Depth Estimation from Indoor Panoramas with Neural Scene Representation

Wenjie Chang, Yueyi Zhang†, Zhiwei Xiong
University of Science and Technology of China
changwj@mail.ustc.edu.cn, {zhyuey, zwxiong}@ustc.edu.cn

Figure 1. Our work aims to implicitly represent a scene with a neural network through fewer panoramas and thus obtain the relative depth measurements. The left diagram illustrates the efficiency and accuracy of our method against NeRF [18] and a traditional geometric-based Structure from Motion approach [12]. All methods adopt 3 panoramas with camera positions as inputs and are evaluated over 10 scenes from Matterport3D [4]. The right shows aerial views of point clouds that are transferred from depth estimations for better visualization. Compared with state-of-the-art methods, measurements from ours have clearer edges and more accurate geometry.

Abstract

Depth estimation from indoor panoramas is challenging due to the equirectangular distortions of panoramas and inaccurate matching. In this paper, we propose a practical framework to improve the accuracy and efficiency of depth estimation from multi-view indoor panoramic images with the Neural Radiance Field technology. Specifically, we develop two networks to implicitly learn the Signed Distance Function for depth measurements and the radiance field from panoramas. We also introduce a novel spherical position embedding scheme to achieve high accuracy. For better convergence, we propose an initialization method for the network weights based on the Manhattan World Assumption. Furthermore, we devise a geometric consistency loss, leveraging the surface normal, to further refine the depth estimation. The experimental results demonstrate that our proposed method outperforms state-of-the-art works by a large margin in both quantitative and qualitative evaluations. Our source code is available at https://github.com/WJ-Chang-42/IndoorPanoDepth.

1. Introduction

Panoramic imaging has emerged as an attractive imaging technique in many fields, such as computer vision and robotics. Different from traditional imaging devices, panoramic cameras capture a holistic scene and present it as a 2D image with equirectangular projection. Indoor panoramas, captured in the interior scenes by panoramic cameras, have been widely used in interior design and decoration. Recovering depth information aligned with RGB panoramic images benefits a line of downstream applications, such as augmented reality and indoor mapping.

Recent works on depth estimation from panoramas employ Convolutional Neural Network (CNN) structures with prior knowledge learned from depth labels and achieve excellent performance. Most of these works adopt a single panoramic image to predict the relative depth map [7,23,29,31,37,39]. These methods require lots of RGB and depth pairs while training and encounter the problem of domain adaptation in practice. There are a few works attempting to employ multiview panoramic images in the depth estimation task [32,38]. They recover depth information by finding the correspondence of different views. However, strict vertical or horizontal position relations are required for input images in these methods.

Panoramas show great distortions when presented as 2D images. Prior works adopt various technologies to overcome this problem, such as processing panoramas [7,26,27,
tributions are summarized as follows:

• We propose an unsupervised method for depth estimation from multi-view indoor panoramic images by utilizing a neural network with a specially designed positional embedding scheme to implicitly learn the SDF of the scene represented by panoramas.

• Inspired by the Manhattan World Assumption, we propose an initialization method for the network weights for better convergence.

• We devise a loss item based on geometric consistency that the geometric information from depth is supposed to be consistent with the surface norm.

• We release a synthetic panoramic RGB-D dataset rendered from photorealistic indoor scenes. Experimental results on our synthetic dataset and two realistic datasets demonstrate that our proposed method achieves superior performance in both quantitative and qualitative ways.

2. Related work

2.1. Indoor Panoramic Depth Estimation

Image-based depth estimation is a fundamental problem in 3D vision [3, 5, 11, 14, 22, 25]. For depth estimation from panoramas, the severe spatial distortion brought by equirectangular projection is challenging. To deal with this deformation, some methods dispose panorama images with a standard projection. Cheng et al. [7] adopted an additional cubemap projection branch which converts a panorama image to six faces of a cube with perspective projection. Wang et al. [31] proposed a bi-projection fusion component to leverage both projections, which was inspired by both peripheral and foveal vision of the human eye. HoHoNet [29] and SliceNet [23] extracted horizontal 1D feature maps from gravity-aligned equirectangular projections and recovered dense 2D predictions.

Others redesign the convolution kernels for panoramic depth estimation. Specifically, SphereNet [8] and DistConv [30] calculated the sampling positions for the convolution kernels with inverse gnomonic projection, and Fernandez-Labrador et al. [10] defined the convolution over the field of view on the spherical surface with longitudinal and latitudinal angles. Eder et al. [9] proposed a more general method to process images of any structured representation by introducing the corresponding mapping function. Zhuang et al. [37] proposed a combined dilated convolution to process panorama images. A few works fulfill the depth estimation task from multiview panorama images. Zioulis et al. [38] proposed a self-supervised method to train a monocular depth estimation network with two vertical panoramic images. Wang et al. [32] developed a stereo depth estimation spherical method which predicts disparity using the setting of top-bottom camera pairs. Both methods demand a strict spatial relation in the vertical or horizontal direction between different views.

2.2. Neural Scene Representation

Neural scene representation, which encodes a 3D scene with a neural network shows superior performance on 3D
reconstruction and free-view rendering tasks [2, 6, 16–19, 21, 33, 35]. In particular, NeRF [18] has opened a series of research combining neural implicit functions together with volume rendering to achieve photo-realistic rendering results. NeRV [28] took a set of images in a scene illuminated by unconstrained known lighting as input and produced a 3D representation that can be rendered from novel viewpoints under arbitrary lighting conditions as output. Ost et al. [20] proposed a learned scene graph representation, which encodes object transformation and radiance, to efficiently render novel arrangements and views of the scene. Wei et al. [34] utilized both conventional SfM reconstruction and learning-based priors over NeRF [18] for multi-view depth estimation. Yariv et al. [36] modeled the SDF values as a function of geometry in contrast to previous works utilizing volume density. Wang et al. [33] proposed a new volume rendering method to train a neural SDF representation.

3. Background

We first introduce the volume rendering mechanism that obtains the expected color of a camera ray from a scene represented by SDF and directional emitted radiance, following NeRF [18], NeuS [33] and techniques in [15]. A camera ray $r(t)$ passing through the scene with near and far bounds $t_n$ and $t_f$ is denoted by

$$r(t) = o + td, \quad r(t), o, d \in \mathbb{R}^3, \quad t \in [t_n, t_f],$$

where $o$ is the camera origin and $d$ is a unit direction vector. With a discrete set of 3D coordinates $\{r(t_i)\}_{i=1}^{N}$, $t_i \in \[t_n, t_f\]$ sampled along the defined camera ray, the expected color is rendered as

$$C(r) = \sum_{i=1}^{N} T_i \alpha_i c_i, \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_j),$$

where $T_i$ is the discrete accumulated transmittance, $c_i$ is the radiance value at the sampled 3D coordinates $r(t_i)$ with direction $d$. $\alpha_i$ is the discrete opacity value, which is defined with SDF values as

$$\alpha_i = \text{max}\left(\frac{\Phi_s(\sigma_i) - \Phi_s(\sigma_{i+1})}{\Phi_s(\sigma_i)}, 0\right),$$

where $\sigma_i$ denotes the SDF value at the sampled location $r(t_i)$, $\Phi_s(x)$ is the sigmoid function equated as $\Phi_s(x) = (1 + e^{-sx})^{-1}$ and $s$ in $\Phi_s(x)$ is a learnable parameter. The rendered color $C(r)$ is then employed for the loss function.

4. Method

4.1. Proposed Network

In this section, we present the pipeline of the proposed method. First, pixels in panoramas are back-projected to a unit sphere and then the relative camera rays are calculated with camera positions. Afterwards, a set of 3D coordinates are sampled from each obtained camera ray. Our constructed network takes the embedded coordinates as input and outputs the corresponding SDF and radiance values. Finally, the expected color of each camera ray is rendered with the SDF and radiance values of the sampled coordinates for optimization. Details of each step are introduced separately.

**Back-projection.** For a point $[m, n]$ in Image Coordinate, the relative polar angle $\theta$ and azimuthal angle $\phi$ are calculated as

$$\theta = 2\pi \frac{m - c_x}{W}, \quad \phi = \frac{n - c_y}{H},$$

where $W$ and $H$ represent the width and height of the panoramic image respectively. $c_x$ and $c_y$ are coordinates of the principal points. The direction of the camera ray passing through $[m, n]$ is calculated as

$$d = [\cos(\phi) \sin(\theta), \sin(\phi), \cos(\phi) \cos(\theta)]^T.$$

**Sampling.** For each obtained camera ray, we sample $N_c + N_f$ coordinates with a coarse-to-fine sampling strategy. In practice, we first sample $N_c$ positions with a stratified sampling approach, which partitions the near and far bounds $[t_n, t_f]$ into $N_c$ evenly spaced bins and then randomly draws one sample $t_i$ from each bin. The mathematical expression of $t_i$ which denotes the distance between the sampled 3D coordinates and the camera origin, is formulated as

$$t_i \sim U[t_n + \frac{i - 1}{N}(t_f - t_n), t_n + \frac{i}{N}(t_f - t_n)].$$

The SDF values of the $N_c$ samples predicted by networks are utilized to calculate a probability density function (PDF) as

$$\text{PDF}(n) = \frac{w_n}{\sum_{i=1}^{N_c} w_i}, \quad w_i = T_i \alpha_i.$$  

Then, additional $N_f$ samples are obtained with the calculated PDF and inverse transform sampling. Finally, the expected color of a camera ray is rendered with $N_c + N_f$ sampled 3D coordinates.

**Positional Embedding.** Rahaman et al. [24] showed that using high frequency functions to map the network’s inputs to a higher dimensional space enables better fitting of data containing high frequency variation. As a result, NeRF [18] adopts a positional embedding scheme for the sampled 3D coordinates, which is commonly followed by Neural Rendering works and represented as

$$\gamma(x) = (\sin(2^0 x), \cos(2^0 x), \cdots, \sin(2^{L-1} x), \cos(2^{L-1} x)).$$
Here $\gamma$ is a mapping from $\mathbb{R}$ to a higher dimensional space $\mathbb{R}^{2L}$. However, two values $x_1$ and $x_2$ would be embedded as the same high dimensional vectors under the following condition

$$|x_1 - x_2| = 2\pi n, n = 0, 1, 2, \cdots.$$  

To avoid this ambiguity, all coordinates in the sampled position $r(t_i)$ are supposed to be in $[-\pi, \pi]$. NeRF [18] resolves this problem by scaling the whole scene to Normal Device Coordinate, encompassing a cube where the $x$, $y$, and $z$ components range from -1 to 1. To address these limitations, we develop a new representation scheme, called Sphere Embedding. First, $r(t_i) = [x, y, z]^T$ is converted in Sphere Coordinate, denoted as $[\theta, \phi]^T$. Then, it is encoded in a fusion representation, which is formulated as

$$
\begin{align*}
\mu(\theta, \phi)^T &= [\cos(\theta), \sin(\theta), \sin(\phi), \\
&\cos(\theta) \cos(\phi), \sin(\theta) \cos(\phi), 1/\rho]^T.
\end{align*}
$$

The input of the MLP network is $\gamma(\mu(r(t_i)))$. For this expression, the value space for $\rho$ without ambiguity is $[\frac{1}{\pi^2}, +\infty)$, which is more appropriate for indoor scenes.

**Network Architecture.** We construct two Multilayer-Perceptrons (MLPs) to learn the geometry field $\mathbb{F}_g$ and color field $\mathbb{F}_c$ separately. The geometry network adopts the embedded position $\gamma(r(t_i))$, outputting SDF value $\sigma$ and geometric feature vector $v \in \mathbb{R}^{256}$. Here, $\sigma$ determines the distance between the 3D location $r(t_i)$ and its closest surface.

Then, we employ a color network to estimate color $c \in \mathbb{R}^3 = [r, g, b]^T$ at point $r(t_i)$. The color network takes the geometric feature vector $v$, the embedded position $\gamma(\mu(r(t_i)))$ and direction vector $\gamma(d)$ as inputs.

Both the geometry field $\mathbb{F}_g$ and the color field $\mathbb{F}_c$ utilize MLPs with 8 fully-connected layers and each layer consists of a linear transformation with the ReLU activation. Specifically, a skip connection operation is applied to the input of the $5^{th}$ layer in MLPs, which concatenates the embedded 3D information ($\gamma(r(t_i))$ for the geometry network and $\gamma(d)$ for the color network) with feature vectors from the $4^{th}$ layer.

**Optimizing.** The constructed networks predict the SDF value $\sigma_i$ and radiance value $c_i$ of the samples $\{r(t_i)\}$, and then the expected color of the camera ray $r$ is calculated with Eq. 2. The main loss function minimizes the difference between the rendered and true pixel colors, which is formulated as

$$
\mathcal{L}_C = \sum_{r \in \mathcal{R}} \|\hat{C}(r) - C(r)\|,
$$

where $\mathcal{R}$ is a batch of camera rays, $\hat{C}$ is the reference color and $C$ is the rendered RGB colors.

**4.2. Initialization.**

Atzmon and Lipman [2] proposed an initialization method that makes MLPs approximate an SDF representing a sphere in the 3D space before training. Indoor scenes are always different from spheres, but generally obey the Manhattan World Assumption, where floors and ceilings are always vertical to the gravity direction. Hence, it is a more reasonable choice to make the geometry field optimized from a shape that approximates the floors and ceilings. Inspired by [2] and the Manhattan World Assumption, we develop an initialization scheme for indoor panoramas and details are introduced in the following.

A single fully-connected layer in MLPs is denoted as

$$f_i(y) = ReLU(W_i y + b_i),$$

where $W_i \in \mathbb{R}^{d_{in} \times d_{out}}$, $b_i \in \mathbb{R}^{d_{out}}$, $y \in \mathbb{R}^{d_{in}}$. Then, the MLP used in our networks is formulated as

$$f([x, y, z]^T; \theta) = w^T f_1 \circ \cdots \circ f_1([x, y, z]^T) + b,$$

where $w \in \mathbb{R}^{d_{out}}$, $b \in \mathbb{R}$, $[x, y, z]$ denotes the input 3D location in Cartesian Coordinate and $\theta =$

---

Figure 2. Illustration of the proposed method. Given the camera positions, we back project the pixels to a unit sphere to get camera rays. Then, a discrete set of sampled points along the ray is fed to our networks after positional embedding. Especially, geometry network adopts the same embedding way with NeRF and color network utilizes the proposed Sphere Embedding. Colors of each ray for optimizing are estimated from predicted SDF $\sigma$ and RGB $c$ with the volume rendering strategy. Init. Geo. is the visualization of the geometry field which is initialized with our proposed scheme before training. Optimized Geo. shows the scene denoted by geometry network after optimized.
Algorithm 1 Initialization Scheme

**Input**: Network Parameters, \((W_l, b_l, \ldots, W_b, b_w, b)\).

**Note**: \(w \in \mathbb{R}^{d_{out}}\), \(b \in \mathbb{R}\), \(W_l \in \mathbb{R}^{d_{out} \times d_{in}}\), \(b_i \in \mathbb{R}^{d_{out}}\), \(W_i = [a_1, a_2, \ldots, a_{d_{out}}]\), \(a \in \mathbb{R}^{d_{out}}\).

1. Let \(i = 1\).
2. **while** \(i \leq l\) **do**
3.  **if** \(i = 1\) **then**
4.   Set \(b_1 = 0, W_1 = 0\).
5.  **else**
6.   Set \(a_2\) in \(W_l\) i.i.d. normal \(\mathcal{N}(0, \frac{\sqrt{2}}{d_{in}})\).
7. **end if**
8. \(i = i + 1\).
9. **end while**
10. Set \(w = \sqrt{\pi}/\sqrt{d_{out}}, b = -c\).

---

![Figure 3: Illustration of the initialization scheme to the geometry field. We visualize the reconstruction results as point clouds in different training iterations. Our method shows a better reconstruction performance and faster convergence speed.](image)

\(\langle W_l, b_l, \ldots, W_b, b_w, b \rangle\) represents the parameters of the MLP. The initialization of parameters of the MLP is illustrated in Algorithm 1.

With the proposed scheme, the MLPs defined by Eq. 13 denotes an SDF \(f([x, y, z]^T, \theta) \approx |y| - c\). As Fig. 3 shows, after initialization, the geometry network is optimized from a shape approximating floors and ceilings in indoor scenes and converges more quickly. The proof of the initialization is provided in the supplementary material.

### 4.3. Geometric Consistency

The loss function in Eq. 11 only optimizes the proposed network with the rendered colors. As a result, we propose a new loss item to constrain the consistency of different properties learned by the geometric network. In addition to depth information, the surface normal is also one of the significant properties utilized to characterize the geometric of a scene. With our constructed geometric network, surface normal could be extracted from the points on the surface and the gradient of the network respectively.

Given a batch of camera rays \(\mathcal{R} = \{r_k(t) = o_k + td_k | 1 \leq k \leq K\}\), the expected depth \(D(r)\), which represents the distance from the camera origin to a point on the surface, is rendered with

\[
D(r) = \sum_{i=1}^{N} T_i \alpha_i t_i, \quad r \in \mathcal{R},
\]

where \(\alpha\) and \(T_i\) are introduced in Sec. 3 and \(t_i\) denotes the distance between the sampled 3D coordinates and the origin of ray \(r\). The 3D coordinates on surface along the ray are calculated by

\[
a_k = o_k + D(r_k)d_k, \quad a_k \in \mathbb{R}^3, \quad 1 \leq k \leq K,
\]

The corresponding matrix is constructed as

\[
A = [a_1, a_2, \ldots, a_K]^T \in \mathbb{R}^{K \times 3}.
\]

Then, the surface normal is calculated with

\[
n_d = \frac{(A^T A)^{-1} A^T 1}{\|{(A^T A)^{-1} A^T 1}\|_2},
\]

where \(1 \in \mathbb{R}^K\) is a vector with all 1 elements.

Meanwhile, the gradient of the geometric network could also be applied to calculate the surface normal. For a set of 3D points \(\{r(t_i) | i = 1, \ldots, N, t_i < t_{i+1}\}\) sampled from a camera ray \(r\), the normal vector at each point \(r(t_i) = [x_i, y_i, z_i]\) is defined by the partial derivative of the predicted SDF value \(\alpha_i\) as

\[
n_i = -\left(\frac{\partial \alpha_i}{\partial x_i}, \frac{\partial \alpha_i}{\partial y_i}, \frac{\partial \alpha_i}{\partial z_i}\right)
\]

Then, the normal vector of the surface hit by camera ray \(r\) is computed by

\[
N(r) = \sum_{i=1}^{N} T_i \alpha_i n_i,
\]

here \(\alpha\) and \(T_i\) are introduced in Sec. 3. Finally, with the normal vectors \(\{N(r_k) | 1 \leq k \leq K\}\) calculated from Eq. 19 and \(n_d\) from Eq. 17, the consistency loss function is formulated as

\[
\mathcal{L}_{GC} = 1 - \cos \left(n_d, \frac{1}{K} \sum_{k=1}^{K} N(r_k)\right),
\]

which minimizes the difference of the estimated surface normal from these two different ways and further refines the depth estimations in our evaluations.
5. Experiments

5.1. Datasets

**IPMP.** To evaluate our approach and other Neural Radiance Fields methods, we construct a synthetic dataset with multiview panorama images named the Indoor Photorealistic Multiview Panoramas dataset. We select 5 different indoor scenes and render 12 photorealistic images in Blender at different viewpoints for each scene. All images are rendered with $512 \times 1024$ pixels from a panoramic camera. Different from other datasets that only offer multiview images at fixed positions with only 3 views, our dataset provides more views at various positions and does not follow a strict spatial relation in the vertical or horizontal direction between different views. Compared with other synthetic panorama RGB-D datasets, IPMP is rendered from high polygon models with the path-tracing algorithm, outputting high quality RGB images and precision depth maps.

**Matterport3D.** Matterport3D [4] is a large-scale RGB-D dataset containing panoramic views from RGB-D images of 90 building-scale scenes. In our experiments, we adopt the dataset processed by [38] and select 10 different scenes for evaluation.

**Stanford2D3D.** Stanford2D3D [1] is collected in 6 large-scale indoor areas that originate from 3 different buildings of educational and official use, providing equirectangular RGB images, as well as their corresponding depths. We select 5 different scenes for evaluation from the remake\(^1\) by [38].

5.2. Implementation details

We implement our network with the PyTorch framework and train 100 epochs on a single NVIDIA RTX3090 GPU with approximately 6 GB video memory in 6.71 hours. Specifically, in the first 80 epochs, we randomly sample 512 camera rays for each optimizing iteration and train the network only with the main color loss $L = L_C$. For the last 20 epochs, the camera rays are obtained from 32 image patches with $4 \times 4$ pixels, which means the $K$ in Eq. 20 is set to 16. The loss function in this stage is $L = L_C + 0.01L_{GC}$. We use the Adam optimizer [13] with a learning rate that begins with $5 \times 10^{-4}$ and decays exponentially to $5 \times 10^{-5}$ during optimization. The hyper-parameter $c$ in the proposed initialization scheme is set to 1.5.

5.3. Evaluation

**Baselines.** To evaluate our approach, we conduct comparisons against scene representation methods which directly learn geometric information from multiview RGB images, a supervised stereo method and a geometric-based structure from motion (SfM) method. For scene representation methods, we compare with NeRF [18], NeuS [33] and VolSDF [36]. Particularly, NeRF adopts volume density to represent the 3D information, and NeuS [33] and VolSDF [36] utilize SDF to denote the surface of a scene. Moreover, we choose 360SD-Net [32], a supervised stereo method which achieves advanced depth estimation results with panoramic images, and ADISM [12], a traditional geometric-based SfM method. In addition to our origin model, we also provide a fast model, named Ours-fast, which is trained for only 10 epochs without the learning rate decay.

Following [31, 39], we use the following metrics to evaluate the performance: mean absolute error (MAE), mean relative error (MRE), mean square error of linear measures (MSE\(^2\)) and relative accuracy $\delta_1$ (the fraction of pixels where the relative error is within a threshold of 1.25). All errors are calculated in meters.

**Quantitative Results.** We compare with the advanced scene representation methods on 2 realistic datasets and the proposed dataset. All methods are trained with 3 views for a scene. As shown in Table 1, NeuS [33] and VolSDF [36] fail to reconstruct correctly and ours achieves the best results.

In addition, comparisons with prior advanced depth estimation works are constructed to demonstrate the outstanding performance of the proposed framework. As Table 3 shows, our approach presents competitive performance

\[\begin{array}{|c|c|c|c|c|}
\hline
\text{Dataset} & \text{Method} & \text{MRE ↓} & \text{MSE ↓} & \text{$\delta_1$ ↑} \\
\hline
\text{IPMP} & \text{NeRF} & 0.1890 & 0.5240 & 0.6712 \\
 & \text{NeuS} & 1.0786 & 13.7711 & 0.4386 \\
 & \text{VolSDF} & 0.5821 & 3.1328 & 0.0970 \\
 & \text{Ours} & 0.0641 & 0.0955 & 0.8975 \\
 & \text{Ours-fast} & 0.1546 & 0.2629 & 0.7483 \\
\hline
\text{M3D} & \text{NeRF} & 0.1006 & 0.5442 & 0.8551 \\
 & \text{NeuS} & 0.8232 & 20.3833 & 0.5163 \\
 & \text{VolSDF} & 0.3018 & 1.9464 & 0.5298 \\
 & \text{Ours} & 0.0258 & 0.0532 & 0.9902 \\
 & \text{Ours-fast} & 0.0801 & 0.1680 & 0.9331 \\
\hline
\text{S2D3D} & \text{NeRF} & 0.1209 & 0.4262 & 0.7960 \\
 & \text{NeuS} & 0.4846 & 8.0183 & 0.6290 \\
 & \text{VolSDF} & 0.5114 & 2.5315 & 0.2442 \\
 & \text{Ours} & 0.0352 & 0.0732 & 0.9790 \\
 & \text{Ours-fast} & 0.0550 & 0.1322 & 0.9423 \\
\hline
\end{array}\]

\(^1\)The processed dataset suffers from depth leakage problems, and we provide more discussion in the supplementary.

\(^2\)Since many works adopt RMSE metric for evaluations, we also report the results with the standard RMSE in the supplementary.
Figure 4. Qualitative comparisons on ours, NeRF [18], NeuS [33] and VolSDF [36]. (a) shows one view from the input images of a scene and (b) is the related ground truth depth map. All depth maps are visualized in a same depth range.

<table>
<thead>
<tr>
<th>Method</th>
<th>Setting</th>
<th>MAE ↓</th>
<th>MRE ↓</th>
<th>MSE ↓</th>
<th>δ₁ ↑</th>
<th>Setting</th>
<th>MAE ↓</th>
<th>MRE ↓</th>
<th>MSE ↓</th>
<th>δ₁ ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>NeRF</td>
<td>3-View</td>
<td>0.4270</td>
<td>0.1890</td>
<td>0.5240</td>
<td>0.6712</td>
<td>6-View</td>
<td>0.1943</td>
<td>0.0938</td>
<td>0.1783</td>
<td>0.9086</td>
</tr>
<tr>
<td>NeuS</td>
<td></td>
<td>2.0442</td>
<td>1.0786</td>
<td>13.7711</td>
<td>0.4386</td>
<td></td>
<td>0.2936</td>
<td>0.1407</td>
<td>1.2934</td>
<td>0.8870</td>
</tr>
<tr>
<td>VolSDF</td>
<td></td>
<td>1.3927</td>
<td>0.5821</td>
<td>3.1328</td>
<td>0.0970</td>
<td></td>
<td>0.5339</td>
<td>0.2403</td>
<td>1.0874</td>
<td>0.5545</td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td>0.1266</td>
<td>0.0641</td>
<td>0.0955</td>
<td>0.8975</td>
<td></td>
<td>0.0953</td>
<td>0.0500</td>
<td>0.0874</td>
<td>0.9494</td>
</tr>
<tr>
<td>NeRF</td>
<td>9-View</td>
<td>0.1411</td>
<td>0.0724</td>
<td>0.1112</td>
<td>0.9394</td>
<td>12-View</td>
<td>0.1231</td>
<td>0.0618</td>
<td>0.0939</td>
<td>0.9478</td>
</tr>
<tr>
<td>NeuS</td>
<td></td>
<td>0.1489</td>
<td>0.0727</td>
<td>0.1659</td>
<td>0.9273</td>
<td></td>
<td>0.1335</td>
<td>0.0635</td>
<td>0.1778</td>
<td>0.9423</td>
</tr>
<tr>
<td>VolSDF</td>
<td></td>
<td>0.3271</td>
<td>0.1356</td>
<td>0.6523</td>
<td>0.7688</td>
<td></td>
<td>0.2133</td>
<td>0.0856</td>
<td>0.3475</td>
<td>0.8504</td>
</tr>
<tr>
<td>Ours</td>
<td></td>
<td>0.0711</td>
<td>0.0392</td>
<td>0.0560</td>
<td>0.9633</td>
<td></td>
<td>0.0747</td>
<td>0.0401</td>
<td>0.0650</td>
<td>0.9656</td>
</tr>
</tbody>
</table>

Table 2. Evaluation results on our proposed dataset. We quantify the performance of each method using different numbers of images. The results are averaged on five scenes with image views used for training. Our method quantitatively outperforms all prior Neural Radiance Fields work in all settings.

against the supervised method and outperforms the traditional SfM method by a large margin. It should be noted that 360SD-Net [32] needs a top-bottom image pair that follows a strict vertical relation and is trained on a fusion dataset with RGB-D labels provided by [38]. Our approach does not require any prior knowledge from the ground truth and is evaluated on two settings. One is trained with 2 views (the same image pair as 360SD-Net) and the other is trained with 3 views. ADfSM is implemented with 3 views and the provided camera positions. 360SD-Net [32] is trained with RGB-D pairs from the total training set and evaluated on scenes selected from the evaluation set. Our method directly works on the panoramas from the evaluation set without any 3D supervision.

Qualitative Results. Fig. 4 shows qualitative results on our synthetic dataset IPMP and two real-world datasets Matterport3D and Stanford2D3D. All depth maps are shown in the same depth range for fair comparisons. The results on the synthetic dataset show that our method can better manage complex lighting conditions and obtain a more reasonable 3D representation (1st and 2nd rows). The results from the real-world dataset show that our method has a better performance in areas with severe distortion (4th and 6th rows), while a more accurate depth estimation result can be ob-
Table 3. Comparisons with 360SD-Net [32] and ADFSM [12]. All metrics are evaluated on the depth estimations of the top image from the 360SD-Net’s inputs and averaged over 10 scenes on Matterport3D. Our method shows comparable performance against the supervised method with one more view.

<table>
<thead>
<tr>
<th>Method</th>
<th>MRE ↓</th>
<th>MSE ↓</th>
<th>δ1 ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADfSM</td>
<td>0.0493</td>
<td>0.1985</td>
<td>0.9598</td>
</tr>
<tr>
<td>360SD-Net</td>
<td>0.0322</td>
<td>0.0740</td>
<td>0.9760</td>
</tr>
<tr>
<td>Ours(2-view)</td>
<td>0.0713</td>
<td>0.1944</td>
<td>0.9427</td>
</tr>
<tr>
<td>Ours(3-view)</td>
<td><strong>0.0277</strong></td>
<td><strong>0.0525</strong></td>
<td><strong>0.9901</strong></td>
</tr>
</tbody>
</table>

Table 4. An ablation study of different positional embedding schemes for the inputs of the color network and the initialization scheme for the geometry network. Metrics are averaged over the 10 scenes from Matterport3D. The averaged evaluation metrics are MRE: 0.0288, MSE: 0.0592 and Δ1: 0.9884. Referring to the 4th row in Table 4, the developed positional embedding scheme, the initialization method and the new loss item for geometric consistency greatly improve the performance of the constructed networks when compared with other methods.

<table>
<thead>
<tr>
<th>Embedding</th>
<th>Init.</th>
<th>MRE ↓</th>
<th>MSE ↓</th>
<th>δ1 ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>[x, y, z]</td>
<td>Ours</td>
<td>0.3606</td>
<td>2.1447</td>
<td>0.3348</td>
</tr>
<tr>
<td>[x/s, y/s, z/s]</td>
<td>Ours</td>
<td>0.4228</td>
<td>2.6902</td>
<td>0.2497</td>
</tr>
<tr>
<td>[θ, ϕ, 1/ρ]</td>
<td>Ours</td>
<td>0.1073</td>
<td>0.4034</td>
<td>0.8682</td>
</tr>
<tr>
<td>S.E.</td>
<td>Ours</td>
<td><strong>0.0258</strong></td>
<td><strong>0.0532</strong></td>
<td><strong>0.9902</strong></td>
</tr>
<tr>
<td>S.E.</td>
<td>None</td>
<td>0.1615</td>
<td>0.1767</td>
<td>0.8897</td>
</tr>
<tr>
<td>S.E.</td>
<td>Sphere</td>
<td>0.0430</td>
<td>0.1032</td>
<td>0.9717</td>
</tr>
</tbody>
</table>

Table 5. Experimental results on scenes that are not strictly following the Manhattan World Assumption. Metrics are averaged over 10 artificially disturbed scenes from Matterport3D and each scene is trained with 3 views.

<table>
<thead>
<tr>
<th>Degree</th>
<th>MAE</th>
<th>MRE ↓</th>
<th>MSE ↓</th>
<th>δ1 ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>0°</td>
<td><strong>0.0731</strong></td>
<td><strong>0.0258</strong></td>
<td><strong>0.0532</strong></td>
<td><strong>0.9902</strong></td>
</tr>
<tr>
<td>1°</td>
<td>0.0785</td>
<td>0.0269</td>
<td>0.0610</td>
<td>0.9892</td>
</tr>
<tr>
<td>3°</td>
<td>0.0840</td>
<td>0.0310</td>
<td>0.0616</td>
<td>0.9826</td>
</tr>
<tr>
<td>5°</td>
<td>0.0843</td>
<td>0.0294</td>
<td>0.0629</td>
<td>0.9886</td>
</tr>
</tbody>
</table>

In this section, we construct a series of experiments to discuss the number of views in training and investigate the effectiveness of the proposed positional embedding scheme, the initialization method and the new loss item for geometric consistency.

**Number of Views.** The effects of using different numbers of image views on the depth estimation results are explored on our synthesized dataset. Table 2 shows the quantitative results. All methods are well converged with 6, 9 and 12 views. However, when fewer images are available, NeuS [33] and VoISDF [36] fail to generate satisfactory results. Our method achieves the best results among all settings and outperforms other methods by a large margin.

**Positional Embedding.** We perform ablation experiments to validate the proposed Sphere Embedding techniques with a series of experiments on different embedding schemes. Quantitative results presented in the 1st – 4th rows of Table 4 demonstrate that the proposed Sphere Embedding greatly improves the performance of the constructed networks when compared with other methods.

**Initialization.** To validate the effectiveness of the proposed initialization scheme, we deploy different initialization strategies for the geometry network, including no initialization, Sphere initialization [2], and our proposed initialization. The results are shown in the 4th – 6th rows of Table 4. Our proposed initialization method leads to the best depth estimation results. Table 5 shows the experiments on the robustness of the proposed initialization method when the normal of the ground in each scene deviates from the gravity direction by a few degrees. Even if the assumption that floors and ceilings are vertical to the gravity direction does not strictly hold, the proposed initialization scheme still has a superior performance.

**Geometric Consistency Loss.** To evaluate the effectiveness of the proposed Geometric Consistency Loss, we train the proposed network with only the color loss LC for 100 epochs on 10 scenes from Matterport3D. The averaged evaluation metrics are MRE: 0.0288, MSE: 0.0592 and Δ1: 0.9884. Referring to the 4th row in Table 4, the developed loss reduces the MSE by 10.14%.

6. Conclusion

In this paper, we propose a framework to estimate depth from multiview indoor panoramas with neural scene representation. The developed positional embedding scheme, initialization and geometric consistency loss improve our networks to obtain accurate measurements efficiently. Experimental results demonstrate that our method outperforms state-of-the-art NeRF-based methods in both qualitative and quantitative ways.

7. Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grants 61901435, 62131003 and 62021001.
References

[12] Sunghoon Im, Hyowon Ha, François Rameau, Hae-Gon Jeon, Gyeongmin Choe, and In So Kweon. All-around depth from small motion with a spherical panoramic camera. In Proceedings of the European Conference on Computer Vision, pages 156–172. Springer, 2016. 1, 6, 8


