Hindawi
Discrete Dynamics in Nature and Society
Volume 2020, Article ID 2528954, 10 pages
https://doi.org/10.1155/2020/2528954

Research Article

Incremental Instance-Oriented 3D Semantic Mapping via RGB-D Cameras for Unknown Indoor Scene

Wei Li,[1] **Junhua Gu**,[2] **Benwen Chen**,[2] **and Jungong Han**[3]

[1]*School of Electrical Engineering, State Key Laboratory of Reliability and Intelligence of Electrical Equipment, Key Laboratory of Electromagnetic Field and Electrical Apparatus Reliability of Hebei Province, Hebei University of Technology, Tianjin 300401, China*
[2]*School of Artificial Intelligence, Key Laboratory of Big Data Computing, Hebei University of Technology, Tianjin 300401, China*
[3]*WMG Data Science, University of Warwick, CV4 7AL, Coventry, UK*

Correspondence should be addressed to Junhua Gu; jhgu@hebut.edu.cn

Received 12 January 2020; Revised 25 February 2020; Accepted 3 March 2020; Published 23 April 2020

Guest Editor: Jinchang Ren

Scene parsing plays a crucial role when accomplishing human-robot interaction tasks. As the "eye" of the robot, RGB-D camera is one of the most important components for collecting multiview images to construct instance-oriented 3D environment semantic maps, especially in unknown indoor scenes. Although there are plenty of studies developing accurate object-level mapping systems with different types of cameras, these methods either process the instance segmentation problem in completed mapping or suffer from a critical real-time issue due to heavy computation processing required. In this paper, we propose a novel method to incrementally build instance-oriented 3D semantic maps directly from images acquired by the RGB-D camera. To ensure an efficient reconstruction of 3D objects with semantic and instance IDs, the input RGB images are operated by a real-time deep-learned object detector. To obtain accurate point cloud cluster, we adopt the Gaussian mixture model as an optimizer after processing 2D to 3D projection. Next, we present a data association strategy to update class probabilities across the frames. Finally, a map integration strategy fuses information about their 3D shapes, locations, and instance IDs in a faster way. We evaluate our system on different indoor scenes including offices, bedrooms, and living rooms from the SceneNN dataset, and the results show that our method not only builds the instance-oriented semantic map efficiently but also enhances the accuracy of the individual instance in the scene.

1. Introduction

Robot vision plays an important role with the development of artificial intelligence industries. With aid of RGB-D cameras (such as Kinect), robots can "see" and analyze the surrounding environment easily. Then, how to make robots accurately and rapidly percept the meaning of objects in real-world environments without a prior knowledge is one of the most important technologies in robotic community. For tasks, such as path planning, object grabbing, or even autonomous driving, we need not only the semantic understanding of a single object but more important, the spatial relationships and layout among individual instances in a 3D environment. It thus leads to the demand of building high-level instance-oriented representations of the scene that would greatly advance the human-robotic interaction. Hence, building progressive semantic instance-level 3D map for indoor scenes with multiview RGB-D images has always been a major project for researchers.

The conventional methods of constructing object-aware semantic maps generally consist of two inseparable aspects: instance segmentation of 3D image and transformation across multiple views. The former focuses on obtaining semantic information via the convolutional neural network [1–6], which is followed by integrating geometric segmentation approach to label 3D objects of the scene. The latter usually carries out simultaneous localization and mapping (SLAM) [7–9], which completes 3D scene reconstruction using RGB-D cameras. Motivated by the mentioned technologies, several works efficiently combine them to generate

a semantically segmented 3D map [10–12] and have achieved impressive results. However, such methods suffer from the oversegment problem or lack of proper data association strategy, and meanwhile, they are computationally inefficient, making them unsuitable for the real-time applications. Some other works focus on processing large-scale video retrieval [13–15], but they mainly deal with the entire scene.

This paper intends to incrementally build instance-orient semantic 3D maps via RGB-D cameras in real time. Without the need of a prior knowledge, the proposed mapping system contains optimized semantic information about the individual object instances from the scene and, meanwhile, integrates semantic probabilities from multiple viewpoints to a globally consistent 3D semantic map. The entire algorithm is basically carried out in three steps. First, RGB images captured by cameras undergo the Mask R–CNN [1] algorithm to generate 2D instance and class predictions. In the second step, the proposed system associates prediction results online into corresponding point cloud mapping by the SLAM system. To improve the instance accuracy, we utilize a Gaussian mixture model with the EM algorithm to cluster and optimize semantical labels predicted from the convolutional neural network. In the last step, we propose a voxel-based Bayesian update strategy towards incremental class update across different frames, which will be incorporated into the truncated signed distance function- (TSDF-) based reconstruction maps for the purpose of accelerating the computational efficiency and reducing time complexity.

The major difference between our system and other works [10, 16, 17] is that we employ the projection relation between voxel and pixel directly to obtain instances semantic in the 3D map instead of using the combination between geometry segmentation on depth images and 2D instance segmentation methods. Doing so helps avoid oversegment with no computation increased. Moreover, our goal is to build an instance-level indoor map consisting of reconstructed object instances with semantic annotation. So, unlike many other dense reconstructions works [18–20] that pursuit accurate instance segmentation, the proposed approach aims to achieve the real-time performance, facilitating real-life robotic applications.

To sum up, the main contributions of this work are as follows:

(i) A novel incremental instance-oriented mapping system that utilizes an RGB-D camera to obtain sequential images and represents as a TSDF-based voxelization map

(ii) An optimization method based on a Gaussian mixture model that clusters the point cloud, further integrating TSDF volumes that contain semantic class and instance IDs

(iii) A voxel-based Bayesian update strategy that tracks and updates class probability distribution across different frames to perform consistent global scene mapping

(iv) Qualitative and quantitative analysis of the proposed system on the SceneNN [21] dataset in multiple scenarios

2. Related Works

2.1. Dense 3D Scene Reconstruction. We can roughly divide 3D reconstruction technologies based on RGB-D images into three categories: feature-based methods, voxel-based methods, and surfel-based methods. Feature-based methods, in general, involve front-end frame-to-frame motion through feature matching and back-end "loop closing" constraints from a heuristic search to perform pose graph optimization. The first popular open-source system was RGB-D SLAM [22] proposed by Endres et al. Subsequent similar methods include DVO-SLAM by Kerl et al. [23] and ORB-SLAM2 by Mur-Artal and Tardos [24]. Although such methods directly consume the point cloud, they could cause incomplete instance segmentation in object-level mapping tasks. Voxel-based methods, such as [8, 25, 26], integrate all depth data of the sensor into a volume model from a 3D space, which uses the iterative closest point (ICP) algorithm to track camera poses and reconstruct dense 3D scene maps.

2.2. Semantic Instance-Aware Mapping. Previous methods have addressed the task of mapping at the level of individual objects. Civera et al. [27] used a monocular SLAM system to create 3D environment maps and then inserted the modeled object from the built database. Similarly, Pavel et al. [28] also required priori 3D object models. Although these methods perform object-oriented semantic mapping, the requirement for priori knowledge of modeling objects makes it difficult for them to be applied in real-time human-robot interaction.

Recent developments in deep learning have also enabled the integration of rich semantic information within real-time simultaneous localization and mapping (SLAM) systems. The work in [11] fuses semantic predictions from a CNN into a dense map built with a SLAM framework. However, conventional semantic segmentation is unaware of object instances, i.e., it does not disambiguate between individual instances that belong to the same category. Thus, the approach in [11] does not provide any information about the geometry and relative placement of individual objects in the scene. A number of other works have addressed the task of detecting and segmenting individual semantically meaningful objects in 3D scenes without predefined shape templates [10, 16, 17, 27, 29–34]. Runz et al. [32] employed the object detector for the first step and then updated the class probabilities of each element consisting of the reconstructed 3D map. As it has a huge time complexity, these methods suggested to only extract semantic information on a subset of the input frames; McCormac et al. [29] utilized the same prediction model but aims at extending the SLAM system by means of object-level pose graph optimizations and relocalizations. [16, 17] are similar to that, but they employ depth segmentation methods to segment 3D instances, which led them to take different approaches and

reach different goals. [22] Proposes an object-oriented mapping system that combines a Single Shot MultiBox Detector (SSD) [6] with ORB-SLAM2 [24]. There are also several object-oriented dense 3D mapping methods [30, 31], the main idea of which is to obtain 2D semantic information by a CNN framework, create associated relationships between 2D semantic and 3D mapping, and then utilize conditional random fields (CRFs) as a postprocessing step to refine the results of semantic segmentation. Another project worth mentioning is [35]. Although it also combines a CNN and SLAM to generate 3D semantic mapping, it adds a recurrent neural network (RNN) [28] in data association.

2.3. Instance Detection and Segmentation. Nowadays, with the rapid development of the convolutional neural network, semantic-related tasks in real-world environments have shown some remarkable results. Beginning with the object detection [3, 28] in RGB images, soon afterwards, Mask R–CNN came out which is further able to predict a per-pixel semantically annotated mask for each of the detected instances, achieving state-of-the-art results on the COCO [36] instance-level semantic segmentation task. Other similar works that are worth to mention, including YOLO [5] and SSD [6], deliver an outstanding performance in terms of accurately segmenting instances. With the help of 2D semantic information, we explore semantical objects in 3D environments.

3. Materials and Methods

The architecture of our system is shown in Figure 1. Each RGB image from the incoming video stream is processed with the Mask R–CNN framework to detect a semantically annotated segmentation mask, then, along with the corresponding depth image, is initialized to the point cloud using the projection method between coordinate frames followed by an optimization strategy using a Gaussian mixture model (GMM) for a more accurate instance label. Next, we employ a voxel-based Bayesian update method to merge class semantic or instance IDs across different frames. Finally, we complete the construction of an incremental instance-oriented semantic mapping system. Details of the proposed system are discussed in the following sections.

3.1. Semantic Instance Segmentation Method. In order to annotate and segment the 3D instances in the scene, we needed to combine the 3D point cloud with its corresponding semantic class distribution and instance IDs. To label objects, we first employed the Mask R–CNN as an object detector to the input image. Mask R–CNN achieved real-time performance while showing high accuracy on the computer vision benchmarks, including the Microsoft COCO dataset [37] and the Pascal VOC collection of datasets [38]. Given the input image $\mathcal{I}_t(\vec{u})$, $\vec{u} = (x, y) \subset \mathbb{Z}^2$, $0 \le x < W$, $0 \le y < H$, Mask R–CNN provides a set of bounding boxes as b_i, $i \subset \mathbb{N}$, $1 \le i \le M$, and class probabilities are assigned to each bounding box as $P(c_i \mid_{I_k}) \subset \mathbb{R}$ by letting $M \in \mathbb{R}^{100*15*15}$ be the number of bounding boxes and $c\varepsilon\mathbb{R}^{100}$ be the class category. Note: although there is a good deal of related research, we chose Mask-R-CNN to achieve the task because of its stability and ability to obtain good results on different datasets. This way, our system can theoretically handle another similar network for an acceleration or accuracy request.

3.2. Incremental 3D Semantic Instance-Oriented Update

3.2.1. 2D-3D Association with Semantic Information. One requirement of the proposed system is to know the camera pose in the target scene. In view of real-time and computing costs, we chose voxel hashing [9] as our SLAM system. This takes advantage of volumetric approaches to achieve dense surface representation while using spatial hashing techniques to avoid memory overhead. The proposed system takes both RGB and depth information as the input and incrementally project them into a single 3D model to achieve the volumetric reconstruction. For each arriving RGB-D frame, the 6-DoF camera pose is estimated by combining ICP [36] and RGB alignment, denoted as $T_{WC} \in \mathrm{SE}(3)$, where W represents the world coordinate and C represents the camera coordinate. Then, we employ the homogeneous transformation matrix $T_{WC}^{-1}(k) = T_{CW}(k)$ to project the transformation from the world coordinate to the camera coordinate. In our case, instead of integrating the original incoming RGB image, the proposed system takes the semantic image $\mathcal{I}_k$ that was processed through the Mask R–CNN as the input, along with corresponding D_k, and then generates the 3D reconstruction with the estimated camera pose. Therefore, the initial point cloud with instance IDs has been generated.

3.2.2. Instance Refinement via the Gaussian Mixture Model. After the rough 2D-3D data association of the SLAM system, point cloud data instances are initially formed, but some false matching points occurred during the projection process. In order to obtain more accurate object representation, we optimized the objects by formulating an accelerated generative model in the form of a GMM with a highly parallel hierarchical expectation-maximization (EM) algorithm, inspired by [39]. Also, there is an alternative clustering approach which can be used for optimization, such as ROC algorithm [12]. As a cluster solution for 3D point cloud data, the advantages of GMM are suited to our work. First, the projected data are embedded into the covariance matrices of GMM, which provides an effective way of processing noisy data. Second, because the storage requirements for a GMM are much lower, the system's ability to perform in real time is not affected. However, due to the computational complexity of the GMM, processing is relatively slow. Normally, the processing method would employ a k-means algorithm to run on the sample set. Because our system already implements 2D-3D association using the projection method of the SLAM system, it generates the corresponding 3D cloud with semantic and instance annotations. This is equal to the process of the sample set, and

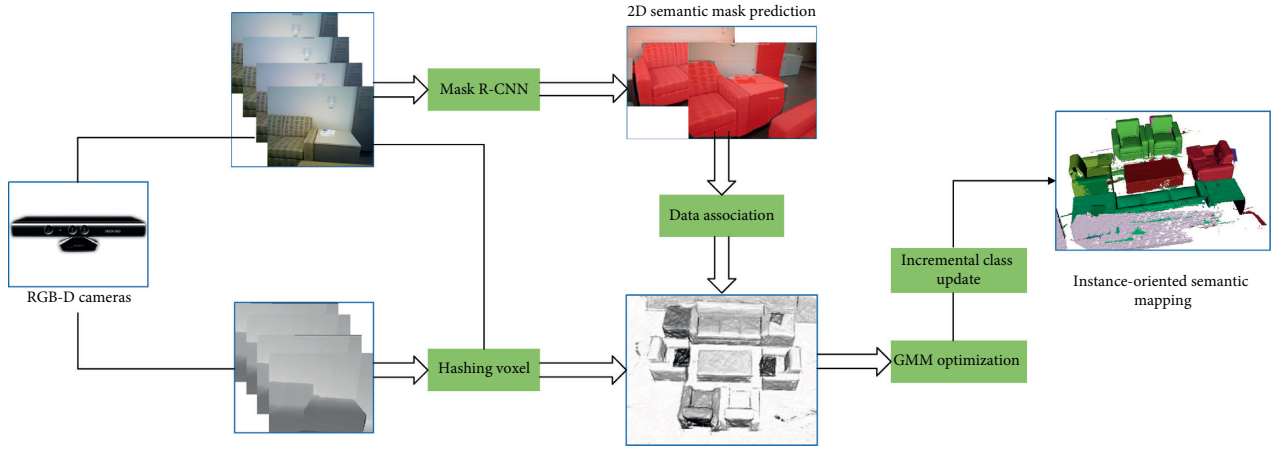

Figure 1: Overview of our incremental instance-level 3D scene reconstruction method. From continuous frames of an RGB-D sensor, our system performs on-the-fly reconstruction and 3D semantic prediction. All of our processing is performed on a frame-by-frame basis in an online fashion, thereby making it useful for real-time applications.

therefore, we can optimize the point cloud data clusters directly with the GMM.

(1) Model Definition. After masks m_j^k are produced by the Mask R–CNN integrated into depth map D_k, we obtained a corresponding point cloud $X = \{x_1, \ldots, x_N\}$ of size N. We assume that there are K classes that can be altered according to the demands of different scenarios. The latent variable represents as $Z = \{z_1, \ldots, z_N\}$, which is a discrete random variable related to sampled point cloud X. In our case, Z indicates classes, the purpose is to index which observed variable belongs to which Gaussian distribution, and the probability of Z represents as $p(Z) = \{p_1, \ldots, p_k\}$. For our formulation, the parameter $\Theta = \{p_k, \mu_k, \Sigma_k\}$ that needs to be estimated with $p_k \varepsilon p(Z)$ represents as class probability and μ_k and Σ_k being the mean and covariance matrix, respectively. Our function describing the generation of incoming point cloud data is a linear combination of Gaussians:

$$p(X \mid \Theta) = \prod_{i=1}^{N} \sum_{k=1}^{K} p_k N\left(x_i \mid \mu_k, \Sigma_k\right), \tag{1}$$

with $\sum_{k=1}^{K} p_k = 1$, and the point cloud data are sets of independent and identically distributed (iid) points.

(2) Executive Parameters. In our case, we are trying to maximize the overall likelihood of a set of Gaussians producing a given point cloud. The general way to compute the maximizer of a parameter is maximum likelihood estimation, but it is only suitable for one Gaussian distribution-contained problem; otherwise, it would not provide an analytical solution. That is why we chose to solve this problem using the EM algorithm, which employs an iterative approach to finding the maximizer of a parameter.

Given initial value $\theta^{(0)}$, the function represents in E-step:

$$\begin{aligned} E_{Z|X,\theta^{(t)}} &= \int_Z^1 \log[p(X, Z \mid \Theta)] p\left(Z \mid X, \theta^{(t)}\right) \mathrm{d}z \\ &= \sum_{k=1}^{K} \sum_{n=1}^{N} \log\left[p_k N\left(x_i \mid \mu_k, \Sigma_k\right)\right] \frac{p_{z_i} N\left(x_i \mid_{\mu_{z_i}^{(t)}, \Sigma_{z_i}^{(t)}}\right)}{\sum_{k=1}^{K} p_k^{(t)} N\left(x_i \mid_{\mu_k^{(t)}, \Sigma_k^{(t)}}\right)}. \end{aligned} \tag{2}$$

In the M-Step, we maximize the expected log-likelihood with respect to θ. The objective function is

$$\theta^{(t+1)} = \arg\max EZ \mid X, \theta^{(t)}. \tag{3}$$

Given a fixed set of expectations, one can solve for the optimal parameters at iteration t:

$$\begin{aligned} p_k^{(t+1)} &= \frac{1}{N} \sum_{i=1}^{N} p\left(z_i = p_k \mid x_i, \theta^{(t)}\right), \\ \mu_k^{(t+1)} &= \frac{\sum_{i=1}^{N} p\left(z_i = p_k \mid x_i, \theta^{(t)} p\left(z_i = p_k\right)\right)}{\sum_{i=1}^{N} p\left(z_i = p_k \mid x_i, \theta^{(t)}\right)}, \\ \sum_k^{(t+1)} &= \frac{\sum_{i=1}^{N} p\left(z_i = p_k \mid x_i, \theta^{(t)}\left(x_i - \mu_k^{(t+1)}\right) p\left(x_i - \mu_k^{(t+1)}\right)^T\right)}{\sum_{i=1}^{N} p\left(z_i = p_k \mid x_i, \theta^{(t)}\right)}. \end{aligned} \tag{4}$$

3.2.3. Voxel-Based Bayesian Class Update Approach. Because frame-wise segmentation processes each incoming RGB-D image pair independently, it lacks any spatiotemporal information about corresponding segments and instances across the different frames. Therefore, we propose an incremental voxel-based Bayesian class update approach. According to Nießner et al. [9], given a series of RGB images $\mathcal{I}_1, \ldots, \mathcal{I}_k$ with semantic and instance IDs, as discussed in Section 3.2.1, and corresponding depth images $D_1, \ldots, D_k$, volumetric representation divides them into a small square called a voxel, v, which stores information such as location, color, and class. In order to update the class distribution of each voxel according to the given classes of pixels from the 2D images, we must first find the correspondence between the voxel and the pixel. This is performed by the SLAM system. Therefore, for the current incoming frame $\mathcal{I}_k$, the world coordinate of the corresponding voxel, $v_k(\vec{u})$, in a 3D map is computed by using backprojection:

$$v_k(\vec{u}) = D_k(\vec{u})K^{-1}\dot{u}, \tag{5}$$

where K denotes the intrinsic camera parameter and $\dot{u}$ denotes the corresponding homogeneous coordinate of the pixel's $\vec{u}$.

Each voxel is then projected onto the RGB image plane via camera projection as follows:

$$\vec{u}(v,k) = \pi\left(T_{WC}^{-1}(k)v_k(\vec{u})\right). \tag{6}$$

When a new image $\mathcal{I}_k$ comes in, the system feeds it to the Mask R–CNN to segment n masks denoted as m_j^k, $j = 1, 2, \ldots, n$. Mask R–CNN outputs masks that may overlap each other, so we do not directly gain a class distribution per pixel, as in semantic segmentation. Therefore, we update the class distribution mask by mask. With the relationship between each pair of voxel and pixel computed from (6), we update the class distribution by an optimized recursive Bayesian update algorithm [11], which fits better with our system:

$$\begin{aligned} P\left(c_v = c_i \mid \mathcal{I}_1, \ldots, \mathcal{I}_k\right) &= \frac{1}{Z}P\left(c_v = c_i \mid \mathcal{I}_1, \ldots, \mathcal{I}_{k-1}\right)P\left(c_v = c_i \mid \mathcal{I}_k\right) \\ &= \frac{1}{Z}P\left(c_v = c_i \mid \mathcal{I}_1, \ldots, \mathcal{I}_{k-1}\right)\prod_{j=1}^{n}P\left(c_{\vec{u}(v,k)} = c_i \mid m_j^k\right). \end{aligned} \tag{7}$$

The instance probability distribution update procedure is similar. Nonetheless, the two distributions are updated independently. We store a list of instance probabilities $P(I_v = I_i)$ for each voxel v with I representing instance IDs. We update the instance distribution according to the segmentation result given by the Mask R–CNN. The general update function for instance distribution adopts a recursive Bayesian update scheme as well:

$$\begin{aligned} &P\left(I_v = I_i \mid \mathcal{I}_1, \ldots, \mathcal{I}_k\right) = P\left(I_v = I_i \mid \mathcal{I}_1, \ldots, \mathcal{I}_{k-1}\right) \\ &\prod_{j=1}^{n}P\left(I_{\vec{u}(v,k)} = I_i \mid m_j^k\right). \end{aligned} \tag{8}$$

3.3. Map Integration. The instance segmentation in the 3D format mentioned above achieves associate class probabilities over multiple camera views. After voxel-based class update approach, every voxel's instance ID has been updated as I_v. For map integration, we attempt to integrate 3D semantic instances into a globally volumetric map with greater speed. To this end, each clustered instance is progressive and integrated into a TSDF-based voxel grid, which is measurement from a depth map, D_k, into a volume, V. V stores at each discrete voxel location, $v = (v_x, v_y, v_z)$, both the current normalized truncated signed distance value, its associated weight, and instance class I_v. And we use raycast, the main method for integrating information from sensor data into TSDF for tracking, data association, and visualization to render depth, normals, vertices, RGB, and object indices as shown in Figure 2. The fusion part of our system is incorporated with Voxblox [40], which is a real-time framework of 3D reconstruction based on volumetric TSDF representation. The main benefit of the Voxblox framework is that it has been extended to the label volume, which can store the instance label related with each voxel in the TSDF grid. At each view, the set of point clouds representing the 3D object with semantics is integrated into the voxel-based representation, and our system ensures consistency among the instance labels across different frames.

4. Results and Discussion

We evaluated the performance of our system on an Ubuntu operating system with an Intel Core i5-6500 CPU at 3.2 GHz and an Nvidia GeForce GTX1080 Ti GPU with 11 GB of RAM. Our system is built on top of ROS open-source middleware. The core function is implemented in Python and uses TensorFlow for instance predictions.

The Mask R–CNN uses ResNet-101 based on the publicly available implementation from Matterport Inc. [41], with the pretrained weights provided for the Microsoft COCO dataset [37].

The input stream is typically a 640×480 resolution RGB-D video. To display the ability of progressive building of instance-aware maps per frame, we perform a Mask R–CNN thread simultaneously with 3D reconstruction upon every frame.

Although there are many 3D databases [42, 43] for different research purposes, we chose the SceneNN dataset [21] to evaluate the 3D object accuracy of the proposed instance-level semantic mapping system, which contains 100 indoor scenes, including offices, bedrooms, living rooms, and kitchens, and scenes with repetitive objects; the SceneNN dataset also provides the annotations with fine-grained information, e.g., axis-aligned bounding boxes, oriented bounding boxes, and object poses. It is suited to the task of reconstruction of the instance-oriented semantic mapping.

4.1. Run-Time Performance. To demonstrate the efficiency of our system, we analyzed its run-time performance and compared it with other state-of-the-art systems, as shown in Table 1. These systems are mainly concentrated on object-level mapping tasks. Our system achieved a speed of 10.8 Hz while performing all processing components on every input frame, thereby outperforming other similar

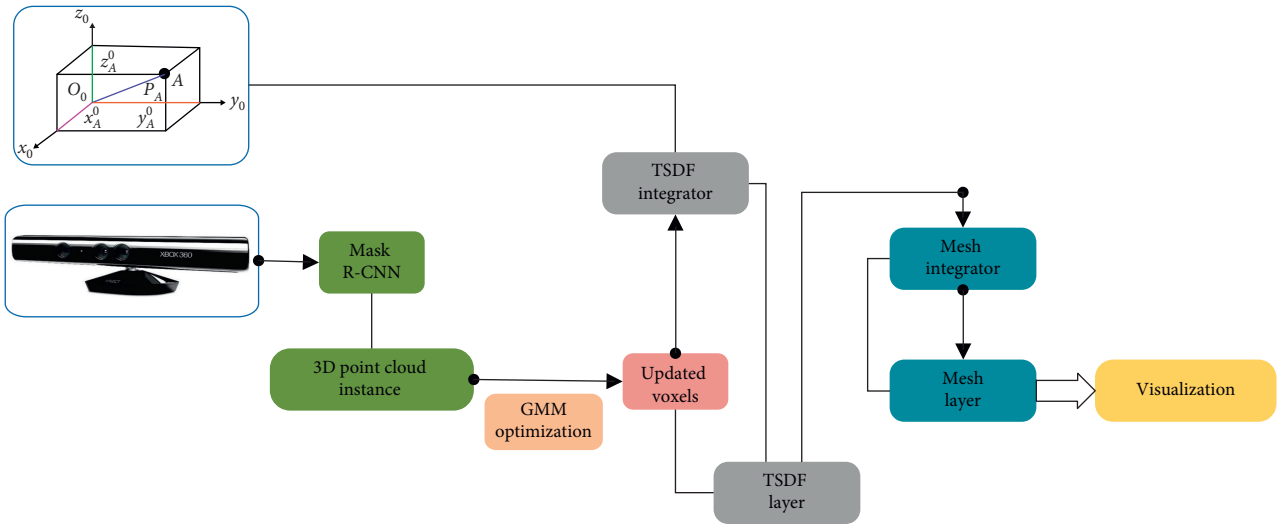

Figure 2: Map integration of the proposed system, showing the interaction between multiple layers and with incoming sensor data through integrators.

Table 1: Comparison of run-time performance. FQ denotes the frequency recognition of when the input frame is performed, and the class probabilities of the 3D map are updated.

Method	Representation	FQ	FPS
SemanticFusion [11]	Dense	Every 10 frames	Under 8 Hz
Hermans et al. [34]	Dense	Every 6 frames	3 Hz
PanopicFusion [44]	Dense	Every 10 frames	4.3 Hz
Voxblox++ [16]	Instance-oriented	Every frame	1 Hz
Pham et al. [45]	Instance-oriented	Every frame	1 Hz
Fusion++ [29]	Instance-oriented	Every frame	4 Hz
Ours	Instance-oriented	Every frame	**10.8 Hz**

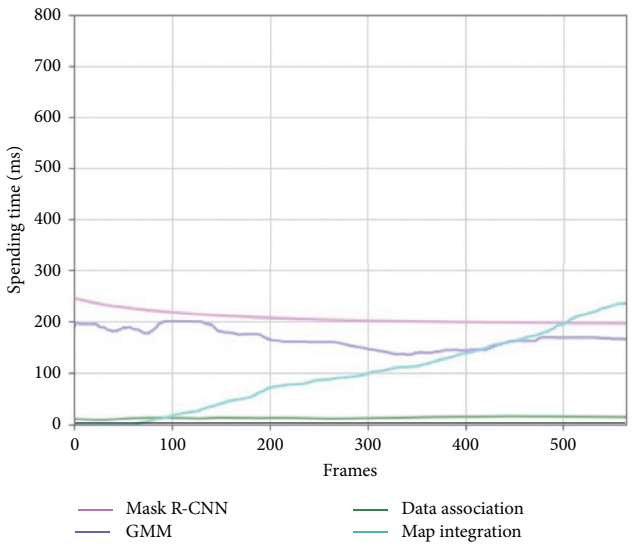

Figure 3: Measured execution times of each stage of the proposed incremental instance-oriented mapping system, averaged over the 5 evaluated sequences from the SceneNN [21].

systems in run-time tests. Compared to the process for utilizing the semantic information from the input image in conventional methods [16, 29, 32], the proposed system has substantially reduced the computational time by exploiting a voxel-based class probability update scheme. All systems were tested on the same sequences of the SceneNN dataset.

Figure 3 shows the evaluation of the execution times upon each individual stage of the proposed incremental instance-level mapping system averaged over five sequences in the SceneNN dataset. Input RGB-D images have 640×480 resolution. Mask R–CNN runs on the GPU, while the rest of the components run on the CPU. The trend lines in the figure showed the data association module running under low rate, which the proposed method effectively improves the operation speed of the system; GMM module maintained on a stable running rate; the map integration module slowed down after 500 frames, ensuring the real-time demand of the system. Note that, by speeding up the system, it is possible to change to a faster object detector network, and the processing of map fusion and Mask R–CNN can occur simultaneously.

4.2. Accuracy. Several recent research projects have focused on semantic instance segmentation of 3D scenes. The majority of these, however, takes as the input the full reconstructed scene, either processing it in chunks or directly as a whole. Because such methods are not constrained to

Table 2: Comparison to the 3D semantic instance segmentation approach from Voxblox++ [16] proposed by Grinvald et al. For 10 sequences from the SceneNN dataset [21], the per-class average precision (AP) is computed using an intersection over union (IoU) threshold of 0.5 over the predicted 3D segmentation masks.

Seq. ID	Method	Bed	Chair	Sofa	Table	Books	Refrigerator	TV	Toilet	Bag
011	Voxblox++	—	75	50	100	—	—	—	—	—
	Ours	—	68.7	**67**	100	—	—	—	—	—
016	Voxblox++	100	0.0	0.0	—	—	—	—	—	—
	Ours	75	0.0	0.0	—	—	—	—	—	—
030	Voxblox++	—	54.4	100	55.6	14.3	—	—	—	—
	Ours	—	**76**	100	50	8.3	—	—	—	—
061	Voxblox++	—	—	100	33.3	—	—	—	—	—
	Ours	—		59.9	33.3					
078	Voxblox++	—	33.3	—	0.0	47.6	100	—	—	—
	Ours	—	**50**	—	**100**	**54.2**	75			
086	Voxblox++	—	80	—	—	0.0	—	—	—	0.0
	Ours	—	66.7	—	—	**25**	—	—	—	**50**
096	Voxblox++	0.0	87.5	—	37.5	0.0	—	0.0	—	50
	Ours	0.0	55.7	—	**39.5**	**11.1**	—	0.0	—	**68.7**
206	Voxblox++	—	58.3	100	60	—	—	—	—	100
	Ours	—	**60**	100	55	—	—	—	—	100
223	Voxblox++	—	12.5	—	75	—	—	—	—	—
	Ours	—	**16.7**	—	75	—	—	—	—	—
255	Voxblox++	—	—	—	—	—	75	—	—	—
	Ours	—	—	—	—	—	75	—	—	—

progressively integrating predictions from partial observations into a global map but can learn from the entire 3D layout of the scene, they are not directly comparable with our work. Among the frameworks that study online, incremental instance-aware semantic mapping, we chose Grinvald et al. [16] as a comparison. Because we relied on a Mask R–CNN model trained on the 80 Microsoft COCO [38] object classes to get the instance IDs, we evaluated the segmentation accuracy on the nine object categories that were common to the SceneNN dataset [21]. The proposed approach was evaluated on the 10 indoor sequences from the SceneNN dataset, the same as Grinvald et al. [16] reported instance-level segmentation results. The results in Table 2 demonstrate that our approach achieves better accuracy in most sequences compared with [16], which is one of the advanced methods focused on real-time incremental instance-aware 3D mapping. It is worth mentioning that further comparing it with [16], our system runs faster and is more suitable for human-robot interaction.

To expand the evaluation of the accuracy of our system, we compared class-averaged mean average precision (mAP) values over the ten evaluated categories with [16, 45]. The results in Table 3 show that the proposed approach outperforms the baseline on six sequences. [45] focuses on building incremental 3D semantic maps of indoor scenes; although it is different from our system, there is an experiment designed for the accuracy of instance classes, and the author explained they only used a simple clustering algorithm to obtain instance semantic so that it can be used as a baseline to compare with similar systems. As the results shown in Table 3, our system highly outperformed in eight scenes compared to their system. Compared to Voxblox++, the proposed system exceeded in six sequences, which proved the advancement of our system. However, it did not perform better in sequences 16, 61, 96, and 206, through analyzing the categories in those sequenced, such as bed and sofa, had more clutter appearances, using the GMM model to optimize might cause oversegment which reduced accuracy. Also, Voxblox++ uses the geometric segmentation method which is better to segment objects with more details, such as chair. We will improve the algorithm in the future.

Table 3: Comparison to the 3D semantic instance-segmentation approach from Voxblox++ [16] and Pham et al. [45] on class-averaged mAP value.

Sequence ID	Voxblox++ [16]	Pham et al. [45]	Ours
011	75.0	52.1	**78.6**
016	33.3	34.2	25.0
030	56.1	56.8	**58.6**
061	66.7	59.1	46.6
078	45.2	34.9	**69.8**
086	20.0	35.0	**47.2**
096	29.2	26.5	26.7
206	79.6	41.7	78.0
223	43.8	40.9	**45.8**
255	75.0	48.6	**75.0**

Furthermore, we showed the qualitative results about the proposed framework on the SceneNN dataset. We presented the incremental instance-oriented 3D semantic mapping generation process in Figure 4. As can be seen, the left image showed the respective progressive semantic segmentation results of our method, the middle image shows the final mapping results, and the right one shows the ground truth segmentation, and the 3D shapes of the object instances, such as chair, sofa, and desk, were incrementally generated

Figure 4: Generation process of incremental instance-oriented semantic mapping in real time.

Proposed system without GMM optimization | Proposed system with GMM optimization | Ground truth

Figure 5: Ablation study on the effects of GMM optimization. The comparison shows the refinement help to improve the segmentation accuracy.

by our system. Because our system is designed to segment instances from the scene, the color of the instance is different from the ground truth, in which the color is assigned according to the classes. As our proposed mapping system focuses primarily on recovering instances of the scene, we have chosen to ignore the background and floor.

4.3. Ablation Analysis. To further illustrate the performance of our GMM model pertaining to the optimized instance cluster, we carried out an ablation analysis to evaluate the effects of accuracy of instance, as shown in Figure 5. Circle A shows that, after GMM optimization, the boundaries of the instance are clearer, and the

segmentation is more accurate. And circle B displays that two different instances are segmented after GMM optimization. The same optimization result is showed in C, and the boundaries of different objects are clearer. This proves that cluster operation in the point cloud based on predicted class information is valid in dense semantic instance-level mapping.

5. Conclusions

Our proposed system is an efficient instance-oriented semantic mapping system. We employed a projection method in the SLAM system that could rapidly associate 2D "masks" and the corresponding depth images to generate a 3D point cloud with instance labels and then used a cluster optimized algorithm to resolve the confusion if projection mismatch occurred. For the 3D reconstruction, the resulting instance-aware semantically annotated volumetric maps are expected to provide benefits in navigation and manipulation planning tasks.

However, as mentioned above, because our system focuses only on recovering 3D instances of an unknown scene, we overlooked the structure of the surrounding environment, such as walls and floors. In the future, we hope to come up with a method that could solve this problem in real time. And also, our system can be used in different applications, such as [44, 46–48]. We intend to research how the segmented instances can serve as semantic landmarks to promote the accuracy of the SLAM system in order to attain a full semantic SLAM system.

Data Availability

The experimental data of the SceneNN and Microsoft COCO dataset used to support the findings of this study are included within the paper.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported in part by Hebei Provincal Innovation Capability Enhancement Project (199676146H).

References

[1] H. Kaiming, G. Georgia, D. Piotr, and G. Ross, "Mask R-CNN," in *Proceedings of the International Conference on Computer Vision (ICCV)*, IEEE, Venice, Italy, Transactions on Pattern Analysis and Machine Intelligence, Venice, Italy, October 2017.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, June 2016.

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[4] R. Girshick, J. Donahue, T. Darrelland, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, June 2014.

[5] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[6] W. Liu, D. Anguelov, D. Erhan et al., "SSD: single shot multibox detector," *Computer Vision—ECCV 2016* in *European Conference on Computer Vision*, vol. 9905, Cham, Switzerland, Lecture Notes in Computer Science, 2016.

[7] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2017.

[8] R. A. Newcombe, S. Izadi, O. Hilliges et al., "Kinectfusion: real-time dense surface mapping and tracking," in *Proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality/ISMAR*, pp. 127–136, Basel, Switzerland, June 2011.

[9] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3D reconstruction at scale using voxel hashing," *ACM Transactions on Graphics*, vol. 32, no. 6, pp. 1–11, 2013.

[10] N. Sunderhauf, T. T. Pham, Y. Latif, M. Milford, and I. Reid, "Meaningful maps with object-oriented semantic mapping," in *Proceedings of the RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5079–5085, IEEE, Vancouver, BC, Canada, September 2017.

[11] J. Mccormac, A. Handa, A. Davison, and S. Leutenegger, "SemanticFusion: dense 3D semantic mapping with convolutional neural networks," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, Singapore, May 2017.

[12] X. Gu, Z. P. Angelov, and Z. Zhao, "A distance-type-insensitive clustering approach," *Applied Soft Computing*, vol. 77, pp. 622–634, 2019.

[13] G. Wu, J. Han, Y. Guo et al., "Unsupervised deep video hashing via balanced code for large-scale video retrieval," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1993–2007, 2018.

[14] G. Wu, J. Han, Z. Lin et al., "Joint image-text hashing for fast large-scale cross-media retrieval using self-supervised deep learning," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9868–9877, 2018.

[15] C. Yan, B. Gong, Y. Wei et al., "Deep multi-view enhancement hashing for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1. In press, 2020.

[16] M. Grinvald, F. Furrer, T. Novkovic et al., "Volumetric instance-aware semantic mapping and 3D object discovery," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 3037–3044, 2019.

[17] Y. Nakajima and H. Saito, "Efficient object-oriented semantic mapping with object detector," *IEEE Access*, vol. 7, p. 3206, 2019.

[18] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt, "BundleFusion," *ACM Transactions on Graphics*, vol. 36, no. 4, p. 1, 2017.

[19] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: dense tracking and mapping in real-time," in *Proceedings of the International Conference on Computer Vision, ICCV*, November 2011.

[20] 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, p. 1697, 2016.

[21] B. S. Hua, Q. H. Pham, D. T. Nguyen et al., "SceneNN: a scene meshes dataset with aNNotations," in *Proceedings of the Fourth International Conference on 3D vision (3DV)*, IEEE Computer Society, Stanford, CA, USA, October 2016.

[22] F. Endres, J. Hess, J. Sturm et al., "3-D mapping with an RGB-D camera," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 177–187, 2017.

[23] C. Kerl, J. Sturm, and D. Cremers, "Dense visual SLAM for RGB-D cameras," in *Proceedings of the RSJ International Conference on Intelligent Robots and Systems*, pp. 2100–2106, IEEE, Tokyo, Japan, November 2013.

[24] R. Mur-Artal and J. D. Tardos, "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.

[25] P. Henry, D. Fox, A. Bhowmik, and R. Mangnia, "Patch volumes: segmentation-based consistent mapping with RGB-D cameras," in *Proceedings of the International Conference on 3D Vision-3DV 2013*, pp. 398–405, IEEE, Seattle, WA, USA, June 2013.

[26] T. Whelan and J. McDonald, "Kintinuous: spatially extended kinectfusion," in *Proceedings of the RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, Cambidge, MA, USA, July 2012.

[27] J. Civera, A. J. Davison, and J. M. M. Montiel, *Structure from Motion Using the Extended Kalman filter*, Springer Science & Business Media, Berlin, Germany, 2011.

[28] M. S. Pavel, H. Schulz, and S. Behnke, "Recurrent convolutional neural networks for object-class segmentation of RGB-D video," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, Killarney, Ireland, July 2015.

[29] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger, "Fusion++: volumetric object-level slam," in *Proceedings of the International Conference on 3D Vision (3DV)*, pp. 32–41, IEEE, Verona, Italy, September 2018.

[30] X. Li and R. Belaroussi, "Semi-dense 3D semantic mapping from monocular slam," 2016, https://arxiv.org/abs/1611.04144.

[31] S. Yang, Y. Huang, and S. Scherer, "Semantic 3D occupancy mapping through efficient high order CRFs)," in *Proceedings of the RSJ International Conference on Intelligent Robots and Systems (IROS*, pp. 590–597, IEEE, Vancouver, BC, Canada, September 2017.

[32] M. Runz, M. Buffier, and L. Agapito, "Maskfusion: real-time recognition, tracking and reconstruction of multiple moving objects," in *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 10–20, IEEE, Munich, Germany, October 2018.

[33] M. Rünz and L. Agapito, "Co-fusion: real-time segmentation, tracking and fusion of multiple objects," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pp. 4471–4478, IEEE, Singapore, May 2017.

[34] A. Hermans, G. Floros, and B. Leibe, "Dense 3D semantic mapping of indoor scenes from RGB-D images," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, pp. 2631–2638, IEEE, Hong Kong, China, May 2014.

[35] Y. Xiang and D. Fox, "DA-RNN: semantic mapping with data associated recurrent neural networks," in *Proceedings of the Robotics: Science and Systems XIII*, Seattle, WA, USA, July 2017.

[36] A. Aldoma, M. Zoltan-Csaba, F. Tombari et al., "Tutorial: point cloud library: three-dimensional object recognition and 6 DOF pose estimation," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, 2012.

[37] T. Y. Lin, M. Maire, S. Belongie et al., "Microsoft COCO: common objects in context," in *Computer Vision—ECCV 2014*, Lecture Notes in Computer Science, vol. 8693, Cham, Switzerland, Springer, 2014.

[38] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes challenge, (VOC2007) results," *Lecture Notes in Computer Science*, vol. 111, no. 1, pp. 98–136, 2007.

[39] B. Eckart and A. Kelly, "REM-Seg: a robust em algorithm for parallel segmentation and registration of point clouds," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Tokyo, Japan, November 2013.

[40] H. Oleynikova, Z. Taylor, M. Fehr et al., "Voxblox: incremental 3D euclidean signed distance fields for on-board MAV planning," in *Proceedings of the RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, September 2016.

[41] https://github.com/matterport/MaskRCNN.

[42] A. Dai, A. X. Chang, M. Savva et al., "ScanNet: richly-annotated 3D reconstructions of indoor scenes," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

[43] I. Armeni, S. Sax, A. R. Zamir et al., "Joint 2D-3D-semantic data for indoor scene understanding," 2017, https://arxiv.org/abs/1702.01105.

[44] Z. Fang, J. Ren, S. Marshall et al., "Triple loss for hard face detection," *Neurocomputing*, 2020, In press.

[45] Q. H. Pham, B. S. Hua, D. T. Nguyen, and S.-K. Yeung, "Real-time Progressive 3D Semantic Segmentation for Indoor Scene," in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, January 2019.

[46] Y. Yan, J. Ren, G. Sun et al., "Unsupervised image saliency detection with Gestalt-laws guided optimization and visual attention based refinement," *Pattern Recognition*, vol. 79, pp. 65–78, 2018.

[47] Y. Yan, J. Ren, H. Zhao et al., "Cognitive fusion of thermal and visible imagery for effective detection and tracking of pedestrians in videos," *Cognitive Computation*, vol. 10, no. 1, pp. 94–104, 2018.

[48] Z. Wang, J. Ren, D. Zhang, M. Sun, and J. Jiang, "A deep-learning based feature hybrid framework for spatiotemporal saliency detection inside videos," *Neurocomputing*, vol. 287, pp. 68–83, 2018.