Real-Time Globally Consistent 3D Reconstruction with Semantic Priors

Shi-Sheng Huang, Haoxiang Chen, Jiahui Huang, Hongbo Fu, and Shi-Min Hu*, *Senior Member, IEEE*

Abstract— Maintaining global consistency continues to be critical for online 3D indoor scene reconstruction. However, it is still challenging to generate satisfactory 3D reconstruction in terms of global consistency for previous approaches using purely geometric analysis, even with bundle adjustment or loop closure techniques. In this paper, we propose a novel real-time 3D reconstruction approach which effectively integrates both semantic and geometric cues. The key challenge is how to map this *indicative* information, i.e. semantic priors, into a *metric space* as *measurable* information, thus enabling more accurate semantic fusion leveraging both the geometric and semantic cues. To this end, we introduce a semantic space with a *continuous* metric function measuring the distance between discrete semantic observations. Within the semantic space, we present an accurate frame-to-model semantic tracker for camera pose estimation, and semantic pose graph equipped with semantic links between submaps for globally consistent 3D scene reconstruction. With extensive evaluation on public synthetic and real-world 3D indoor scene RGB-D datasets, we show that our approach outperforms the previous approaches for 3D scene reconstruction both quantitatively and qualitatively, especially in terms of global consistency.

Index Terms—3D Reconstruction, Semantic Fusion, Semantic Tracker, Semantic Pose Graph.

1 Introduction

Accurate 3D scene reconstruction and understanding is a fundamental research topic, which benefits a wide range of applications such as intelligent robotics, virtual or augmented reality, and computer games etc., and thus has been receiving continuous research attention in the past decades. Most of early works have focused on either 3D reconstruction [1], [2], [3], [4], [5], [6], [7], [8], [9] or 3D semantic segmentation [10], [11], [12], [13], [14], [15], [16], [17] separately. Recently, the joint analysis of 2D semantics and 3D geometry has been introduced to improve the 3D semantic segmentation in the category level [18], [19], [20] or instance level [21]. However, the problem of how to boost the 3D reconstruction quality using both the geometry and semantic priors remains to be explored for real-time 3D scene reconstruction, though the fusion of both two priors offers great potentials [22].

For a real-time 3D reconstruction system, there are three main aspects that influence the final 3D reconstruction quality, including effective 3D scene representation, precise depth fusion, and accurate camera pose estimation. The implicit function (truncated signed distance function, i.e., TSDF [23]) and recent neural implicit functions (e.g., DeepSDF [24], Convolutional Occupancy Networks [25], DeepLS [26]) provide effective representations for heterogeneous 3D objects or scenes, and play a fundamental role for the *complete* and *detailed* 3D reconstruction. On the other hand, the state-of-the-art depth fusion approaches (e.g., RoutedFusion [27], NeuralFusion [28]) contribute on precise depth fusion mechanisms (given camera poses), and achieve impressively high-fidelity 3D reconstruction quality. But for accurate reconstruction of a whole 3D scene in a global manner, there are continuous pursuits [1], [3], [5], [6], [8], [29], [30] for accurate camera pose estimation, aiming at globally consistent 3D reconstruction.

The previous real-time 3D reconstruction approaches, such as KinectFusion [1], VoxelHashing [3], BundleFusion [6] etc., are still unable to generate satisfactory reconstruction of globally consistent 3D scenes, especially for cluttered 3D scenes with texture-less objects or challenging lighting conditions. The main drawback of these approaches is that they only perform ego-motion (or bundle adjustment) with geometric cues, such as sparse [31], [32] or dense [1], [3], [5] landmarks. Although they achieve impressive results, the pure geometric information limits the capability of high-quality data association between heterogeneous RGB-D scans, thus potentially causing drift in camera pose estimation. Without high-quality data association, the global drift introduced by the camera pose estimation could not be rectified even with bundle adjustment [6] or loop closure [29] techniques.

The semantic priors provide essential description of 3D scene contents, and thus could be potentially used for accurate camera pose estimation as shown by visual SLAM techniques [33], [34], [35]. However, the loosely coupled usage of semantic priors as data association guidance for feature points [34] or instance landmarks [33], [35] is not suitable for online 3D reconstruction, since these data associations could be sensitive for the camera pose estimation in the frame-to-frame or frame-to-model ICP registration [34], [35] framework. The problem is that the semantic information is *indicative* but not *quantitative*, thus causing difficulty to directly use such information for camera pose estimation. An appropriate mapping that transforms the indicative se-

- *Manuscript received XX XX 2021; revised XX XX, 2021.*
- **Corresponding author.*
- *Shi-Sheng Huang is with School of Artificial Intelligence, Beijing Normal University. Email: huangss@bnu.edu.cn*
- *Haoxiang Chen, Jiahui Huang, and Shi-Min Hu are with BNRist, Department of Computer Science and Technology, Tsinghua University. Email: {chx20,huang-jh18}@mails.tsinghua.edu.cn, shimin@tsinghua.edu.cn.*
- *Hongbo Fu is with the School of Creative Media, City University of Hong Kong, Hong Kong, China. Email: hongbofu@cityu.edu.hk.*

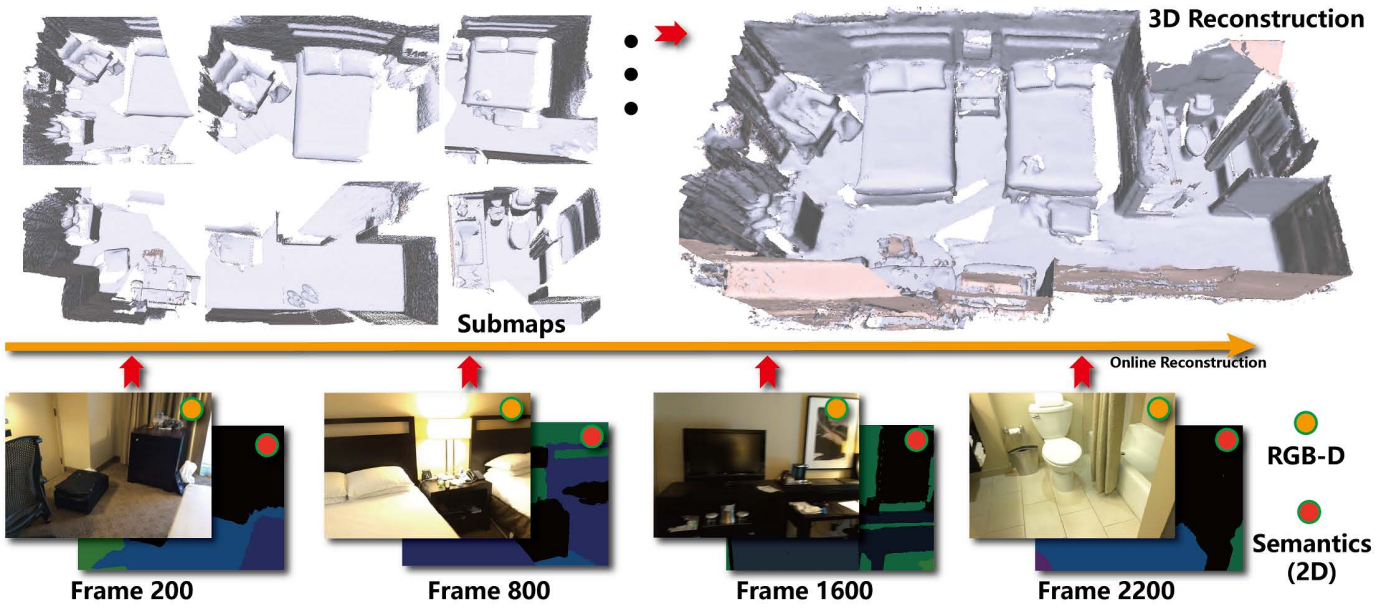


Fig. 1. We propose a new online 3D reconstruction approach which tightly uses the semantic priors together with RGB-D frames for globally consistent surface reconstruction. Our approach takes the RGB-D frames with 2D category labels (inferred by 2D CNNs, Bottom rows) as input, fuses them into a serials of submaps (Top-Left), and achieves a 3D reconstructed scene in a globally consistent manner (Top-Right). The semantic priors help improve both the camera pose estimation and the submaps' global pose correction. The data in this example is from the scene435 sequence of the ScanNet dataset.

mantic priors into a descriptive metric space as *measurable* priors is thus essentially needed, such that the semantic priors can be tightly integrated for globally consistent 3D reconstruction. However, such a mapping still remains unexplored by previous works yet.

In this paper, we propose a novel approach to convert the semantic priors into quantitative information within a metric space, i.e., semantic space, in which a *continuous* metric function is defined to measure the distance between discrete semantic observations. Within the semantic space, we propose a frame-to-model semantic tracker, which tightly incorporates both the geometric and semantic cues for an accurate camera pose estimation. In the back-end, we manage the 3D reconstruction with semantic submaps and build a global pose graph with reliable semantic links computed by semantic registration. This pose graph helps rectify the global drift between submaps effectively for globally consistent 3D reconstruction.

Benefiting from the compact use of semantic priors, we show that our approach outperforms previous approaches, evaluated on the public 3D indoor scene RGB-D datasets including both a synthetic dataset (SceneNetRGB-D [36]) and real-world scan datasets (ScanNet [22] and SceneNN [37]), in terms of both quantity and quality for globally consistent 3D reconstruction. To our best knowledge, we are the first to contribute such a real-time 3D reconstruction approach tightly coupling the geometric and semantic priors for globally consistent surface reconstruction, as shown in Fig. 1. Besides, our tightly-coupled multi-modal fusion enables 25fps processing rate with concise 2D semantics instead of time-consuming instance prediction [35], [38] for accurate camera pose estimation. Overall, our approach achieves globally more consistent 3D reconstruction results than previous approaches, and thus becomes a new state-of-the-art real-time 3D reconstruction approach. We summarize our main contributions as:

1) We introduce semantic space, which gives a *continuous* metric to precisely measure the distance between discrete semantic observations.
2) By tightly coupling the geometry and semantic priors, we provide a real-time 3D reconstruction approach, which relies on the semantic TSDF tracker for accurate camera tracking and a semantic pose graph for globally consistent 3D reconstruction.

2 Related Work

2.1 Real-time 3D Reconstruction

Real-time 3D reconstruction has achieved much progress since the pioneer work of KinectFusion [1]. There are two kinds of 3D scene representations for current mainstream 3D reconstruction approaches, i.e., TSDF on volumetric voxels [23] and discretized surfels [29], [39]. For TSDF-based approaches, VoxelHash [3] and its variations [40] introduce efficient a sparse voxel allocation mechanism, making it feasible to reconstruct large-scale 3D scenes. The subsequent approaches such as global pose graph (InfiniTAM [5]) and bundle adjustment (BundleFusion [6], Noise-Resilient Fusion [8]), focusing on reconstructing 3D scenes in a global manner. For surfel-based approaches, a deformable loop closure technique [29] has also been introduced to rectify both the camera pose estimation drift and the surfel representation for consistent 3D reconstruction.

The main drawback of the previous real-time 3D reconstruction approaches is that they perform the camera pose estimation based on geometry cues only, thus limiting the improvement of reconstruction quality especially in terms of

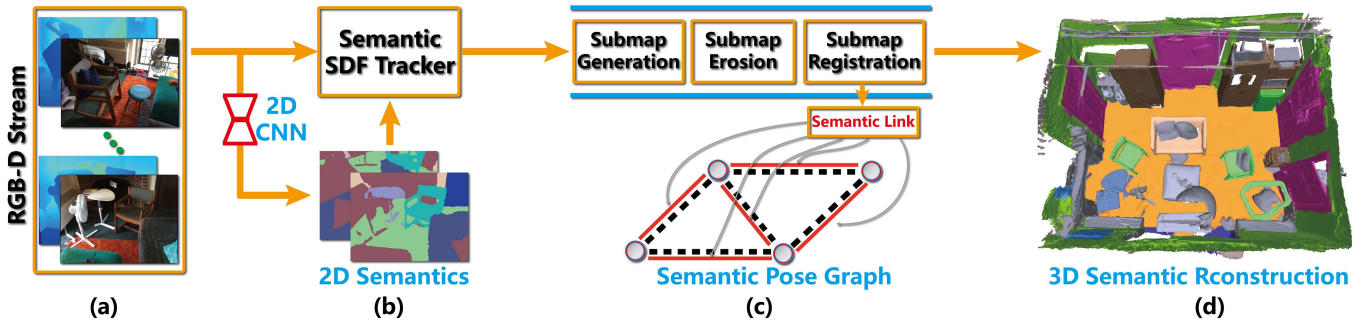


Fig. 2. Overview of our approach. Given an RGB-D data stream (a), our approach first estimates the camera poses using our semantic tracker (b), which takes a compact use of semantic priors for accurate camera tracking. To achieve globally consistent reconstruction, we build a semantic pose graph (c) in the back-end to further rectify the submaps' global poses for globally consistent 3D reconstruction as well as 3D semantics, i.e. 3D semantic reconstruction (d).

global consistency. To address this issue, our approach advocates semantic priors to the real-time 3D reconstruction, aiming at more accurate camera pose estimation in global consistency.

2.2 Deep 3D Representation and Reconstruction

The current widely used 3D scene representation, i.e., implicit function on volumetric voxels like TSDF [23], is still redundant in memory storage and ineffective in geometry prior representation, thus often leading to a heavy 3D reconstruction system. With the huge progress of deep geometry learning [41], DeepSDF [24] was proposed as a neural implicit function for 3D shape representation, which enables effective single-view 3D reconstruction and shape interpolation. DeepLS [26] and LIG [42] encode the complex geometry priors in local shapes or local grids, and thus enhance the ability to reconstruct complex objects or scenes. Convolutional Occupancy Network [25] relies on a more flexible neural implicit representation by combining the convolutional encoders and implicit occupancy decoders together, thus providing high-fidelity reconstruction of objects or large scale 3D scenes. DI-Fusion [30] is one of the first approaches to leverage a deep 3D representation (i.e., PLIVox) for online 3D reconstruction, and achieves impressive 3D reconstruction results. One the other hand, the recent works of RoutedFusion [27], NeuralFusion [28] introduce a precise depth fusion mechanism using deep neural networks, and accurately integrate the *noisy* depth data into a high-fidelity 3D surface. Explicit structural priors can be applied [43] to enable reconstruction from sparse views.

Different from these previous works, which aim at expressive deep 3D representations or depth fusion, our work aims at producing consistent 3D reconstruction in a global manner and contributes to accurate camera pose estimation by tightly fusing both geometry and semantic priors.

2.3 Semantic SLAM

Our work is also relevant to the techniques of visual SLAM (Simultaneous Localization and Mapping). Visual SLAM techniques have a long research history with many popular works. We focus on visual SLAM approaches using semantics and refer readers to [44] for an insightful survey of visual SLAM progress in the past few decades.

SLAM++ [33] was proposed for the first time to utilize object priors to detect object landmarks for camera tracking, though object priors are simply obtained by retrieving from a set of manually collected 3D shapes. The subsequent works such as Fusion++ [35] and MID-Fusion [38] directly use object masks predicted from 2D CNNs to build object landmarks for camera pose estimation, and formulate object-level bundle adjustment to further rectify global pose estimation. Besides, recent works adopt to utilize category or instance labels predicted from 2D CNNs to guide the data association of sparse feature points [45] or instance landmarks [46], [47], [48] for camera pose estimation.

Unlike using the semantic information as explicit data association in the above approaches, our approach directly uses the semantic information as measurement within a unified semantic space, leading to a tightly-coupled semantic fusion for more accurate 3D reconstruction in terms of global consistency. Moreover, our approach enables a real-time depth fusion system on the semantic priors only with a concise representation, i.e., 2D category labels, without the need of time-consuming instance inference systems [35], [38], which prevent real-time performance. Besides, the latest SLAM techniques leverage deep neural networks to predict depth priors (such as CodeSLAM [49], SceneCode [50], Mobile3DRecon [51]) or end-to-end pose prediction (such as SfMLearner [52], SAVO [53]) in monocular scenarios for accurate camera tracking. Their goals are different from our goal of high-fidelity reconstruction of 3D scenes.

2.4 Deep 3D Registration

Point cloud registration is a classic problem, which has been extensively studied in the past decades [54]. The recent research focus has shifted to deep learning based approaches on two critical issues, i.e., keypoint detection and keypoint description. For the former, 3DFeat-Net [55] provides a patch-wise detection approach, which encodes the spatial context for 3D point clouds. USIP [56] presents a covariant keypoint detection approach with unsupervised learning. For the task of keypoint description, the recent work of 3DMatch [57] introduces a volumetric convolution approach to encode keypoint descriptors for 3D scan data, with which a RANSAC [58] based point cloud registration is performed for 3D reconstruction. D3Feat [59] performs the

keypoint detection and description jointly with one convolution neural network for more accurate keypoint detection and description. FCG [60] provides fully-convolutional geometric features to perform the accurate 3D registration in a fast way. 3DRegNet [61] performs robust DNN registration for efficient transformatin estimation. DGR [62] introduces a differentiable framework for pairwise registration for robust, accurate, and fast real world scan registration. On the other hand, [63] introduces an end-to-end learnable point cloud registration system, which aims at globally consistent multi-view 3D registration. Such an idea is further extended to the multibody setting by [64].

Although these deep learning based 3D registration methods could be applied to online 3D reconstruction in global pose graph construction or loop closure, we propose efficient semantic registration to formulate an accurate semantic pose graph during the back-end of the real-time 3D reconstruction system. Compared with those deep 3D registrations, our approach does not require the computation of any keypoints using deep learning networks, and can achieve the same level of 3D registration accuracy than the state-of-the-art deep 3D registration approaches (DGR [62]). According to the comparison we made in Sec. 4.4, our semantic registration can be more suitable than these deep 3D registration approaches for the submap registration in the task of creating semantic pose graph, considering the balance of 3D registration accuracy, memory footprint and time cost during the real-time 3D reconstruction system.

3 METHOD

3.1 System Overview

Given a sequential RGB-D stream, we leverage a 2D CNN (FuseNet [65] in our implementation) to extract 2D category labels for the RGB-D frames. During online 3D reconstruction, we first map both the RGB-D observations and 3D reconstructions (in a representation of TSDF) into a semantic space. Within the semantic space, we perform frame-to-model camera tracking with a semantic SDF tracker. In the back-end, we maintain the reconstructed 3D scene as a collection of semantic submaps, and construct a global pose graph with semantic links between the submaps to further rectify their global pose, aiming at globally consistent 3D reconstruction. To build the global pose graph, we propose a few novel operations on the semantic submaps, including semantic submap generation, semantic submap erosion, and semantic registration etc, to efficiently register the submaps with reliable semantic links. An overview of our approach is given in Fig. 2.

Notation. For a given RGB-D data stream, we denote $F_k = \{I_k, Z_k, L_k\}$ as the k-th frame with I_k, Z_k, L_k being the intensity, depth, and 2D category labels respectively, and $T_k \in SE(3)$ as its camera pose. Here T_k is a rigid transformation, which can be represented as an exponential function of a vector $\xi \in \mathfrak{se}(3)$, i.e., $T_k(\xi) = Exp(\xi^\wedge) \in SE(3)$ [66] , where $\wedge$ is a *hat* operator [66] (see our supplementary materials for more details). Given the camera's intrinsic parameters, we back-project frame F_k to the currently reconstructed 3D points $V_k = \pi'(Z_k)$, with $\pi(\cdot)$ being the 3D-to-2D projection and $\pi'(\cdot)$ the inverse. Here we use $\mathcal{L}$ to represent the category label set with m categories, i.e., $L_k := \{L_k(u) \in \mathcal{L}|u\}$, with u representing 2D coordinates in the intensity image I_k.

3.2 Semantic Space

We introduce a mapping that embeds category labels from the 2D domain to the 3D space via 2D-to-3D inverse-projection. Specifically, we introduce a semantic mapping $\mathcal{M}$ that maps a pixel $p \in R^2$ with a label $l \in \mathcal{L}$ to a point $P \in R^3$ affiliated with the label l as $\mathcal{M} : \{p, l\} \in R^2 \times \mathcal{L} \rightarrow \{P, l\} \in R^3 \times \mathcal{L}$ if $\pi(P) = p$, where $\pi(\cdot)$ is the 3D-to-2D projection. Here we denote $\bar{\mathcal{S}} = R^3 \times \mathcal{L}$ as the semantic space. For a hyper point in the semantic space $\bar{P} = \{P, l\} \in \bar{\mathcal{S}}$, we extend a rigid transformation T in R^3 to $\bar{\mathcal{S}}$ as an operator $\otimes$, which transforms a hyper point $\bar{P}$ to another hyper point $\bar{P}'$ by $\bar{P}' = T \otimes \bar{P} = \{TP, l\} \in \bar{\mathcal{S}}$.

Distance Function. We introduce a distance function $\Gamma : \bar{\mathcal{S}} \times \bar{\mathcal{S}} \rightarrow R$ to measure the distance between two hyper points $\bar{P}_i = \{P_i, l_i\}$ and $\bar{P}_j = \{P_j, l_j\}$ in the semantic space $\bar{\mathcal{S}}$ as: $\Gamma(\bar{P}_i, \bar{P}_j) = \frac{1}{2}(\sum_{\bar{P}_k}^{\Omega_i} G(|P_k - P_j|)\Phi(l_k, l_j) + \sum_{\bar{P}_k}^{\Omega_j} G(|P_i - P_k|)\Phi(l_i, l_k))$, where Ω_i and Ω_j are the neighborhoods of $\bar{P}_i$ and $\bar{P}_j$, respectively, and $\bar{P}_k = \{P_k, l_k\}$ represents a hyper point in Ω_i or Ω_j. We approximate the neighborhood Ω_i for each hyper point $\bar{P}_i$ as a $5 \times 5 \times 5$ voxel grid for computation efficiency when evaluating the distance function $\Gamma(\cdot)$. $G(x) = \frac{1}{\sqrt{2\pi}\sigma} Exp\{-\frac{x^2}{\sigma^2}\} \sim \mathcal{N}(0, \sigma)$ is a normal Gaussian function (σ is the standard deviation), where we use the Euclid position distance $x_{ij} = |P_i - P_j|$ as the input in the distance function. Besides, $\Phi(l_i, l_j)$ is an indicator function: $\Phi(l_i, l_j) = 0$ if $l_i = l_j$, otherwise $\Phi(l_i, l_j) = 1$.

Properties of Semantic Space. From the construction of the semantic space, we summarize its properties in two folds: (1) the semantic space is a closed space over the transformation operator $\otimes$ (please see the formal proof of Lemma I in the supplementary materials). and (2) the distance metric function $\Gamma(\cdot)$ is continuous over each hyper point $\bar{P} = \{P, l\}$'s position $P \in R^3 \subset \bar{S}$ at everywhere (please see the formal proof of Lemma II in the supplementary materials). According to such two properties, we can first calculate the analytic derivation of operator $\otimes$ over the rigid transformation $T(\xi)$ by $\begin{bmatrix} \frac{\partial TP}{\partial \xi} \\ \mathbf{0}_{1\times 6} \end{bmatrix}_{4\times 6}$, where $\frac{\partial TP}{\partial \xi} = [-(TP) \quad I]_{3\times 6}$. Besides, the analytic derivation of $\Gamma(\cdot)$ over hyper point $\bar{P}_i$ is calculated as $\frac{\partial \Gamma}{\partial \bar{P}_i} = [\sum_{v_k \in \Omega_j} \frac{\partial G}{\partial P_i} \Phi(l_i, l_k) \quad 0]_{1\times 4}$. By incorporating the derivation of operator $\otimes$ over the rigid transformation, we can calculate the analytic derivation of the distance function $\Gamma(\cdot)$ over rigid transformation $T(\xi)$ as $\frac{\partial \Gamma}{\partial \xi} = \frac{\partial \Gamma}{\partial \bar{P}} \frac{\partial \bar{P}}{\partial \xi}$, which is also continuous with respect to rigid transformations $T(\xi)$.

This continuous distance function $\Gamma(\cdot)$ enables us to perform frame-to-model camera tracking directly in the semantic space. Our insight is that we can estimate the camera pose by registering the hyper points from the observations to the reconstructed 3D scene directly within the semantic space, by embedding both the RGB-D (and labels) observations and the reconstructed 3D scene (in the TSDF voxel representation) to the semantic space, in which both the geometry and semantic cues are tightly fused. Embedding the observations to the hyper observations is straightforward,

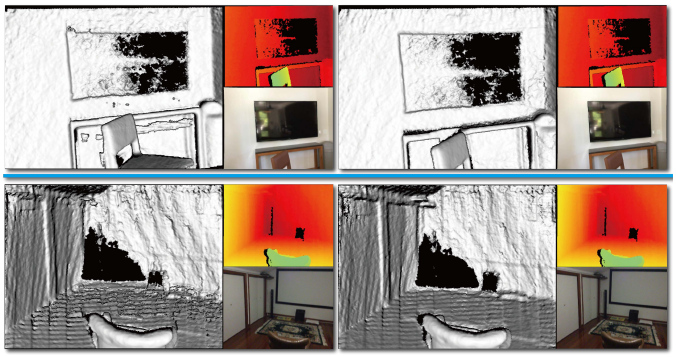


Fig. 3. Two tiny examples of camera tracking for Scene0011 (Top) and Scene0012 (Bottom) sequences in the ScanNet dataset, with (right column) and without (left column) semantic priors in our semantic SDF tracker. Without semantic priors, we can see obvious artifacts in the reconstructions (the 'chair' region in Scene0011 and the 'floor' region in Scene0012, left column), while the reconstructions are much better with our semantic error term (right column) at the same frame respectively.

i.e., by applying the semantic mapping $\mathcal{M}$ to frame F_k as $\mathcal{M}(F_k) \rightarrow \bar{F}_k = \{(v_i, l_i) | v_i \in V_k, l_i \in L_k\}$. But how to embed the reconstructed 3D scene to the semantic space remains to be a problem. Here we propose to perform this embedding following a semantic TSDF representation.

Semantic TSDF Representation. We extend the truncated signed distance function (TSDF) [23] from R^3 to the semantic space $\bar{\mathcal{S}}$ to represent the reconstructed 3D scene during the depth fusion, which we name as a Semantic TSDF representation. Specifically, in our representation $\mathcal{S}$ we record the SDF value $D(v)$ (weight $W_d(v)$), intensity value $I(v)$ (weight $W_i(v)$), and semantic label $L(v)$ (weight $W_l(v)$) for each volumetric voxel v, i.e. $\mathcal{S} = \{D(v), I(v), L(v) | v\}$, and update $\mathcal{S}^k \rightarrow \mathcal{S}^{k+1}$ as: $D^{k+1}(v) = \frac{D^k(v)W_d^k(v)+D(v)}{W_d^k(v)+1}$, $W_d^{k+1}(v) = W_d^k(v) + 1$, $I^{k+1}(v) = \frac{I^k(v)W_i^n(v)+I(v)}{W_i^k(v)+1}$, $W_i^{k+1}(v) = W_i^k(v) + 1$, $L^{k+1}(v) = L^k(v)$, if $W_l^k(v) > 1$, otherwise $L^{k+1}(v) = L(v)$, $W_l^{k+1}(v) = W^k(v) + 1$ if $L^k(v) = L(v)$, otherwise $W_l^{k+1}(v) = W^k(v) - 1$. $L(v)$ is obtained by projecting the category labels predicted from 2D CNNs to the 3D space. Note that although a probability distribution of labels could be predicted by the 2D CNNs, we only store one label (with the most possibility) for each volumetric voxel for the system's efficiency in memory storage.

3.3 Semantic SDF Tracker

Now we introduce our semantic SDF tracker, which estimates the camera pose directly based on the semantic TSDF representation within the semantic space. Specifically, for the currently reconstructed scene $\mathcal{S}^k$, we aim at estimating the camera pose $T_{k+1}(\xi)$ of frame F_{k+1} by optimizing an objective function $E(\mathcal{S}^k, F_{k+1}, T_{k+1}(\xi))$ for the optimized pose $T_{k+1}(\xi^*)$ with

$$\xi^* = \arg\min_{\xi} |E(\mathcal{S}^k, F_{k+1}, T_{k+1}(\xi))|.$$

Within the semantic space, we design the objective function by considering the registration errors from both the geometry and semantic cues when aligning the points from frame F_{k+1} to the geometry surface of $\mathcal{S}^k$. Specifically, our objective function consists of three error terms, namely, SDF error term, intensity error term and semantic error term. Mathematically, it is formulated as follows:

$$\begin{aligned} E(\mathcal{S}^k, F_{k+1}, T_{k+1}(\xi)) = \sum_u |D^k(T_{k+1}(\xi)V_{k+1}(u))|^2 + \\ \alpha \sum_u |I^k(\pi(T_{k+1}(\xi)V_{k+1}(u))) - I^{k+1}(u)|^2 + \\ \beta \sum_{\bar{P}_u \in \mathcal{M}(F_{k+1})} W^{k+1}(T_{k+1}(\xi)V_{k+1}(u)) |\Gamma(T_{k+1}(\xi) \otimes \bar{P}_u, \bar{P}_v)|^2 \end{aligned} \quad (1)$$

where $\mathcal{S}^k = \{D^k(v), I^k(v), L^k(v) | v\}$ is the semantic TSDF representation, $F_{k+1} = \{I_{k+1}(u), Z_{k+1}(u), L_{k+1}(u) | u\}$ is the latest observation at time t_{k+1}, $W^{k+1}(\cdot)$ is the label weight stored in the Semantic TSDF representation, and α and β are balancing weights (see in Section 4.7). The semantic mapping $\mathcal{M}$ maps observation F_{k+1} to the semantic space, with the distance function $\Gamma(\cdot)$ described in Section 3.2.

Since all of the error terms in the objective function have continuous derivations over the camera pose T, this objective function can be optimized using Gauss-Newton optimization, where we search the optimized perturbation $\delta\xi$ in each iteration by solving:

$$\delta\xi = -H^{-1}g,$$

where H is the Hessian matrix and g is the gradient of the objective function. The camera pose is updated using the perturbation $\delta\xi$ as $T(\xi^{n+1}) = T(\delta\xi)T(\xi^n)$. Please refer to the supplementary materials for the derivation of Hessian matrix H and error vector g. Fig. 3 shows an example with and without using the semantic error term, showing its effectiveness in reducing the drift in camera pose estimation.

3.4 Semantic Pose Graph

In parallel with the semantic SDF tracker, we also build a global pose graph to reduce the drift for globally consistent 3D reconstruction. We adopt the mechanism that divides a reconstructed 3D scene into overlapping submaps and adjust their global poses using bundle adjustment in the back-end [5]. Our global pose graph contains not only geometric links (Fig. 2(c) dash black links) but also semantic links (Fig. 2(c) red links), which are calculated using our semantic registration for globally more consistent 3D reconstruction. We call it as a semantic pose graph (Fig. 4(c)). A semantic link is a relative pose constraint between a pair of submaps measured from the semantics priors. We introduce a semantic erosion operation (Fig. 4(a)) to *select* an overlapping region with the same *category* label for the submap pair, and perform the registration on the overlapping *object* region with semantic registration (Fig. 4(b)) using the both geometry and semantic cues. Below we introduce the details of each step.

Semantic Erosion. For a submap pair $(\mathcal{S}^i, \mathcal{S}^j)$, we introduce a semantic erosion operation $\phi : (\mathcal{S}^i, \mathcal{S}^j) \rightarrow (\tilde{\mathcal{S}}^i, \tilde{\mathcal{S}}^j)$ where $\tilde{\mathcal{S}}^i$ and $\tilde{\mathcal{S}}^j$ are the subsets of $\mathcal{S}^i$ and $\mathcal{S}^j$, respectively, such that $\tilde{\mathcal{S}}^i$ and $\tilde{\mathcal{S}}^j$ have regions with the same object labels. More specifically, let $L_{ij} = L_i \cap L_j$ with L_i and L_j being the

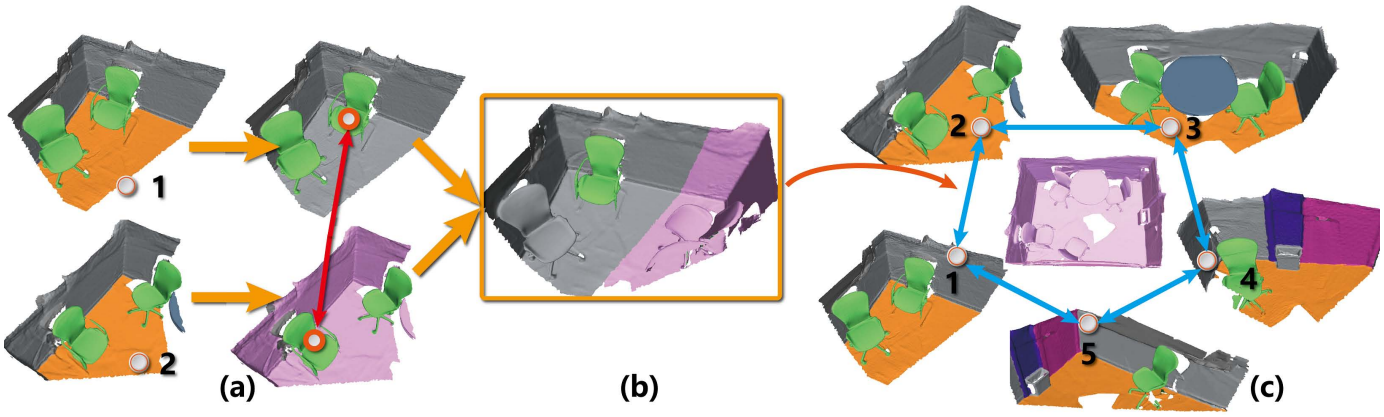


Fig. 4. An illustration of our semantic pose graph's formulation. We introduce a semantic erode operation (a) to perform semantic registration (b) between submaps, thus providing accurate semantic links (c) to build our semantic pose graph. Benefiting from the semantic erode operation, the registration can be accurately performed on object regions sharing the same semantic information rather than the whole geometry region like previous purely geometry-based ICP registration techniques, which would easily fail when the geometry of two submaps varies a lot.

semantic label sets of the semantic TSDF representations of submap pair $\mathcal{S}^i$ and $\mathcal{S}^j$, respectively. The subsets $\tilde{\mathcal{S}}^i$ and $\tilde{\mathcal{S}}^j$ are the subset regions of $\mathcal{S}^i$ and $\mathcal{S}^j$ that involve the semantic labels in L_{ij}, respectively. Fig. 4(a) shows an example of the semantic erode operation to obtain an overlapping region with the same *objects*.

Semantic Registration. We register the overlapping regions $\tilde{\mathcal{S}}^i$ and $\tilde{\mathcal{S}}^j$ filtered by our semantic erosion operation ϕ_{ij} to calculate the relative pose $\tilde{T}_{ij}$, which serves as the semantic link between submap pair $\mathcal{S}^i$ and $\mathcal{S}^j$. This is achieved by minimizing the following objective function:

$$\tilde{E}_{ij}(\tilde{\mathcal{S}}^i, \tilde{\mathcal{S}}^j, \tilde{T}_{ij}) = \sum |D(\tilde{T}_{ij} \otimes \bar{P}_l)|^2 + \gamma |\Gamma^i(\tilde{T}_{ij} \otimes \bar{P}_l, \bar{P}_k)|^2 \tag{2}$$

where $\Gamma^i(\cdot)$ is the semantic distance function and $\bar{P}_k \in \tilde{\mathcal{S}}^i$ and $\bar{P}_l \in \tilde{\mathcal{S}}^j$ are hyper points belonging to the semantic TSDF representations of $\mathcal{S}^i$ and $\mathcal{S}^j$, respectively, and γ is a weight parameter to balance the two error terms.

Semantic Submap Generation. We divide the reconstructed 3D scene into a series of submaps to balance enough in both the geometry and semantic priors, such that an accurate global pose graph can be achieved. So during the online 3D scanning, we create a new semantic submap $\mathcal{S}^{n+1}$ when a new-arrival frame F_k satisfies the following conditions: (1) the overlapping ratio between frame F_k's view observation and the current submap $\mathcal{S}^n$ is less than a threshold $\theta_1 = 0.2$; (2) the semantic gain that frame F_k adds to the current submap $\mathcal{S}^n$ is larger than a threshold $\theta_2 = 1.0$. Here we define the semantic gain $\Theta(\mathcal{S}^n|F_k)$ as the semantic uncertainty reduction that frame F_k takes to $\mathcal{S}^n$: $\Theta(\mathcal{S}^n|F_k) = \sum_{v\in\mathcal{S}^n} W_l^n(v) log(W_l^n(v)) - \sum_{v\in\mathcal{S}^n} W_l^{n+1}(v) log(W_l^{n+1}(v))$, where $W_l^n(v)$ is the label weight of each voxel v described and updated in Section 3.3. In this way, we divide the reconstructed 3D scene into a series of semantic submaps with enough view gaps and low semantic uncertainty.

Semantic Pose Graph Formulation. Finally, we formulate the semantic pose graph among the semantic submaps $\mathfrak{S} = \{\mathcal{S}^i | i = 1, ..., n\}$ with each submap $\mathcal{S}^i$ serving as a graph node, and build both the geometric links and the semantic links as graph edges between node pairs (Fig. 4(c)). Based on the semantic pose graph, we further adjust each submap' global pose with bundle adjustment. Specifically, supposing that the global poses set for the submaps are $\mathfrak{T} = \{T_i | i = 1, ..., n\}$, we seek to adjust their poses to the optimized global poses $\mathfrak{T}^* = \{T_i^* | i = 1, ..., n\}$ such that:

$$\mathfrak{T}^* = \arg\min_{\mathfrak{T}'} \sum_{<i,j>\in\mathcal{E}} log(T_i' T_j'^{-1} \Delta T_{ij}) + log(T_i' T_j'^{-1} \Delta \tilde{T}_{ij}) \tag{3}$$

where $\mathcal{E}$ is the whole link set with $\Delta T_{ij} = T_j^{-1} T_i$ representing the geometric links measurement and $\Delta \tilde{T}_{ij} = \tilde{T}_{ij}$ for the semantic link measurement of a submap pair $< i, j >\in \mathcal{E}$, and $log(\cdot)$ is the logarithmic function of a transformation $T \in SE(3)$ [66].

3.5 Loop Closure

We also perform loop closure detection between the submaps to further rectify the global drift between RGB-D scans with loops. Specifically, for every RGB-D frame during scanning, we encode them as feature vectors using Random Ferns [67] and perform the loop closure detection for the similar frames in the loop. Once the loop closure is detected, we perform semantic registration between the corresponding two submaps and add a semantic link between such two submaps in the semantic pose graph (see Sec. 3.4) for subsequent bundle adjustment in the semantic pose graph between submaps with loops.

4 Experiments and Evaluations

In this section, we first compare our approach with previous real-time 3D reconstruction approaches (including InfiniTAM [5], ElasticFusion [29], and BundleFusion [6]) through a quantitative evaluation on a synthetic dataset (Sec. 4.1) and a qualitative evaluation on a real-world scan dataset (Sec. 4.2). We also perform quantitative and qualitative evaluations (Sec. 4.3) to demonstrate the difference between our approach and the state-of-the-art deep 3D reconstruction approaches (including DI-Fusion [30] and RoutedFusion [27]). The comparison between our semantic registration method and the state-of-the-art deep 3D registration approaches

(including 3DMatch [57], D3Feat [59] and DGR [62]) is made (Sec. 4.4), to justify the effectiveness of our semantic registration for formulating accurate submap links. We also compare our approach with instance-level visual SLAM approaches (Sec. 4.5). Furthermore, we give a comprehensive evaluation on our whole system under 2D semantic annotations with different quality (Sec. 4.6) and different choices of key parameters (Sec. 4.7) and we also evaluate its time efficiency in Sec. 4.8. Finally, we summarize the main limitations of our approach and give possible directions to further improve our performance in the future (Sec. 4.9).

System Implementation. We implemented our approach based on the framework of InfiniTAM [5] and modified the voxels to store the category labels (and weights) as the semantic TSDF representation during the online 3D reconstruction. We adopt the state-of-the-art 2D CNN, i.e., FuseNet [65], to detect the 2D category labels for RGB-D frames. The FuseNet is pre-trained on the ScanNet dataset [22] with category number $m = 21$.

Hardware Configurations. All the experiments are performed on a desktop PC with an i7-6850K CPU, 32 GB RAM, and an Nvidia Titan Xp GPU graphics card. Note that for BundleFusion [6], we use two Nvidia Titan Xp GPUs due to the huge GPU memory consumption for sparse feature point detection and camera pose estimation, with the other hardware configurations keep the same as the original implementation.

4.1 Quantitative Evaluation on Synthetic Dataset

We first perform a quantitative evaluation of our approach on a public synthetic RGB-D dataset SceneNetRGB-D [36], that contains 3D surface annotations as ground-truth surfaces to evaluate the surface reconstruction quality. We choose three publicly available real-time 3D reconstruction approaches for comparison, i.e., InfiniTAM [5], ElasticFusion [29], and BundleFusion [6]. Here, BundleFusion is the state-of-the-art real-time 3D reconstruction method. We adopt the public source code for the other three approaches (InfiniTAM[1], ElasticFusion[2] and BundleFusion[3]) with the default configurations (voxel size, truncated value, etc).

Dataset Collection. The SceneNetRGB-D provides a collected RGB-D dataset containing *5M* RGB-D frames[4]. However, each camera trajectory (with 5 minutes for each) only has sparsely rendered 300 view frames, and thus is not suitable for online 3D reconstruction. So alternatively, we choose to collect our own synthetic RGB-D stream dataset using the photorealistic rendering tool SceneNetRGB-D[5] provided. Specifically, considering that SceneNetRGB-D provides only 5 scene types (including 'Bathroom', 'Bedroom', 'Kitchen', 'Living Room', and 'Office'), we choose to render RGB-D frames with generated camera trajectories and synthetic scenes (created by the Chrono Engine provided by SceneNetRGB-D) uniformly sampled from those 5 scene types, and finally collect 80 RGB-D stream data for the evaluation.

1. InfiniTAM: https://github.com/victorprad/InfiniTAM
2. ElasticFusion: https://github.com/mp3guy/ElasticFusion
3. BundleFusion: https://github.com/niessner/BundleFusion
4. https://robotvault.bitbucket.io/scenenet-rgbd.html
5. https://bitbucket.org/dysonroboticslab/scenenetrgb-d/src/master/

Accuracy Metric. When evaluating the final reconstruction quality in terms of both *accuracy* and *completion*, besides the commonly used accuracy metric like RMSE surface error, we also use more recognized metrics including *F-score* (with both *Precision* and *Recall*) used in [68] and Mean Squared Error (MSE), Mean Absolute Distance (MAD), Accuracy (Acc), Intersection-over-Union (IoU) used in NeuralFusion [28]. All of these metrics are effective measurements to measure both the accuracy and completion for the geometry quality of the reconstructed 3D scenes. When calculating the metrics for MSE, MAD, IoU, and Acc, we use the mesh-to-sdf library[6] to convert the reconstructed mesh (or GT mesh) to SDF samples, and set the voxel resolution as 5mm, as done in NeuralFusion [28].

Comparison Results. During the evaluation, we perform 3D scene reconstruction for the 80 sequences in the SceneNetRGB-D dataset using the four compared approaches separately, and calculate the six metrics for each reconstructed 3D mesh against its corresponding ground-truth mesh.

Table 1 shows the average accuracy scores using the above six metrics for the four approaches. For the RMSE, MSE, and MAD metrics (the lower the better), our approach consistently achieves lower scores as 5.29mm, 7.93e-5 and 6.03e-2 respectively, than BundleFusion (5.70mm, 8.31e-5 and 6.21e-2), ElasticFusion (7.69mm, 9.83e-5 and 7.55e-2) and InfiniTAM (7.33mm, 9.92e-5 and 7.91e-2). For the IOU, ACC, and F-score metrics (the larger the better), our approach also achieves consistently the highest metrics scores as 68.4%, 78.9% and 66.3%, respectively (cf. BundleFusion: 64.2%, 76.6% and 64.4%; ElasticFusion: 52.8%, 62.2% and 55.3%; InfiniTAM: 52.1%, 61.3% and 54.8%. In the F-score metric, the Precision (P) score for BundleFusion (73.1%) is higher than ours (61.6%). This is mainly due to that BundleFusion keeps only reconstructed regions with high accuracy and removes the badly reconstructed regions. But this is at the cost of the decreasing *completion* quality, as reflected by the significantly lower Recall (R) value of BundleFusion (58.3%) than ours (73.4%). Overall, our approach achieves consistently better accuracy than the other three approaches in terms of all the six metrics.

4.2 Qualitative Evaluation on Real World Dataset

We also perform an evaluation on the ScanNet dataset [22] to demonstrate how our approach behaves in reconstructing real-world scenes.

Evaluation Description. Although ScanNet also provides rich annotations of 3D surface reconstruction, those 3D surface reconstruction annotations are generated by BundleFusion[7]. Considering that BundleFusion is also an approach being compared with our approach, we do not use the 3D surface annotations in ScanNet for quantitative evaluation. Alternatively, we only evaluate the visual quality of the reconstructed 3D meshes in ScanNet by comparing our approach with the other three online 3D reconstruction approaches, i.e. InfiniTAM [5], ElasticFusion [29] and BundleFusion [6]. Besides, although the 2D annotation for each frame is provided in every RGB-D sequence of the ScanNet

6. https://github.com/marian42/mesh_to_sdf
7. http://www.scan-net.org

TABLE 1
The quantitative comparisons between our approach and five existing approaches evaluated on the SceneNet RGB-D synthetic dataset, including three previous real-time 3D reconstruction approaches: InfiniTAM (*IM*) [5], ElasticFusion (*EF*) [29], BundleFusion (*BF*) [6], and two deep 3D reconstruction approaches: DI-Fusion (*DF*) [30] and RoutedFusion (*RF*) [27]. The results are measured using six different metrics (from left to right), including RMSE, Mean Squared Error (MSE), Mean Absolute Distance (MAD), Accuracy (Acc), Intersection-over-Union (IoU), and *F-score* (with both *Precision* (P) and *Recall* (R)). '↑' means 'the larger the better' for the underlying metrics and vice versa '↓' means 'the smaller the better'. The numbers in boldface indicate the best performance.

M	RMSE ↓		MSE ↓		MAD ↓		IOU ↑		ACC ↑		*F-score* ↑		
	Mean (mm)	Std -	Mean [e-5]	Std -	Mean [e-2]	Std -	Mean (%)	Std -	Mean (%)	Std -	P (%)	R (%)	F (%)
IM [5]	7.33	1.54	9.92	1.06	7.91	3.21	52.1	3.10	61.3	1.51	55.0	54.7	54.8
EF [29]	7.69	1.01	9.83	1.66	7.55	3.74	52.8	1.14	62.2	0.73	51.2	61.8	55.3
BF [6]	5.70	0.92	8.31	1.56	6.21	3.40	64.2	3.41	76.6	1.25	**73.1**	58.3	64.4
DF [30]	6.95	0.95	9.49	1.03	7.12	1.25	53.1	1.34	64.5	2.04	57.7	56.9	57.2
RF [27]	6.45	0.72	8.60	1.04	6.75	1.02	58.6	1.87	70.1	0.94	65.7	55.6	60.2
Ours	**5.29**	1.03	**7.93**	1.48	**6.03**	3.13	**68.4**	1.35	**78.9**	0.74	61.6	**73.4**	**66.3**

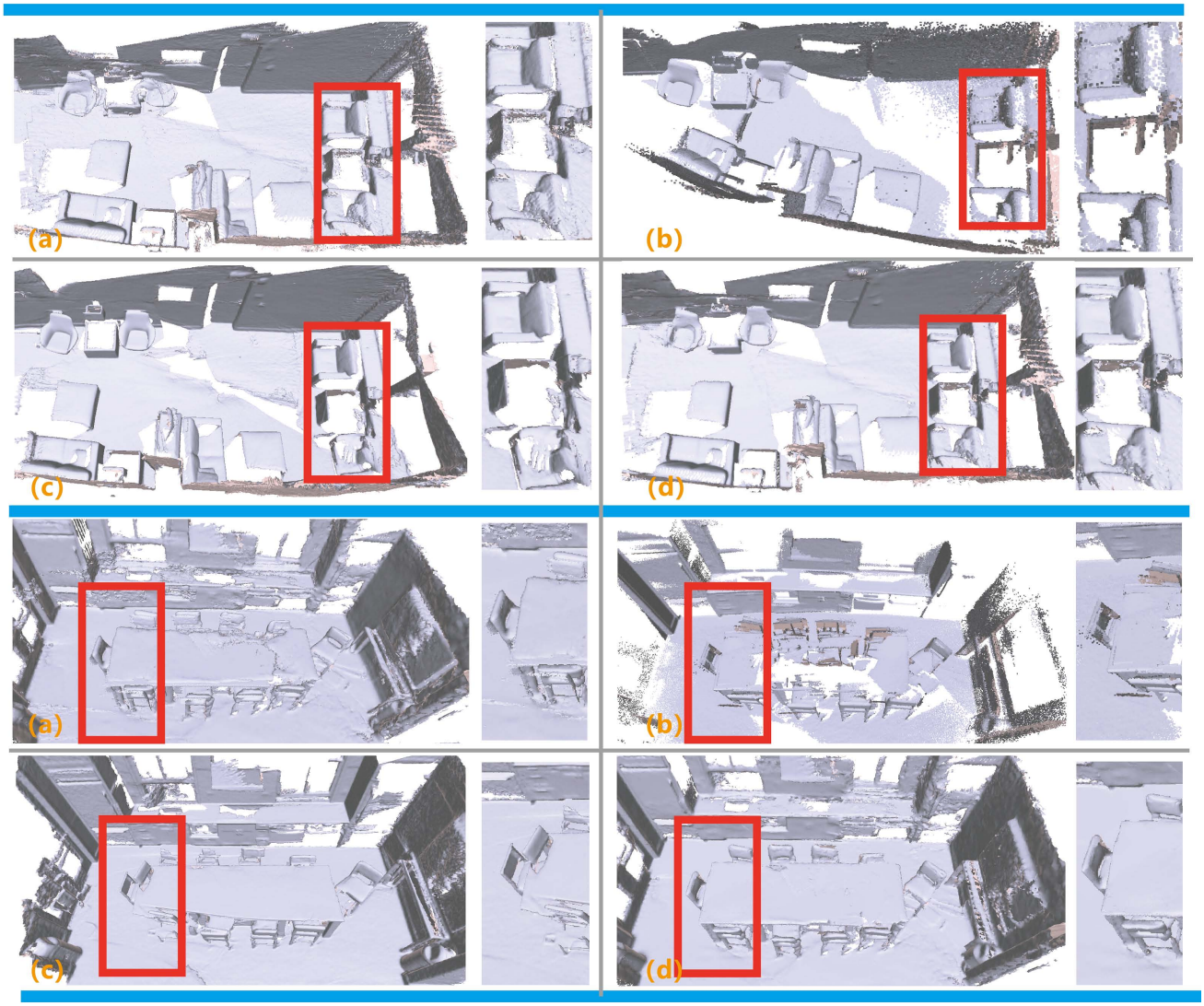


Fig. 5. The visual comparisons of 3D reconstructed geometry surfaces of scene0008 (Top), scene0011 (Bottom) in the ScanNet dataset using the four compared approaches, including InfiniTAM (a), ElasticFusion (b), BundleFusion (c), and ours (d). Our approach achieves higher 3D reconstruction quality for object regions (see the highlighted region in each result) with globally more consistent scene reconstructions than the other three approaches.

dataset, considering that such annotations are obtained by projecting the 3D semantic reconstruction annotations to each 2D frame, we also do not use these 2D annotations as the semantic information but directly detect the 2D annotations by using FuseNet. Lastly, since ElasticFusion [29] reconstructs 3D geometry using surfels but not TSDF voxels

as like InfiniTAM [5], BundleFusion and ours, we use the original implementation of ElasticFusion and maintain the surfel-based geometry surface reconstruction during the qualitative evaluation for a fair comparison.

Comparison Results. Fig. 5 shows two visual comparisons of the reconstructed 3D meshes of 3D scenes in the ScanNet dataset by the four compared approaches. Our approach achieves consistently better 3D reconstruction of object regions, such as sofa, chair, table, etc., as well as background regions such as flatten floor, wall, etc., than the other three approaches. This indicates that our approach can achieve globally more consistent 3D reconstruction of indoor scenes than the other three approaches. Our approach specially takes effect for accurate camera tracking when scanning texture-less object regions, such as table regions in scene0011 as shown in Fig. 5 (Bottom). In such scenarios, it would be challenging for the other approaches like BundleFusion to reliably extract sparse 2D feature points for accurate camera tracking.

Evaluation on SceneNN Dataset. To test the generalization ability of our approach across different real-world scan datasets, we also perform another evaluation on the SceneNN dataset [37], which is another public real world 3D indoor scene RGB-D dataset. Fig. 6 show some visual comparison results generated by the four compared real-time 3D reconstruction approaches. Our approach can consistently achieves better object region reconstruction, thus indicating globally more consistent 3D reconstruction than the other three approaches.

4.3 Comparison with Deep 3D Reconstruction

We also compare our approach with the recent deep 3D reconstruction approaches to see the difference between our semantic-guided 3D reconstruction and the deep-learning-based 3D reconstruction in terms of global consistency of reconstruction. Here we choose two deep 3D reconstruction approaches including: (1) DI-Fusion [30], which leverages the effective neural implicit representation for online 3D reconstruction and (2) RoutedFusion [27], which provides an accurate depth fusion mechanism for precise 3D reconstruction. For DI-Fusion, we use the publicly released code with the pre-trained model in the default parameter configuration[8]. We also use the open-sourced code of RoutedFusion[9]. Since RoutedFusion does not provide the camera pose estimation, we choose to use the traditional baseline approach (InfiniTAM [5]) to perform the camera pose estimation. Besides, although NeuralFusion [28] might provide better depth fusion quality than RoutedFusion, we did not choose it for comparison, partially due to that the two approaches achieve high depth fusion quality in the same level, while RoutedFusion is more widely accessible for comparison.

Table 1 (middle rows) shows the quantitative comparison results between DI-Fusion, RoutedFusion, and our approach, evaluated on the SceneNetRGB-D synthetic dataset in terms of the six accuracy metrics. Our approach achieves consistently better accuracy scores in all of the six metrics than both DI-Fusion and RoutedFusion. This is mainly because our approach focuses on more accurate camera pose estimation with the aid of semantic cues, though the scene representations by DI-Fusion and depth fusion mechanism by RoutedFusion would be more advanced. Fig. 7 shows several visual comparisons of reconstruction results by the three compared approaches tested on the SceneNetRGB-D dataset. Compared with DI-Fusion and RoutedFusion, our approach leads to visually more similar to the ground truth annotations. Please see more visual comparison results in the supplementary materials.

8. https://github.com/huangjh-pub/di-fusion
9. https://github.com/weders/RoutedFusion

4.4 Comparison with Deep 3D Registration

The 3D registration techniques based on deep neural networks are also related with our approach, especially our semantic submap registration in semantic pose graph generation. An alternative way to generate the global pose graph is to replace the semantic submap registration with an existing 3D deep registration method. To test how our semantic registration behaves in comparison with such deep 3D registration approaches, we perform an evaluation on 3D registration quality between consecutive submap pairs. For the 3D deep registration, we choose the three state-of-the-art approaches, i.e., 3DMatch [57], D3Feat [59], and DGR [62], as the representative 3D deep registration approaches. For the evaluation, we first collect 50 pairs for consecutive submaps randomly generated during the online 3D reconstruction process from the SceneNetRGB-D dataset, and then perform the 3D registration on these submap pairs with the four compared approaches respectively. We calculate the average Precision score to evaluate the 3D registration quality of each compared approach. Besides, we also calculate the average memory footprint and run time for different approaches in the evaluation.

Precision Score. For a submap pair (M_1, M_2) after 3D registration, we calculate the precision score $p = \frac{2|M_1 \cap M_2|}{|M_1|+|M_2|}$ with $|\cdot|$ is the point cloud number for a submap. The overlap region $M_1 \cap M_2$ is calculated as overlap points that have nearest neighbor point with the distance under a threshold (5mm).

Table 2 shows the average Precision scores, memory footprint, and run times for the four compared approaches. Our approach can achieve slightly better 3D registration quality in Precision (65.6%) than DGR (62.2%), D3Feat (60.7%), and better than 3DMatch (55.4%). For rum times, although DGR [62] performs fast pairwise 3D registration (about twice as fast as RANSAC with 2M iterations), our semantic registration is faster (0.25s) than DGR (0.76s), and much faster than 3DMatch (1.5s) and D3Feat (10.0s). In addition, since our semantic registration does not require the computation of any keypoints using deep learning networks as needed by the other three deep registration approaches, our approach needs much lower GPU consumption. It is thus more suitable for the semantic submap registrations in our task. Fig. 8 shows several visual comparisons of submap registration by the four compared approaches. Our approach achieves better registration especially for object regions. Please refer to our supplementary materials for more visual results of submap registration.

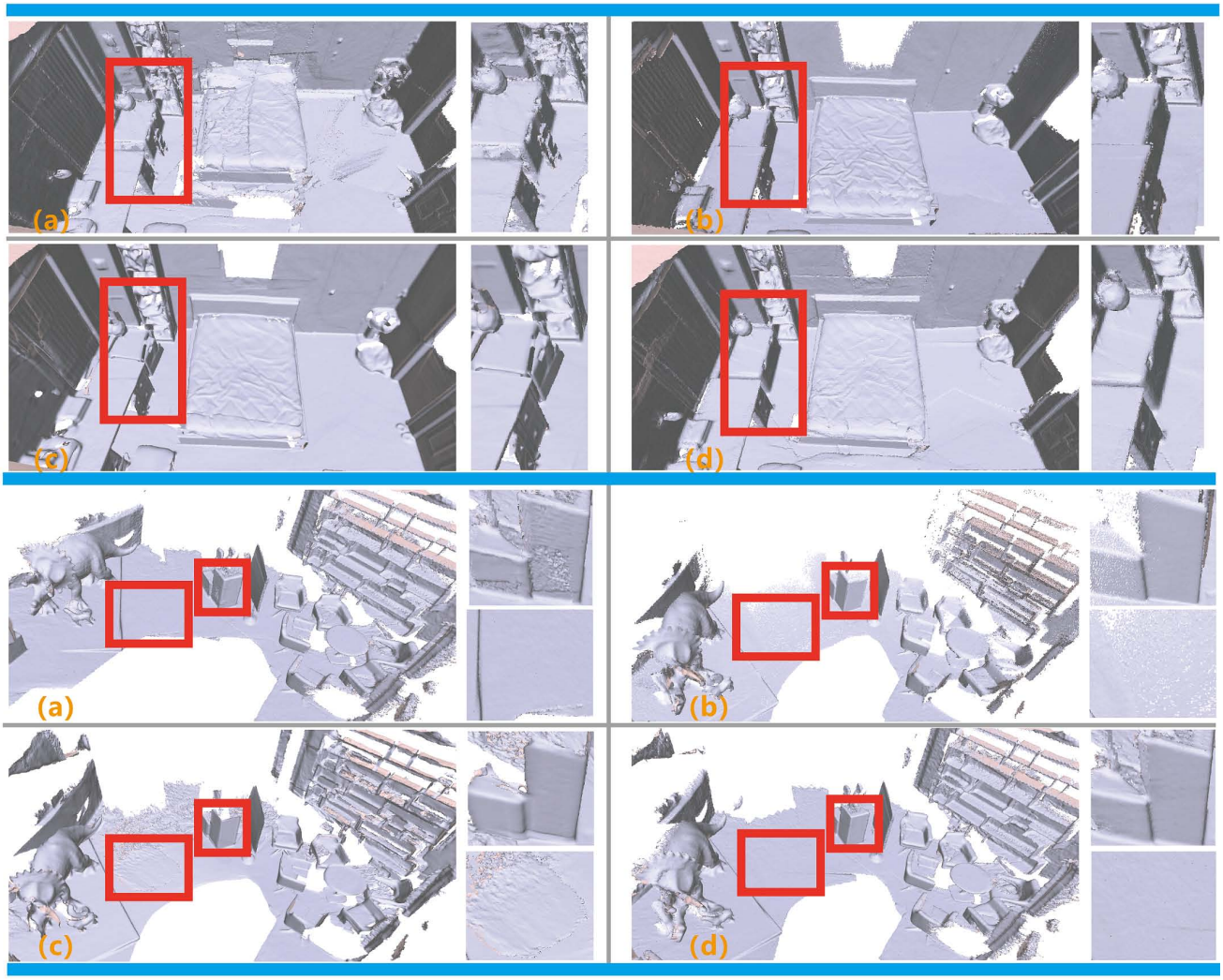


Fig. 6. Visual comparison of surface reconstruction results from the SceneNN dataset using the four compared approaches, including InfiniTAM (a), ElasticFusion (b), BundleFusion (c), and Ours (d). The close-ups corresponding to the highlight red boxes are listed on the right for each result.

TABLE 2
The comparison results between our semantic registration (Ours) and three deep 3D registration approaches, including 3DMatch [57], D3Feat [59], and DGR [62]. For each approach, the precision, memory footprint, and run time quantities are listed respectively.

Method	3DMatch	D3Feat	DGR	Ours
Precision (%)↑	55.4	60.7	62.2	**65.6**
Memory Footprint (G) ↓	5.76	8.04	8.19	**0.42**
Run Time (s) ↓	1.5	10.0	0.76	**0.25**

4.5 Instance-level Visual SLAM

There are several impressive visual SLAM approaches, which take object instances as explicit landmarks for camera pose estimation, including Fusion++ [35] and MID-Fusion [38] as the state-of-the-art instance-level visual SLAM approaches. One of the main drawbacks for these instance-level visual SLAM techniques is that they heavily rely on the instance detection accuracy for camera pose estimation. Although our approach as a depth fusion method has a different goal from those visual SLAM techniques, to evaluate the benefit of our approach for the camera pose estimation, we perform an evaluation on the accuracy of camera pose estimation by comparing these approaches with ours on the TUM RGB-D dataset [69]. Specifically, we calculate the ATE RMSE error between the estimated camera trajectories and the ground-truth camera trajectories of each RGB-D sequence using the three compared approaches, i.e., Fusion++, MID-Fusion, and ours. Besides, we choose a typical TSDF odometry approach (KinectFusion [1]) as the baseline approach in the comparison.

Table 3 shows the average ATE RMSE errors of the four compared approaches for the RGB-D sequences in the TUM RGB-D dataset. To make a fair comparison, we use the ATE RMSE errors of the TSDF odometry, Fusion++, and MID-Fusion reported in the original paper of Fusion++ [35]. Our approach achieves the lowest errors in 'fr1-d', 'fr1-r', 'fr2-x', and 'fr3-l' RGB-D sequences and the second lowest errors in 'fr1-d2' (only higher than the TSDF odometry) and 'fr2-d' (only higher than Fusion++) RGB-D sequences. In average, our approach achieves lower average ATE RMSE error (0.109) than the TSDF odometry (0.193), MID-Fusion (0.171), and Fusion++ (0.113), indicating that our approach can provide higher camera pose estimation accuracy than the other three approaches (though only slightly higher

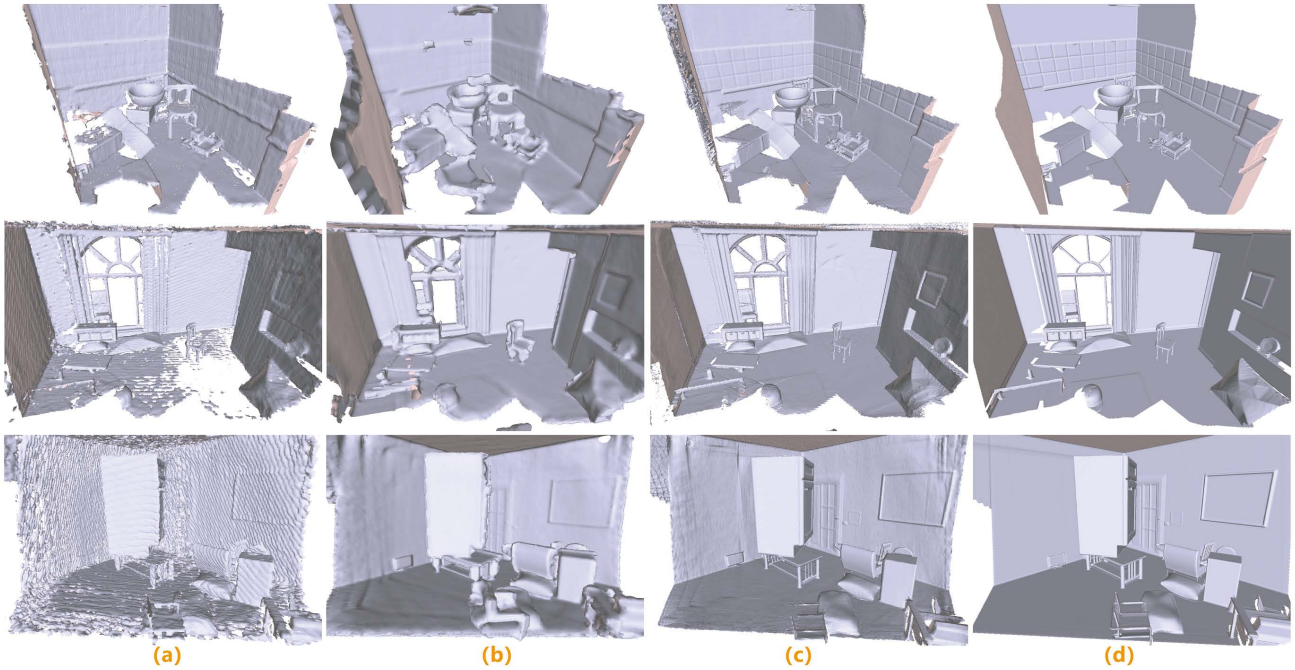


Fig. 7. Visual comparison of surface reconstruction results from the SceneNetRGB-D [36] dataset by DI-Fusion (a), RoutedFusion (b), Ours (c), and the ground-truth meshes (d).

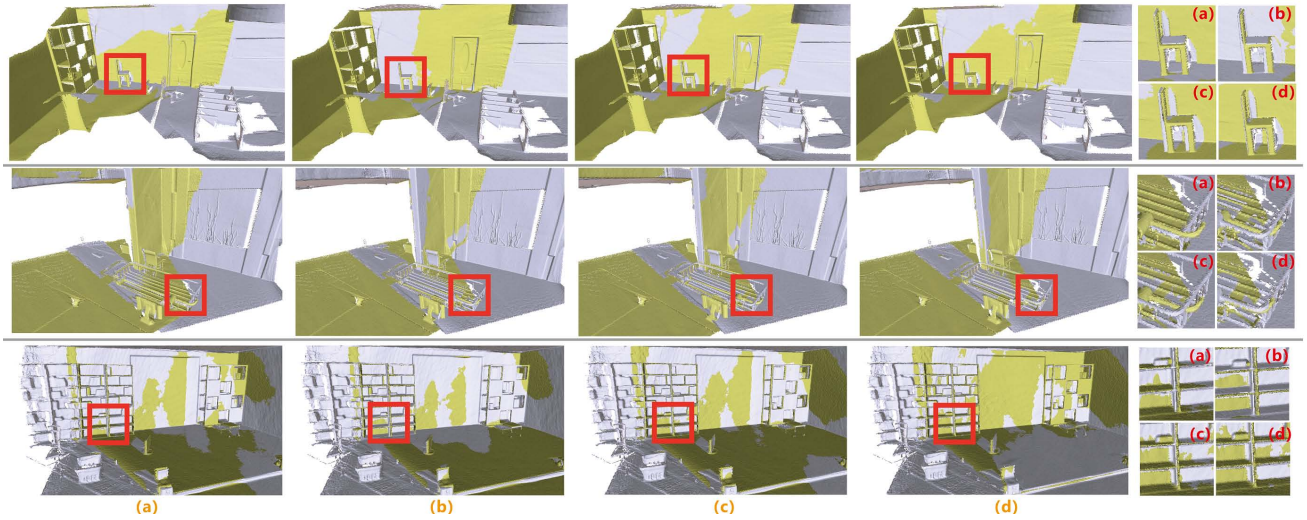


Fig. 8. Representative visual comparisons of 3D submap registration by four different approaches, including 3DMatch (a), D3Feat (b), DGR (c), and our semantic submap registration method (d). Our method produces more accurate 3D registration, especially for object regions. The close-ups corresponding to the highlight red boxes are listed in the right column for each result respectively.

than Fusion++ in the average ATE RMSE error). Note that our approach achieves real-time performance (see Sec. 4.8) at 25fps processing rate with only category-level semantic information, and is much faster than MID-Fusion (2-3fps) and Fusion++ (4-8fps) due to their use of time-consuming instance inference.

The main reason that our approach does not consistently outperform the other approaches for the camera pose estimation accuracy in 'fr1-d2' and 'fr2-d' RGB-D sequences is due to the undesired semantic information quality predicted by FuseNet, especially for the cluttered small objects on the desk in the 'fr2-d' RGB-D sequence. Note that our approach could be further improved by fine-tuning the FuseNet on the TUM RGB-D dataset, or by using other state-of-the-art 2D CNNs that are more suitable for the semantic prediction of TUM RGB-D frames.

4.6 Evaluation on Annotation Quality

The quality of semantic priors (2D annotation) plays an important role for accurate camera tracking in our approach. To study how our approach behaves in respect with the 2D annotation quality, we provide an evaluation on our approach (1) by using different prediction quality

TABLE 3
The quantitative ATE RMSE accuracy (m) of the camera trajectories of TUM RGB-D dataset sequences using different 3D reconstruction approaches, including TSDF Odometry (Odom) [1], MID-Fusion (MF) [38], Fusion++ (F++ for short) [35], and ours.

Method	fr1-d	fr1-d2	fr1-r	fr2-d	fr2-x	fr3-l	avg
Odom	0.066	**0.146**	0.305	0.342	0.022	0.281	0.193
MF	0.058	0.182	0.257	0.268	0.026	0.237	0.171
F++	0.049	0.153	0.235	**0.114**	0.020	0.108	0.113
Ours	**0.040**	0.152	**0.207**	0.148	**0.015**	**0.097**	**0.109**

of FuseNet and (2) by replacing FuseNet with different 2D CNN baselines.

Different FuseNet Quality. We re-train $K = 4$ versions of FuseNet with the number of epochs set as 10, 20, 50, and 100, which are termed as FuseNet-10, FuseNet-20, FuseNet-50, and FuseNet-100, respectively. The mIoU accuracies of these four versions of FuseNet in the test dataset are 0.60 (FuseNet-10), 0.65 (FuseNet-20), 0.70 (FuseNet-50) and 0.79 (FuseNet-100), respectively. The 2D semantic label predictions using these four versions of FuseNet represent different 2D annotation qualities. Note that the final version we use is FuseNet-150 with 0.81 mIoU accuracy.

Different 2D CNN Baselines. We also choose three commonly used 2D CNN approaches to replace FuseNet in our system, including SSMA [70], SegNet [71] and LinkNet [72], to see how our approach behaves with different kinds of 2D semantic prediction baselines. We choose to perform the evaluation on the SceneNetRGB-D synthetic dataset, by comparing the final reconstruction quality for the above mentioned different versions of our system. Besides, we also implement our system without using any semantic priors, and set it as a baseline system. For an efficient evaluation, we calculate two accuracy metrics, i.e., RMSE and F-score, to evaluate the final 3D reconstruction quality.

Results. Table 4 (upper rows) shows the average RMSE sore and F-score of the four different versions of FuseNet integrated in our system. In general, the average RMSE errors increase as the semantic prediction accuracy decreases along with the different versions of FuseNet. Correspondingly, the F-score increases as the semantic prediction accuracy increases. This makes sense since the quality of 2D semantic prediction takes a positive effect on the two important modules of our approach, i.e., camera tracking and semantic pose graph optimization. The more accurate the 2D annotation provided by 2D CNNs, the better 3D registration quality obtained in both the semantic SDF tracker component and the semantic pose graph optimization component. Table 4 (middle rows) shows the average RMSE scores and F-scores of three different 2D CNN baselines integrated in our system. Similarly, the use of different 2D CNN baselines will also influence the final reconstruction quality. If the 2D CNNs decrease significantly (like SSMA), the final surface reconstruction quality would decrease accordingly. But all of these different systems can achieve better surface reconstruction quality than the baseline system, which shows that our approach fusing both semantic and geometry cues takes effects than the baseline system without using any semantic priors.

TABLE 4
The quantitative comparisons between our system and different FuseNet versions (upper rows) and different 2D CNN baselines (middle rows), including SSMA [70] (termed as F1), SegNet [71] (termed as F2), and LinkNet [72] (termed as F3), measured using two different metrics including RMSE and *F-score* (with both *Precision* (P) and *Recall* (R)). The quantitative scores for the baseline system are also included (termed as Baseline). '↑' means 'the larger the better' for the underlying metrics and vice versa '↓' means 'the smaller the better'. The numbers in boldface indicate the best performance.

M	RMSE ↓		*F-score* ↑		
	Mean (mm)	Std [e-5]	P (%)	R (%)	F (%)
FuseNet-10	7.41	1.09	54.1	58.3	56.1
FuseNet-20	6.76	1.02	56.9	58.9	57.8
FuseNet-50	5.94	0.76	60.3	66.9	63.4
FuseNet-100	5.64	0.93	60.9	71.5	65.7
F1	7.22	0.89	53.6	59.1	56.2
F2	6.84	1.14	58.2	57.1	57.6
F3	5.81	0.96	60.1	70.8	65.0
Baseline	7.71	0.97	52.8	58.6	55.4
Ours	**5.29**	1.03	**61.6**	**73.4**	**66.3**

4.7 Parameter Study

In our implementation, σ used in the distance metric function $\Gamma(\cdot)$, α (for the intensity error term) and β (for the semantic error term) used in the semantic SDF tracker, and γ used in the semantic registration are four key parameters, which influence the final 3D reconstruction quality. To evaluate the impact of these parameters, we perform a parameter study experiment. Since it would not be feasible to traverse all of the parameter configurations, we choose to study the impact of each parameter on the reconstruction quality one by one. Specifically, we uniformly sample one parameter in a certain range and randomly sample 100 configurations for the remaining parameters. Using such parameter configurations, we perform 3D reconstruction on the SceneNetRGB-D synthetic dataset and calculate the reconstruction accuracy using the average RMSE surface reconstruction metric. For efficiency, we set the range of each parameter as [0,2.0] for α, [0,2.0] for β, [0,0.20] for γ, and [0.1a, 4a] for σ (where a is the voxel size), respectively.

Fig. 9 shows the RMSE metric curves for the four parameters, respectively. For parameters β, γ, and σ, which involves with the semantic priors, we can see that the RMSE error decreases in general along with the increasing values of those three parameters. This makes sense since the semantic priors have a positive impact in our approach. For the parameter α that is related with the intensity error term of the semantic SDF tracker, the RMSE error curve does not show a clear trend in terms of the parameter. Considering both the reconstruction quality and system efficiency, we set $\sigma = 1.5a$, $\alpha = 1.25$, $\beta = 0.75$ and $\gamma = 0.12$ in all of our experiments.

4.8 Time Analysis

As shown in Table 5, our semantic SDF tracker takes in average 30ms to perform the camera tracking. To generate the global pose graph, our semantic registration takes average 25ms to calculate a semantic link between two consecutive submaps. For time efficiency, we perform the semantic submap registration at every $N = 5$ frames. Besides, the

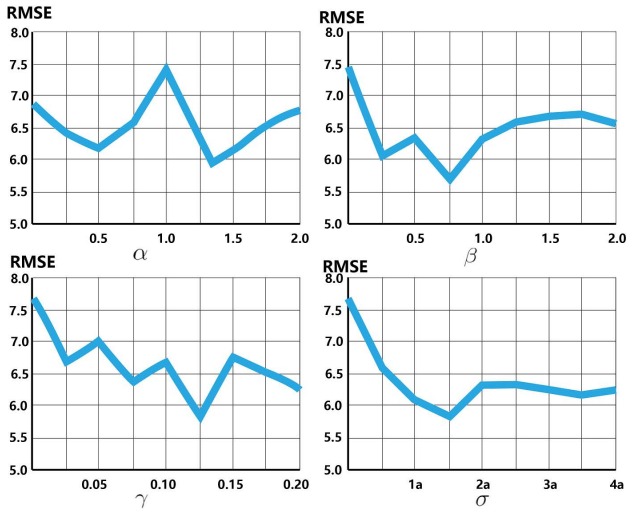


Fig. 9. The RMSE metric curves for the different main parameters in our approach.

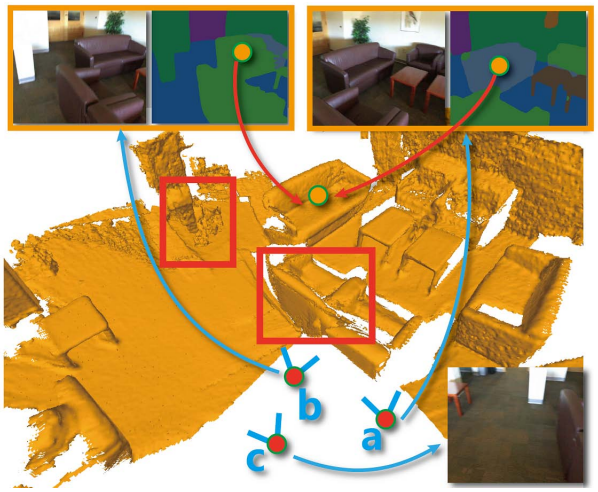


Fig. 10. A representative failure case of our approach in scene0001 sequence of the ScanNet dataset. The sofa predictions in two RGB-D scan places (a and b) are totally different, leading to undesired reconstruction of another sofa (highlighted red). RGB-D scan c contains a strong background region but without enough object regions, also causing failure by our approach.

FuseNet takes 25ms to perform one 2D semantic prediction. Since the FuseNet semantic prediction runs at a separate thread in parallel to the main reconstruction thread, including semantic SDF tracker and semantic submap registration etc, we maintain an average camera tracking processing rate (front-end) at 25fps in our system. Please refer to the accompany video for the real-time processing of our system during the 3D reconstruction.

TABLE 5
The time cost for each key module in our full system.

Module	Time	Module	Time
Semantic SDF Tracker	30ms	Semantic Registration	25ms
FuseNet	25ms	System	25fps

GPU Memory Storage. There are three modules that consume GPU memory storage in our full system: (1) the FuseNet module takes 500M GPU memory storage for 2D semantic label prediction; (2) the TSDF 3D reconstruction module takes 400M GPU memory storage for each single semantic TSDF submap, and we allocate 20 submaps on average for an indoor 3D scene with $10m \times 10m$ room size; (3) our fusion approach takes on average 300M GPU for semantic SDF tracker and 400M GPU for semantic submap registration. So in total our full system takes 10G GPU memory storage on average for a typical indoor scene 3D reconstruction with $10m \times 10m$ room size.

4.9 Limitations and Discussion

One of the main limitations for our approach is that our current solution could not correct the totally wrong 2D semantic prediction from the 2D CNNs. This would lead to undesired semantic distance measurement during the semantic SDF tracker, thus causing camera tracking drift. Such camera tracking drift could not be further rectified even with our semantic registration, thus failing to achieve globally consistent 3D reconstruction, as show in Fig. 10 (scans at a and b). Besides, when the RGB-D scan at place contains a strong background but without enough object regions, our approach degenerates to previous online 3D reconstruction using pure geometry information such as InfiniTAM and ElasticFusion. Such a failure case is shown in Fig. 10 (scan c).

One interesting direction to further improve the performance of our approach would be to correct the 3D semantics by using the 2D-to-3D mapping approaches such as SemanticFusion [18], ProgressiveFusion [19] or the 2D-3D joint learning approach SupervoxelConv [20]. With more consistent 3D semantic priors, both the semantic SDF tracker and submap registration in our approach would be subsequently improved. Another meaningful point is to explore more effective loop closure techniques by considering the spatial information between objects in the 3D scenes, potentially leading to more efficient and accurate loop closure. Besides, it would be interesting to apply our approach to the 3D outdoor scene reconstruction scenario with the visual-LiDAR data streams.

5 Conclusion

In this paper, we have provided an accurate real-time 3D reconstruction approach with a tight coupling of geometric and semantic priors, stepping towards a globally consistent 3D reconstruction. Benefiting from the use of semantic priors, our approach outperforms the state-of-the-art methods for 3D scene reconstruction on the public benchmarks in terms of both quantity and quality, especially in the terms of global consistency. We hope that our approach could inspire the subsequent works to explore the tightly-coupled fusion of both the geometry and semantic cues, in category-level or instance-level, for more advanced 3D scene reconstruction and understanding techniques in this community. Besides, it is also an interesting direction to predict both the geometric appearance and semantic information (segmentation or classification) jointly using a unified end-to-end deep neural networks in the future work.

ACKNOWLEDGEMENTS

We thank the anonymous reviewers for the constructive comments. This work was supported by the Natural Science Foundation of China (Project Number 61521002, 61902210) , Research Grant of Beijing Higher Institution Engineering Research Center and Tsinghua-Tencent Joint Laboratory for Internet Innovation Technology. Hongbo Fu was partially supported by a grant from City University of Hong Kong (Project No. 7005729). Shi-Sheng Huang was supported by "the Fundamental Research Funds for the Central Universities".

REFERENCES

[1] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. W. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *ISMAR*, 2011, pp. 127–136.

[2] Q.-Y. Zhou and V. Koltun, "Dense scene reconstruction with points of interest," *ACM TOG*, vol. 32, no. 4, pp. 112:1–112:8, 2013.

[3] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3d reconstruction at scale using voxel hashing," *ACM TOG*, vol. 32, no. 6, pp. 169:1–169:11, 2013.

[4] Y. Zhang, W. Xu, Y. Tong, and K. Zhou, "Online structure analysis for real-time indoor scene reconstruction," *ACM TOG*, vol. 34, no. 5, pp. 159:1–159:13, 2015.

[5] O. Kähler, 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 devices," *IEEE TVCG*, vol. 21, no. 11, pp. 1241–1250, 2015.

[6] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt, "Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration," *ACM TOG*, vol. 36, no. 3, pp. 24:1–24:18, 2017.

[7] J. Huang, A. Dai, L. J. Guibas, and M. Nießner, "3dlite: towards commodity 3d scanning for content creation," *ACM TOG*, vol. 36, no. 6, pp. 203:1–203:14, 2017.

[8] Y.-P. Cao, L. Kobbelt, and S.-M. Hu, "Real-time high-accuracy three-dimensional reconstruction with consumer RGB-D cameras," *ACM TOG*, vol. 37, no. 5, pp. 171:1–171:16, 2018.

[9] Y. Jin, D. Jiang, and M. Cai, "3d reconstruction using deep learning: a survey," *Communications in Information and Systems*, vol. 20, no. 4, pp. 389–413, 2020.

[10] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *IEEE CVPR*, 2017, pp. 77–85.

[11] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *NIPS*, 2017, pp. 5099–5108.

[12] B. Graham, M. Engelcke, and L. van der Maaten, "3d semantic segmentation with submanifold sparse convolutional networks," in *IEEE CVPR*, 2018, pp. 9224–9232.

[13] W. Wu, Z. Qi, and F. Li, "Pointconv: Deep convolutional networks on 3d point clouds," in *IEEE CVPR*, 2019, pp. 9621–9630.

[14] J. Hou, A. Dai, and M. Nießner, "3d-sis: 3d semantic instance segmentation of RGB-D scans," in *IEEE CVPR*, 2019, pp. 4421–4430.

[15] C. B. Choy, J. Gwak, and S. Savarese, "4d spatio-temporal convnets: Minkowski convolutional neural networks," in *IEEE CVPR*, 2019, pp. 3075–3084.

[16] Z. Hu, X. Bai, J. Shang, R. Zhang, J. Dong, X. Wang, G. Sun, H. Fu, and C.-L. Tai, "Vmnet: Voxel-mesh network for geodesic-aware 3d semantic segmentation," in *IEEE ICCV*, 2021, p. Accepted.

[17] J. Gong, Z. Ye, and L. Ma, "Neighborhood co-occurrence modeling in 3d point cloud segmentation," *Computational Visual Media*, vol. 8, no. 2, pp. 303–315, 2022.

[18] J. McCormac, A. Handa, A. J. Davison, and S. Leutenegger, "Semanticfusion: Dense 3d semantic mapping with convolutional neural networks," in *IEEE ICRA*, 2017, pp. 4628–4635.

[19] Q.-H. Pham, B.-S. Hua, D. T. Nguyen, and S.-K. Yeung, "Real-time progressive 3d semantic segmentation for indoor scenes," in *IEEE WACV*, 2019, pp. 1089–1098.

[20] S. Huang, Z. Ma, T. Mu, H. Fu, and S. Hu, "Supervoxel convolution for online 3d semantic segmentation," *ACM TOG*, vol. 40, no. 3, pp. 34:1–34:15, 2021.

[21] L. Han, T. Zheng, L. Xu, and L. Fang, "Occuseg: Occupancy-aware 3d instance segmentation," in *IEEE ICRA*, 2020, pp. 2937–2946.

[22] A. Dai, A. X. Chang, M. Savva, M. Halber, T. A. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *IEEE CVPR*, 2017, pp. 2432–2443.

[23] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *ACM SIGGRAPH*, 1996, pp. 303–312.

[24] J. J. Park, P. Florence, J. Straub, R. A. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *IEEE CVPR*, 2019, pp. 165–174.

[25] S. Peng, M. Niemeyer, L. M. Mescheder, M. Pollefeys, and A. Geiger, "Convolutional occupancy networks," in *ECCV*, vol. 12348. Springer, 2020, pp. 523–540.

[26] R. Chabra, J. E. Lenssen, E. Ilg, T. Schmidt, J. Straub, S. Lovegrove, and R. A. Newcombe, "Deep local shapes: Learning local SDF priors for detailed 3d reconstruction," in *ECCV*, vol. 12374. Springer, 2020, pp. 608–625.

[27] S. Weder, J. L. Schonberger, M. Pollefeys, and M. R. Oswald, "Routedfusion: Learning real-time depth map fusion," in *IEEE CVPR*, 2020, pp. 4886–4896.

[28] S. Weder, J. L. Schönberger, M. Pollefeys, and M. R. Oswald, "Neuralfusion: Online depth fusion in latent space," in *IEEE CVPR*, 2021, pp. 3162–3172.

[29] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "Elasticfusion: Real-time dense SLAM and light source estimation," *International Journal of Robotics Research*, vol. 35, no. 14, pp. 1697–1716, 2016.

[30] J. Huang, S. Huang, H. Song, and S. Hu, "Di-fusion: Online implicit 3d reconstruction with deep priors," in *IEEE CVPR*, 2021, pp. 8932–8941.

[31] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[32] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE PAMI*, vol. 29, no. 6, pp. 1052–1067, 2007.

[33] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison, "SLAM++: simultaneous localisation and mapping at the level of objects," in *IEEE CVPR*, 2013, pp. 1352–1359.

[34] S. Yang, Z.-F. Kuang, Y.-P. Cao, Y.-K. Lai, and S.-M. Hu, "Probabilistic projective association and semantic guided relocalization for dense reconstruction," in *IEEE ICRA*, 2019.

[35] J. McCormac, R. Clark, M. Bloesch, A. J. Davison, and S. Leutenegger, "Fusion++: Volumetric object-level SLAM," in *3DV*, 2018, pp. 32–41.

[36] J. McCormac, A. Handa, S. Leutenegger, and A. J.Davison, "Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation?" 2017.

[37] B. Hua, Q. Pham, D. T. Nguyen, M. Tran, L. Yu, and S. Yeung, "Scenenn: A scene meshes dataset with annotations," in *3DV*, 2016, pp. 92–101.

[38] B. Xu, W. Li, D. Tzoumanikas, M. Bloesch, A. J. Davison, and S. Leutenegger, "Mid-fusion: Octree-based object-level multi-instance dynamic SLAM," in *IEEE ICRA*, 2019, pp. 5231–5237.

[39] T. Schöps, T. Sattler, and M. Pollefeys, "Surfelmeshing: Online surfel-based mesh reconstruction," *IEEE PAMI*, vol. 42, no. 10, pp. 2494–2507, 2020.

[40] J. Chen, D. Bautembach, and S. Izadi, "Scalable real-time volumetric surface reconstruction," *ACM TOG*, vol. 32, no. 4, pp. 113:1–113:16, 2013.

[41] Y. Xiao, Y. Lai, F. Zhang, C. Li, and L. Gao, "A survey on deep geometry learning: From a representation perspective," *Computational Visual Media*, vol. 6, no. 2, pp. 113–133, 2020.

[42] C. M. Jiang, A. Sud, A. Makadia, J. Huang, M. Nießner, and T. A. Funkhouser, "Local implicit grid representations for 3d scenes," in *IEEE CVPR*, 2020, pp. 6000–6009.

[43] J. Huang, Z. Kuang, F. Zhang, and T. Mu, "Wallnet: Reconstructing general room layouts from RGB images," *Graph. Model.*, vol. 111, p. 101076, 2020.

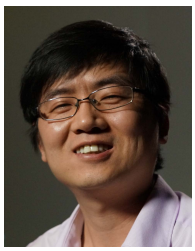
[44] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-

perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.
[45] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, "Probabilistic data association for semantic SLAM," in *IEEE ICRA*, 2017, pp. 1722–1729.
[46] J. Zhang, M. Gui, Q. Wang, R. Liu, J. Xu, and S. Chen, "Hierarchical topic model based object association for semantic SLAM," *IEEE TVCG*, vol. 25, no. 11, pp. 3052–3062, 2019.
[47] M. Strecke and J. Stückler, "Em-fusion: Dynamic object-level SLAM with probabilistic data association," in *IEEE ICCV*, 2019, pp. 5864–5873.
[48] J. Huang, S. Yang, T. Mu, and S. Hu, "Clustervo: Clustering moving instances and estimating visual odometry for self and surroundings," in *IEEE CVPR*, 2020, pp. 2165–2174.
[49] M. Bloesch, J. Czarnowski, R. Clark, S. Leutenegger, and A. J. Davison, "Codeslam - learning a compact, optimisable representation for dense visual SLAM," in *IEEE CVPR*, 2018, pp. 2560–2568.
[50] S. Zhi, M. Bloesch, S. Leutenegger, and A. J. Davison, "Scenecode: Monocular dense semantic reconstruction using learned encoded scene representations," in *IEEE ICRA*, 2019, pp. 11 776–11 785.
[51] X. Yang, L. Zhou, H. Jiang, Z. Tang, Y. Wang, H. Bao, and G. Zhang, "Mobile3drecon: Real-time monocular 3d reconstruction on a mobile phone," *IEEE TVCG*, vol. 26, no. 12, pp. 3446–3456, 2020.
[52] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *IEEE CVPR*, 2017, pp. 6612–6619.
[53] S. Li, F. Xue, X. Wang, Z. Yan, and H. Zha, "Sequential adversarial learning for self-supervised deep visual odometry," in *IEEE ICCV*, 2019, pp. 2851–2860.
[54] N. J. Mitra, N. Gelfand, H. Pottmann, and L. J. Guibas, "Registration of point cloud data from a geometric optimization perspective," in *Symposium on Geometry Processing*, vol. 71, 2004, pp. 22–31.
[55] Z. J. Yew and G. H. Lee, "3dfeat-net: Weakly supervised local 3d features for point cloud registration," in *ECCV*, vol. 11219. Springer, 2018, pp. 630–646.
[56] J. Li and G. H. Lee, "USIP: unsupervised stable interest point detection from 3d point clouds," in *ICCV*, 2019, pp. 361–370.
[57] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. A. Funkhouser, "3dmatch: Learning local geometric descriptors from RGB-D reconstructions," in *IEEE CVPR*, 2017, pp. 199–208.
[58] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
[59] X. Bai, Z. Luo, L. Zhou, H. Fu, L. Quan, and C.-L. Tai, "D3feat: Joint learning of dense detection and description of 3d local features," in *IEEE CVPR*, 2020, pp. 6358–6366.
[60] C. Choy, J. Park, and V. Koltun, "Fully convolutional geometric features," in *ICCV*, 2019, pp. 8957–8965.
[61] G. D. Pais, S. Ramalingam, V. M. Govindu, J. C. Nascimento, R. Chellappa, and P. Miraldo, "3dregnet: A deep neural network for 3d point registration," in *CVPR*, 2020, pp. 7191–7201.
[62] C. Choy, W. Dong, and V. Koltun, "Deep global registration," in *IEEE CVPR*, 2020, pp. 2511–2520.
[63] Z. Gojcic, C. Zhou, J. D. Wegner, L. J. Guibas, and T. Birdal, "Learning multiview 3d point cloud registration," in *CVPR*, 2020, pp. 1756–1766.
[64] J. Huang, H. Wang, T. Birdal, M. Sung, F. Arrigoni, S. Hu, and L. J. Guibas, "Multibodysync: Multi-body segmentation and motion estimation via 3d scan synchronization," in *IEEE CVPR*, 2021, pp. 7108–7118.
[65] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "Fusenet: incorporating depth into semantic segmentation via fusion-based cnn architecture," in *ACCV*, November 2016.
[66] T. D. Barfoot, *State estimation for robotics*. Cambridge University Press, 2017.
[67] B. Glocker, J. Shotton, A. Criminisi, and S. Izadi, "Real-time RGB-D camera relocalization via randomized ferns for keyframe encoding," *IEEE TVCG*, vol. 21, no. 5, pp. 571–583, 2015.
[68] A. Knapitsch, J. Park, Q. Zhou, and V. Koltun, "Tanks and temples: benchmarking large-scale scene reconstruction," *ACM TOG*, vol. 36, no. 4, pp. 78:1–78:13, 2017.
[69] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *IEEE IROS*, Oct. 2012.
[70] A. Valada, R. Mohan, and W. Burgard, "Self-supervised model adaptation for multimodal semantic segmentation," *IJCV*, vol. 128, no. 5, pp. 1239–1285, 2020.
[71] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE PAMI*, vol. 39, no. 12, pp. 2481–2495, 2017.
[72] J. Cai, T. Mu, Y. Lai, and S. Hu, "Linknet: 2d-3d linked multi-modal network for online semantic segmentation of RGB-D videos," *Computers & Graphics*, vol. 98, pp. 37–47, 2021.

Shi-Sheng Huang is currently a lecturer in school of Artificial Intelligence, Beijing Normal University. Before this, he was a PostDoc researcher in Tsinghua University. He got his Ph.D. degree in computer science and technology from Tsinghua University, Beijing, in 2015. His primary research interests include fields of computer graphics, computer vision and visual SLAM, and has published many relevant research works in ACM TOG/IEEE TVCG/IEEE CVPR etc.

HaoXiang Chen received his bachelor's degree in computer science from JiLin University in 2020. He is currently a PhD candidate in the Department of Computer Science and Technology, Tsinghua University. His research interests include 3D reconstruction and 3D computer vision

Jiahui Huang Jiahui Huang received his B.S. degree in computer science and technology from Tsinghua University in 2018. He is currently a Ph.D. candidate in computer science in Tsinghua University. His research interest include computer vision, robotics and computer graphics.



Hongbo Fu received a BS degree in information sciences in 2002 from Peking University, China and a PhD degree in computer science from the Hong Kong University of Science and Technology in 2007. He is a Professor at the School of Creative Media, City University of Hong Kong. His primary research interests fall in the fields of computer graphics and human computer interaction. He has served as an Associate Editor of The Visual Computer, Computers & Graphics, and Computer Graphics Forum.

Shi-Min Hu received the Ph.D. degree from Zhejiang University, in 1996. He is currently a Professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, and Director of State Key Laboratory of Virtual Reality Technology and Systems at Beihang University, Beijing. He has authored over 100 papers in journals and refereed conference. His research interests include digital geometry processing, video processing, rendering, computer animation, and computer-aided geometric design. He is the Editor-in-Chief of Computational Visual Media, and on the Editorial Board of several journals, including Computer Aided Design (Elsevier) and Computer & Graphics (Elsevier).