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. Reconstruction of 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 \( q \in \mathbb{R}^3 \), the neural network predicts the density \( \sigma(q) \in \mathbb{R}^+ \) and color \( c(q, d) \in \mathbb{R}^3 \), where \( d \) indicates the ray direction passing \( q \) which enables to model view-dependent effects such as reflections.

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

\[
C = \sum_{i=1}^{I} T_i (1 - \exp(-\sigma(q_i) \delta_i)) c(q_i, d),
\]

where \( T_i = \exp(-\sum_{j<i} \sigma(q_j) \delta_j) \) is the accumulated transmittance along the ray and \( \delta_i \) is the Euclidean distance between \( q_{i+1} \) and \( q_i \). 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 \( q_i \) into a higher dimensional space using sinusoidal functions with a frequency band. Formally, the encoding function \( \gamma(q_i) \) is defined below,

\[
(sin(\omega_1 q_i), cos(\omega_1 q_i), ..., sin(\omega_L q_i), cos(\omega_L q_i)),
\]
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.

## 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 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
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 \( e_j \in \mathbb{R}^3 \), where \( j \in [1, J] \) and \( J = R^3 \). We denote the set of quantized coordinates as \( E = \{ e_j | j \in [1, R^3] \} \), and use nearest interpolation in \( E \) to discretize each query \( q_i \) into a discrete coordinate \( \tilde{q}_i \), as formulated by,

\[
\tilde{q}_i = e_k, \quad \text{where} \quad k = \arg \min_j ||q_i - e_j||_2. \tag{4}
\]

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

\[
(sin(\omega_1 \tilde{q}_i), \cos(\omega_1 \tilde{q}_i), ..., sin(\omega_L \tilde{q}_i), \cos(\omega_L \tilde{q}_i)), \tag{5}
\]

where \( \{\omega_1, ..., \omega_L\} \) is also a band containing \( L \) frequencies, and \( \omega_L = 2^{L-1}\pi \). Accordingly, with discrete coordinates \( \tilde{q}_i \), we reformulate volume rendering as,

\[
C = \sum_{i=1}^{t} T_i(1 - \exp(-\sigma(\tilde{q}_i)\delta_i))c(\tilde{q}_i, d), \tag{6}
\]

where \( T_i = \exp(-\sum_{j<i} \sigma(\tilde{q}_i)\delta_j) \) is the accumulated transmittance along the ray and \( \delta_i \) is still the Euclidean distance between continuous coordinates \( q_{j+1} \) and \( q_j \). 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,

\[
\min_{\theta} ||C_{GT} - \sum_{i=1}^{t} V(\{f_\theta(\tilde{q}_i)\})C_{\theta}(\tilde{q}_i, d)||_2^2. \tag{7}
\]

**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 discretize 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 512003 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 (f) and (h).

**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.
We report numerical evaluations in DTU in the continuous coordinates to get precise color from other coordinates. For the warping in NeuralWarp, we still use the normals at the discrete coordinates to calculate positional encodings as in Eq. 5. For NeuS and NeuralWarp to produce our results with discrete coordinates, we use the official code released by UNISURF, NeuS, and NeuralWarp. 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. 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 that 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
Table 1. Numerical comparisons with the latest methods in DTU dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>Chamfer-L1</th>
<th>Precision↑</th>
<th>Recall↑</th>
<th>F-score↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>COLMAP [54]</td>
<td>0.141</td>
<td>0.711</td>
<td>0.441</td>
<td>0.537</td>
</tr>
<tr>
<td>UNISURF [43]</td>
<td>0.359</td>
<td>0.212</td>
<td>0.362</td>
<td>0.267</td>
</tr>
<tr>
<td>NeuS [63]</td>
<td>0.194</td>
<td>0.313</td>
<td>0.275</td>
<td>0.291</td>
</tr>
<tr>
<td>VolSDF [66]</td>
<td>0.267</td>
<td>0.321</td>
<td>0.394</td>
<td>0.346</td>
</tr>
<tr>
<td>NeuRIS [62]</td>
<td>0.050</td>
<td>0.717</td>
<td>0.669</td>
<td>0.692</td>
</tr>
<tr>
<td>MonoSDF [68]</td>
<td>0.042</td>
<td>0.799</td>
<td>0.681</td>
<td>0.733</td>
</tr>
<tr>
<td>Ours</td>
<td>0.039</td>
<td>0.794</td>
<td>0.750</td>
<td>0.770</td>
</tr>
</tbody>
</table>

Table 2. Numerical comparisons in ScanNet.

<table>
<thead>
<tr>
<th>Method</th>
<th>Chamfer-L1</th>
<th>Precision↑</th>
<th>Recall↑</th>
<th>F-score↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>MonoSDF [68]</td>
<td>92.11</td>
<td>2.94</td>
<td>86.12</td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>93.86</td>
<td>2.76</td>
<td>90.16</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Numerical comparison with MonoSDF in Replica.

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.

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-
Table 4. Statistical analysis for our improvements over baselines.

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

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

Table 5. Effect of resolution for quantized coordinates.

<table>
<thead>
<tr>
<th>Resolution</th>
<th>1024</th>
<th>25600</th>
<th>38400</th>
<th>51200</th>
<th>76800</th>
<th>102400</th>
<th>Continuous</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chamfer ↓</td>
<td>1.67</td>
<td>0.70</td>
<td>0.64</td>
<td><strong>0.59</strong></td>
<td>0.62</td>
<td>0.69</td>
<td>0.79</td>
</tr>
</tbody>
</table>

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-
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 reveals 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


[17] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oguz. 3dgen: Triplane latent diffusion for textured mesh generation, 2023. 2


[22] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions, 2023. 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


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


[58] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson,


Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. HF-NeuS: Improved surface reconstruction using high-frequency details. 2022. 1, 2, 6, 7, 9


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