HHS Public Access

Author manuscript

IEEE Trans Vis Comput Graph. Author manuscript; available in PMC 2023 July 01.

Published in final edited form as:

IEEE Trans Vis Comput Graph. 2022 July ; 28(7): 2668–2681. doi:10.1109/TVCG.2020.3036868.

ManhattanFusion: Online Dense Reconstruction of Indoor Scenes from Depth Sequences

Mahdi Yazdanpour,

Department of Physics, Geology and Engineering Technology, Northern Kentucky University.

Guoliang Fan*,

School of Electrical and Computer Engineering, Oklahoma State University.

Weihua Sheng

School of Electrical and Computer Engineering, Oklahoma State University.

Abstract

We present a new framework for online dense 3D reconstruction of indoor scenes by using only depth sequences. This research is particularly useful in cases with a poor light condition or in a nearly featureless indoor environment. The lack of RGB information makes long-range camera pose estimation difficult in a large indoor environment. The key idea of our research is to take advantage of the geometric prior of Manhattan scenes in each stage of the reconstruction pipeline with the specific aim to reduce the cumulative registration error and overall odometry drift in a long sequence. This idea is further boosted by local Manhattan frame growing and the local-to-global strategy that leads to implicit loop closure handling for a large indoor scene. Our proposed pipeline, namely ManhattanFusion, starts with planar alignment and local pose optimization where the Manhattan constraints are imposed to create detailed local segments. These segments preserve intrinsic scene geometry by minimizing the odometry drift even under complex and long trajectories. The final model is generated by integrating all local segments into a global volumetric representation under the constraint of Manhattan frame-based registration across segments. Our algorithm outperforms others that use depth data only in terms of both the mean distance error and the absolute trajectory error, and it is also very competitive compared with RGB-D based reconstruction algorithms. Moreover, our algorithm outperforms the state-of-the-art in terms of the surface area coverage by 10%−40%, largely due to the usefulness and effectiveness of the Manhattan assumption through the reconstruction pipeline.

Keywords

Manhattan frame; Volumetric reconstruction; Planar alignment; Local pose optimization

1. Introduction

With the prevalence of low-cost RGB-D cameras, even available in the new generation of smartphones, 3D sensing technologies such as face recognition, facial payment, 3D

*Corresponding Author guoliang.fan@okstate.edu.

modeling, semantic mapping, gesture recognition, robot navigation, and augmented and virtual reality are becoming increasingly feasible among consumers. In the past decade, there has also been extensive research on various 3D sensing applications in computer vision and robotics communities. Dense mapping and volumetric reconstruction of extended scale indoor scenes are active research topics in these communities and creating high fidelity dense models from RGB-D streams has become one of the most active and interesting research topics in many related areas. In general, in the robotics community, the research is focused on Simultaneous Localization and Mapping (SLAM) that generates sparse 3D maps using visual features and depth information where the main goal is to minimize the absolute trajectory error with less concern about the quality and geometric details of the reconstructed models [8], [19], [20], [24]. By contrast, the research objective in the computer vision community is often about dense volumetric reconstruction where the fidelity and quality of the created models are of the main interest [1], [15], [21], [47], [49].

Over the past decade, many reconstruction algorithms have been proposed for generating 3D models from real scene objects or environments. While creating dense reconstruction of a single object can be performed without much difficulty, large scale indoor scene reconstruction particularly with complex trajectories can be a challenging task. In this regard, a few key research problems have been studied that led to the development of some new and powerful methods for 3D dense modeling. Specifically, the cumulative registration error and overall geometric drift are two major problems in large scale scene reconstruction. Researchers in both vision and robotics communities proposed various online approaches [1], [2], [6], [12], [19], [46], [47], [49] and offline methods [15], [21], [52] to address these two issues. Usually, online and real-time approaches can reconstruct a scene model incrementally and progressively from the input RGB-D stream where additional constraints (such as loop closure or visual feature matching) are used to minimize the drifting problem. On the other hand, the offline methods usually can generate more accurate and detailed models, but they require more processing time for global optimization over all input frames. Many existing frameworks involve both color and depth (RGB-D) data in scene reconstruction and some methods use only depth data. In this work, we are particularly interested in depth-based indoor scene reconstruction.

In this paper, we present a new pipeline, namely ManhattanFusion, for 3D dense reconstruction of extended scale indoor scenes from only depth data by taking advantage of the Manhattan World (MW) assumption [39], as shown in Fig. 1. This research is particularly useful in cases with a poor light condition or in a nearly featureless environment. The lack of RGB images makes long-range camera pose estimation difficult in large indoor scenes. The MW assumption available in most structured indoor scenes provides a reliable geometry prior for 3D scene mapping and has been used in many SLAM systems. Different from other approaches where the MW assumption is mainly used for data rectification or model representation (e.g., [22], [23], [53], [54]), our goal is to reduce the accumulative drift error over an extended scale by exploiting the MW assumption as a geometric prior at multiple stages in the proposed pipeline. Specifically, our approach has three major components. First, we propose a new local Manhattan frame growing strategy that sequentially propagates the geometric cue across depth frames to enhance the registration accuracy and efficiency without using any visual information. Second, we

further incorporate the Manhattan constraint between adjacent keyframes for local pose optimization that leads to accurate reconstruction of local segments. Third, we develop Manhattan keyframe-based model registration where geometric planar alignment between two local segments mitigates the discontinuity problem in large area surface reconstruction due to noise and error in depth data, leading to smoother and more complete planar surfaces in the final reconstructed model.

2 Related Work

The major challenges in 3D scene reconstruction of large scale indoor environments include the cumulative drift due to the registration error and the inaccuracy in pose estimation. Specifically, in extended scale environments with complex trajectories, this odometry information is prone to error. There has been abundant research on 3D mapping and dense modeling that are motivated by minimizing this error and creating drift-free 3D reconstruction. The acquisition of robust pose estimation is generally possible by using visual feature matching in conjunction with bundle adjustment [26] and/or pose graph optimization [27] to minimize the reprojection error or to refine pose estimation. A comparison between the state-of-the-art in 3D mapping and dense reconstruction systems is shown in Table 1, which is an extended version of the analogy performed by [50].

One of the foremost approaches that had a remarkable impact on the related research, is KinectFusion [3], [5]. This system demonstrated real-time dense mapping and tracking for volumetric reconstruction of objects and small indoor scenes. KinectFusion was further extended to handle large scale scenes. For example, the spatially extended KinectFusion, i.e., Kintinuous [9], and its extension [10] were developed for mesh mapping of large scale indoor environments using dynamic shift of the voxel grid in real-time. A new hierarchical data structure has been proposed in [14] to address the scalability problem in real-time volumetric surface reconstruction of large scale environments. Voxel hashing [13] proposed a spatial hashing scheme for scalable volumetric reconstruction in real-time. Point-based fusion [18] presented a new system for online reconstruction of large scale scenes without the need of a spatial data structure. Moving volume KinectFusion [43] was developed to handle large volumes in outdoor environments. The open source Kinfu large scale [40] has been implemented based on the Point Cloud Library (PCL) for producing textured meshes from large areas using handheld commodity range sensors. These reconstruction systems are normally limited to relatively simple trajectories and may not detect the loops and handle the loop closure problem in extended scale environments.

Thereafter, a number of 3D mapping and dense modeling approaches have been developed for handling loop closures in complicated trajectories. For example, RGB-D mapping [24], patch volumes [25], RGB-D SLAM [29], [30], deformation-based dense SLAM [11], and multi-resolution surface reconstruction [33] have been proposed to handle loop closures using visual feature matching or dense image registration in long trajectories and to create large scale maps in real-time. Many volumetric methods have also been developed to generate high quality reconstructions using global optimization. Offline systems such as elastic fragments [16], dense scene reconstruction with points of interest [15], and Redwood robust reconstruction [21] present the new frameworks, which can handle loop closures and

produce globally consistent volumetric reconstructions. A novel local-toglobal hierarchical optimization framework called BundleFusion has been proposed in [1] that handles loop closures implicitly and reintegrates the scene on the fly, and generates highly detailed dense models in real-time from RGB-D data.

Our ManhatanFusion approach is aimed at taking advantage of the MW assumption with multiple purposes to generate volumetric reconstructions of extended scale indoor scenes with fine-grained details. Compared with existing methods where the MW idea was used [22], [23], [28], [53], [54], our approach has three unique features. First, our approach only uses the depth data where the MW assumption plays an important role for long-range camera pose estimation, and local-to-global model integration along with implicit loop-closure handling, while others mainly rely on RGB images for camera pose estimation and loop closure detection. Second, our approach creates a dense and detailed volumetric representation while others generate a CAD-like model, a floorplan model, or colored point clouds. Third, our Manhattan frame growing strategy propagates the geometric constraint across the long depth sequences, and supports pose graph optimization along with local model reconstruction and local-to-global model integration. This is in contrast with previous methods where the MW assumption is mainly used for a specific purpose, such as data rectification, model representation or frame alignment.

3 Proposed Method

We first provide an overview of the proposed ManhattanFusion framework. Then we discuss the major technical components in the pipeline, including Manhattan frame growing, local robust pose optimization, Manhattan frame-based model registration and final model integration.

3.1 Overview

The pipeline of ManhattanFusion is shown in Fig. 2. It begins by performing a preprocessing on depth sequences to compute the normals and depth maps. The core idea of this framework is using Manhattan frame growing to extend the first estimated Manhattan frame to the adjacent depth keyframe, leading to a rotated depth keyframe aligned with the dominant plane, called Manhattan keyframe (MKF), which has a significant role in all reconstruction steps. In our previous work [42], we perform local Manhattan frame growing to create regional segments as long as the dominant plane is detectable in the scene. The system starts creating a new local model when the dominant plane disappears or a new dominant plane is observed. In our new framework, to drive more accurate information and minimize geometric drift in each segment, we create local segments for every N consecutive MKFs (e.g., $N = 100$). According to our pose tracking, a new keyframe is identified when a significant translation occurs in the camera motion, then the previous Manhattan keyframe is extended to the new keyframe and used for local segment creation.

To facilitate the Manhattan frame growing process, we utilize surface normal adjustment to produce highly persistent distribution of the surface normals for Manhattan frame estimation [4] that yields dominant planes for each depth keyframe (as shown in Fig. 3). The first identified Manhattan frame will initiate the Manhattan frame growing scheme over

adjacent depth keyframes. Afterwards, we deploy reliable planar pre-alignment between MKFs across dominant planes for surface registration initialization in local segment creation as well as global model integration. At the same time, we construct a segmental pose graph and then incrementally optimize the local odometry in each local segment using a MKF-based constraint, which reduces considerably the overall geometric drift error in the final dense model. Moreover, we use local pose optimization and MKF-based depth-to-model registration to create robust local segments by using refined camera poses retrieved from the local pose optimization step. Finally, we create a complete volumetric representation of the scene by integrating all local segments into a global framework via MKF-initialized registration between consecutive segments. Our proposed approach is a depth-based reconstruction system and we do not use RGB channels in our pipeline. We present all major steps in details below.

3.2 Manhattan Frame Growing

The geometric representation of the structured indoor scenes using the Manhattan world assumption provides reliable information in 3D modeling, scene understanding, and semantic segmentation applications. The MW assumption states that all objects in an indoor scene are aligned with one of three mutually orthogonal planes. Estimating the Manhattan frame based on this assumption provides the reliable geometric properties, which benefits the reconstruction process of indoor environments. The Manhattan frame estimation methods can be classified into two groups. The first group includes the RGB-based approaches, which rely on extracting lines, edges, and orthogonal vanishing points from RGB frames [38], [44]. The second group is RGB-D-based methods, which use the 3D perspective information like surface normals computed from point clouds or depth frames [36], [37]. In the majority of instances, the observed scene is not an ideal Manhattan scene with absolute Manhattan elements, which are aligned with the principal axes. In addition, due to noise, inconsistency in depth information, and computational error, estimating the reliable Manhattan frame can be an arduous task.

In our new framework, we propose a new Manhattan frame growing scheme to extend the first estimated Manhattan frame to the next identified depth keyframes along the dominant surface planes of the scene. This proposed technique is extended based on the original Manhattan frame estimation method introduced by [36]. The main idea of Manhattan frame estimation is to find the best rotation matrix and use it to transform the original surface normals to be aligned with at least one of three main perpendicular axes as follows:

$$MF = \underset{R,\,X}{\operatorname{argmin}} \ \frac{1}{2} \parallel (R \cdot N - X) \parallel_F^2 \ + \lambda \parallel X \parallel_{1,1} \tag{1}$$

where $N \in \mathbb{R}^{3 \times m}$ is the matrix of the original surface normals, $R \in SO(3)$ is the rotation matrix, and X is the sparse matrix result of applying R to matrix of the surface normals N. The second term acts as a sparsity regularizer and is the sum of the ℓ_1 norms of the columns in matrix X. $\parallel X \parallel_{p,q}$ shows the $\ell_{p,q}$ matrix norm of X, and the parameter λ operates as a trade-off between sparsity and error sensitivity. This non-convex optimization problem does not have a globally optimal solution and the local minimum is attainable via alternating

optimization, where the solution for two variables R and X is updated iteratively, while the other variable is kept fixed. In this alternating optimization, the current estimation of the rotation matrix R applied to the original surface normals N is updated using the singular value decomposition function and the sparse matrix X is updated using a soft-thresholding operator to achieve a higher sparseness. Finally, the best estimated rotation matrix will be applied to align the surface normals to the main principal axes of the scene.

In our Manhattan frame growing strategy, we use the original Manhattan frame estimation method to find the best rotation matrix R for the first identified keyframe in the scene and then use it as a reliable initialed rotation matrix for the surface normals of the next adjacent depth keyframe f_i^k as follows:

$$MFG = \underset{R_i, X_i}{\operatorname{argmin}} \frac{1}{2} \|(X_i - R_i \cdot N_i) + (X_i - X_{i-1})\|_F^2 + \lambda \|X_i\|_{1,1}, \tag{2}$$

where N_i is the matrix of the original surface normal vectors of the new keyframe, R_i is the rotation matrix, which is initialized from the previous desirable rotation matrix R_{i-1}, X_i is the sparse matrix result of applying R_i to the set of surface normal vectors, and X_{i-1} is the sparse matrix belongs to the previous MKF. X_i should be continuance of X_{i-1} to grow dominant planes. The second term is used as a regularizer and helps to avoid the overfitting problem and to achieve the higher sparseness. We rotate every MKF to be aligned with the dominant plane of the scene before using in the reconstruction system, as shown in Fig. 4.

3.3 Local Robust Pose Optimization

The importance of the pose optimization and graph correction for minimizing the odometry drift in large scale environments has been emphasized in many SLAM-based and volumetric reconstruction systems. The camera tracking and pose estimation can be fulfilled based on the visual odometry or just depth information. The reconstructed models by visual odometry approaches, which use visual feature correspondences via a frame-to-frame tracking and matching scheme, and volumetric fusion systems with a frame-to-model tracking and registration model based on the geometric features, are inherently prone to accumulate the drift error. This problem led researchers to use different auxiliary systems like pose graph optimization, bundle adjustment, and loop closure detection for pose estimation correction. Performing global optimization over all frames is a gradual process and increases the computational cost.

In our local pose optimization method, each segment will be locally optimized within on frames by considering the additional constraint between two Manhattan keyframes. The local pose information is usually more reliable to be obtained and optimized. We incrementally optimize the pose estimates to achieve consistent frame poses. We align the identified depth keyframe with the estimated Manhattan frame of the scene and rely on the geometric similarity to find the translation between two successive Manhattan keyframes in order to find the definitive pose estimation and to reduce the accumulation of the local drift error. These geometric camera pose constraints help to have more robust and accurate pose estimation. Furthermore, this technique is faster than traditional SLAM-based methods

that use the visual feature matching and explicit loop closure detection system to refine the camera trajectory recursively. We optimize the pose graph through minimizing the following objective function:

$$X^* = \underset{X}{\operatorname{argmin}} \; E(X), \tag{3}$$

we formulate the objective function in the following form, where $E(X)$ and e_{ij} are defined as:

$$E(X) = \sum_{i,j} e_{ij}^T \Omega_{ij} e_{ij} \tag{4}$$

where

$$e_{ij} = f(x_i, x_j) - d_{ij},$$

where $X = \{x_0, \ldots, x_n\}$ is a set of estimated frame poses, $f(x_i, x_j)$ is the relative transformation between two consecutive poses x_I and x_j obtained from camera tracking system, d_{ij} is an observed constraint derived from two adjacent MKFs f_i^k and f_j^k, and Ω_{ij} represents uncertainty and is the covariance matrix of the relative transformation between consecutive frames. In the quaternion representation of pose $x_n = \left[p_n^T, q_n^T\right]^T$, p_n represents the position and q_n is the orientation represented as an Eigen quaternion. Using quaternion can speed-up the optimization process, since this representation requires less computation than a regular rotation matrix. Specifically, $f(x_i, x_j)$ is computed from the absolute poses returned by the camera tracker and represented a quaternion form as $f(x_i, x_j) = \left[\hat{p}_{ij}^T, \hat{q}_{ij}^T\right]^T$, where p_{ij} and q_{ij} are the translation and relative rotation between two poses, respectively. To minimize the energy function and correct the pose estimation error, we add an observed Manhattan-based constraint d_{ij} from f_i^k and f_j^k, which is derived from two adjacent MKFs and shows the transformation between these two. According to our Manhattan growing algorithm, a new MKF is identified when a significant translation in pose tracking occurs and a new node is added to the local reconstructed graph. Afterwards, d_{ij} is calculated based on the position and orientation information of two adjacent MKFs and the error between the measured and observed poses is corrected according to the assigned pose IDs to keyframes. We use Ceres Solver [48] for optimizing the local trajectories and to minimize sequential constraints between depth keyframes. We assign pose IDs to estimated MKFs and use these indices to retrieve the corrected pose estimations. These refined poses will be used in the sequential registration process in a MKF-based depth-to-model registration scheme to create robust fused segments.

3.4 Manhattan Keyframe-based Model Registration

We effectively enhance the robustness of keyframe registration by benefiting from the local geometric information of the scene. We use pre-planar alignment based on the identified MKFs as a robust initializer for surface registration. Using MKFs provides a reliable geometric constraint to reduce the overall registration error. We also assign pose IDs to

estimated MKFs and use these indices to retrieve the refined pose estimations from the local pose optimization step. These refined poses will be used in the sequential registration process. We initially use geometric registration to efficiently align dominant planes in two successive depth f_i^k and f_j^k. We run the planar alignment by constraining the registration for those points located on the dominant plane of the consecutive MKFs. For the MKF-based planar alignment, the metric distance between the point sets on two dominant surfaces is minimized by solving:

$$T_{ij} = \underset{T_{ij}}{\operatorname{argmin}} \sum_{i,j} \| T_{ij}\left(p_i^k\right) - p_j^k \|^2, \tag{5}$$

where $p_i^k \in P_i$ and $p_j^k \in P_j$ are two set of points on the dominant planes P_i and P_j located in two adjacent Manhattan keyframes f_i^k and f_j^k, and T_{ij} is the transformation matrix that minimizes the distance between two planes. The dominant plane in each scene is first detected using the plane detection algorithm available in the point cloud library. It is the plane with the most points and is available in the sequential depth frames until a new MKF is identified as a dominant plane. Furthermore, after performing the planar initialization, point-to-plane surface registration similar to (6) is used to fully register the Manhattan depth keyframes. This planar alignment reduces the computational complexity and enhances the accuracy and speed of the surface registration by providing a robust and reliable initiation and reducing the number of iterations. For the final registration between local segments, we use the following geometric registration on the overlapping parts of two consecutive segments.

$$T_S = \underset{T_S}{\operatorname{argmin}} \sum_{i} \| N_i (T_S(p_i') - q_i') \|^2, \tag{6}$$

where $p_i' = \left(p_{ix}', p_{iy}', p_{iz}', 1\right)^T$ and $q_i' = \left(q_{ix}', q_{iy}', q_{iz}', 1\right)^T$ are a sample point and its corresponding point on the surface of two successive local segments, $N_i = \left(N_{ix}, N_{iy}, N_{iz}, 0\right)^T$ is the unit normal vector at destination point q_i', and T_S is the transformation matrix to align two set of points. After performing this iterative registration, two neighbor segments will be aligned so that we can integrate two adjacent local segments by constraining point registration in the overlapping parts of two local segments.

MKF-based model registration can handle loop closure implicitly due to the continuous propagation of Manhattan prior segment-by-segment over a long sequence. For example, given the first MKF from the first segment and the last MKF from the last segment (Fig. 5(a)), let us assume we have a loop closure between these two segments. We initially use geometric registration to robustly align the dominant planes (colored) extracted from two MKFs. Then we perform planar alignment by constraining point registration for P_i and P_j located on the dominant planes (Fig. 5 (b)). Then the overlap part is identified where point-to-plane surface registration is performed to align two local segments. As the result, the loop closure is handled without being detected explicitly since the drift problem is mitigated by sequentially applying the Manhattan prior in multiple stages (Fig. 5(d)).

3.5 Final Model Integration

After obtaining the optimized pose estimation for each keyframe, a MKF-based depth-to-model registration scheme using a non-uniform weighting strategy is used to integrate the depth keyframes into the previously reconstructed TSDF (Truncated Signed Distance Function) model. The TSDF model is represented in GPU memory on a volumetric grid as a 3D array of voxels. The integration of the new surface measurements is accomplished in a similar way to KinectFusion [3], [5]. Assume each voxel at location p contains a signed distance TSDF value $v(\mathrm{p})$ and a voxel weight $w(\mathrm{p})$. To integrate an i^{th} incoming Manhattan keyframe with optimized pose estimation into the reconstructed model, the value of each voxel is updated by:

$$v_i(\mathrm{p}) = \frac{v_{i-1}(\mathrm{p})w_{i-1}(\mathrm{p}) + v_i(\mathrm{p})w_i(\mathrm{p})}{w_{i-1}(\mathrm{p}) + w_i(\mathrm{p})}, \tag{7}$$

where $w_i(\mathrm{p})$ denotes the weighting of the TSDF to surface measurement uncertainty and is defined by

$$w_i(\mathrm{p}) = min(w_{i-1}(\mathrm{p}) + w_i(\mathrm{p}), w_{max}), \tag{8}$$

In our implementation, we set $w_i(\mathrm{p}) = 1$, as a simple average, and $w_{max} = 128$. After this integration step, the 3D dense segments are reconstructed using the refined pose estimation results retrieved from optimized camera trajectory.

4 Experimental Results

In this section, different types of datasets, the employed evaluation methods consisting of our new proposed metric, and all quantitative and qualitative experimental results are presented. Not only the proposed ManhattanFusion method is evaluated by comparing against many state-of-the-art SLAM algorithms, but also we show the progressive improvement over our early attempts [4], [17], [42]. Specifically, the Manhattan frame-based reconstruction (MFR) algorithm proposed in [4] only applies the MW assumption to find dominant planes to assist frame alignment and point-to-plane registration. MFR was enhanced with pose graph optimization (MFR + PGO) in [17] with improved reconstruction accuracy. The Local Manhattan frame growing method (LMFG) proposed in [42] extends MFR + PGO sequentially and incrementally with improved accuracy and robustness and is able to handle loop closure implicitly. Essentially, ManhattanFusion furthers LMFG by introducing local model reconstruction and local-to-global integration that integrates and streamlines previous key techniques in one unified pipeline as presented in Fig. 2.

4.1 Datasets

We evaluated our proposed approach on a variety of synthetic and real scene datasets including various RGB-D sequences from indoor environments with both simple and complicated trajectories.

4.1.1 Synthetic Scenes—First, we have evaluated MahattanFusion performance on the ICL-NUIM dataset provided by [34], which has been released for the evaluation of RGB-D

visual odometry, 3D reconstruction and SLAM systems. This dataset includes two different synthetic models of two virtual indoor scenes, a living room and an office, with different camera trajectories. The ICL-NUIM dataset has also provided a ground truth surface model for the living room, for benchmarking surface reconstruction accuracy evaluation.

Then, we have used the augmented ICL-NUIM dataset provided by [21], which augmented two synthetic models based on the original ICL-NUIM dataset. These virtual scenes have a higher number of frames with longer and more complicated trajectories for the whole scene 3D reconstruction. The augmented ICL-NUIM dataset has also provided a dense point-based surface model for the office scene, which enables the measurement of the surface reconstruction accuracy.

4.1.2 Real World Scenes—In addition to the synthetic datasets, we have used different real scene sequences to evaluate the performance of our proposed approach in 3D reconstruction and camera tracking. We have used BundleFusion dataset provided by [1] that consists of eight large scale indoor environments with a high number of frames and very long trajectories. We have also tested our approach on a few depth sequences from the SceneNN dataset [45] and TUM RGB-D dataset provided by [35], which is a novel benchmark for the evaluation of 3D reconstructions and RGB-D SLAM algorithms.

4.2 Evaluation Methods

We have evaluated the performance of our proposed approach on multifarious synthetic and real depth sequences from three different perspectives, the pose tracking accuracy, the dense reconstruction preciseness, and the surface area coverage.

4.2.1 Absolute Trajectory Error—The global consistency of the estimated trajectory is very important in RGB-D mapping and dense modeling systems. The Absolute Trajectory Error (ATE) proposed by [35] is used to evaluate this consistency by computing the absolute distances of the estimated trajectory to the ground truth trajectory.

4.2.2 Surface Reconstruction Error—The Surface Reconstruction Error (SRE) is another metric to evaluate the performance of the SLAM-based frameworks and dense reconstruction systems. This evaluation provides the mean distance of the generated model to the ground-truth surface. In our work, we have used the CloudCompare [41] tool to compute the surface reconstruction accuracy.

4.2.3 Surface Area Coverage—The Surface Area Coverage (SAC) shows the capability of a generated model to cover the whole surface area of an indoor scene. It is our expectation that the local geometric information of the scene and MKF-based planar alignment would enable ManhattanFusion to cover more surface area and to mend holes, gapes, and discontinuities.

4.3 Quantitative Comparison

For the quantitative comparison, we have computed the mean distance of the generated models by ManhattanFusion on the original and augmented synthetic ICL-NUIM datasets

to the ground-truth surface for the living room and point-based dense map for the office and compared the results with the numeric values released by [21] and [1], as shown in Tables 2 and 3. Our results are compared to Kintinuous [9], DVO SLAM [19], RGB-D SLAM [29], MRSMap [31], ElasticFusion [6], [7], BundleFusion [1], and Redwood [21] on the original synthetic ICL-NUIM sequences. They are also compared to Kintinuous, DVO SLAM, Redwood, SUN3D SfM [20], MFR [4], MFR + PGO [17], and LMFG [42] on two augmented synthetic ICL-NUIM scenes. The type of the input frames are specified in the tables for a better comparison. ”RGB-D” means the system uses both color and depth information in the reconstruction process, ”D” means the system uses only depth information, and ”D+RGB” means the system uses depth information in the reconstruction pipeline and color information for loop closure detection.

It is evident that, ManhattanFusion outperforms Kintinuous, DVO SLAM, RGB-D SLAM, SUN3D SfM, MRSMap, Redwood, MFR, MFR + PGO, and LMFG reconstruction systems. In addition, it has very close results to the offline Redwood robust reconstruction and BundleFusion approaches. This comparison confirms that our proposed framework reduces the average mean distance dramatically by factors of 4.3 relative to Kintinuous and RGB-D SLAM, 4.1 relative to DVO SLAM, 8.5 relative to MRSMap, and 1.2 relative to Redwood on the original synthetic ICL-NUIM dataset. In the same way, it reduces the mean distance by factors of 3.3 relative to Kintinuous, 2.6 relative to DVO SLAM, 2 relative to SUN3D SfM, 2.7 relative to MFR, 1.4 relative to MFR + PGO, and 1.2 relative to LMFG on the augmented synthetic ICL-NUIM sequences and also has very close results to the Redwood reconstructed models. In other words, the bottom half of Table 3 also serves as an ablation study to show the progressive improvement from MFR to MFR + PGO, from MFR + PGO to LMFG, and from LMFG to ManhattanFusion.

Additionally, we further evaluate our pose tracking ability on the original and augmented ICL-NUIM, and TUM RGB-D trajectories. The measured absolute trajectory error shows our trajectory estimation performance on these datasets, which is on par with or better than the camera tracking of the existing state-of-the-art systems. Our computed ATE has a noticeable improvement compared to other approaches, near 68% improvement over Kintinuous, 70% over DVO SLAM, 71% over RGB-D SLAM, 92% over MRSMap, and 63% over Redwood on the synthetic ICL-NUIM dataset, as shown in Table 4. We have also compared our estimated ATE on the augmented ICL-NUIM dataset, as shown in Table 5, and it is conspicuous that ManhattanFusion has near 69% improvement over Kintinuous, 78% over DVO SLAM, 61% over SUN3D SfM, 14% over Redwood, and 75% over Elastic Fusion. Tables 6 and 7 also indicate the accuracy of our pose tracking on the TUM RGB-D dataset. Our approach relies only on geometric registration between depth keyframes. The geometric-based approaches like ManhattanFusion fail on fr3/nst sequence, since this scene is just a textured flat wall with lack of depth variation where our geometric-based approach cannot create a valid model due to the inherent ambiguity.

We have also proposed a new metric to evaluate the ability of the reconstruction process to cover the possible discontinuities and inconsistencies in the reconstructed surfaces. Tables 8, 9, and 10 show the surface area coverage ability of our approach compared to the Redwood and BundleFusion systems on the augmented synthetic ICL-NUIM, SceneNN, and

BundleFusion datasets respectively. Taking advantage of the Manhattan world assumption available in the most indoor scenes and geometric planar alignment helps us to cover more areas and resolve the discontinuity problem in the planar surfaces.

4.4 Qualitative Performance

The reconstructed models from augmented ICL-NUIM and BundleFusion datasets, shown in Figs. 6, 7, and 8 demonstrate the robustness of our proposed approach for dense reconstruction of large scale indoor environments using depth sequences compared to the state-of-the-art reconstruction systems. Our method relies on the local geometric structure of the scene and local pose refinement for surface reconstruction. This precise characteristic helps to preserve the geometry, to mend gaps and discontinuities, and to cover more surface areas and generate more accurate 3D models, as shown in Fig. 9. Our system is geometry-based and does not use RGB channels. It fails on the apt2 depth sequences from BundleFusion dataset due to a significant depth gap caused by sensor occlusion.

4.5 Additional Results

We tested our framework on several real scene depth sequences. Considering the absence of the ground truth surface for the real environments, we just compared our dense models with different approaches qualitatively. For instance, Fig. 8 shows the qualitative differences between the reconstructed models by BundleFusion and ManhattanFusion on different depth sequences from BundleFusion dataset. Moreover, Fig. 11 shows 3D models generated by Redwood and ManhattanFusion on a few sequences from SceneNN dataset. The dense models created by our framework are on par with the state-of-the-art BundleFusion and Redwood systems in terms of quality. In addition, Fig. 10 demonstrates the advantage of using Manhattan keyframes in our volumetric reconstruction pipeline.

4.6 Computation Complexity

There are many factors that determine the computational complexity, among which two key ones are the kind of data used and the size of space volume covered. Most existing SLAM algorithms are based on RGB-D data which have computational advantage over those using depth data only (like ours), since RGB images can support efficient feature tracking and loop closure handling which are essential for camera pose estimation in a large environment. On the other hand, the quality of RGB images is influenced by light condition and the richness of visual features in the scene. Using depth data only has some advantages in the cases where the environment is almost featureless or has poor light conditions. However, depth data tend to be sparse and noisy which could be problematic for long-range camera pose optimization and large scale volume reconstruction. That is one of the main reasons why the famous KinectFusion works best for a relatively small volume. Our research is mainly focused on depth-based volumetric reconstruction for a large indoor environment where RGB data are not used or may not be available. Our current implementation can build the 3D volumetric model online with some delay depending the length of depth sequences and the scene complexity, and it does have potential to be real-time if it can be fully GPU-enabled. Given our current PC specification (CPU: Intel Core i7–4770 3.4 GHz; RAM: 20GB; GPU: GeForce GT 640 3GB DDR3), the delay of sequences with around 1000 frames is around 10–15 seconds, and that of those with 6000–10000 frames is around 1–2 minutes. Moreover,

our research is mainly focused on the indoor scene where the MW assumption is usually valid. In the case where the MW assumption is weak or absent, the advantage of the proposed method will diminish, and it could be reduced into a KinectFusionlike framework that may have some limitations in handling a large environment as reported in [9], [10] .

5 Conclusion

We have presented a new 3D dense surface modeling framework, ManhattanFusion, for volumetric reconstruction of extended scale indoor environments by using only depth sequences. Different from other methods that apply the Manhattan World (MW) assumption to process RBG-D data, ManhattanFusion takes advantage of the MW assumption available in an indoor scene to improve the quality of scene reconstruction with a few key characteristics. First, it implicitly handles loop closure during the reconstruction process by local Manhattan frame growing and incremental local pose optimization. Second, Manhattan keyframes (MKFs) are estimated from depth keyframes by extending the first identified Manhattan frame along the main dominant axis in the scene for each local segment. These MKFs are not only used for planar pre-alignment to initialize depth-to-model surface registration as well as the final model integration, but also serve as geometric constraints for local pose optimization. The incorporation of MKFs in the pipeline at different stages reduces the accumulative registration error and improves the accuracy and fidelity of generated models. Third, ManhattanFusion adequately handles the discontinuity problem and enhances the surface area coverage in large scale scene reconstruction. The experimental results on both synthetic and real-world depth data demonstrate the advantage of our proposed approach to reduce the accumulation of the registration error and overall geometric drift in 3D dense modeling. Our algorithm outperforms recent ones that use depth data only in terms of both the mean distance error and the absolute trajectory error, and it is also very competitive compared with RGB-D based SLAM algorithms. Our ManhattanFusion algorithm also significantly outperforms the state-of-the-art in terms of the surface area coverage, largely due to the effectiveness and usefulness of the MW assumption through the reconstruction pipeline.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions that are very helpful to improve this paper. This work is supported in part by the US National Science Foundation (NSF) Grant IIS-1427345, the US National Institutes of Health (NIH) Grant R15 AG061833 and the Oklahoma Center for the Advancement of Science and Technology (OCAST) Health Research Grant HR18–069.

Biographies

Mahdi Yazdanpour is an Assistant Professor of Mechatronics Engineering Technology in the Department of Physics, Geology and Engineering Technology at Northern Kentucky University. He received his Ph.D. in Electrical Engineering from Oklahoma State University, USA, in 2019. He obtained his M.S. degree in Industrial Engineering from Amirkabir University of Technology (Tehran Polytechnic), Iran, in 2012, and the B.S. degree in Computer Engineering from Qazvin Azad University, Iran, in 2003. His research interests include Computer Vision, 3D Dense Reconstruction, Intelligent Mechatronic Systems, Medical Robotics, Brain-Controlled Robots, Machine Vision, and Industrial Automation. He was the recipient of the CEAT Dean's Outstanding ECE Graduate Student Award from Oklahoma State University in 2018.

Guoliang Fan (S'97–M'01–SM'05) received the B.S. degree in Automation Engineering from the Xi'an University of Technology, Xi'an, China, the M.S. degree in Computer Engineering from Xidian University, Xi'an, China, and the Ph.D. degree in Electrical Engineering from the University of Delaware, Newark, DE, USA, in 1993, 1996, and 2001, respectively. He is the Cal and Marilyn Vogt Professor in Engineering with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, USA. His research interests include Image Processing, Pattern Recognition, and Computer Vision. He is an Associate Editor for the IEEE Journal of Biomedical and Health Informatics and EURASIP Journal on Image and Video Processing. He was the recipient of the Young Alumni Achievement Award from the Department of Electrical and Computer Engineering, University of Delaware in 2015.

Weihua Sheng is an Associate Professor at the School of Electrical and Computer Engineering, Oklahoma State University (OSU), USA. He is the Director of the Laboratory for Advanced Sensing, Computation and Control (ASCC Lab, http://ascc.okstate.edu) at OSU. Dr. Sheng received his Ph.D. degree in Electrical and Computer Engineering from Michigan State University in May 2002. He obtained his M.S and B.S. degrees in Electrical Engineering from Zhejiang University, China in 1997 and 1994, respectively. During 2002–2006, he taught in the Electrical and Computer Engineering Department at Kettering University (formerly General Motor Institute). He is the author of more than 200 papers in major journals and international conferences. Eight of them have won best paper or best student paper awards in major international conferences. His current research interests include Social Robots, Wearable Computing, Human Robot Interaction, and Intelligent

Transportation Systems. Dr. Sheng is a senior member of IEEE and served as an Associate Editor for IEEE Transactions on Automation Science and Engineering from 2013–2019.

References

[1]. Dai A, Nießner M, Zollöfer M, Izadi S, and Theobalt C, ”BundleFusion: Real-time globally consistent 3D reconstruction using On-the-fly surface re-integration,” In ACM TOG, vol. 36, no. 3, 2017.

[2]. Yang S, Li B, Liu M, Lai Y, Kobbelt L. and Hu S. ”HeteroFusion: dense scene reconstruction integrating multi-sensors,” In TVCG, 2019.

[3]. Newcombe RA, Davison AJ, Izadi S, Kohli P, Hilliges O, Shotton J, Molyneaux D, Hodges S, Kim D, and Fitzgibbon A, ”KinectFusion: real-time dense surface mapping and tracking,” In ISMAR, pp. 127–136, 2011.

[4]. Yazdanpour M, Fan G, and Sheng W, ”Real-time volumetric reconstruction of Manhattan indoor scenes,” In VCIP, 2017.

[5]. Izadi S, Kim D, Hilliges O, Molyneaux D, Newcombe R, Kohli P, Shotton J, Hodges S, Freeman D, Davison A, and Fitzgibbon A. ”KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera,” In UIST Proceedings of the 24th annual ACM symposium on User interface software and technology, pp. 559–568, 2011.

[6]. Whelan T, Salas-Moreno RF, Glocker B, Davison AJ, and Leutenegger S, ”ElasticFusion: real-time dense SLAM and light source estimation,” In Int. J. of Robotics Research, 2015.

[7]. Whelan T, Leutenegger S, Salas-Moreno RF, Glocker B, and Davison AJ ”ElasticFusion: dense SLAM without a pose graph,”. In RSS, 2015.

[8]. Whelan T, Kaess M, Johannsson H, Fallon M, Leonard JJ, and McDonald J, ”Real-time large scale dense RGB-D SLAM with volumetric fusion,” In Int. J. of Robotics Research, vol. 34, pp. 598–626, 2015.

[9]. Whelan T, Kaess M, Fallon M, Johannsson H, Leonard J, and Mc-Donald J,”Kintinuous: spatially extended Kinectfusion,” In RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras, 2012.

[10]. Whelan T, Johannsson H, Kaess M, Leonard J, and Mc-Donald J. ”Robust real-time visual odometry for dense RGB-D mapping,” In ICRA, 2013.

[11]. Whelan T, McDonald J, Kaess M, and Leonard JJ ”Deformation-based loop closure for large scale dense RGB-D SLAM,” In IROS, 2013.

[12]. Zhang H. and Xu F. ”MixedFusion: Real-Time reconstruction of an indoor scene with dynamic objects,” In TVCG, vol. 24, no. 12, pp. 3137–3146, 2018. [PubMed: 29990141]

[13]. Nießner M, Zollofer M, Izadi S, and Stamminger M, ”Real-time 3D reconstruction at scale using voxel hashing,” In ACM TOG, vol. 32, no. 6, 2013.

[14]. Chen J, Bautembach D, and Izadi S. ”Scalable real-time volumetric surface reconstruction,” In ACM TOG, vol. 32, no 4, 2013.

[15]. Zhou Q-Y and Koltun V. ”Dense scene reconstruction with points of interest,” In ACM TOG, vol. 32, no. 4, 2013.

[16]. Zhou Q-Y, Miller S, and Koltun V. ”Elastic fragments for dense scene reconstruction,” In CVPR, pp. 473–480, 2013.

[17]. Yazdanpour M, Fan G, and Sheng W, ”Online Manhattan keyframe-based dense reconstruction from indoor depth sequences,” In VCIP, 2019.

[18]. Keller M, Lefloch D, Lambers M, Izadi S, Weyrich T, and Kolb A. ”Real-time 3D reconstruction in dynamic scenes using point-based fusion,” In 3DV, pp. 1–8, 2013.

[19]. Kerl C, Sturm J, and Cremers D. ”Dense visual SLAM for RGB-D cameras,” In IROS, 2013.

[20]. Xiao J, Owens A, and Torralba A. ”SUN3D: a database of big spaces reconstructed using SfM and object labels,” In ICCV, 2013.

[21]. Choi S, Zhou Q, and Koltun V. ”Robust reconstruction of indoor scenes,” In CVPR, 2015.

[22]. Yaguchi H, Takaoka Y, Yamamoto T, and Inaba M. ”A method of 3D model generation of indoor environment with Mmanhattan world assumption using 3D camera,” In IEEE/SICE Int. Symposium on System Integration, 2013.

[23]. Wolters D. ”Automatic 3D reconstruction of indoor Manhattan world scenes using Kinect depth data,” In GCPR, 2014.

[24]. Henry P, Krainin M, Herbst E, Ren X, and Fox D. ”RGB-D Mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments,” In Int. J. Robotics Research, vol. 31, no. 5, pp. 647–663, 2012.

[25]. Henry P, Fox D, Bhowmik A, and Mongia R. ”Patch volumes: Segmentation-based consistent mapping with RGB-D cameras,” In 3DV, 2013.

[26]. Triggs B, McLauchlan P, Hartley R, and Andrew Fitzgibbon. ”Bundle adjustment—a modern synthesis,” In Vision Algorithms: Theory and Practice, pp. 298–372, 2000.

[27]. Kummerle R, Grisetti G, Strasdat H, Konolige K, and Burgard W. ”g^2o: A general framework for graph optimization,” In ICRA, 2011.

[28]. Wang L. and Wu Z. ”RGB-D SLAM with Manhattan frame estimation using orientation relevance,” In Sensors, vol. 19, no. 5, 2019.

[29]. Endres F, Hess J, Engelhard N, Sturm J, Cremers D, and Burgard W. ”An evaluation of the RGB-D SLAM system,” In ICRA, pp. 1691–1696, 2012.

[30]. Endres F, Hess J, Sturm J, Cremers D, and Burgard W. ”3-D mapping with an RGB-D camera,” In IEEE Transactions on Robotics, vol. 30, no. 1, 2014.

[31]. Stuckler J. and Behnke S. ”Multi-resolution surfel maps for efficient dense 3D modeling and tracking,” In Visual Communication Image Representation, vol. 25, no. 1, pp. 137–147, 2014.

[32]. Maier R, Sturm J, and Cremers D. ”Submap-based bundle adjustment for 3D reconstruction from RGB-D data,” In GCPR, 2014.

[33]. Steinbrucker F, Kerl C, and Cremers D. ”Large-scale multiresolution surface reconstruction from RGB-D sequences,” In ICCV, 2013.

[34]. Handa A, Whelan T. McDonald J. and Davison AJ ”A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM,” In ICRA, 2014.

[35]. Sturm J, Engelhard N, Endres F, Burgard W. and Cremers D. ”A benchmark for the evaluation of RGB-D SLAM systems,” In IROS, 2012.

[36]. Ghanem B, Thabet A, Niebles JC and Heilborn FC ”Robust Manhattan frame estimation from a single RGB-D image,” In CVPR, 2015.

[37]. Joo K, Oh T-H, Kim J. and Kweon IS ”Globally optimal Manhattan frame estimation in real-time,” In CVPR, 2016.

[38]. Furukawa Y, Curless B, Seitz SM and Szeliski R. ”Manhattan-world stereo,” In CVPR, 2009.

[39]. Coughlan JM and Yuille AL, ”Manhattan world: Compass direction from a single image by Bayesian inference,” In ICCV, 1999.

[40]. Heredia F. and Favier R. ”Using Kinfu large scale to generate a textured mesh,” 2012. [Online]. Available: http://pointclouds.org/documentation/tutorials/usingkinfulargescale.php

[41]. Girardeau-Montaut D. ”CloudCompare: 3D point cloud and mesh processing software Open Source Project,” 2015. [Online]. Available: http://cloudcompare.org

[42]. Yazdanpour M, Fan G, and Sheng W, ”Online reconstruction of indoor scenes With local Manhattan frame growing,” In CVPR Workshops, 2019.

[43]. Roth H. and Vona M. ”Moving volume KinectFusion,” In BMVC, 2012.

[44]. Choi W, Chao Y-W, Pantofaru C. and Savarese S. ”Understanding indoor scenes using 3D geometric phrases,” In CVPR, 2013.

[45]. Hua B-S, Pham Q-H, Nguyen DT, Tran M-K, Yu L-F and Yeung -K ”SceneNN: a scene meshes dataset with aNNotations,” In 3DV, 2016.

[46]. Golodetz S, Cavallari T, Lord NA, Prisacariu VA, Murray DW and Torr PHS, ”Collaborative Large-Scale Dense 3D Reconstruction with Online Inter-Agent Pose Optimisation,” In TVCG, vol. 24, no. 11, pp. 2895–2905, 2018. [PubMed: 30334761]

[47]. Wang H, Wang J, and Wang L. ”Online reconstruction of indoor scenes from RGB-D streams,” In CVPR, 2016.

[48]. Agarwal S, Mierle K, and Others. 2013. ”Ceres Solver,” 2013. [Online]. Available: http://ceres-solver.org

[49]. Dong W, Wang Q, Wang X, and Zha H. ”PSDF fusion: probabilistic signed distance function for on-the-fly 3D data fusion and scene reconstruction,” In ECCV, 2018.

[50]. Zollhofer M, Stotko P, orlitz AG, Theobalt C, Nießner M, Klein R, and Kolb Andreas ”State of the art on 3D reconstruction with RGB-D cameras,” In J. of Computer Graphics Forum, vol. 37, pp. 625–652, 2018.

[51]. Engel J, Schops T, and Cremers D. ”LSD-SLAM: large-scale direct monocular SLAM,” In ECCV, 2014.

[52]. Zhou Q-Y and Koltun V. ”Simultaneous localization and calibration: self-calibration of consumer depth cameras,” In CVPR, 2014.

[53]. Huang J, Dai A, J Guibas L, Nießner M. ”3Dlite: towards commodity 3D scanning for content creation,” In ACM Trans. Graph, Vol. 36, No. 6, pp203–1, 2017.

[54]. Halber M, Funkhouser T. ”Fine-to-coarse global registration of RGB-D scans,” In CVPR, 2017.

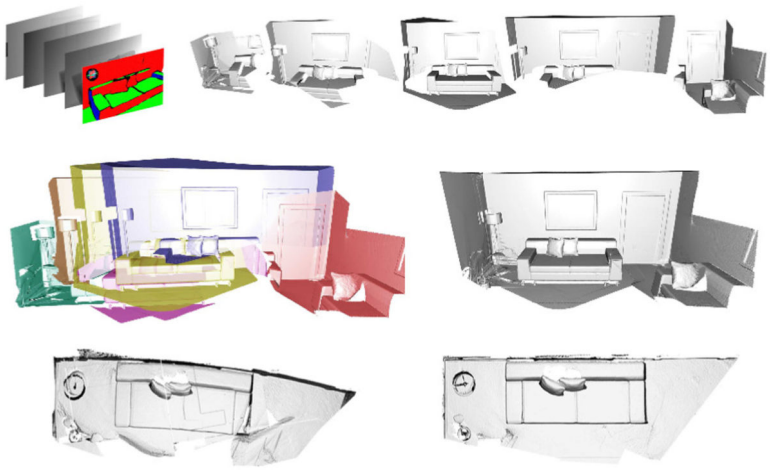

Fig. 1.

The 1st row shows (from left to right) a depth sequence with a colored Manhattan keyframe (MKF) and multiple local segments. The 2nd row shows the registration result of multiple local segments along the dominant planes (colored) in each segment and the final reconstructed model. The 3rd row presents the top-down view of the reconstruction model without using MKF and the one generated by ManhattanFusion.

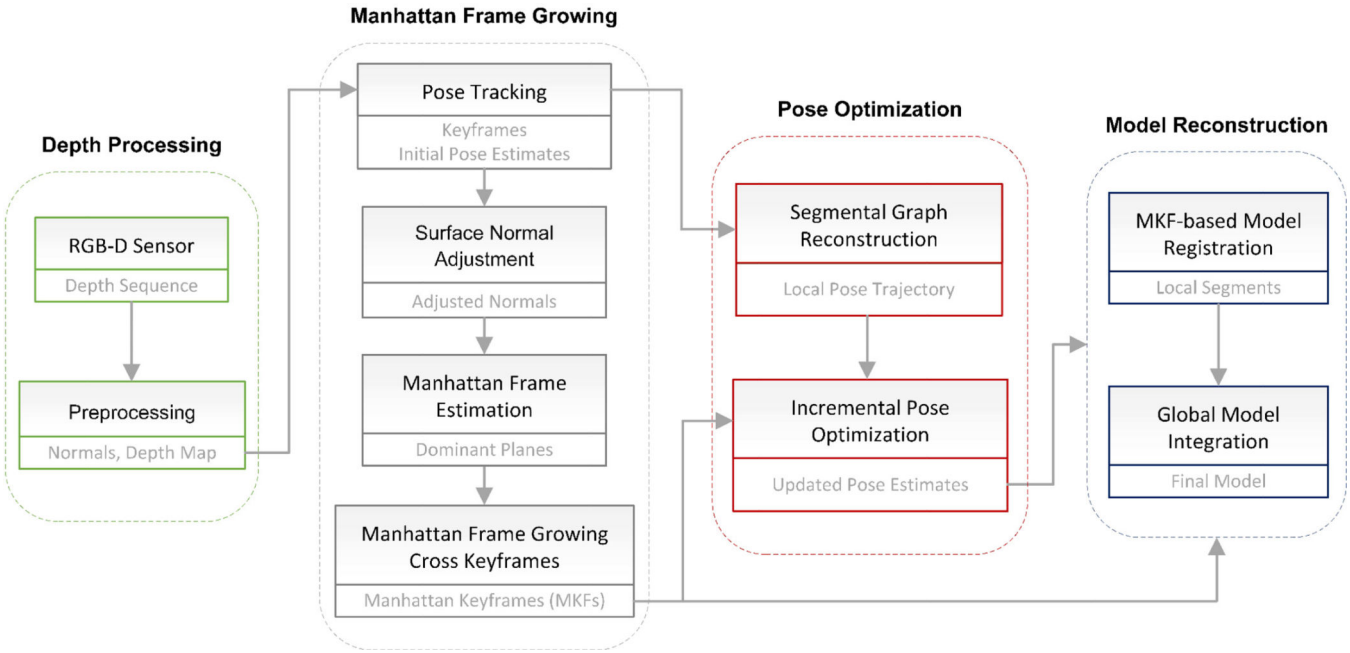

Fig. 2.

The general overview of our proposed framework. The top part shows the process and the bottom part shows the output. The ManhattanFusion estimates the Manhattan keyframes using a Manhattan frame growing scheme over depth keyframes after a normal surface adjustment and use it as a reliable initial planar alignment in a keyframe to model registration system to create the local segments using the optimized pose estimates. The final volumetric model will be reconstructed by integrating the local fused models into a global framework.

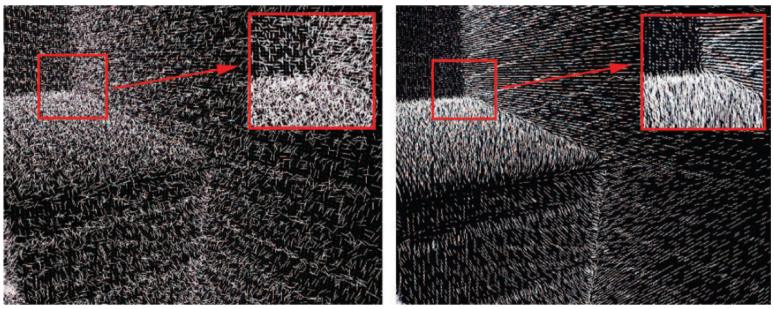

Fig. 3.
Surface normals before (left) and after (right) adjustment.

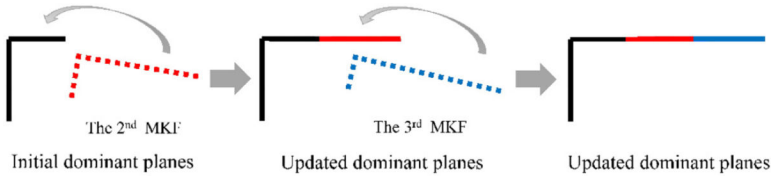

Fig. 4.

A 2D illustration of Manhattan frame growing over two MKFs where the initial dominant plane is extended over two adjacent Manhattan keyframes (MKFs).

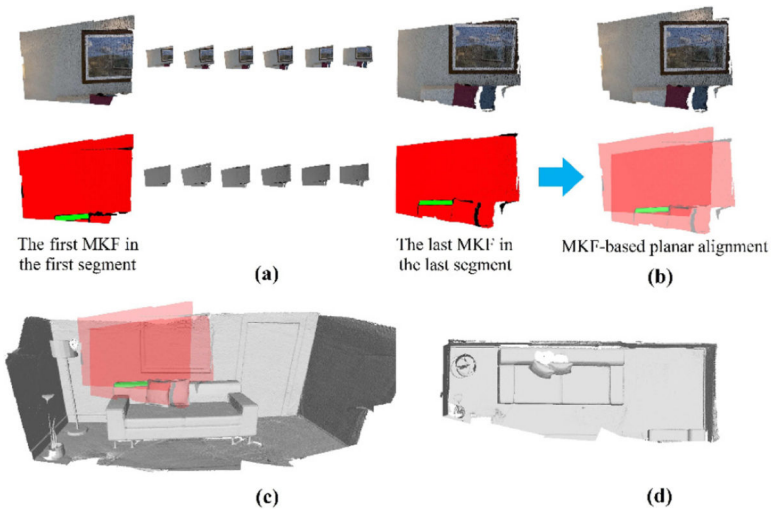

Fig. 5.

(a) Given two MKFs belong to two local segments in a loop closure, the dominant planes (red and green) are extracted in each MFK for planar alignment. (b) Point-to-plane surface registration in the overlapping parts. (c) A top-down view of the loop closure portion.

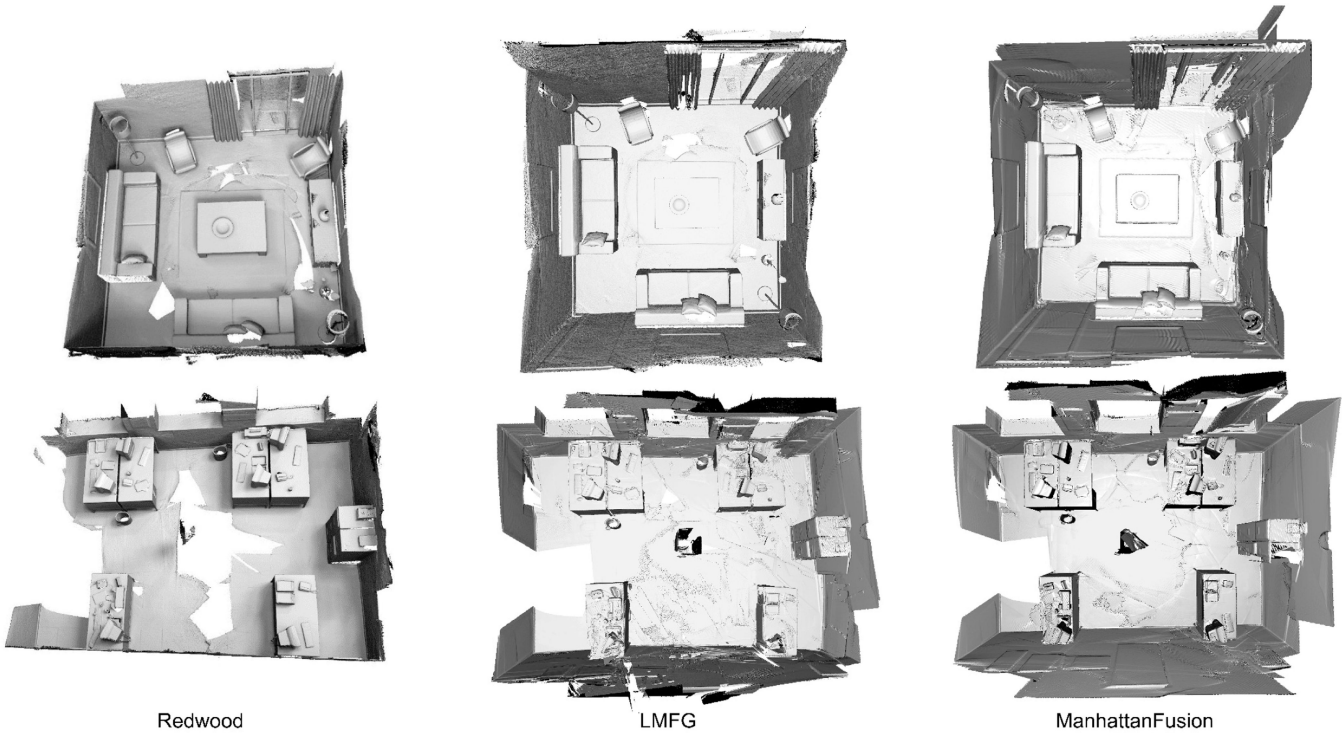

Fig. 6.

The reconstructed models of Living Room 2 (above) and Office 2 (below) by offline Robust Reconstruction (Redwood) [21], Local Manhattan Frame Growing (LMFG) [42], and ManhattanFusion.

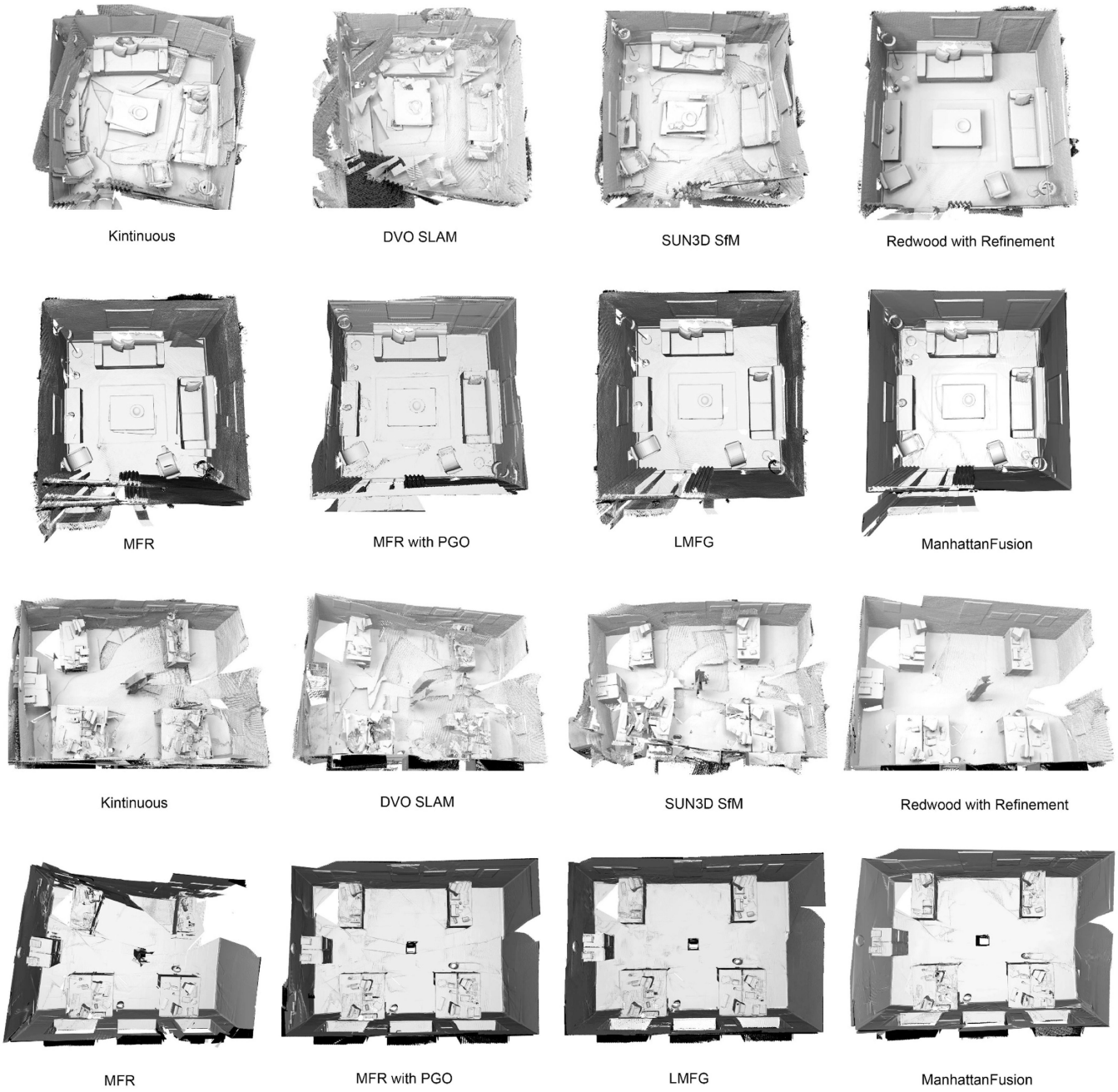

Fig. 7.

The reconstructed models of Living Room 1 (above) and Office 1 (below), by Kintinuous [9], DVO SLAM [19], SUN3D SfM [20], Offline Robust Reconstruction (Redwood) with an optional refinement [21], Manhattan Frame Reconstruction (MFR) [4], MFR with global pose optimization [17], Local Manhattan Frame Growing (LMFG) [42], and ManhattanFusion.

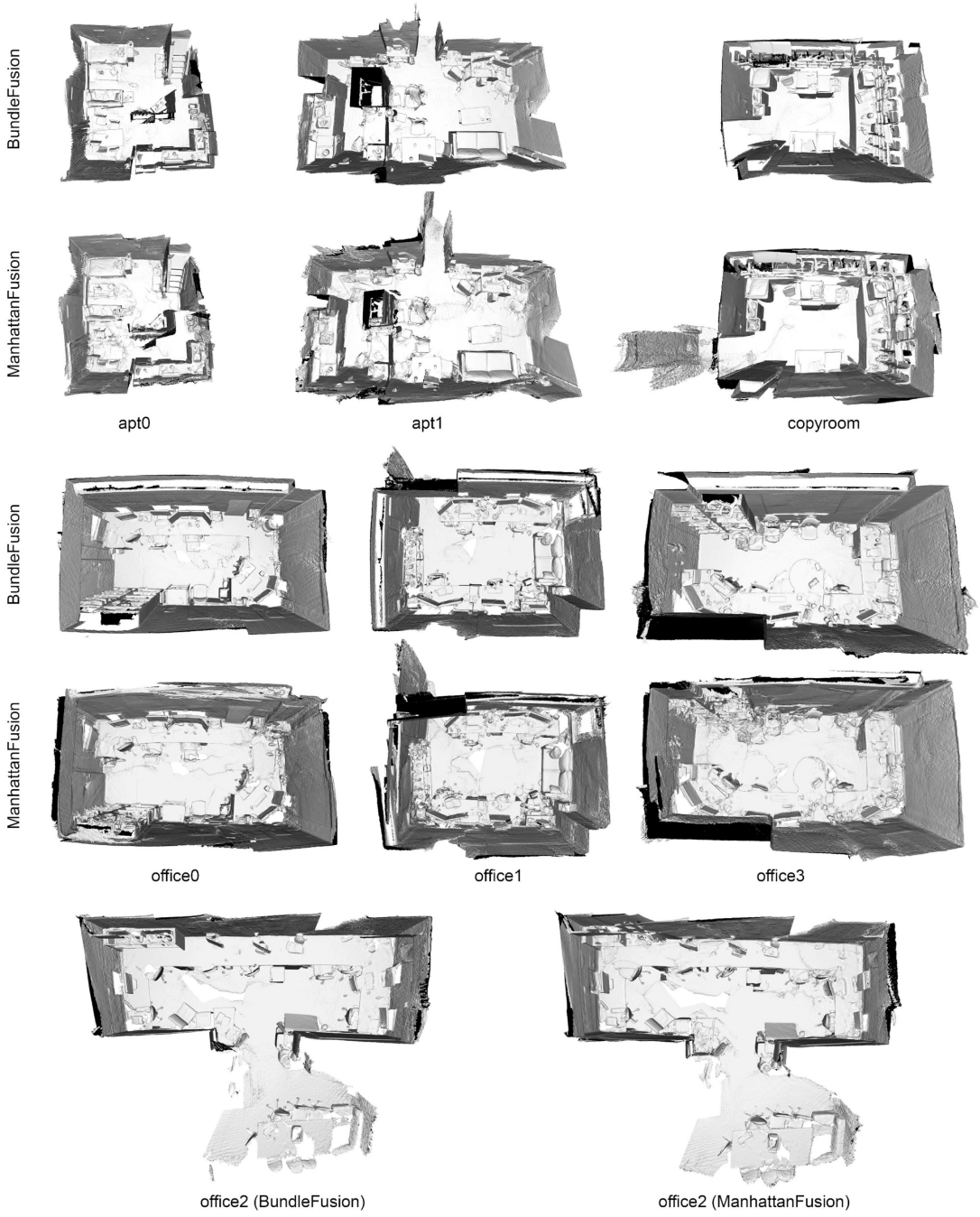

Fig. 8.

The reconstructed models by BundleFusion and our approach on the BundleFusion dataset. The generated models by ManhattanFusion are on par with BundleFusion approach without using visual features and explicit loop closure detection.

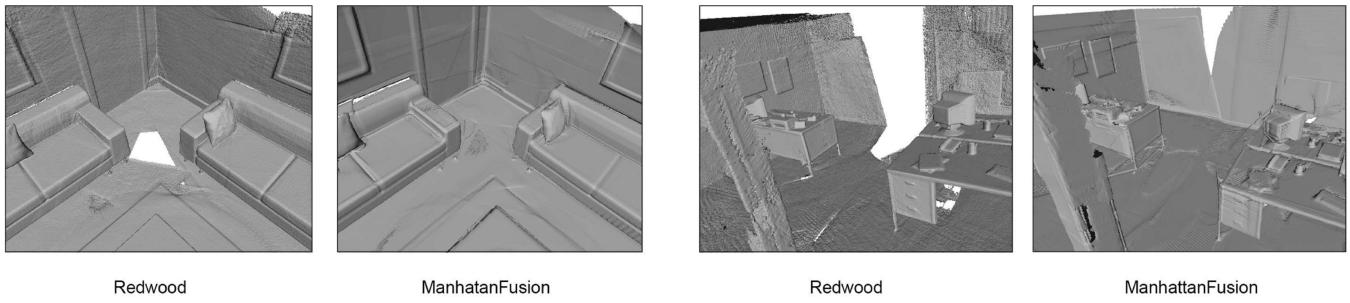

Fig. 9.

The reconstructions created by offline robust reconstruction (Redwood) and our approach from Living Room 2 (left) and Office 2 (right). Our method preserves the local geometric structure of the planar surfaces in the scene.

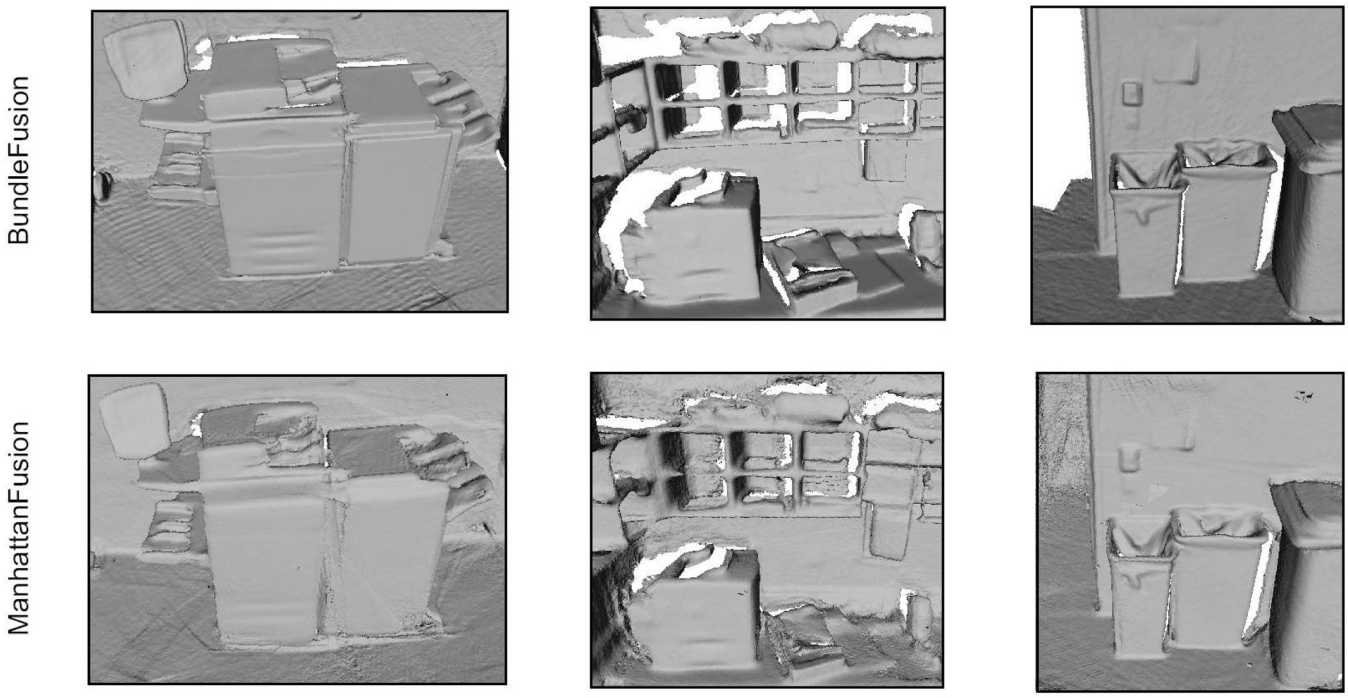

Fig. 10.

The reconstructions created by BundleFusion and our approach from *copyroom*. Our method preserves the local geometric structure of the planar surfaces in the scene.

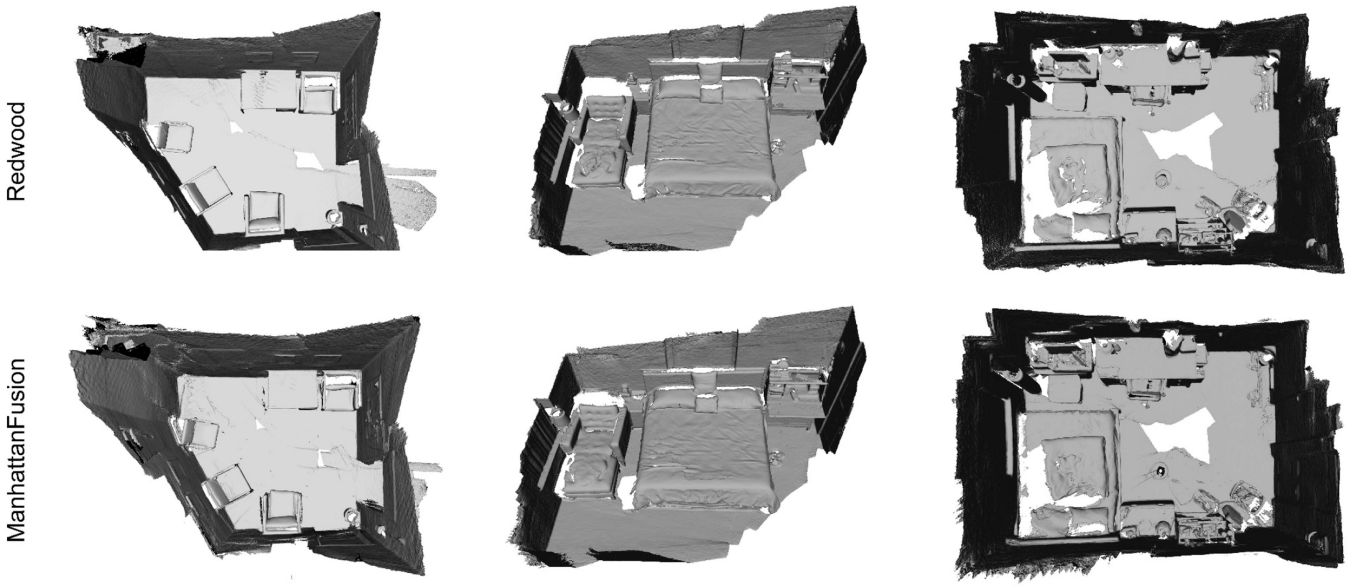

Fig. 11.

The reconstructed models created by offline Robust Reconstruction (Redwood) and our approach on SceneNN dataset. From left to right: Generated models from scenes 11, 16, and 21. Our results are on par with the reconstructions generated by this offline approach.

TABLE 1

f the state-of-the-art in 3D mapping and volumetric reconstruction systems.

	Run Mode			Input		Map Representation			Tracking			Data Structure				Data Association				Pose Correction		
	Real-time	Online	Offline	RGB	Depth	TSDF	Surfel	Points	Frame-to-Frame	Frame-to-Model	Global Optimization	Dense Grid	Sparse Hash	Sparse Octree	Sparse Points	Point-to-Point	Point-to-Plane	Feature Points	Dense Color	Local	Global	Loop Closure
n	✓				✓	✓				✓		✓					✓					
9]	✓				✓	✓				✓		✓					✓	✓				
2	✓				✓	✓				✓		✓					✓	✓			✓	✓
I	✓			✓	✓			✓	✓						✓		✓				✓	✓
I		✓		✓	✓			✓	✓						✓	✓		✓				✓
ng	✓			✓	✓		✓		✓						✓		✓	✓			✓	✓
on	✓				✓	✓				✓				✓			✓					
est			✓		✓	✓					✓	✓					✓	✓			✓	✓
on	✓			✓	✓	✓			✓					✓			✓		✓			✓
]	✓			✓	✓		✓			✓				✓			✓		✓			✓
	✓				✓		✓			✓					✓		✓					
g	✓				✓	✓				✓			✓				✓					
6]			✓		✓	✓					✓	✓					✓	✓			✓	✓
n	✓			✓	✓		✓			✓					✓		✓		✓		✓	✓
]			✓		✓	✓					✓	✓				✓					✓	✓
n	✓			✓	✓	✓					✓		✓				✓	✓	✓	✓	✓	
		✓			✓	✓				✓		✓					✓			✓		

IEEE Trans Vis Comput Graph. Author manuscript; available in PMC 2023 July 01.

	Run Mode			Input		Map Representation			Tracking			Data Structure				Data Association				Pose Correction		
	Real-time	Online	Offline	RGB	Depth	TSDF	Surfel	Points	Frame-to-Frame	Frame-to-Model	Global Optimization	Dense Grid	Sparse Hash	Sparse Octree	Sparse Points	Point-to-Point	Point-to-Plane	Feature Points	Dense Color	Local	Global	Loop Closure
Kintinuous		✓			✓	✓				✓							✓			✓		

TABLE 2

Mean distance of the reconstructed models to the ground-truth surface (in centimeters) on the ICL-NUIM dataset.

	Input	kt0	kt1	kt2	kt3	Average
DVO SLAM [19]	RGB-D	3.2	6.1	11.9	5.3	6.6
RGB-D SLAM [29]	RGB-D	4.4	3.2	3.1	16.7	6.9
MRSMap [31]	RGB-D	6.1	14	9.8	24.8	13.7
ElasticFusion [6], [7]	RGBD	0.7	0.7	0.8	2.8	1.3
BundleFusion [1]	RGB-D	0.5	0.6	0.7	0.8	0.7
Redwood [21]	D+RGB	2.0	2.0	1.3	2.2	1.9
Kintinuous [9]	D	1.1	0.8	0.9	24.8	6.9
ManhattanFusion	D	1.4	1.1	1.7	2.2	1.6

TABLE 3

Mean distance of the reconstructed models to the ground-truth surface (in centimeters) on the ICL-NUIM dataset.

	Input	Living Room1	Living Room2	Office1	Office2	Average
DVO SLAM [19]	RGB-D	21.0	6.0	11.0	10.0	12.0
SUN3D SfM [20]	RGB-D	9.0	7.0	13.0	9.0	9.5
Redwood [21]	D+RGB	4.0	7.0	3.0	4.0	4.5
Kintinuous [9]	D	22.0	14.0	13.0	13.0	15.5
MFR [4]	D	11.3	9.1	12.8	17.0	12.6
MFR + PGO [17]	D	7.2	7.6	4.7	6.5	6.5
LMFG [42]	D	6.3	7.3	2.8	5.2	5.4
ManhattanFusion	D	4.7	6.6	2.7	4.8	4.7

TABLE 4

Absolute Trajectory Error of reconstructed models to the ground-truth trajectory (in centimeters) on the ICL-NUIM dataset.

	Input	kt0	kt1	kt2	kt3	Average
DVO SLAM [19]	RGB-D	10.4	2.9	19.1	15.2	11.9
RGB-D SLAM [29]	RGB-D	2.6	0.8	1.8	43.3	12.1
MRSMap [31]	RGB-D	20.4	22.8	18.9	109	42.8
ElasticFusion [6], [7]	RGB-D	0.9	0.9	1.4	10.6	3.5
BundleFusion [1]	RGB-D	0.6	0.4	0.6	1.1	0.7
Redwood [21]	D+RGB	25.6	3.0	3.3	6.1	9.5
Kintinuous [9]	D	7.2	0.5	1.0	35.5	11.1
ManhattanFusion	D	2.0	0.9	7.6	3.7	3.5

TABLE 5

Absolute Trajectory Error of the reconstructed models to the ground-truth trajectory (in centimeters) on the ICL-NUIM dataset.

	Input	Living Room1	Living Room2	Office1	Office2	Average
DVO SLAM [19]	RGB-D	102.0	14.0	11.0	11.0	34.5
SUN3D SfM [20]	RGB-D	21.0	23.0	24.0	12.0	20.0
ElasticFusion [6], [7]	RGB-D	62.0	37.0	13.0	13.0	31.2
BundleFusion [1]	RGB-D	0.6	0.5	15.3	1.4	4.4
Redwood [21]	D+RGB	10.0	13.0	6.0	7.0	9.0
Kintinuous [9]	D	27.0	28.0	19.0	26.0	25.0
ManhattanFusion	D	8.1	11.6	5.0	6.2	7.7

TABLE 6

Absolute Trajectory Error of the reconstructed models to the ground-truth surface (in centimeters) on the TUM RGB-D dataset. BundleFusion uses a robust tracking system for relocalization using a local-to-global hierarchical optimization per RGB-D frames.

	Input	fr1/desk	fr2/xyz	fr3/office	fr3/nst	Average
DVO SLAM [19]	RGB-D	2.1	1.8	3.5	1.8	2.3
RGB-D SLAM [29]	RGB-D	2.3	0.8	3.2	1.7	2.0
MRSMap [31]	RGB-D	4.3	2.0	4.2	201.8	53.1
Submap BA [32]	RGB-D	2.2	-	3.5	-	2.9
LSD-SLAM [51]	RGB-D	-	1.5	-	-	1.5
ElasticFusion [6], [7]	RGB-D	2.0	1.1	1.7	1.6	1.6
BundleFusion [1]	RGB-D	1.6	1.1	2.2	1.2	1.5
Redwood [21]	D+RGB	2.7	9.1	3.0	192.9	51.9
Kintinuous [9]	D	3.7	2.9	3.0	3.1	3.1
Voxel Hashing [13]	D	2.3	2.2	2.3	8.7	3.9
ManhattanFusion	D	2.3	1.3	2.1	-	1.9

TABLE 7

Absolute Trajectory Error of the reconstructed models to the ground-truth surface (in centimeters) on the TUM RGB-D dataset.

	Input	fr1/360	fr1/floor	fr3/long office household	Average
RGB-D SLAM [29]	RGB-D	10.3	6.1	8.2	8.2
MF RGB-D SLAM [28]	RGB-D	8.2	5.4	5.2	6.3
ManhattanFusion	D	7.1	3.5	4.6	5.3

TABLE 8

Surface Area Coverage of the reconstructed models on the Augmented Synthetic ICL-NUIM dataset.

	Input	Living Room1	Living Room2	Office1	Office2
Redwood [21]	D+RGB	141.78	92.98	150.42	92.95
ManhattanFusion	D	104.95	99.95	156.89	139.53

TABLE 9

The comparison of surface area coverage on the SceneNN dataset.

	Input	Scene 11	Scene 16	Scene 21	Scene 700
Redwood [21]	RGB-D	65.78	31.1	54.54	83.84
ManhattanFusion	D	70.58	33.19	54.17	114.11

TABLE 10

The comparison of surface area coverage on the BundleFusion dataset.

	BundleFusion [1]	ManhattanFusion
apt0	103.54	105.38
apt1	84.78	101.9
copyroom	48.39	55.72
office0	79.05	82.63
office1	81.62	101.88
office2	89.71	95.62
office3	83.67	92.61