

BNV-Fusion: Dense 3D Reconstruction using Bi-level Neural Volume Fusion

Kejie Li¹, Yansong Tang^{1,2}, Victor Adrian Prisacariu¹, Philip H.S. Torr¹
¹University of Oxford, ²Tsinghua-Berkeley Shenzhen Institute, Tsinghua University

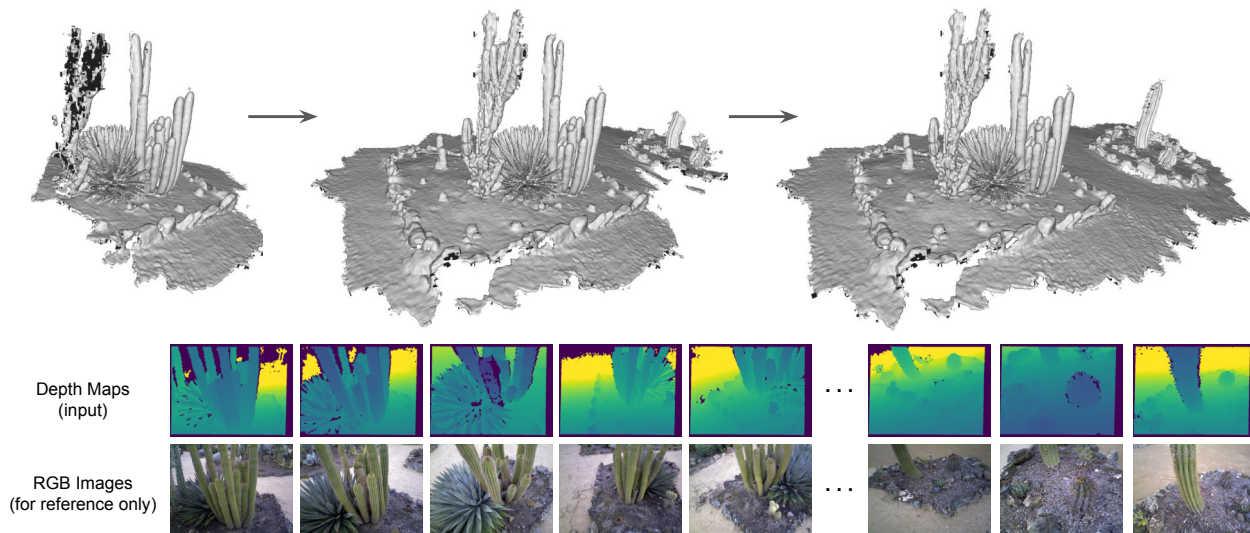


Figure 1. Bi-level Neural Volume Fusion (BNV-Fusion) incrementally integrates noisy depth images to a global model of geometry.

Abstract

Dense 3D reconstruction from a stream of depth images is the key to many mixed reality and robotic applications. Although methods based on Truncated Signed Distance Function (TSDF) Fusion have advanced the field over the years, the TSDF volume representation is confronted with striking a balance between the robustness to noisy measurements and maintaining the level of detail. We present Bi-level Neural Volume Fusion (BNV-Fusion), which leverages recent advances in neural implicit representations and neural rendering for dense 3D reconstruction. In order to incrementally integrate new depth maps into a global neural implicit representation, we propose a novel bi-level fusion strategy that considers both efficiency and reconstruction quality by design. We evaluate the proposed method on multiple datasets quantitatively and qualitatively, demonstrating a significant improvement over existing methods.

1. Introduction

Dense 3D reconstruction from images is one of the most long-standing tasks in the computer vision community. While there is a large body of research focusing on

reconstruction using RGB-only images [13, 15], the increasing popularity of depth sensors in commodity devices (e.g. Microsoft Kinect [45], Apple LiDAR scanner [1]) has enabled researchers to develop reconstruction algorithms taking advantage of depth maps [9, 26, 33].

However, the representation used in these methods – Truncated Signed Distance Function (TSDF) Volume – is known to lose fine details at sub-voxel scale (e.g. thin surfaces) [3, 41] because it discretizes the scene geometry at a pre-defined resolution. In addition to the limitation of the representation, each depth measurement is integrated into the volume independently using voxel-wise weighted averaging without any local context, which makes the fusion process vulnerable to noisy depth measurements.

In contrast, emerging neural implicit representations, which show promising results in novel view synthesis [25], and shape modelling [24, 29, 31], have the potential of being a better alternative to TSDF volume-based reconstruction in an online setting. In essence, these representations are deep neural networks that map continuous 3D coordinates to a task-dependent scene property, such as the color or the distance to the nearest surface. As a result, a surface can be extracted at any resolution given the implicit function represented by the network, without any increase in memory

usage. Another advantage of neural implicit representations is that the network can be trained as a generative model to capture prior knowledge of a family of surfaces. These appealing aspects of the neural implicit representations have motivated recent works [2, 3, 17] to develop surface reconstruction methods in an offline setting. Nonetheless, while TSDF volume-based methods [26, 33] have demonstrated voxel-wise weighted averaging is real-time capable, how to incrementally integrate new depth measurements using neural implicit representations is still an open question.

Inspired by traditional volumetric fusion approaches, we present Bi-level Neural Volume Fusion (BNV-Fusion) for *high-quality* and *online* 3D reconstruction in this paper. Given a sequence of depth maps and the associated poses, BNV-Fusion incrementally integrates depth measurements into a global neural volume. The novelty of BNV-Fusion is the combination of a *local*-level fusion and a *global*-level fusion. At the local level, a new depth map is first mapped to latent codes, each representing local geometry in the latent space. They are then fused into the global neural volume by weighted averaging, which resembles the efficient update in traditional volumetric fusion methods. However, the local-level fusion is susceptible to depth outliers as it only integrates the measured surfaces and their surroundings to the volume. Furthermore, although several works on shape modelling [12, 43] suggest that arithmetic operations in the latent space correspond to the actual geometry change to some degree, updating the global representation in the latent space using an additive scheme does not always lead to correct geometry. To this end, we propose to optimize the global volume using neural rendering, where we penalize the discrepancies between the SDFs extracted from the global volume and those of depth measurements. This optimization is coined as *global*-level fusion as it encourages a coherent reconstruction globally.

Overall, the key realization in BNV-Fusion is that the local- and global-level fusions are complementary. While the local-level fusion efficiently integrates new information and initializes the global-level fusion, the reconstruction quality is improved significantly by the global-level fusion. To summarize, our contributions are threefold:

- We propose BNV-Fusion, a novel and state-of-the-art dense 3D reconstruction pipeline that represents the geometry of a scene by an implicit neural volume.
- We design a novel bi-level fusion algorithm that efficiently and effectively updates the neural volume given new depth measurements.
- We conduct extensive experiments, including an evaluation on 312 sequences of various indoor environments in ScanNet [8], to validate that BNV-Fusion improves existing approaches significantly and is truly generalizable to arbitrary scenes.

2. Related Work

Neural implicit representations can be categorized into global and local representations. In Sec. 2.1, we provide an overview of both categories with a focus on the latter one as the neural implicit volume in this work is inspired by methods in this category. We then, in Sec. 2.2, introduce previous works in dense 3D reconstruction and describe how our method is different from existing approaches that also leverage neural implicit representations.

2.1. Neural Implicit Representations

Global representations. DeepSDF [29], Occupancy Networks [24], and IM-Net [5] are pioneering works in neural implicit representations for object shapes. Follow-up works [27, 37] remove the requirement of 3D ground-truth supervision. They train their network by minimizing the discrepancies between input images and color (and depth) images rendered from the implicit representation. Mildenhall *et al.* [25], a seminal work in novel view synthesis using a neural implicit representation, takes a step further by representing the geometry and appearance of a scene as a Neural Radiance Field (NeRF). It learns to map 3D coordinates and viewing directions to occupancy and RGB values. The idea of using neural rendering as a supervision signal to learn a neural implicit representation has inspired subsequent works in 3D reconstruction [2, 40], including ours.

Local representations. Recent advances on implicit representations suggest that using an MLP to represent the geometry of a scene or an object is not scalable, and the prior knowledge at the object level is not generalizable [3, 17]. Therefore, they propose to learn neural implicit representation for local geometry structures, which is easily generalizable to objects of novel categories. The geometry of an entire scene can be decomposed into a grid of local latent codes, each of which represents the geometry in the local region. Similarly, Genova *et al.* [11] propose a network to predict both a set of local implicit functions and their 3D locations when given a set of depth maps for object reconstruction. In the application of novel view synthesis, Liu *et al.* [21] also show that decomposing a scene into a set of local latent codes, arranged in an Octree volume, improves rendering quality and speed. Our neural implicit volume is inspired by the work of Jiang *et al.* [17] and Chabra *et al.* [3]. While they are designed to map a complete point cloud of a scene to a set of latent codes, we can update the volume when given new information. More importantly, their frameworks do not handle outliers explicitly as they try to fit all observed surfaces to local latent codes. We instead filter out outliers in the proposed global-level fusion.

2.2. Dense 3D Reconstruction

Traditional approaches. The seminal work by Curless and Levoy [7] presents the idea of “TSDF Fusion”, which fuses

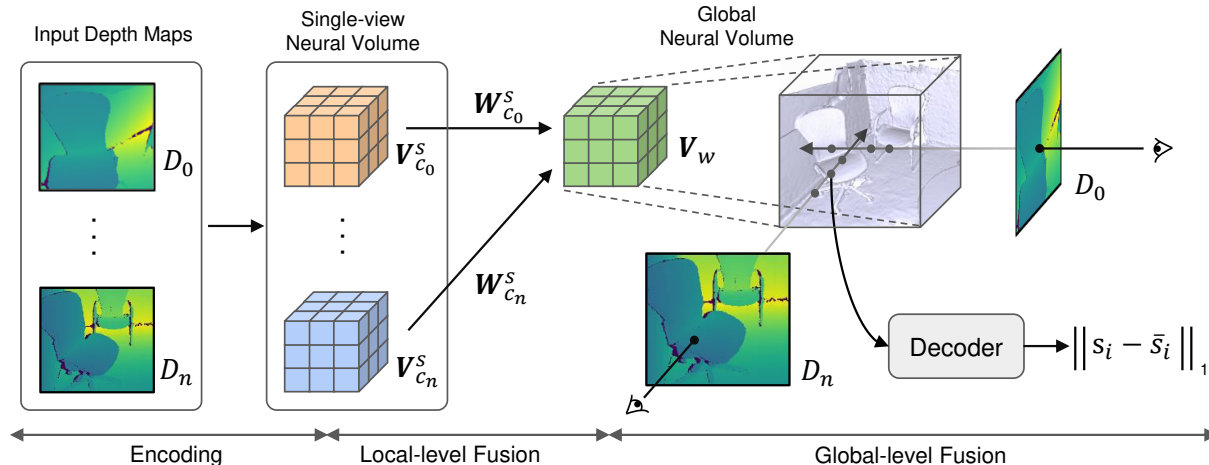


Figure 2. The architecture of BNV-Fusion. At frame t , BNV-Fusion first maps a depth map D_t to a single-view neural volume $V_{c_t}^s$, where each voxel contains a latent code that represents the local geometry in a latent space (Sec. 3.2). BNV-Fusion then integrates the single-view neural volume into the global neural volume using the proposed bi-level fusion (Sec. 3.3). At the *local* level, $V_{c_t}^s$ is integrated into the global volume V_w using running average weighted by $W_{c_t}^s$. At the *global* level, we iteratively optimize the global volume by minimizing the discrepancies between the SDFs decoded from the volume s_i and the actual measurements \bar{s}_i along a camera ray.

depth maps into a TSDF volume by averaging. KinectFusion [26] revisits and extends this idea to develop a real-time dense SLAM system with commodity-level depth cameras, such as Microsoft Kinect [45]. Subsequent works [9, 10, 18, 23, 28, 33, 35, 38] improve scalability, reconstruction quality, loop closure and various aspects in the fusion pipeline. Apart from the methods based on TSDF volume, several works [32, 36, 39, 42, 44] resort to surface-based representations that only model the surface of geometry for map compression. These classic approaches have promoted the development of 3D reconstruction in the past decades.

Reconstruction with neural implicit representations.

Azinović *et al.* [2] use an MLP to represent the geometry of a scene and train the MLP by comparing the rendered and input RGBD images. Sucar *et al.* propose iMAP [40], a dense SLAM system using a single MLP as the only representation for both mapping and tracking. Despite shrinking the training time from days as in Azinović *et al.* [2] to near real-time performance, the limitations of iMAP are as follows. First, the reconstruction does not scale well to the size of a scene and tends to lose details. This is because, although the MLP converges quickly to low-frequency shapes, it takes a much longer time to attend to high-frequency details, as noted by the authors of iMAP [40]. Second, they train a new MLP for each scene, thereby being inefficient and prone to noise in depth measurements. The proposed method circumvents these issues by using a volume of latent codes that encode local geometry in a shape embedding. The volume-based representation can improve the level of detail in reconstruction because the MLP that is conditioned on a latent code only needs to learn local surface patterns rather than the geometry of a

scene. Moreover, by only optimizing the latent codes while freezing the MLP’s parameters in the global-level fusion, we effectively leverage prior knowledge of local geometry embedded in MLP.

More closely related to our work as to representation are NeuralFusion [41] and DI-Fusion [16], both of which rely on a grid of latent codes. However, our work differs significantly from these methods in updating the latent codes given new measurements. Instead of only integrating new measurements in the domain of latent codes, we achieve a more globally consistent reconstruction by also optimizing the latent codes via neural rendering.

To summarize, although there are pioneering works [16, 40, 41] that try to apply neural implicit representations to reconstruction in an online setting, they improve efficiency at a great cost of reconstruction quality. Instead, the proposed method can reconstruct fine details that even traditional volumetric fusion approaches tend to miss while running in near real-time (~ 2 Hz without proper code optimization).

3. Method

Given a sequence of depth maps $\{D_0, \dots, D_n\}$ and the associated extrinsic parameters $\{\mathbf{T}_{c_0}^w, \dots, \mathbf{T}_{c_n}^w\}$ ($\mathbf{T}_{c_n}^w$ denotes a rigid transformation from camera c_n to the world coordinate), BNV-Fusion aims to reconstruct the geometry of a scene represented by a global implicit neural volume V_w (defined in Sec. 3.1). BNV-Fusion processes each depth map in three main steps, as outlined in Fig. 2. The encoding step converts a depth map into a single-view neural volume that comprises a set of latent codes in an embedding of local shapes (Sec. 3.2). The single-view neural volume is in-

tegrated into the global volume using the proposed bi-level fusion (Sec. 3.3).

3.1. Implicit Neural Volume

An implicit neural volume contains a set of local latent codes spatially organized in a grid structure. Specifically, the implicit neural volume takes the following form: $\mathbf{V} = \{v = (\mathbf{p}, \mathbf{l})\}$, where each voxel v contains its 3D position in the space \mathbf{p} , and a latent code \mathbf{l} that implicitly represents the local geometry in an embedding of local shapes. To recover the Signed Distance Function (SDF) value of a 3D point \mathbf{x} from the volume, we first retrieve 8 neighboring voxels $v_{0\dots7}$ in the volume and transform \mathbf{x} to the local coordinate with respect to each neighboring voxel: $\bar{\mathbf{x}}_i = \mathbf{x} - \mathbf{p}_i$. Given the latent code and the local coordinate, a shape decoder $\mathcal{D}(\cdot, \cdot)$ predicts the SDF value as follows.

$$s = \sum_{i=0}^{N=7} w(\bar{\mathbf{x}}_i, \mathbf{p}_i) \mathcal{D}(\mathbf{l}_i, \bar{\mathbf{x}}_i), \quad (1)$$

where $w(\cdot, \cdot)$ is the weight of trilinear interpolation. A mesh depicting the geometry of a scene can be extracted using the Marching Cubes algorithm [22] given the SDFs decoded from the volume.

3.2. Learning the Local Shape Embedding

The key of the implicit neural volume to represent geometry effectively is a data-driven embedding of local shapes, which we learn using an AutoEncoder-like network. Technically, a depth encoder \mathcal{E} , which is modified from PointNet [34], takes as input a point cloud unprojected from a depth map and extracts deep features of each point by aggregating information within a local region. These features are then mapped to a set of latent codes in the embedding. The decoder \mathcal{D} is a Multilayer Perceptron (MLP) with 4 fully connected layers. It takes a latent code and a 3D coordinate as input and predicts the input coordinate’s SDF value.

Training. We train the encoder and decoder jointly in a supervised manner using object CAD models in ShapeNet [4]. The loss function given a pair of training sample (\mathbf{p}, \mathbf{x}) is

$$L(\theta_{\mathcal{E}}, \theta_{\mathcal{D}}) = \|\mathcal{D}_{\theta_{\mathcal{D}}}(\mathcal{E}_{\theta_{\mathcal{E}}}(\mathbf{p}), \mathbf{x}) - s_x^{gt}\|_1, \quad (2)$$

where \mathbf{p} is a local point cloud with normals of a local surface patch, and \mathbf{x} is a sampled point around the surface with its ground-truth SDF value s_x^{gt} . We further detail the training process in Sec. 3.4.

Inference. At frame t , the depth map D_t is first unprojected to a 3D point cloud using the known intrinsic parameters. The point cloud is then segmented into overlapping local point clouds $\{\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n\}$, each of which is taken

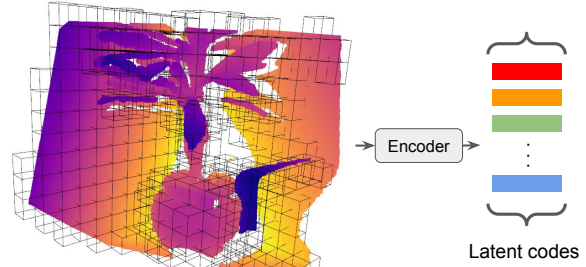


Figure 3. Encoding a depth map to the embedding of local shapes. A 3D point cloud, unprojected from a depth map, is segmented into local point clouds (bounded by 3D cuboids). The depth encoder takes as input a local point cloud and predicts a latent code.

by the encoder to map to a latent code $\mathbf{l} = \mathcal{E}(\mathbf{p})$, as shown in Fig. 3. The latent codes are aggregated into a single-view implicit volume $\mathbf{V}_{c_t}^s$ that represents D_t in the domain of latent space. The implicit volume is accompanied by a weight volume $\mathbf{W}_{c_n}^s$, where the values are set to the number of 3D points associated with a voxel.

3.3. Bi-level Fusion

Given a single-view neural volume, the global volume is updated sequentially by running the local- and global-level fusions. The latent codes in the single-view neural volume are first integrated into the global volume by weighted averaging at the local level. After the local update, the global volume is optimized via neural rendering to ensure a globally consistent reconstruction. We detail the bi-level fusion in the rest of this section.

Local-level fusion. At frame t , the single-view neural volume $\mathbf{V}_{c_t}^s$ is transformed to the world coordinate using the camera extrinsic parameters: $\mathbf{V}_w^s = \mathbf{T}_{c_t}^w \mathbf{V}_{c_t}^s$. The computation flow in the local-level fusion is similar to that of the traditional volumetric Fusion [7] except we are averaging latent codes rather than TSDF values, as shown below:

$$\mathbf{V}_w^t = \frac{\mathbf{W}^{t-1} \mathbf{V}_w^{t-1} + \mathbf{W}_{c_t}^s \mathbf{V}_w^s}{\mathbf{W}^{t-1} + \mathbf{W}_{c_t}^s}, \quad (3)$$

$$\mathbf{W}^t = \mathbf{W}^{t-1} + \mathbf{W}_{c_t}^s, \quad (4)$$

where \mathbf{V}_w and \mathbf{W} denote the global implicit volume and its weight respectively. The superscript $t-1$ and t means prior update and post update.

Global-level fusion. Although the local update is efficient, it is susceptible to outliers in measurements. Moreover, arithmetic operations in the latent space do not align perfectly with the actual geometry changes. To resolve these problems, we enforce a global consistency of geometry by rendering SDF values from the global volume and compare them with depth measurements. The discrepancies between the rendered values and the actual observations are then

used as supervision signals to optimize the global volume iteratively. At each iteration, we randomly sample a set of pixels in a depth map. Along the ray unprojected from each sampled pixel, we sample N 3D points $\{x_0, x_1, \dots, x_n\}$ using a hierarchical sampling strategy inspired by Mildenhall *et al.* [25] where we sample more 3D points that are closer to the measured surface. The projective TSDF of each 3D point given a depth measurement p on a ray is computed as: $\bar{s}_i = \min(\max(\|p - x_i\|_2, -\delta), \delta)$, where δ is the truncation threshold. After transforming the 3D points to the world coordinate, the SDF values of these points can be extracted using Eq. (1). Lastly, the optimization objective becomes:

$$\arg \min_{V_w} \sum_{i=0}^n \|s_i - \bar{s}_i\|_1, \quad (5)$$

where s_i is the SDF value decoded from the neural volume V_w using Eq. (1).

3.4. Implementation Details

Implicit neural volume. The voxel resolution is 2 cm (*i.e.* a latent code represents the geometry of a 2cm^3 volume around its position), and the dimension of the latent codes is set to 8.

Network training. We train the depth encoder and shape decoder using object CAD models from two categories (chairs and lamps) in ShapeNet [4]. To generate the input point clouds for the encoder, we first render 20 depth maps from random viewpoints for each CAD model. A depth map is unprojected to a 3D point cloud. We then randomly select 2000 seed points in each point cloud. A local point cloud is created by retrieving neighboring points within a local region centered at each seed point. We apply a Gaussian noise on the surface points and perturb their normal directions to simulate noise in depth measurements. We also randomly sample 1000 training points and calculate their SDF in the local region in order to train the decoder. The encoder contains 4 fully connected layers with sizes [128, 128, 128, 8], which are interconnected by ReLU except for the last layer. We aggregate information within a local region using an average pooling layer. The decoder is an MLP with 4 fully connected layers with sizes [128, 128, 128, 1].

Global-level fusion. We sample 5000 pixels per image in each iteration. For each camera ray unprojected from a sampled pixel, we sample 5 3D points per meter at the coarse-level sampling, and 20 3D points at the fine-level sampling. Latent codes in the global neural volume are optimized by the Adam optimizer [20] in Pytorch [30] for 5 iterations for each image.

4. Experiments

4.1. Datasets and Metrics

We evaluate the proposed method extensively on three datasets: 3D Scene Dataset [6], ICL-NUIM RGBD benchmark [14] (under the Creative Commons 3.0 license), and ScanNet [8] (under the MIT license). These datasets cover both synthetic scans with ground-truth 3D models, and real-world scans with pseudo ground-truth 3D models. To evaluate reconstruction quality, we uniformly sample 100,000 points from the ground-truth meshes and reconstructed meshes, respectively, then report the following metrics. *Accuracy* (denoted as Accu. in tables) measures the fraction of points from the reconstructed mesh that are closer to points from the ground-truth mesh than a threshold distance, which is set to 2.5 cm. *Completeness* (denoted as Comp. in tables) calculates the fraction of points from the ground-truth mesh that are closer to points from the reconstructed mesh than 2.5 cm. *F1 score* (denoted as F1 in tables) is the harmonic mean of accuracy and completeness, which quantifies the overall reconstruction quality. When reporting per-sequence quantitative results on ICL-NUIM [14] and the 3D Scene dataset [6], we run experiments 5 times using different frames and sample surface points twice independently to ensure the quantitative results are statistically significant.

4.2. Baselines

In the following experiments, we demonstrate the effectiveness of BNV-Fusion by comparing against three strong baseline methods that use traditional TSDF fusion or build upon modern neural implicit representations.

“TSDF Fusion” in this section denotes an implementation of TSDF Fusion in the Open3D library [46] based on KinectFusion [26]. “DI-Fusion” [16] is a reconstruction pipeline using a volumetric neural implicit representation. We use the code published by the authors in our experiments. “iMAP” is our reimplementation of the paper iMAP from Sucar *et al.* [40] because we do not have access to the official implementation at the time of submission. Running our system on their dataset is not preferable since they use synthetic depth maps without noise in evaluation, which have a domain gap to real-world scans. To isolate the mapping components in all compared methods, we use camera poses provided by the datasets and take the same images as input in the evaluation.

4.3. Evaluation on 3D Scene Dataset

The 3D Scene dataset, which comprises several real-world RGBD sequences, is a popular benchmark in the reconstruction community. Unlike synthetic datasets (*e.g.* the ICL-NUIM dataset), where ground-truth 3D models are available, in order to provide a quantitative comparison, we follow a common practice in prior art [3, 41]. Specifically,

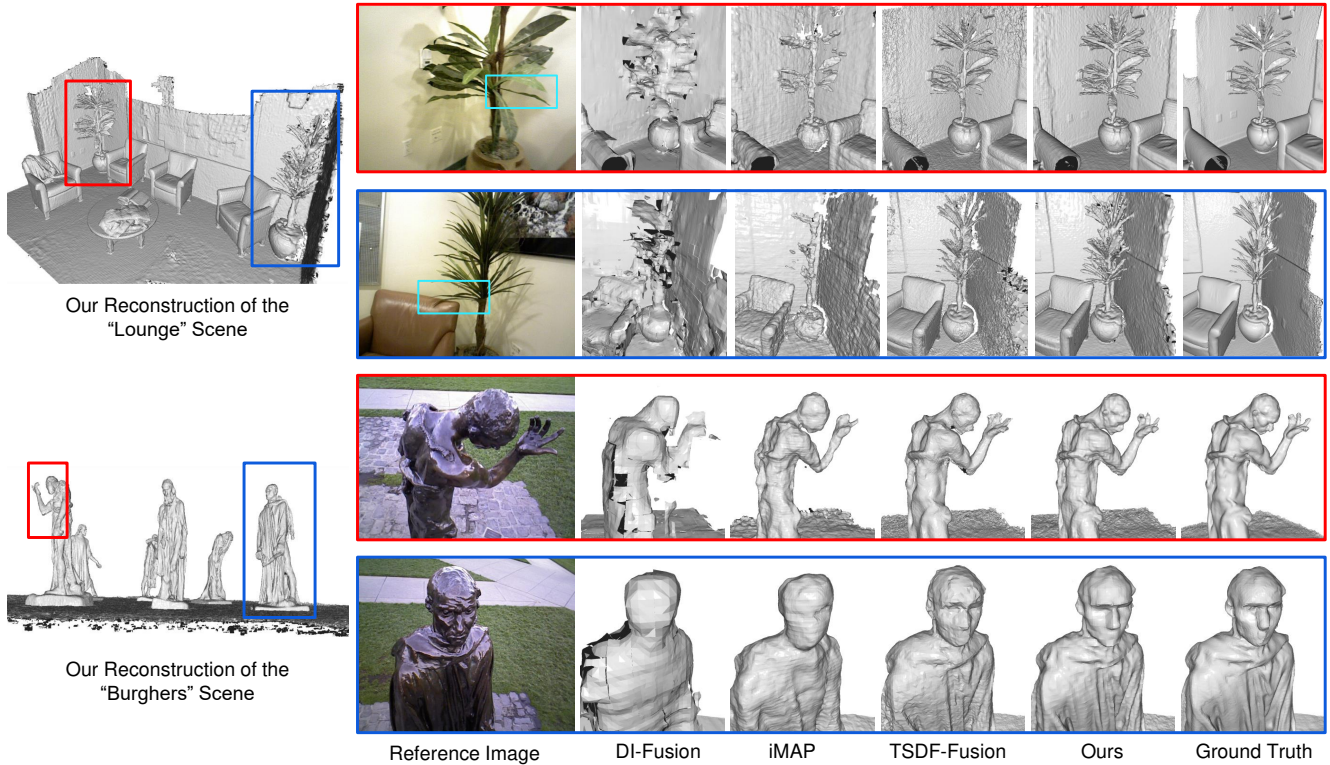


Figure 4. Qualitative comparison on the Scene3D dataset [6]. The proposed method can reconstruct more fine details accurately than previous traditional and learning-based methods. Notably, in the scene on top (the “Lounge” scene), our method can reconstruct some leaves and branches of the plants, which even the ground-truth mesh provided by the dataset fails to reconstruct (highlighted by the cyan bounding boxes in the reference RGB image). Compared to other methods in the scene at the bottom (the “Burghers” scene), our method can faithfully reconstruct the statue’s face. The differences are more visible when zooming in.

Method	Lounge	CopyRoom	CactusGarden	StoneWall	Burghers
	Accu. / Comp. / F1	Accu. / Comp. / F1	Accu. / Comp. / F1	Accu. / Comp. / F1	Accu. / Comp. / F1
TSDF Fusion [46]	86.16 / 93.46 / 89.66	89.88 / 90.22 / 90.05	75.62 / 94.17 / 83.84	88.73 / 94.34 / 91.45	72.78 / 82.35 / 77.26
iMAP [40]	85.76 / 87.98 / 86.85	83.94 / 80.22 / 82.04	73.04 / 85.01 / 78.57	85.82 / 85.83 / 85.82	70.23 / 71.71 / 70.96
DI-Fusion [16]	67.76 / 79.09 / 72.98	85.24 / 78.22 / 81.58	58.00 / 68.70 / 62.90	82.36 / 89.97 / 85.90	63.10 / 65.90 / 64.47
BNV-Fusion (Ours)	87.53 / 94.77 / 91.01	88.56 / 90.32 / 89.43	78.62 / 94.33 / 85.75	92.57 / 94.19 / 93.37	75.98 / 82.44 / 79.08

Table 1. Quantitative evaluation on the 3D Scene Dataset [6]. Our BNVF shows superior performance over the state-of-the-arts.

we consider the reconstructions provided by the dataset as ground truth. Since those models are created by running a TSDF Fusion method on all available frames in each sequence with post-processing, we take only every 10th frame as input images when running compared methods in the evaluation.

Tab. 1 quantifies the reconstruction quality on 5 scenes in the 3D Scene dataset, from which it is clear that our method outperforms other methods in all sequences. We highlight the differences in reconstructions produced by different methods in Fig. 4. Although DI-Fusion [16] is able to reconstruct smooth surfaces given noisy depth measurements, it fails to capture any fine details. This suggests

that integrating depth measurements in the latent space only does not utilize the depth measurements effectively. While decreasing the voxel size in DI-Fusion seems to be an alternative to facilitate accurate reconstruction, we present an ablation study on voxel size in Sec. 4.6 to show that there is a problematic trade-off if reducing voxel size. iMAP [40], another method based on a neural implicit representation, is unable to attend to details either because they use a single MLP to represent the entire scene. It is also visible in Fig. 4 that iMAP is more susceptible to noise in depth maps than both our method and DI-Fusion. This is due to the lack of prior knowledge since they train a new network for each scene. Compared to the TSDF-Fusion approach, our

Method	Livingroom0	Livingroom1	Office0	Office1
	Accu. / Comp. / F1	Accu. / Comp. / F1	Accu. / Comp. / F1	Accu. / Comp. / F1e
TSDF Fusion [46]	54.66 / 62.52 / 58.31	60.13 / 72.19 / 65.28	49.21 / 53.71 / 51.41	56.61 / 59.18 / 57.79
iMAP [40]	61.23 / 62.31 / 61.76	65.12 / 65.40 / 62.25	47.26 / 47.18 / 47.22	64.96 / 59.91 / 57.33
DI-Fusion [16]	61.52 / 64.43 / 62.55	69.69 / 67.81 / 68.00	50.06 / 48.99 / 49.79	54.53 / 50.41 / 52.36
BNVF (Ours)	71.26 / 73.86 / 72.54	80.01 / 81.94 / 81.02	58.17 / 60.33 / 59.23	63.93 / 63.89 / 63.91

Table 2. Comparison of accuracy (Accu.), completeness (Comp.) and F1 score (F1) on the augmented ICL-NUIM dataset [6]. The proposed BNV-Fusion outperforms all methods by a large margin in all metrics in 4 sequences except the accuracy in the ‘‘Office1’’ sequence.

Methods	Thresholds		Every x^{th} frames @ 2.5cm		Estimated Poses
	1cm	5cm	$x = 1$	$x = 30$	
TSDF [46]	33.63	90.87	85.26	79.81	55.20
iMAP [40]	28.43	89.72	82.07	74.12	55.96
DI-Fusion [16]	23.53	88.83	74.40	73.81	58.33
Ours	35.34	93.29	87.60	80.57	61.07

Table 3. F1 scores under various experimental settings.

method reconstructs challenging structures more accurately, such as thin leaves, fingertips, and human faces. Furthermore, by referring to the RGB images, it is encouraging to see that our method can even reconstruct thin structures (highlighted in the cyan bounding boxes in Fig. 4) that the ground-truth meshes miss. Note that $10\times$ more depth images are used to generate the ground-truth meshes. In addition, Tab. 3 evaluates the methods under different experiment settings (e.g. using SLAM to track camera, different thresholds, different frame rates).

4.4. Evaluation on ICL-NUIM Dataset

The ICL-NUIM dataset is a synthetic dataset with ground-truth 3D models. We use the synthetic sequences rendered by Choi *et al.* [6] because their rendering considers a more comprehensive noise model (e.g. disparity-based quantization, realistic high-frequency noise, and low-frequency distortion based on real depth cameras) to simulate the noise in real depth images. Our reconstructions are more accurate and complete, supported by the high accuracy and completeness in Tab. 2. Similar to the results in the 3D Scene Dataset, both DI-Fusion [16] and iMAP [40] struggle to reconstruct fine details in the geometry. We outperform TSDF-Fusion [46] by a larger margin than what we have in the 3D Scene Dataset. We believe this is because the synthetic noise in ICL-NUIM dataset is higher than the actual noise in a real depth sensor for large depth values, which indicates that our method is robust to greater noise in measurements. We present qualitative results of all compared methods in the supplementary material.

4.5. Evaluation on ScanNet

Both the ICL-NUIM and 3D Scene datasets have only a handful of sequences, which might not be diverse enough to test the generalization of a learning-based method. To this end, we evaluate our method on the validation set of ScanNet [8], comprising 312 sequences captured in diverse in-

Method	Accu.	Comp.	F1
TSDF Fusion [46]	73.83	85.85	78.84
iMAP	68.96	82.12	74.96
DI-Fusion	66.34	79.65	72.97
BNVF (Ours)	74.90	88.12	80.56

Table 4. Comparison between our proposed BNV-Fusion and TSDF-Fusion on ScanNet [8]. It demonstrates that the proposed method can generalize well to various scenes.

Method	Accu.	Comp.	F1
w/o global-level fusion	30.29	39.64	34.31
w/o local-level fusion	33.61	38.50	36.55
Ours (local + global)	68.34	70.01	69.17

Table 5. A quantitative comparison of different fusion algorithms. The bi-level fusion significantly outperforms both baselines.

door environments, such as living rooms, conference rooms, and offices. Since ScanNet also does not have ground-truth models of the scenes, we again use every 10^{th} frame as input and consider the meshes provided by the dataset as pseudo ground truth. The quantitative comparison is summarized in Tab. 4. Outperforming TSDF Fusion in various scenes in ScanNet demonstrates the excellent generalization of the proposed method. We present a few example reconstructions in the supplementary material.

4.6. Ablation studies

The proposed method differs from previous online reconstruction frameworks that use neural implicit representations thanks to the combination of the local- and global-level fusion. Therefore, we present more analysis on the bi-level fusion in this section using sequences in the augmented ICL-NUIM dataset [6].

Global-level fusion. Compared to prior art [16, 41] that only fuses information in the latent space, we optimize the global volume using neural rendering to achieve consistent reconstruction. In addition to the comparison against prior art reported in Tab. 1 and Tab. 2, we validate the necessity of the global-level fusion by disabling it in our system.

The quantitative comparison is reported in Tab. 5, and the contrast is visualized in Fig. 5. Using local-level fusion only with small voxel size suffers from depth outliers

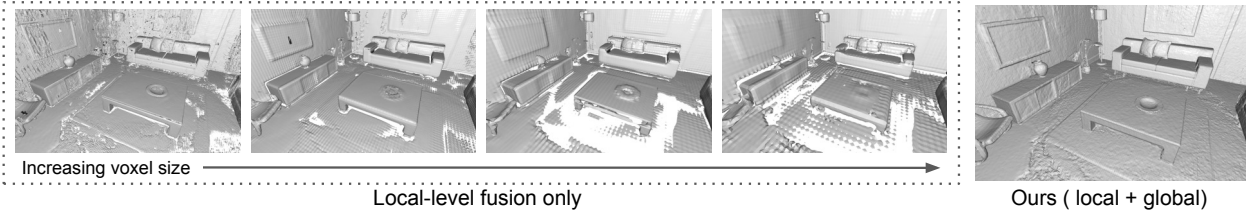


Figure 5. A visual comparison between local-level fusion only and the proposed method. Reconstructions using local-level fusion only clearly suffer from outliers and missing structures. Increasing the size of implicit voxels, from 5 cm to 20 cm with a step size 5 cm, as shown in the figure, reduces outliers at the cost of losing even more details.

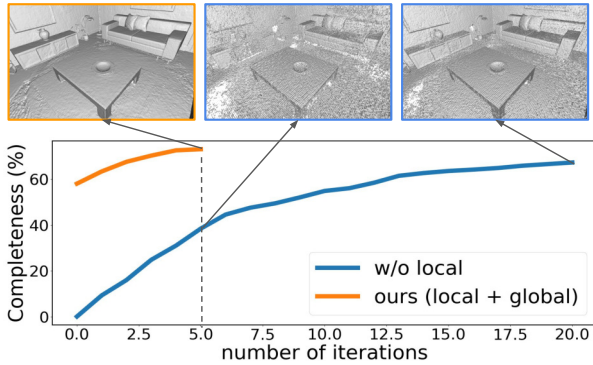


Figure 6. Improvement in completeness as the global-level fusion progresses. The global-level fusion initialized by the local-level fusion (shown in orange line) converges in 5 iterations. In contrast, the global-level fusion initialized randomly (shown in blue line) takes a much longer time to converge, and it is still not as good as the proposed method even after 20 iterations.

because it updates each implicit voxel independently in the domain of latent codes. We also test an approach used in DI-Fusion [16], which is increasing the voxel size. However, as shown in Fig. 5, this is also undesirable since more details are lost as the voxel size increases.

Local-level fusion. One of the key contributions of the local-level fusion is to initialize the global-level fusion. We compare our method against a baseline that initializes the global-level fusion randomly. Fig. 6 illustrates that the baseline requires more iterations to converge than our method. We also quantitatively compare the reconstructions of the baseline and that of our method by running the same number of iterations in Tab. 5, which clearly shows that the local-level fusion is crucial for dense reconstruction in an online setting.

4.7. Runtime analysis

We break down the runtime of each component as follows. Encoding a depth map to the latent space and the local-level fusion takes 0.1 seconds in total, the majority of which is taken by the encoding step. Running the global-level fusion for 2 iterations takes 0.5 seconds. Mesh extraction is excluded since it can be run on a separate thread.

Overall, BNV-Fusion runs at almost 2 frames per second (fps) on a 1080Ti GPU. DI-Fusion [16] runs at 10 fps on the same device. iMAP [40] is not real-time capable because we run on all frames rather than selected keyframes in the original paper.

4.8. Limitations

There are two limitations to be considered. First, the proposed method is still slower than traditional volumetric fusion approaches, which are heavily engineered. For instance, InfiniTAM v3 [19] can easily run at over 30 fps on a mobile device. Future research will need to be done to develop a neural-implicit-based reconstruction pipeline that can be run as fast as the traditional approaches. Second, severe noise in depth measurements sometimes causes discontinuities between neighboring implicit voxels, even though we try to improve the border consistency by using trilinear interpolation among neighboring implicit voxels.

5. Conclusion

We present BNV-Fusion, a novel online approach that effectively uses implicit neural volumes to represent geometry, for 3D reconstruction. The core of BNV-Fusion is the bi-level fusion algorithm: 1) The local-level fusion efficiently fuses new depth maps into the global volume; 2) The global-level fusion, framed as neural rendering, facilitates a consistent reconstruction. We evaluate BNV-Fusion in multiple reconstruction benchmarks, where it shows significant improvements in accuracy and completion over both a traditional volumetric fusion approach and recent learning-based approaches. This verifies BNV-Fusion’s capability of reconstructing the geometry faithfully. Furthermore, we justify the bi-level fusion in the ablation studies. Despite limitations discussed in Sec. 4.8, the superior performance of BNV-Fusion is still encouraging, hence we believe this work shows notable progress in dense 3D reconstruction.

Acknowledgement This work is supported by the UKRI grant: Turing AI Fellowship EP/W002981/1 and EP-SRC/MURI grant: EP/N019474/1. We would also like to thank the Royal Academy of Engineering and FiveAI.

References

- [1] Apple LiDAR scanner. <https://www.apple.com/newsroom/2020/03/apple-unveils-new-ipad-pro-with-lidar-scanner-and-trackpad-support-in-ipados>. 1
- [2] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. *arXiv preprint arXiv:2104.04532*, 2021. 2, 3
- [3] Rohan Chabra, Jan E Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *European Conference on Computer Vision*, pages 608–625. Springer, 2020. 1, 2, 5
- [4] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 4, 5
- [5] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2
- [6] Sungjoon Choi, Qian-Yi Zhou, and Vladlen Koltun. Robust reconstruction of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015. 5, 6, 7
- [7] 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. 2, 4
- [8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017. 2, 5, 7
- [9] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics (ToG)*, 36(4):1, 2017. 1, 3
- [10] Simon Fuhrmann and Michael Goesele. Fusion of depth maps with multiple scales. *ACM Trans. Graph.*, 30(6):148:1–148:8, 2011. 3
- [11] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas A. Funkhouser. Local deep implicit functions for 3d shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4856–4865, 2020. 2
- [12] Rohit Girdhar, David F Fouhey, Mikel Rodriguez, and Abhinav Gupta. Learning a predictable and generative vector representation for objects. In *European Conference on Computer Vision*, pages 484–499. Springer, 2016. 2
- [13] Xian-Feng Han, Hamid Laga, and Mohammed Bannamoun. Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(5):1578–1604, 2021. 1
- [14] A. Handa, T. Whelan, J.B. McDonald, and A.J. Davison. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *IEEE Intl. Conf. on Robotics and Automation, ICRA*, Hong Kong, China, May 2014. 5
- [15] Andrew Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003. 1
- [16] Jiahui Huang, Shi-Sheng Huang, Haoxuan Song, and Shi-Min Hu. Di-fusion: Online implicit 3d reconstruction with deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8932–8941, 2021. 3, 5, 6, 7, 8
- [17] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6001–6010, 2020. 2
- [18] Olaf Kähler, Victor Adrian Prisacariu, Carl Yuheng Ren, Xin Sun, Philip Torr, and David Murray. Very high frame rate volumetric integration of depth images on mobile devices. *IEEE transactions on visualization and computer graphics*, 21(11):1241–1250, 2015. 3
- [19] O. Kahler, V. A. Prisacariu, C. Y. Ren, X. Sun, P. H. S Torr, and D. W. Murray. Very High Frame Rate Volumetric Integration of Depth Images on Mobile Device. *IEEE Transactions on Visualization and Computer Graphics*, 22(11), 2015. 8
- [20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015. 5
- [21] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *arXiv preprint arXiv:2007.11571*, 2020. 2
- [22] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 4
- [23] Nico Marniok, Ole Johannsen, and Bastian Goldluecke. An efficient octree design for local variational range image fusion. pages 401–412, 2017. 3
- [24] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019. 1, 2
- [25] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020. 1, 2, 5
- [26] 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. IEEE, 2011. 1, 2, 3, 5
- [27] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In

- Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. [2](#)
- [28] Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Trans. Graph.*, 32(6):169:1–169:11, 2013. [3](#)
- [29] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019. [1](#), [2](#)
- [30] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019. [5](#)
- [31] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. [1](#)
- [32] Hanspeter Pfister, Matthias Zwicker, Jeroen van Baar, and Markus H. Gross. Surfels: surface elements as rendering primitives. In Judith R. Brown and Kurt Akeley, editors, *SIGGRAPH*, pages 335–342. [3](#)
- [33] Victor Adrian Prisacariu, Olaf Kähler, Stuart Golodetz, Michael Sapienza, Tommaso Cavallari, Philip HS Torr, and David W Murray. Infinitam v3: A framework for large-scale 3d reconstruction with loop closure. *arXiv preprint arXiv:1708.00783*, 2017. [1](#), [2](#), [3](#)
- [34] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. [4](#)
- [35] V. Reijgwart, A. Millane, H. Oleynikova, R. Siegwart, C. Cadena, and J. Nieto. Voxgraph: Globally consistent, volumetric mapping using signed distance function submaps. *IEEE Robotics and Automation Letters*, 2020. [3](#)
- [36] Thomas Schöps, Torsten Sattler, and Marc Pollefeys. Surfelmeshing: Online surfel-based mesh reconstruction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(10):2494–2507, 2020. [3](#)
- [37] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *arXiv preprint arXiv:1906.01618*, 2019. [2](#)
- [38] Frank Steinbrücker, Christian Kerl, and Daniel Cremers. Large-scale multi-resolution surface reconstruction from RGB-D sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3264–3271, 2013. [3](#)
- [39] Jörg Stückler and Sven Behnke. Multi-resolution surfel maps for efficient dense 3d modeling and tracking. *J. Vis. Commun. Image Represent.*, 25(1):137–147, 2014. [3](#)
- [40] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. imap: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6229–6238, 2021. [2](#), [3](#), [5](#), [6](#), [7](#), [8](#)
- [41] Silvan Weder, Johannes L Schonberger, Marc Pollefeys, and Martin R Oswald. Neurfusion: Online depth fusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3162–3172, 2021. [1](#), [3](#), [5](#), [7](#)
- [42] Thomas Whelan, Renato F. Salas-Moreno, Ben Glocker, Andrew J. Davison, and Stefan Leutenegger. Elasticfusion: Real-time dense SLAM and light source estimation. *Int. J. Robotics Res.*, 35(14):1697–1716, 2016. [3](#)
- [43] Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T Freeman, and Joshua B Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 82–90, 2016. [2](#)
- [44] Zhixin Yan, Mao Ye, and Liu Ren. Dense visual SLAM with probabilistic surfel map. *IEEE Trans. Vis. Comput. Graph.*, 23(11):2389–2398, 2017. [3](#)
- [45] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE Multim.*, 19(2):4–10, 2012. [1](#), [3](#)
- [46] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3d: A modern library for 3d data processing. *arXiv preprint arXiv:1801.09847*, 2018. [5](#), [6](#), [7](#)