MuSHRoom: Multi-Sensor Hybrid Room Dataset for Joint 3D Reconstruction and Novel View Synthesis

Xuqian Ren,[1] Wenjia Wang,[2] Dingding Cai,[1] Tuuli Tuominen,[1] Juho Kannala,[3] Esa Rahtu[1]
[1]Tampere University, Finland [2]The University of Hong Kong, China [3]Aalto University, Finland
{xuqian.ren,dingding.cai,tuuli.tuominen,esa.rahtu}@tuni.fi wwj2022@connect.hku.hk
Juho.Kannala@aalto.fi

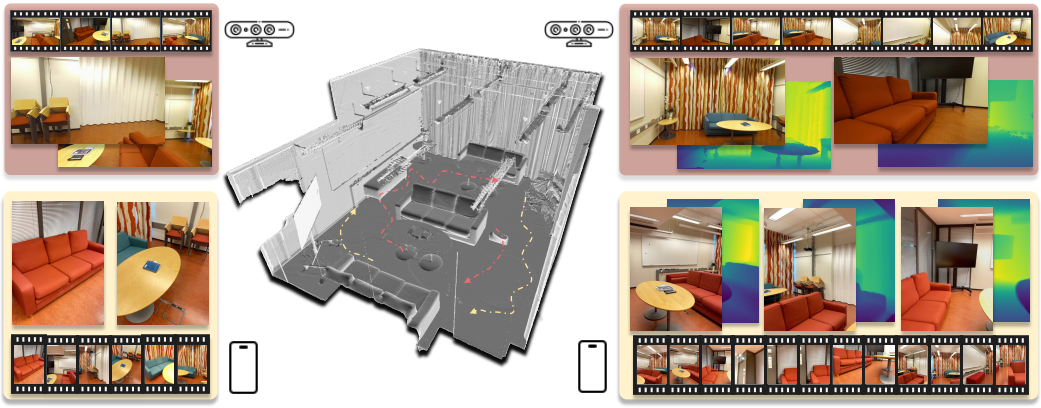

Figure 1. The proposed MuSHRoom dataset includes 10 rooms captured by consumer devices Kinect and iPhone, and each room provides ground-truth mesh models obtained by a Faro scanner. Both Kinect and iPhone capture one long and one short RGB-D sequence for simulating a typical VR/AR use case. The MuSHRoom dataset provides camera poses and point clouds for Kinect and iPhone sequences. The dash lines demonstrate the rough capture trajectories. This dataset is intended for benchmarking room-scale 3D reconstruction and novel view synthesis.

Abstract

*Metaverse technologies demand accurate, real-time, and immersive modeling on consumer-grade hardware for both non-human perception (e.g., drone/robot/autonomous car navigation) and immersive technologies like AR/VR, requiring both structural accuracy and photorealism. However, there exists a knowledge gap in how to apply geometric reconstruction and photorealism modeling (novel view synthesis) in a unified framework. To address this gap and promote the development of robust and immersive modeling and rendering with consumer-grade devices, first, we propose a real-world **Mu**lti-**S**ensor **H**ybrid **Room** Dataset (MuSHRoom). Our dataset presents exciting challenges and requires state-of-the-art methods to be cost-effective, robust to noisy data and devices, and can jointly learn 3D reconstruction and novel view synthesis instead of treating them as separate tasks, making them ideal for real-world applications. Second, we benchmark several famous pipelines on our dataset for joint 3D mesh reconstruction and novel view synthesis. Finally, in order to further improve the overall performance, we propose a new method that achieves a good trade-off between the two tasks. Our dataset and benchmark show great potential in promoting the improvements for fusing 3D reconstruction and high-quality rendering in a robust and computationally efficient end-to-end fashion. The dataset and code are available at the project website:* https://xuqianren.github.io/publications/MuSHRoom/.

1. Introduction

An effective way for artificial intelligence to understand and interact with the tangible realm is to simulate and extrapolate physical objects into a digital environment with the help of sensory input signals, such as RGB images or RGB-D images captured by cameras. To realize the task of creating virtual representations of tangible entities, geo-

metric reconstruction (3D reconstruction) and photorealism modeling (novel view synthesis, NVS) tasks have been proposed, and both of them play a significant role in the development of VR/AR [13, 17]. Nonetheless, current room-scale datasets do not support evaluating the two tasks jointly in a quantitative way, which hinders the state-of-the-art methods applied to VR/AR applications that require both geometry accuracy and photorealism.

Most of the current room-scale datasets [4, 32] either only contain RGB/RGB-D inputs without ground truth meshes for 3D reconstruction comparison, or are over-cleaned [2, 33] and cannot fully reflect the challenges in the real world. Redwood Scan Dataset [25] provides RGB-D inputs of real-world room scenes and industrial laser scans for mesh reference. However, it only uses a single-capturing device in each room to capture a single sequence, which is not enough for simulating the real VR/AR use case.

Considering the lack of proper benchmark and datasets, we propose a real-world **Mu**lti-**S**ensor **H**ybrid **Room** Dataset (**MuSHRoom**). Our dataset focuses on indoor room-scale scenarios and raises interesting real-world challenges on occlusion, motion blur, reflection, transparency, sparseness, illumination diversity, etc. Each room is captured with the Azure Kinect and iPhone consumer device for RGB-D sequences as inputs and an industrial laser scanner as geometry ground truth reference. For each consumer device, we capture two sequences: one long capture with most of the details inside the room and another shorter sequence captured with an independent trajectory.

Based on the MuSHRoom dataset, we propose a new benchmark, aiming to evaluate both the reconstruction and NVS ability of methods. Furthermore, we also propose a new protocol for practical NVS evaluation. When evaluating NVS, previous methods [4, 32] usually uniformly sample frames from the whole sequences as the test set, which does not reflect the real case in VR/AR. In our comparison protocol, we use the long sequences as the training set and the short sequences as the test set, which raises challenges in robustness since the camera positions and view directions have a large gap between these two captures when observing the same objects. This evaluation protocol is common in AR/VR applications where the users will scan the whole room for the first time, and then VR glasses will render the reality according to the positions and view directions of users.

Most existing pipelines are designed to either perform excellent geometry modeling or photorealistic rendering. Based on the proposed MuSHRoom dataset and benchmark, we provide an extensive comparison of previous pipelines for both reconstruction and rendering quality. Moreover, we propose a new method, employing a two-head structure and enriching the training dataset through data augmentation. Our method can obtain a trade-off between geometry and synthesis accuracy. The comparison also shows that the need for achieving both reconstruction and NVS tasks at the same time is clear and a long way.

Our contributions can be summarized as follows:

- We make one of the first attempts to construct a dataset collected with multiple sensors for joint 3D reconstruction and novel view synthesis. We provide a detailed pipeline and program codes for capturing and processing the data, including information on the hardware setup, data acquisition, and post-processing steps. Our pipeline serves as a comprehensive guide for researchers interested in creating similar datasets.

- We provide an extensive comparison of our proposed method with previous methods based on our new benchmark. Our evaluation provides insights into the strengths and limitations of each pipeline and their applicability to real-world scenarios.

- Our dataset raises new real-world challenges and practical evaluation protocol for the state-of-the-art methods to apply to real applications and encourages further exploration of the challenges and opportunities presented by our dataset.

2. Related Work

There are numerous datasets for the 3D reconstruction or novel view synthesis tasks. Therefore, we limit our discussion to the most related scene-level datasets and introduce benchmarks used for modeling and rendering.

3D Room-Level Datasets. Redwood Scan [25] captures five real-world rooms with one single RGB-D camera and Faro scanner. It is the most similar dataset to ours, but it only used one device to get the color and depth images. Neural RGB-D Synthesis Datasets [2, 33] are unified synthesized datasets that can be used for joint 3D reconstruction and novel view synthesis comparison. To simulate real-world captures, noise and artifacts are manually added to the depth images, and BundleFusion [12] is used to generate the estimated pose annotations. However, real-world noise caused by motion blur, shaking, reflection, etc., cannot be easily simulated. Thus, the domain gap between these datasets and real scenes still remains. ETH3D [30] releases ground truth laser scans with registered images captured by multiple devices. However, they only provide high-resolution RGB images without depth, and the other device only provides grayscale low-resolution images. ScanNet [11] is targeted for 3D scene understanding. It contains a large volume of RGB-D sequences and is a valuable dataset for room-scale 3D reconstruction. However, the evaluation can only be conducted qualitatively due to the lack of ground truth mesh models.

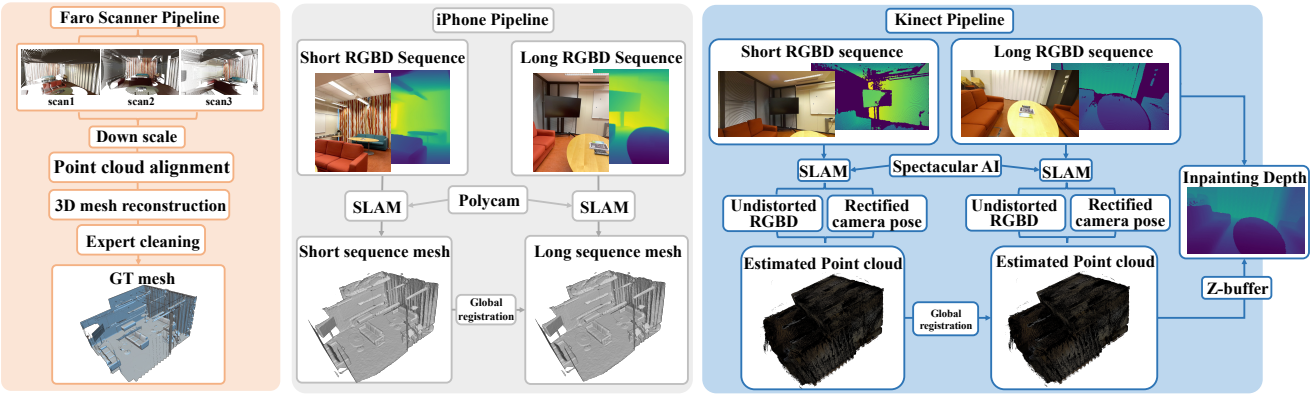

Figure 2. The process pipeline. We use a Faro Scanner to obtain point clouds of the room from different locations and stitch them to create a complete model of the room, compensating for occluded areas. We use spectacular AI SDK to extract the undistorted RGB-D and camera pose for Kinect sequences and use the z-buffer to project point clouds into pixel coordinates to in-paint the raw depth. iPhone sequences are processed and registered by Polycam pose. Long/short captures of each consumer device are registered with global registration and further refined by COLMAP [29] bundle adjustment.

With the rapid prosperity of research on NeRF [21], new datasets are proposed for novel view synthesis. Nerfstudio Dataset [32] contains object-scale and room-scale scenes captured with a mobile phone or mirrorless camera. Mip-NeRF 360 Dataset [4] includes five outdoor scenes and four indoor scenes, among which only one sequence is captured in the room-scale scenario. Note that only RGB images are provided in the last two datasets. Some datasets [6,8,20,31,36] are commonly used for scene reconstruction and rendering. However, they are either synthetic datasets [8,31] or lack ground truth meshes [20,36]. [6] provides real RGB-D images with ground truth meshes but is only captured by a single device.

3D Reconstruction and NVS Methods. Commercial software applications, such as Pixel4D [26] and Reality Capture [28], can be used for image-based reconstruction. However, they require dense input sequences to guarantee precision and inevitably suffer significant performance degradation when the inputs are very sparse, limiting their applicability in room reconstruction with commercial devices. Traditional methods like volumetric fusion [10], BundleFusion [12], KinectFusion [22] reconstruct 3D models from image sets based on geometric vision and graphics principles, but they are lack robustness of some complex scenes. Based on volumetric rendering, Nerf++ [41], MipNeRF 360 [4], Nerfstudio [32], and zip-NeRF [5] extend original NeRF [21] to real-scene applications. NICE-SLAM [45] and NICER-SLAM [44] combine NeRF with simultaneous localization and mapping (SLAM) method, enabling real-time dense RGB-D SLAM system that can be applied to large-scale scenes. NeuS [34], VolSDF [38], Neural RGB-D [3], GO-Surf [33], and BakedSDF [39] combine truncated signed distance function (TSDF) and volumetric rendering to minimize the geometry ambiguity. Many of these pipelines prioritize either geometric accuracy or synthesis enhancement. This specialization can limit their effectiveness in VR/AR applications, which demand both structural accuracy and realistic immersion. Therefore, we propose a new method that provides a trade-off between the reconstruction and rendering quality and further improves the performance by overcoming one of the challenges provided by our dataset.

3. The MuSHRoom Dataset

This section first presents the procedures for recording real-world indoor room data using the Kinect, iPhone, and Faro scanner. Then, we describe the post-processing steps we applied to the captured data before the evaluation. Lastly, we highlight the key challenges of the obtained dataset.

3.1. Data Collection

To create a diverse dataset, we selected rooms with varying shapes, colors, and indoor objects. We have chosen 10 real-world rooms while further details on the selected rooms can be found in the supplementary material. Prior to recording, we take measures to ensure that any personal privacy concerns are addressed and that the rooms do not reveal any confidential information. During the recording process, we ensured that objects within the rooms remained stationary to maintain consistency across devices and that the objects recorded by each device were in the same position.

3.1.1 Raw data capturing

In Figure 2, we briefly illustrate the data-capturing pipelines for the three devices. Comparisons of our dataset with others can be found in Table 1. Compared with other datasets,

Dataset	N_{room}	Device	RGB-D	N_{seq}	Resolution	Pose estimation method	Geometry ground truth format
Tanks&Temples [19]	4	FARO scanner X330 [15]; Sony A7SM2 camera		4	1920×1080	COLMAP [29]	point cloud
Redwood Scan [25]	5	FARO scanner X330; Asus Xtion Live camera	✓	5	640×480	color ICP [25]	point cloud
ETH3D [30]	7	FARO scanner X330; Nikon D3X DSLR camera; Global-shutter camera		9	high-res: 6048×4032; low-res: 752×480	COLMAP [29]	point cloud
Neural RGB-D synthesis dataset [2]	10	Synthetic camera	✓	10	640×480	BundleFusion [12]	Blender [7] mesh
Mip-NeRF 360 [4]	1	Fujifilm X100V camera		1	3114×2075	COLMAP [29]	-
Nerfstudio [32]	1	mobile phone		1	994×738	Polycam [27]	-
MuSHRoom	10	Faro scanner X130; Azure Kinect v2; iPhone 12 Pro Max	✓	40	Kinect: 1280×720 iPhone: 994×738	Kinect: Spectacular AI [1] & COLMAP [29] iPhone: Polycam [27] & COLMAP [29]	mesh

Table 1. Comparison between 3D reconstruction datasets. We only counted the number of indoor rooms from each datasets. MuSHRoom dataset provides the most indoor scenes captured by multiple sensors.

MuSHRoom provides the most indoor scenes captured with multiple RGB-D devices and has expert-cleaned reference meshes. All the raw color/depth images, in-painted depth, the estimated pose and point cloud extracted by Spectacular AI SDK [1] and Polycam [27], and the expert-cleaned reference mesh will be provided for further research. We use three devices to record each room. A faro scanner is used for high-precision point cloud collection for geometry comparison, and consumer device Azure Kinect and iPhone are used to collect RGB-D sequences.

Kinect. We use an Azure Kinect depth camera to get synchronized depth and color images at 30 Hz with a laptop. The depth images are captured with a resolution of 512x512, and color images at 1280x720 pixels. We use the wide FoV mode of the depth camera with 2x2 binning to increase the field of view for better room reconstruction. Inertial Measurement Unit (IMU) data was recorded at 1.6kHz. For color image capturing, we fixed the white balance for each room and the auto-exposure for 8 rooms except the sauna and olohuone room, which have large illumination variations inside the room. During capturing, to increase the possibility of capturing all the details of the room for the long capture, we use a visualization system developed by Spectacular AI SDK [1] to inspect the integrity of the reconstructed point cloud extracted from the captured RGB-D images in real-time.

When evaluating the novel view synthesis, most of the previous methods [4, 32] select keyframes from the sequences uniformly. However, this may not reflect the real case in AR/VR. It is common in real-world applications for users to first scan an entire room with a device and then wear AR glasses to interact with the environment from random positions and directions. Our goal is to simulate this scenario in order to create a more realistic evaluation method. We recorded two sequences inside each room. For the long capture, we try to include all the parts of the whole scene, and when capturing the short one, we attempt to follow a different motion trajectory.

iPhone. We use an iPhone 12 pro max to record iPhone data with the Polycam app [27]. During capture, a UI system provided by Polycam is also used to guarantee all the objects, ceiling, floor, and walls have been covered within one capture video. Auto-exposure and auto-white balance are used by default. To ensure the stability of the iPhone, we fix it on a DJI OSMO Mobile 3 handle [14]. Following the same pattern with the Kinect device, we collect the second sequence with the iPhone as a test dataset.

Faro scanner. To obtain the geometry ground truth reference mesh of each room, Faro Focus 3D X130 Laser Scanner [15] fixed on a tripod is used to collect a high-resolution XYZRGB point cloud. The reach of the laser ranges from 0.6m to 130m. We have selected the indoor capture mode, which has a range of more than 10 meters and a ranging noise of 0.15 millimeters. Each scan was set with 360° horizontal, 170° vertical (-60° to 90°) with 1/5 resolution, which takes around 9 minutes to record. The resolution of each scan is 8192x3414 pixels, with a maximum of 28 million points. We opt for the horizontal weighted metering mode for the camera, which utilizes the light from the horizontal direction to determine the optimal exposure setting. This mode is particularly well-suited for indoor rooms with bright ceiling lights. In order to capture a comprehensive view of the room's interior surface, we perform scanning from 4-5 positions for regular rooms and 7-10 positions for larger rooms. Each position was strategically selected to maximize the coverage of areas that have not been scanned.

3.1.2 Post-processing

Kinect. After acquiring the raw data, we used Spectacular AI SDK to extract the 6-degree-of-freedom (6DoF) pose in the OpenCV coordinate system [23]. Spectacular AI SDK fuses data from RGB-D cameras and IMU sensors and outputs a robust and accurate 6DoF pose for the keyframes

extracted from the whole sequence. It also exports a reconstructed point cloud registered with multi-view information. To address the issue of raw depth images containing multiple holes with invalid depth values, we utilize the z-buffer [16] to render depth images from the point cloud and then perform hole in-painting. We perform global registration by using COLMAP [29] to re-calculate the poses for all the images from the long and short capture with bundle adjustment, and then we re-scale and rotate the COLMAP pose to align with the original Spectacular AI pose. Pose and point cloud optimized by bundle adjustment are used as *estimated pose* and *estimated point cloud* for reconstruction.

iPhone. The raw images have a resolution of 1024×768 and a raw depth of 256×192 pixels. We use Polycom to extract poses for each keyframe with global optimization. Then, we use scripts in Nerfstudio [32] to pre-process the RGB-D images to get cropped color images as well as upscaled aligned depth images with a resolution of 994×738 pixels. To register long and short sequences, we also use COLMAP to re-calculate the COLMAP pose as *estimated pose* and align them to the Polycam pose coordination.

Faro Scanner. We register scans captured inside one room with FARO SCENE Software. To further reduce the size of the point cloud without excessive loss of accuracy, we down-sample 3x for each registered point cloud. When getting mesh from these high-resolution point clouds, most of the previous datasets [25, 30] utilize Poisson reconstruction [18], which is not suitable for our scenes with complex objects and high occlusion. To ensure the quality of the mesh, we use Reality Capture [28] to triangulate mesh from point clouds. However, there are still artifacts in the reference mesh due to occlusion and complex reflective surface, which are detrimental to the evaluation. These artifacts are again manually refined by removing wrong vertices and completing holes in MeshLab [9, 24] and Blender [7], and will not contribute to the final comparison.

When comparing different benchmarks, the reconstructed mesh needs to be aligned with the ground truth reference meshes. We use the estimated point cloud of each room and each device to register the reference mesh automatically. For other pipelines that can expose camera poses, we align the predicted result to the estimated point cloud. Similar to the alignment procedure proposed in Tanks and Temples [19], we first initialize a global scaling and alignment with RANSAC and then refine the registration with color ICP to get the final alignment $\mathbf{T}$.

3.2. Challenges of the MuSHRoom dataset

The MuSHRoom dataset introduces several practical challenges, including sparse occlusion, motion blur, reflection, transparent objects, and significant illumination variations, which are detrimental to the training of the reconstruction and rendering models. In Figure 3, we illustrate examples of the challenges observed in our dataset.

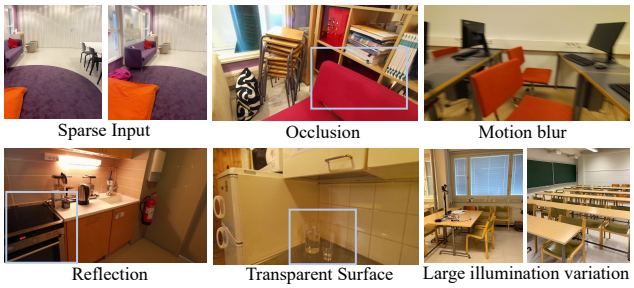

Figure 3. The challenges observed in the MuSHRoom dataset.

Sparseness To ensure the accuracy of the entire room reconstruction, we only optimize the poses for keyframes with specific view gaps. As a result, the extracted keyframes of each device and room are relatively sparse. This characteristic is not ideal for methods such as Neural Radiance Fields (NeRF), which benefit from dense images as input.

Occlusion The layout of objects within each room often includes narrow spaces, making it challenging to capture the backside of many objects. As a result, artifacts can occur during the reconstruction process, as the NeRF models are required to guess the appearance of unseen areas randomly.

Motion Blur Unsteady walking patterns and shaky hands can cause images to appear blurry, which will influence the training process.

Reflection Reflection usually occurs on metal surfaces, like the stove, TV, or mirror, where depth is hard to capture. The invalid depth is detrimental to the learning for both reconstruction and synthesis tasks.

Transparency Transparency is a difficult attribute to learn because of the wrong depth value. These regions are usually completely missing from the mesh model.

Large illumination variations Due to uneven light conditions inside one room, the illumination may vary significantly, making it hard to learn the illumination circumstances and synthesize images as close as possible to the real images.

Evaluation gap When training and testing models with different captures and trajectories, the directions and positions of the camera in the training and test set may have large pose differences, which stimulates the pipelines to be robust.

4. Benchmark

We propose a new benchmark for developing unified frameworks that focus on realizing both immersive and structurally accurate modeling under real-world constraints. These frameworks are optimized for consumer-grade hardware and operate in an end-to-end fashion. They take as input RGB-D sequences captured by consumer devices and output both accurate 3D mesh models and photorealistic im-

ages synthesized from novel views. We compare the methods for both the mesh reconstruction and novel view synthesis quality quantitatively and qualitatively.

4.1. Evaluating Geometric Reconstruction

Metrics. We evaluate the predicted mesh models with the reference mesh from both accuracy and completeness aspects. Following the mesh evaluation protocol introduced in GO-Surf [33], we measure accuracy (Acc), completion (Comp), Chamfer distance (C- ℓ_1), normal consistency (NC) and F-score metrics when evaluating reconstruction results. The comparison is conducted between the point cloud sampled from the predicted mesh and reference mesh at a density of 1 point per cm^2. The threshold for computing the F-score is 5 cm. More details can be seen in the Supplementary Materials.

Mesh Culling. In previous methods [2, 33], the mesh will be culled according to whether the surfaces have been observed, occluded, or have valid depth before evaluation. We follow this protocol for Kinect sequences, and we also cut the predicted mesh outside the silhouette of the reference mesh. Because in MuSHRoom some rooms are unbounded or non-square scenes, which cannot be simply culled by square-style bounding boxes and previous assumptions. We project the predicted mesh and reference mesh to the xy, yz, and xz planes separately and remove vertices and their corresponding triangles of the predicted mesh that out of the contours of the projection of the reference mesh. After getting a predicted mesh that is not influenced by the outside surface, we only compare the parts owned by reference mesh. For iPhone sequences, we only apply this cutting method, and the details about the culling strategy can be seen in the Supplementary Materials.

Training and evaluation strategy. To compare mesh quality, we utilize frames from the long capture of each device and room as inputs and compare the resulting mesh with the cleaned ground truth mesh.

4.2. Evaluating Novel View Synthesis

We test novel view synthesis with two comparison methods: testing within a single sequence and testing with a different sequence. When comparing with the testing within a single sequence method, we extract keyframes from one sequence as test data and training methods on the other frames of the same sequence. However, the uniform sampling method usually used in previous methods [4, 32] is not practical in VR/AR applications that require random trajectories. Thus, we propose a new evaluation protocol, testing with a different sequence, which uses one sequence for training and the other individual sequence for testing. The distances and directions of the camera from the same object will be significantly different in the two sequences, which poses a great challenge to the rendering robustness of the pipelines.

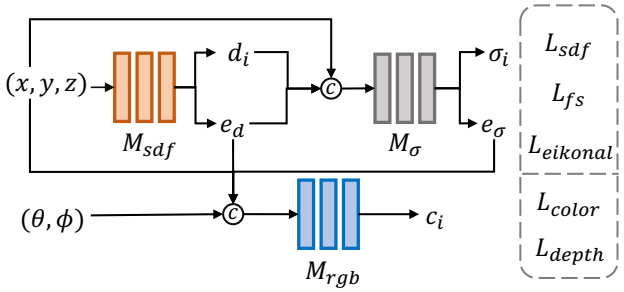

Figure 4. The visualization of the structure of our method. Data augmentation (DA) generates pseudo images/depth to enrich the training set. Our structure predicts SDF and density in an end-to-end fashion, which adds flexibility to the density and further composite background content. (x, y, z) and (θ, ϕ) is the position and direction of the sampling point along the camera ray. d_i, c_i, σ_i is the predicted signed distance, color, and density values of each sampling point correspondingly. e_d, e_σ is the corresponding signed distance and density embedding feature.

Metrics. We compare images synthesized from novel views with PSNR, SSIM [35], and LPIPS [43] evaluation metrics.

Training and evaluation strategy. When testing within a single sequence, we select 10% frames from the long sequence uniformly as the test set, and others are used as training datasets. When testing with a different sequence, we use the long capture for training and the short capture for testing based on the MuSHRoom dataset.

5. Proposed Method

We develop a new method for joint estimating 3D reconstruction and novel view synthesis. Currently, NeRF [21] utilizes volume rendering to achieve impressive fidelity for novel view synthesis, but its intense focus on photorealistic rendering tends to compromise geometric accuracy. On the other hand, pipelines based on SDF prediction can accurately capture surfaces. However, when directly synthesizing RGB images from these pipelines, density transformed from SDF will lose color fidelity and flexibility, leading to underfitting in appearance learning.

We first adopt a two-head structure to provide a trade-off between rendering quality and reconstruction quality. Inspired by ResNeRF [37], we reimplement Neus-facto [40] with a two-head structure that employs both volume density field for photorealistic rendering and SDF for geometry accuracy. In Neus-facto, SDF is predicted from the feature of sampling points and transformed to density for the background content synthesis [42]. Instead of getting density from SDF, we directly predict it from multilayer perceptron (MLP), which gives the density much more freedom and provides a trade-off between photorealism synthesis and structural accurate modeling.

Device	Methods	Reconstruction quality					Rendering quality					
							Test within a single sequence			Test with a different sequence		
		Acc ↓	**Comp**↓	**C-ℓ_1** ↓	**NC** ↑	**F-score** ↑	**PSNR** ↑	**SSIM** ↑	**LPIPS** ↓	**PSNR** ↑	**SSIM** ↑	**LPIPS** ↓
Kinect	Volumetric Fusion [10]	0.0354	0.0341	0.0347	0.8159	0.8439	13.84	0.6628	0.4208	13.10	0.6509	0.4423
	GO-Surf [33]	0.0355	0.0367	0.0361	0.8664	0.8620	20.56	0.7708	0.3095	20.01	0.7693	0.2812
	Nerfacto [32]	0.0570	0.1485	0.1027	0.6878	0.5715	22.08	0.7971	0.2479	22.60	0.8457	0.1822
	NeuS-facto [40]	0.0294	0.0253	0.0274	0.8738	0.9025	20.61	0.7799	0.2760	21.50	0.8285	0.2094
	Ours	0.0295	0.0258	0.0276	0.8736	0.8995	22.26	0.8040	0.2608	22.45	0.8423	0.2012
iPhone	Volumetric Fusion [10]	0.0521	0.0207	0.0364	0.7867	0.8050	11.74	0.5519	0.5126	11.80	0.5525	0.5094
	GO-Surf [33]	0.0630	0.0305	0.0468	0.8401	0.7818	17.50	0.6292	0.4620	17.64	0.6247	0.4832
	Nerfacto [32]	0.0592	0.1450	0.1021	0.6661	0.5973	20.53	0.7555	0.2696	20.72	0.7625	0.2670
	NeuS-facto [40]	0.0659	0.0453	0.0556	0.8135	0.7200	17.11	0.6696	0.4501	16.83	0.6504	0.4608
	Ours	0.0629	0.0448	0.0539	0.8231	0.7281	19.29	0.7130	0.3898	18.29	0.6819	0.3985

Table 2. The average metrics of reconstruction and rendering quality for all rooms. The best results are highlighted in pink. The second best results are marked in yellow. Test within a single sequence means we uniformly sample test frames from a single sequence and train on all left frames. Test with a different sequence means we train on one sequence and test on another individual sequence.

As shown in Figure 4, the camera ray of each pixel starting from the camera position $\mathbf{o}$ will travel in the direction of the camera's orientation $\mathbf{r}$. Neural networks M_{rgb}, M_σ sample N points $\mathbf{s}_i = \mathbf{o} + d_i\mathbf{r}, \mathbf{s}_i \in S_N = \{\mathbf{s}_1, ..., \mathbf{s}_N\}$ along each ray and predict color c_i and density σ_i. To get the density, instead of directly transforming from predicted signed distance d_i generated from M_{sdf}, we employ another density prediction MLP M_σ, accepts concatenated d_i, e_d and (x, y, z) to get the volume density σ_i and density embedding e_σ. (θ, ϕ), the direction of the sampling point, concatenated with (x, y, z), e_d, e_σ are imported to M_{rgb} to predict color c_i. We utilize the background model [42] for compositing appearance, σ_i is used to accumulate background color, and the final color $\hat{\mathbf{C}}$ is a combination of accumulated c_i and background color c_{bg}. To control the training process, we apply SDF loss L_{sdf}, free space regulation loss L_{fs} [3], and eikonal loss $L_{eikonal}$ [33] for predicted SDF.

$$\ell_{sdf} = \frac{1}{|S_{tri}|} \sum_{x \in S_{tri}} (d(\mathbf{s}) - b(\mathbf{s}))^2 \tag{1}$$

where b is the observed signed distance, which is truncated by a distance $t = 5cm$ from the captured depth. $S_{tri} = |\mathbf{D} - d_i| <= t$ is the set of sampling points between the front and back truncation surfaces.

$$\ell_{fs} = \frac{1}{|S_{fs}|} \sum_{x \in S_{fs}} (d(\mathbf{s}) - t)^2 \tag{2}$$

$S_{fs} = |\mathbf{D} - d_i| > t$ is the set of sampling points that are distributed between the ray start position and truncation surface.

$$\ell_{eikonal} = \frac{1}{|S_N|} \sum_{x \in S_N} (1 - \|\nabla d(\mathbf{s})\|)^2 \tag{3}$$

where ∇d is the gradient of d. RGB loss L_{color} and depth loss L_{depth} are used for color and depth regulation during training.

$$\ell_{color} = |\mathbf{C} - \hat{\mathbf{C}}|, \quad \ell_d = |\mathbf{D} - \hat{\mathbf{D}}| \tag{4}$$

where $\mathbf{C}/\hat{\mathbf{C}}$ and $\mathbf{D}/\hat{\mathbf{D}}$ are the real/predicted color and depth of each pixel.

We further apply data augmentation to solve one of the challenges released by our dataset, sparseness, to further boost the performance of our baseline. The large content gap between adjacent frames can destabilize the training process. This often results in the model initially learning from diverse directions and becoming unstable. To control the learning process, we interplate n new poses along the trajectory between every two successive frames, rendering corresponding images and depth images from trained models that can let us get the best synthesis and depth effect in the experiments to further enrich the training dataset[1]. These pseudo-RGB-D images work as a regulation, accelerating the convergence and limiting the inaccuracies caused by sparse data in the training process.

6. Experiments

We compare our baseline with several representative pipelines for both reconstruction and rendering quality, and the detailed instructions for these methods can be found in the Supplementary Materials.

6.1. Quantitative Evaluation

We calculate the average metrics of reconstruction and rendering quality for all rooms and show them in Table 2. Nerfacto [32] is excellent in rendering quality but is much worse than others in terms of mesh completeness. GO-Surf [33] and NeuS-facto [40] predict SDF, which regulates the mesh without ambiguity, but their synthesis qualities are worse than NeRF. Nevertheless, our method provides a good trade-off for the reconstruction and rendering quality. The results also highlight that the inherent complexities of our dataset impede the enhancement of rendering fidelity,

[1]To render pseudo-RGB images, we use Nerfacto for both Kinect and iPhone. To render pseudo-depth images, we use NeuS-facto to render depth for Kinect sequences and GO-Surf to render depth for iPhone sequences

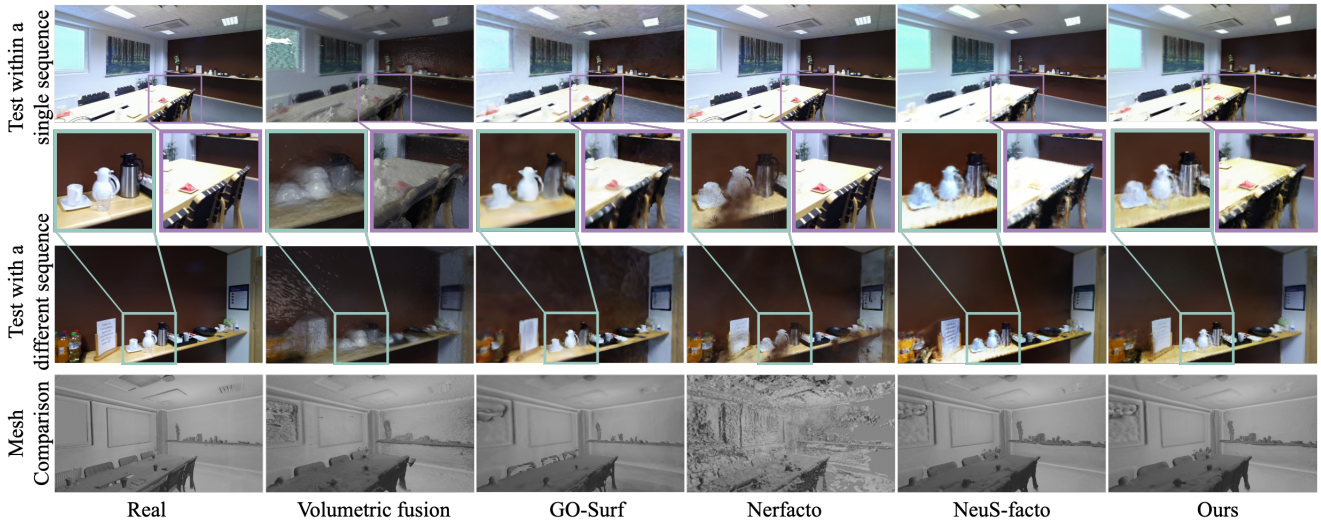

Figure 5. The qualitative comparison on Kinect sequences. Test within a single sequence means we uniform sample test frames from a single sequence and train on all left frames. Test with a different sequence means we train on one sequence and test on another individual sequence. We also visualize the mesh qualitatively. Please zoom in to see the details.

which requires further advanced methods developed for real scenarios.

6.2. Qualitative Evaluation

We show the qualitative results of testing within one sequence and testing with different sequences, mesh quality of the Kinect sequences in Figure 5. The images produced by Volumetric Fusion [10] have a large domain gap compared to real images, as they are directly synthesized from mesh. Nerfacto provides the most details and fine-grained images, but they are not very robust when views change dramatically, as seen in the third row of Figure 5. NeuS-facto and our method are relatively much more robust, but they still lack details. More visualization of Kinect and iPhone sequences can be found in the Supplementary Materials.

6.3. Ablation Study

We provide the ablation study of the two-head structure and data augmentation techniques of some rooms in our dataset by measuring the rendering quality. The two-head structure contributes to the overall improvement, and the data augmentation method further regulates the training process. Data augmentation can work more efficiently for large rooms that have more obvious sparse frames, like the activity room. However, directly applying this method is not very solid. When the sequence is not so sparse, pseudo images/depth will actually hinder the training process. This technique requires further research.

7. Conclusion

We have proposed a real-world dataset and a new benchmark with multiple sensors for evaluating pipelines on both 3D reconstruction accuracy and novel view synthesis quality. The new dataset poses more realistic challenges and supports more practical evaluation. With consumer-grade devices to collect inputs, pipelines are encouraged to be robust, generalized, and computationally efficient. We also propose a new method and evaluate it with several popular pipelines, revealing the aim to realize both geometry accuracy and immersion still has a long way to go. Our dataset can serve as a foundation for the development of a unified framework training in an end-to-end fashion.

Device	Room	Methods	Test within a single sequence			Test with a different sequence		
			PSNR ↑	**SSIM ↑**	**LPIPS ↓**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**
Kinect	activity	NeuS-facto	19.53	0.7258	0.3356	20.18	0.7707	0.2750
		w/o DA	20.30	0.7379	0.3222	21.06	0.7864	**0.2531**
		Ours	**20.89**	**0.7509**	**0.3221**	**21.33**	**0.7919**	0.2534
	honka	NeuS-facto	19.40	0.7759	0.2658	22.45	0.8505	0.1616
		w/o DA	20.70	0.7962	**0.2440**	**24.04**	0.8671	**0.1401**
		Ours	**20.86**	**0.8000**	0.2584	23.59	**0.8683**	0.1527
iPhone	computer	NeuS-facto	16.59	0.6636	0.4197	15.83	0.6327	0.4321
		w/o DA	19.45	0.7248	0.3661	17.66	0.6769	0.3835
		Ours	**20.10**	**0.7411**	**0.3530**	**18.10**	**0.6894**	**0.3649**
	sauna	NeuS-facto	17.37	0.6791	0.5354	16.78	0.6585	0.5330
		w/o DA	19.31	**0.7043**	0.4716	**18.20**	0.6742	**0.4760**
		Ours	**19.82**	0.7126	**0.4700**	18.61	**0.6805**	0.4778

Table 3. The ablation study of two-head structure and data augmentation (DA) in rendering quality. Two-head structure can improve the overall performance. DA is much more effective for larger rooms.

8. Acknowledgement

The work was supported by the Academy of Finland projects #324346 and #353139, and also carried out with the support of Centre for Immersive Visual Technologies (CIVIT) research infrastructure, Tampere University, Finland.

References

[1] Spectacular ai sdk. https://www.spectacularai.com, 2021. 4

[2] Dejan Azinovic, Ricardo Martin-Brualla, Dan B Goldman, Matthias Niebner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, May 2022. 2, 4, 6

[3] Dejan Azinovic, Ricardo Martin-Brualla, Dan B Goldman, Matthias Niebner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, May 2022. 3, 7

[4] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, pages 5470–5479, 2022. 2, 3, 4, 6

[5] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-nerf: Anti-aliased grid-based neural radiance fields. *arXiv preprint arXiv:2304.06706*, 2023. 3

[6] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Gebauer Thomas, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. Arkitscenes – a diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. *NeurIPS*, Jun 2021. 3

[7] Blender. Blender. https://www.blender.org. 4, 5

[8] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *2017 International Conference on 3D Vision (3DV)*, Sep 2017. 3

[9] Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. MeshLab: an Open-Source Mesh Processing Tool. In Vittorio Scarano, Rosario De Chiara, and Ugo Erra, editors, *Eurographics Italian Chapter Conference*. The Eurographics Association, 2008. 5

[10] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996. 3, 7, 8

[11] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, Jun 2017. 2

[12] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface re-integration. *arXiv: Graphics,arXiv: Graphics*, Apr 2016. 2, 3, 4

[13] Nianchen Deng, Zhenyi He, Jiannan Ye, Budmonde Duinkharjav, Praneeth Chakravarthula, Xubo Yang, and Qi Sun. Fov-nerf: Foveated neural radiance fields for virtual reality. *IEEE TVCG*, 28(11):3854–3864, 2022. 2

[14] DJI. Dji osmo mobile 3 handle. https://www.dji.com/fi/osmo-mobile-3. 4

[15] faro. Faro focus laser scanner. https://www.faro.com/en/Products/Hardware/Focus-Laser-Scanners. 4

[16] Xiaoyang Huang, Yi Zhang, Bingbing Ni, Teng Li, Kai Chen, and Wenjun Zhang. Boosting point clouds rendering via radiance mapping. In *AAAI*, volume 37, pages 953–961, 2023. 5

[17] Yi-Hua Huang, Yue He, Yu-Jie Yuan, Yu-Kun Lai, and Lin Gao. Stylizednerf: consistent 3d scene stylization as stylized nerf via 2d-3d mutual learning. In *CVPR*, pages 18342–18352, 2022. 2

[18] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM TOG*, 32(3):1–13, 2013. 5

[19] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM TOG*, 36(4):1–13, 2017. 4, 5

[20] Andreas Meuleman, Yu-Lun Liu, Chen Gao, Jia-Bin Huang, Changil Kim, MinH. Kim, and Johannes Kopf. Progressively optimized local radiance fields for robust view synthesis. Mar 2023. 3

[21] 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. *Communications of the ACM*, 65(1):99–106, 2021. 3, 6

[22] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, 2011. 3

[23] opencv. Opencv: Open source computer vision library. https://github.com/opencv/opencv. 4

[24] Cignoni Paolo, Muntoni Alessandro, Ranzuglia Guido, and Callieri Marco. Meshlab. 10.5281/zenodo.5114037. 5

[25] Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Colored point cloud registration revisited. In *ICCV*, pages 143–152, 2017. 2, 4, 5

[26] Pixel4D. Pixel4D. https://www.pix4d.com. 3

[27] Polycam. Polycam. https://poly.cam. 4

[28] Capturing reality. Reality Capture. https://www.capturingreality.com. 3, 5

[29] Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, May 2016. 3, 4, 5

[30] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, pages 3260–3269, 2017. 2, 4, 5

[31] Julian Straub, ThomasA. Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob Engel, Raul Mur-Artal, CarlYuheng Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, BrianChristopher Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, NigelP. Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke Strasdat, RenzoDe Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The replica dataset: A digital replica of indoor spaces. *Cornell University - arXiv,Cornell University - arXiv*, Jun 2019. 3

[32] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, et al. Nerfstudio: A modular framework for neural radiance field development. *arXiv preprint arXiv:2302.04264*, 2023. 2, 3, 4, 5, 6, 7

[33] Jingwen Wang, Tymoteusz Bleja, and Lourdes Agapito. Go-surf: Neural feature grid optimization for fast, high-fidelity rgb-d surface reconstruction. In *2022 International Conference on 3D Vision (3DV)*, pages 433–442. IEEE, 2022. 2, 3, 6, 7

[34] 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. *NeurIPS*, 2021. 3

[35] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 6

[36] Jianxiong Xiao, Andrew Owens, and Antonio Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *ICCV*, Nov 2013. 3

[37] Yuting Xiao, Yiqun Zhao, Yanyu Xu, and Shenghua Gao. Resnerf: Geometry-guided residual neural radiance field for indoor scene novel view synthesis. *arXiv preprint arXiv:2211.16211*, 2022. 6

[38] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *NeurIPS*, Dec 2021. 3

[39] Lior Yariv, Peter Hedman, Christian Reiser, Dor Verbin, PratulP. Srinivasan, Richard Szeliski, JonathanT. Barron, and Ben Mildenhall. Bakedsdf: Meshing neural sdfs for real-time view synthesis. Feb 2023. 3

[40] Zehao Yu, Anpei Chen, Bozidar Antic, Songyou Peng Peng, Apratim Bhattacharyya, Michael Niemeyer, Siyu Tang, Torsten Sattler, and Andreas Geiger. Sdfstudio: A unified framework for surface reconstruction, 2022. 6, 7

[41] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv: Computer Vision and Pattern Recognition,arXiv: Computer Vision and Pattern Recognition*, Oct 2020. 3

[42] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv: Computer Vision and Pattern Recognition,arXiv: Computer Vision and Pattern Recognition*, Oct 2020. 6, 7

[43] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[44] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, MartinR. Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. Feb 2023. 3

[45] 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 *CVPR*, May 2022. 3