Volumetric Instance-Aware Semantic Mapping and 3D Object Discovery

Margarita Grinvald, Fadri Furrer, Tonci Novkovic, Jen Jen Chung, Cesar Cadena, Roland Siegwart, Juan Nieto

***Abstract*—To autonomously navigate and plan interactions in real-world environments, robots require the ability to robustly perceive and map complex, unstructured surrounding scenes. Besides building an internal representation of the observed scene geometry, the key insight towards a truly functional understanding of the environment is the usage of higher-level entities during mapping, such as individual object instances. This work presents an approach to incrementally build volumetric object-centric maps during online scanning with a localized RGB-D camera. First, a per-frame segmentation scheme combines an unsupervised geometric approach with instance-aware semantic predictions to detect both recognized scene elements as well as previously unseen objects. Next, a data association step tracks the predicted instances across the different frames. Finally, a map integration strategy fuses information about their 3D shape, location, and, if available, semantic class into a global volume. Evaluation on a publicly available dataset shows that the proposed approach for building instance-level semantic maps is competitive with state-of-the-art methods, while additionally able to discover objects of unseen categories. The system is further evaluated within a real-world robotic mapping setup, for which qualitative results highlight the online nature of the method. Code is available at https://github.com/ethz-asl/voxblox-plusplus.**

***Index Terms*—RGB-D Perception; Object Detection, Segmentation and Categorization; Mapping**

I. INTRODUCTION

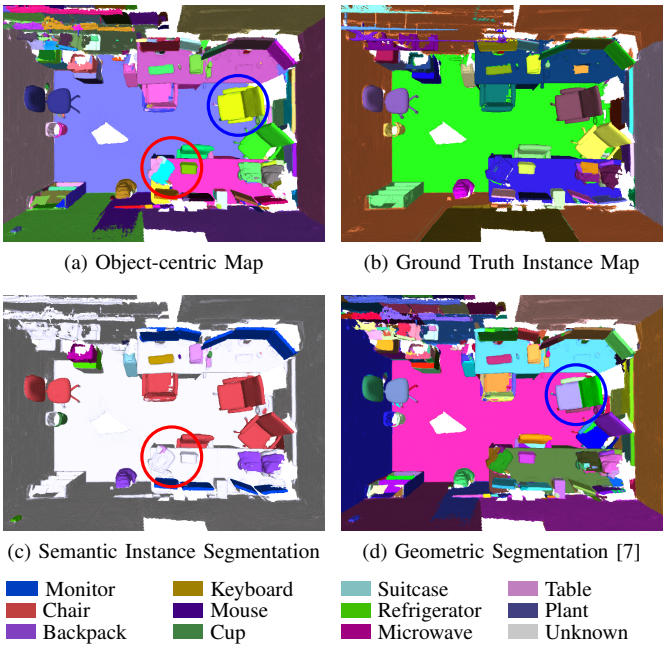
ROBOTS operating autonomously in unstructured, real-world environments cannot rely on a detailed *a priori* map of their surroundings for planning interactions with scene elements. They must therefore be able to robustly perceive the complex surrounding space and acquire task-relevant knowledge to guide subsequent actions. Specifically, to learn accurate 3D object models for tasks such as grasping and manipulation, a robotic vision system should be able to discover, segment, track, and reconstruct objects at the level of the individual instances. However, real-world scenarios exhibit large variability in object appearance, shape, placement, and location, posing a direct challenge to robotic perception. Further, such settings are usually characterized by open-set conditions, i.e. the robot will inevitably encounter novel objects of previously unseen categories.

Manuscript received: February 24, 2019; Revised: May 10, 2019; Accepted: May 29, 2019.

This paper was recommended for publication by Editor Eric Marchand upon evaluation of the Associate Editor and Reviewers' comments. This work was partially supported by ABB Corporate Research, the Amazon Research Awards program, and the Swiss National Science Foundation (SNF) through the National Centre of Competence in Research on Digital Fabrication.

The authors are with the Autonomous Systems Lab, ETH Zurich, 8092 Zurich, Switzerland {mgrinvald, fadri, ntonci, chungj, cesarc, rsiegwart, nietoj}@ethz.ch.

Digital Object Identifier (DOI): see top of this page.



(a) Object-centric Map (b) Ground Truth Instance Map

(c) Semantic Instance Segmentation (d) Geometric Segmentation [7]

Fig. 1: Reconstruction and object-level segmentation of an office scene using the proposed approach. Besides accurately describing the observed surface geometry, the final object-centric map in Figure (a) carries information about the location and 3D shape of the individual object instances in the scene. As opposed to a geometry-only segmentation from our previous work [7] shown in Figure (d), the proposed framework prevents over-segmentation of recognized articulated objects and segments them as one instance despite their non-convex shape (blue circle), assigning each a semantic category shown in Figure (c). At the same time, the proposed approach discovers novel, previously unseen object-like elements of unknown class (red circle). Note that different colors in Figure (a) and Figure (b) represent the different instances, and that a same instance in the prediction and ground truth is not necessarily of the same color. Progressive mapping of sequence 231 from the SceneNN [8] dataset is shown in the accompanying video available at http://youtu.be/Jvl42VJmYxg.

Computer vision algorithms have shown impressive results for the tasks of detecting individual objects in RGB images and predicting for each a per-pixel semantically annotated mask [1], [2]. On the other hand, dense 3D scene reconstruction has been extensively studied by the robotics community. Combining the two areas of research, a number of works successfully locate and segment semantically meaningful objects in reconstructed scenes while dealing with substantial intra-class variability [3]–[6]. Still, these methods can only detect objects from a fixed set of classes used during training, thus limiting interaction planning to a subset of the observed elements. In contrast, purely geometry-based methods [7], [9] are able to discover novel, previously unseen scene elements, under open-set conditions. However, such approaches tend to over-segment the reconstructed objects and additionally fail to

provide any semantic information about them, making high-level scene understanding and task planning impractical.

This paper presents an approach to incrementally build geometrically accurate volumetric maps of the environment that additionally contain information about the individual object instances observed in the scene. In particular, the proposed object-oriented mapping framework retrieves the pose and shape of recognized semantic objects, as well as of newly discovered, previously unobserved object-like instances. The proposed system builds on top of the incremental geometry-based scene segmentation approach from our previous work in [7] and extends it to produce a complete instance-aware semantic mapping framework. Figure 1 shows the sample object-centric map of an office scene reconstructed with the proposed approach.

The system takes as input the RGB-D stream of a depth camera with known pose.[1] First, a frame-wise segmentation scheme combines an unsupervised geometric segmentation of depth images [9] with semantic object predictions from RGB [1]. The use of semantics allows the system to infer the category of some of the 3D segments predicted in a frame, as well as to group segments by the object instance to which they belong. Next, the tracking of the individual predicted instances across multiple frames is addressed by matching per-frame predictions to existing segments in the global map via a data association strategy. Finally, observed surface geometry and segmentation information are integrated into a global Truncated Signed Distance Field (TSDF) map volume. To this end, the Voxblox volumetric mapping framework [10] is extended to enable the incremental fusion of class and instance information within the reconstruction. By relying on a volumetric representation that explicitly models free space information, i.e. distinguishes between unknown space and observed, empty space, the built maps can be directly used for safe robotic navigation and motion planning purposes. Furthermore, object models reconstructed with the voxel grid explicitly encode surface connectivity information, relevant in the context of robotic manipulation applications.

The capabilities of the proposed method are demonstrated in two experimental settings. First, the proposed instance-aware semantic mapping framework is evaluated on office sequences from the real-world SceneNN [8] dataset to compare against previous work on progressive instance segmentation of 3D scenes. Lastly, we show qualitative results for an online mapping scenario on a robotic platform. The experiments highlight the robustness of the presented incremental segmentation strategy, and the online nature of the framework.

The main contributions of this work are:

- A combined geometric-semantic segmentation scheme that extends object detection to novel, previously unseen categories.
- A data association strategy for tracking and matching instance predictions across multiple frames.
- Evaluation of the framework on a publicly available dataset and within an online robotic mapping setup.

[1] Please note that the current work focuses entirely on mapping, hence localization of the camera is assumed to be given.

II. RELATED WORK

A. Object detection and segmentation

In the context of object recognition in real-world environments, computer vision algorithms have recently shown some impressive results. Driven by the advances in deep learning using Convolutional Neural Network (CNNs), several architectures have been proposed for detecting objects in RGB images [11], [12]. Beyond simple bounding boxes, the recent Mask R-CNN framework [1] 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 instance-level semantic segmentation task [13].

One of the major limitations of learning-based instance segmentation methods is that they require extensive amounts of training data in the form of annotated masks for the specified object categories. Such annotated data can be expensive or even infeasible to acquire for all possible categories that may be encountered in a real-world scenario. Moreover, these algorithms can only recognize the fixed set of classes provided during training, thus failing to correctly segment and classify other, previously unseen object categories.

Some recent works aim to relax the requirement for large amounts of pixel-wise semantically annotated training data. Mask^X R-CNN [14] adopts a transfer method which only requires a subset of the data to be labeled at training time. SceneCut [15] and its Bayesian extension in [2] also operate under open-set conditions and are able to detect and segment novel objects of unknown classes. However, beyond detecting object instances in individual image frames, these methods alone do not provide a comprehensive 3D representation of the scene and, therefore, cannot be directly used for planning tasks such as manipulation or navigation.

B. Semantic object-level mapping

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 [16] 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 [16] does not provide any information about the geometry and relative placement of individual objects in the scene. Similar work in [17] additionally proposes to incrementally segment the scene using geometric cues from depth. However, geometry-based approaches tend over-segment articulated scene elements. Thus, without instance-level information, a joint semantic-geometric segmentation is not enough to group parts of the scene into distinct separate objects. Indeed, the instance-agnostic semantic segmentation in these works fails to build semantically meaningful maps to model individual object instances.

Previous work has addressed the task of mapping at the level of individual objects. SLAM++ [18] builds object-oriented maps by detecting recognized elements in RGB-D data, but is limited to work with a database of objects for which exact

geometric models need to be known in advance. A number of other works have addressed the task of detecting and segmenting individual semantically meaningful objects in 3D scenes without predefined shape templates [3]–[7], [9]. Recent learning-based approaches segment individual instances of semantically annotated objects in reconstructed scenes with little or no prior information about their exact appearance while at the same time handling substantial intra-class variability [3]–[6]. However, by relying on a strong supervisory signal of the predefined classes during training, a purely learning-based segmentation fails to discover novel objects of unknown class in the scene. As a result, these methods either fail to map objects that do not belong to the set of known categories and for which no semantic labels are predicted [3], [4], [6], or wrongly assign such previously unseen instances to one of the known classes [5]. In a real-world scenario, detecting objects only from a fixed set of classes specified during training limits interaction planning to a subset of all the observed scene elements.

In contrast, purely geometry-based methods operate under open-set conditions and are able to discover novel, previously unobserved objects in the scene [7], [9]. The work in [9] provides a complete and exhaustive geometric segmentation of the scene. Similarly, the Incremental Object Database (IODB) in [7] performs a purely geometric segmentation from depth data to reconstruct the shape of individual segments and build a consistent database of unique 3D object models. However, as mentioned previously, geometry-based approaches can result in unwanted over-segmentation of non-convex objects. Furthermore, by not providing semantic information, the two methods disallow high-level interaction planning. In addition to a complete geometric segmentation of the scene, the work in [19] performs object recognition on such segments from a database of known objects. While able to discover new, previously unseen objects and to provide for some semantic information, the main drawback lies in the requirement for exact 3D geometric models of the recognized objects to be known. This is not applicable to real-world environments, where objects with novel shape variations are inevitably encountered on a regular basis.

Closely related to the approach presented in this paper is the recent work in [20], with the similar aim of building dense object-oriented semantic maps. The work presents an incremental geometry-based segmentation strategy, coupled with the YOLO v2 [11] bounding box detector to identify and merge geometric segments that are detected as part of the same instance. One of the key differences to our approach is the choice of scene representation. Their system relies on the RGB-D SLAM system from [21] and stores the reconstructed 3D map through a surfel-based representation [22]. While surfels allow for efficient handling of loop closures, they only store the surface of the environment and do not explicitly represent observed free space [23]. That is, a surfel map does not distinguish between unseen and seen-but-empty space, and thus cannot be directly used for planning in robotic navigation or manipulation tasks where knowledge about free space is essential for safe operation [24]. Further, visibility determination and collision detection in surfel clouds can be significantly harder due to the lack of surface connectivity information. Therefore, as with all other approaches relying on sparse point or surfel clouds representations [3], [4], the object-oriented maps built in [20] cannot be immediately used in those robotic settings where an explicit distinction between unobserved space and free space is required.

Conversely, the volumetric TSDF-based representation adopted in this work does not discard valuable free space information and explicitly distinguishes observed empty space from unknown space in the 3D map. In contrast to all previous approaches, the proposed method is able to incrementally provide densely reconstructed volumetric maps of the environment that contain shape and pose information about both recognized and unknown object elements in the scene. The reconstructed maps are expected to directly benefit navigation and interaction planning applications.

III. METHOD

The proposed incremental object-level mapping approach consists of four steps deployed at each incoming RGB-D frame: (i) geometric segmentation, (ii) semantic instance-aware segmentation refinement, (iii) data association, and (iv) map integration. First, the incoming depth map is segmented according to a convexity-based geometric approach that yields segment contours which accurately describe real-world physical boundaries (Section III-A). The corresponding RGB frame is processed with the Mask R-CNN framework to detect object instances and compute for each a per-pixel semantically annotated segmentation mask. The per-instance masks are used to semantically label the corresponding depth segments and to merge segments detected as belonging to the same geometrically over-segmented, non-convex object instance (Section III-B). A data association strategy matches segments discovered in the current frame and their comprising instances to the ones already stored in the map (Section III-C). Finally, segments are integrated into the dense 3D map, where a fusion strategy keeps track of the individual segments discovered in the scene (Section III-D). An example illustrating the individual stages of the proposed approach is shown in Figure 2.

A. Geometric segmentation

Building on the assumption that real-world objects exhibit overall convex surface geometries, each incoming depth frame is decomposed into a set of object-like convex 3D segments following the geometry-based approach introduced in [7]. First, surface normals are estimated at every depth image point. Next, angles between adjacent normals are compared to identify concave region boundaries. Additionally, large 3D distances between adjacent depth map vertices are used to detect strong depth discontinuities. Surface convexity and the 3D distance measure are then combined to generate, at every frame t, a set $\mathcal{R}_t$ of closed 2D regions r_i in the current depth image and a set $\mathcal{S}_t$ of corresponding 3D segments s_i. Figure 2 shows the sample output of this stage.

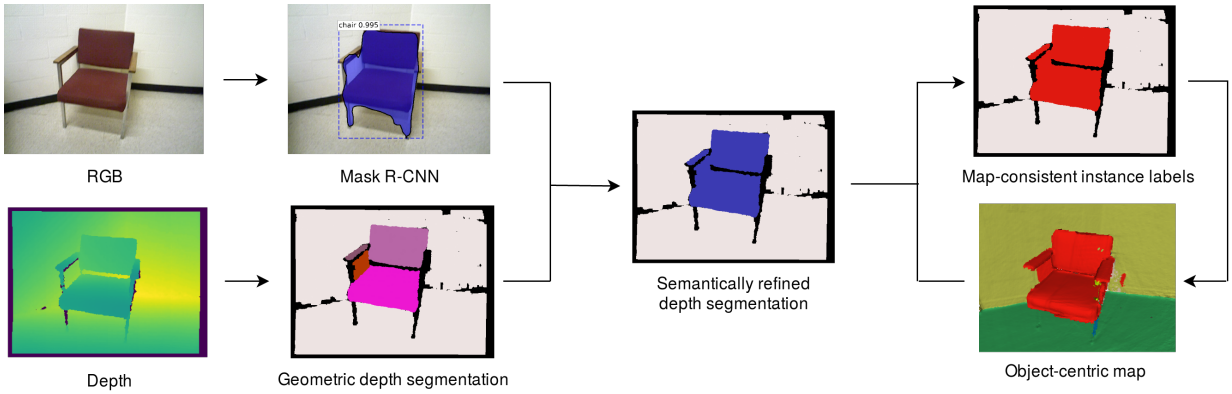


Fig. 2: The individual stages of the proposed approach for incremental object-level mapping are illustrated here with an example. At each new frame, the incoming RGB image is processed with the Mask R-CNN network to detect object instances and predict for each a semantically annotated mask. At the same time, a geometric segmentation decomposes the depth image into a set of convex 3D segments. The predicted semantic mask is used to infer class information for the corresponding depth segments and to refine over-segmentation of non-convex objects by grouping segments by the object instance they belong to. Next, a data association strategy matches segments predicted in the current frame to their corresponding instance in the global map to retrieve for each a map-consistent label. Finally, dense geometry and segmentation information from the current frame are integrated into the global map volume.

B. Semantic instance-aware segmentation refinement

To complement the unsupervised geometric segmentation of each depth frame with semantic object instance information, the corresponding RGB images are processed with the Mask R-CNN framework [1]. The network detects and classifies individual object instances and predicts a semantically annotated segmentation mask for each of them. Specifically, for each input RGB frame the output is a set of object instances, where the k-th detected instance is characterized by a binary mask M_k and an object category c_k. Figure 2 shows the sample output of Mask R-CNN.

The segmentation masks offer a straightforward way to associate each of the detected instances with one or more corresponding 3D depth segments $s_i \in \mathcal{S}_t$. Pairwise 2D overlaps $p_{i,k}$ between each $r_i \in \mathcal{R}_t$ and each predicted binary mask M_k are computed as the number of pixels in the intersection of r_i and M_k normalized by the area of r_i:

$$p_{i,k} = \frac{|r_i \cap M_k|}{|r_i|} \ . \tag{1}$$

For each region $r_i \in \mathcal{R}_t$ the highest overlap percentage p_i and index $\hat{k}_i$ of the corresponding mask M_k are found as:

$$p_i = \max_k \ p_{i,k} \tag{2}$$

$$\hat{k}_i = \arg\max_k \ p_{i,k} \ . \tag{3}$$

If $p_i > \tau_p$, the corresponding 3D segment s_i is assigned the object instance label $o_i = \hat{k}_i$ and a semantic category $c_i = c_{\hat{k}_i}$. Multiple segments in $\mathcal{S}_t$ assigned to the same object instance label o_i value indicate an over-segmentation of non-convex, articulated shapes being refined through semantic instance information. The unique set of all object instance labels o_i assigned to segments $s_i \in \mathcal{S}_t$ in the current frame is denoted by $\mathcal{O}_t$. All segments $s_i \in \mathcal{S}_t$ for which no mask M_k in the current frame exhibits enough overlap are assigned $o_i = c_i = 0$, denoting a geometric segment for which no semantic instance information could be predicted.

C. Data association

Because the frame-wise segmentation processes each incoming RGB-D image pair independently, it lacks any spatio-temporal information about corresponding segments and instances across the different frames. Specifically, this means that it does not provide an association between the set of predicted segments $\mathcal{S}_t$ and the set of segments $\mathcal{S}_{t+1}$. Further, segments belonging to the same object instance might be assigned different o_i label values across two consecutive frames, since these represent mask indices valid only within the scope of the frame in which such masks were predicted.

A data association step is proposed here to track corresponding geometric segments and predicted object instances across frames. To this end, we define a set of persistent geometric labels $\mathcal{L}$ and a set of persistent object instance labels $\mathcal{O}$ which remain valid throughout the entire mapping session. In particular, each s_j from the set of segments $\mathcal{S}$ stored in the map is defined by a unique geometric label $l_j \in \mathcal{L}$ through a mapping $L(s_j) = l_j$. At each frame we then look for a mapping $L_t(s_i) = l_j$ that matches predicted segments $s_i \in \mathcal{S}_t$ to corresponding segments $s_j \in \mathcal{S}$. Similarly, within the scope of a frame we seek to define a mapping $I_t(o_i) = o_m$ that matches object instances $o_i \in \mathcal{O}_t$ to persistent instance labels $o_m \in \mathcal{O}$ stored in the map.

To track spatial correspondences between segments $s_i \in \mathcal{S}_t$ identified in the current depth map and the set $\mathcal{S}$ of segments in the global map it is only necessary to consider the set $\mathcal{S}_v \subset \mathcal{S}$ of map segments visible in the current camera view. The pairwise 3D overlap $\Pi_{i,j}$ is computed for each $s_i \in \mathcal{S}_t$ and each $s_j \in \mathcal{S}_v$ as the number of points in segment s_i that, when projected into the global map frame using the known camera pose, correspond to a voxel which belongs to segment s_j. For each segment $s_j \in \mathcal{S}_v$, the highest overlap measure Π_j and the index $\hat{i}_j$ of the corresponding segment $s_i \in \mathcal{S}_t$ are found as,

$$\Pi_j = \max_i \ \Pi_{i,j} \tag{4}$$

$$\hat{i}_j = \arg\max_i \ \Pi_{i,j} \ . \tag{5}$$

Sequence ID	Bed	Chair	Sofa	Table	Books	Refrigerator	Television	Toilet	Bag	Average	Pham *et al.* [5]
011	-	75.0	50.0	100	-	-	-	-	-	**75.0**	52.1
016	100	0.0	0.0	-	-	-	-	-	-	33.3	**34.2**
030	-	54.4	100	55.6	14.3	-	-	-	-	56.1	**56.8**
061	-	-	100	33.3	-	-	-	-	-	**66.7**	59.1
078	-	33.3	-	0.0	47.6	100	-	-	-	**45.2**	34.9
086	-	80.0	-	0.0	0.0	-	-	-	0.0	20.0	**35.0**
096	0.0	87.5	-	37.5	0.0	-	0.0	-	50	**29.2**	26.5
206	-	58.3	100	60.0	-	-	-	-	100	**79.6**	41.7
223	-	12.5	-	75.0	-	-	-	-	-	**43.8**	40.9
255	-	-	-	-	-	75.0	-	-	-	**75.0**	48.6

TABLE I: Comparison to the 3D semantic instance-segmentation approach from Pham *et al.* [5]. Per-class AP is evaluated using an IoU threshold of 0.5 for each of the 10 evaluated sequences from the SceneNN [8] dataset. The class-averaged mAP value is compared to the results presented in [5]. The proposed approach improves over the baseline for 7 of the 10 sequences evaluated, however it is worth noting that the reported mAP values are evaluated on a smaller set of classes compared to the ones from [5].

Each segment $s_j \in \mathcal{S}_v$ with $\Pi_j > \tau_\pi$ determines the persistent label mapping for the corresponding maximally overlapping segment $s_{\hat{i}_j} \in \mathcal{S}_t$ from the current depth frame, i.e. $L_t(s_{\hat{i}_j}) = L(s_j)$. The τ_π threshold value is set to 20, and is used to prevent poorly overlapping global map segment labels from being propagated to the current frame. All segments $s_i \in \mathcal{S}_t$ that did not match to any segment $s_j \in \mathcal{S}_v$ are assigned a new persistent label l_{new} as $L_t(s_i) = l_{new}$. It is worth noting that, in contrast to previous work on segment tracking across frames [9], the proposed formulation disallows matching multiple segments in $\mathcal{S}_t$ to the same segment $s_j \in \mathcal{S}_v$. Without such constraint, information about a region in the map that was initially segmented as one now being segmented in two or more parts in the current frame would be lost, thus making it impossible to fix incorrect under-segmentations over time.

We introduce here the notation $\Phi(l_j, o_m)$ to denote the pairwise count in the global map between a persistent segment label $l_j \in \mathcal{L}$ and a persistent instance label $o_m \in \mathcal{O}$. $\Phi(l_j, o_m)$ is used here to determine the mapping $I_t(o_i) = o_m$ from instance labels $o_i \in \mathcal{O}_t$ to instance labels $o_m \in \mathcal{O}$. Specifically, for each segment $s_i \in \mathcal{S}_t$ with a corresponding $o_i \neq 0$ and no $I_t(o_i)$ defined yet, the persistent object label $\hat{o}_m$ with the highest pairwise count $\Phi(L_t(s_i), o_j) > 0$ is identified. The object label o_i is then mapped to $\hat{o}_m$ as $I_t(o_i) = \hat{o}_m$. Remaining o_i with no mapping $I_t(o_i)$ found are assigned a new persistent instance label o_{new} as $I_t(o_i) = o_{new}$. Following a similar reasoning as above, multiple labels $o_i \in \mathcal{O}_t$ are prevented from mapping to the same persistent label $o_m \in \mathcal{O}$ in order not to discard valuable instance segmentation information from the current frame.

The result of this data association step is a set of 3D segments $s_i \in \mathcal{S}_t$ from the current frame, each assigned a persistent segment label $l_j = L(s_i)$. Further, the corresponding object instance label is matched to a persistent label $o_m = I_t(o_i)$. Additionally, each segment $s_i \in \mathcal{S}_t$ is associated with the semantic object category c_i predicted by Mask R-CNN (Section III-B).

D. Map integration

The 3D segments discovered in the current frame, including some which are enriched with class and instance information, are fused into a global volumetric map. To this end, the Voxblox [10] TSDF-based dense mapping framework is extended to additionally encode object segmentation information. After projecting the segments into the global TSDF volume using the known camera pose, voxels corresponding to each projected 3D point are updated to store the incoming geometric segment label information, following the approach introduced in [7]. Additionally, for each $s_i \in \mathcal{S}_t$ integrated into the map at frame t with corresponding $o_i \neq 0$, the pairwise count between $l_j = L(s_i)$ and the object instance $o_m = I_t(o_i)$ and the pairwise count between l_j and the class c_i are incremented as,

$$\Phi(l_j, o_m) = \Phi(l_j, o_m) + 1 \tag{6}$$

$$\Psi(l_j, c_i) = \Psi(l_j, c_i) + 1 \ . \tag{7}$$

Each 3D segment $s_j \in \mathcal{S}$ in the global map volume is then defined by the set of voxels assigned to the persistent label l_j. If the segment represents a recognized, semantically annotated instance then it is also associated with an object label $\hat{o}_m = \arg\max_{o_m} \Phi(l_j, o_m)$ and a corresponding semantic class $\hat{c}_j = \arg\max_{c_j} \Psi(l_j, c_j)$.

IV. EXPERIMENTS

The proposed approach to incremental instance-aware semantic mapping is evaluated on a Lenovo laptop with an Intel Xeon E3-1505M eight-core CPU at 3.00 GHz and an Nvidia Quadro M2200 GPU with 4 GB of memory only used for the Mask R-CNN component. The Mask R-CNN code is based on the publicly available implementation from Matterport,[2] with the pre-trained weights provided for the Microsoft COCO dataset [13]. In all of the presented experimental setups, maps are built from RGB-D video with a resolution of 640x480 pixels.

[2]https://github.com/matterport/Mask_RCNN

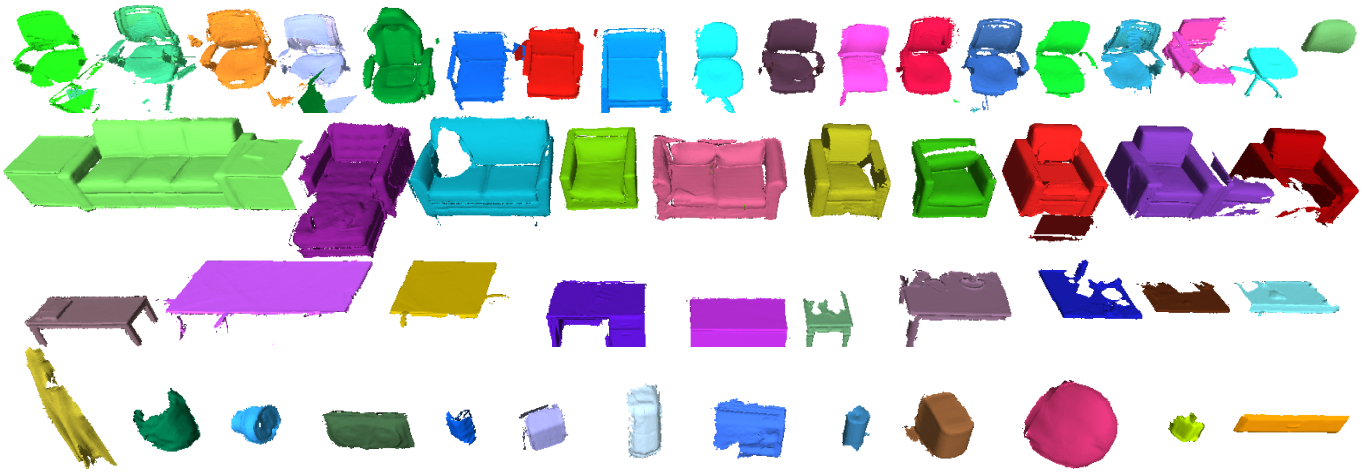


Fig. 3: Sample inventory of scene objects discovered during reconstruction of 10 indoor sequences from the SceneNN [8] dataset. By virtue of a combined geometric-semantic segmentation scheme, the proposed mapping framework is able to detect recognized elements from a set of known categories and simultaneously discover novel, previously unseen objects in the scene. Accordingly, the shown collection features selected elements of predicted class *chair* (first row), *couch* (second row), and *table* (third row), as well as a set of newly discovered objects without an associated semantic label (fourth row). Namely, the discovered objects correspond to (from left to right): a jacket, a plastic bag, two types of fans, a loudspeaker, a cardboard box, a computer case, a heater, a tissue paper roll, a kitchen appliance, a pillow, a tissue box, and a drawer. The individual models, shown here in the form of meshes, densely describe the reconstructed object shapes and provide detailed smooth surface definitions.

To compare against previous work in [5], we evaluate the 3D segmentation accuracy of the proposed dense object-level semantic mapping framework on real-world indoor scans from the SceneNN [8] dataset, improving over the baseline for most of the evaluated scenes. A sample inventory of object models discovered in these scenes is shown to contain recognized, semantically annotated elements, as well as newly discovered, previously unseen objects. Lastly, we report on the runtime performance of the proposed system.

The framework is further evaluated within an online setting, mapping an office floor traversed by a robotic platform. Although the system operates at only 1 Hz, qualitative results in the form of a semantically annotated object-centric reconstruction validate the online nature of the approach and show its benefits in real-world, open set conditions.

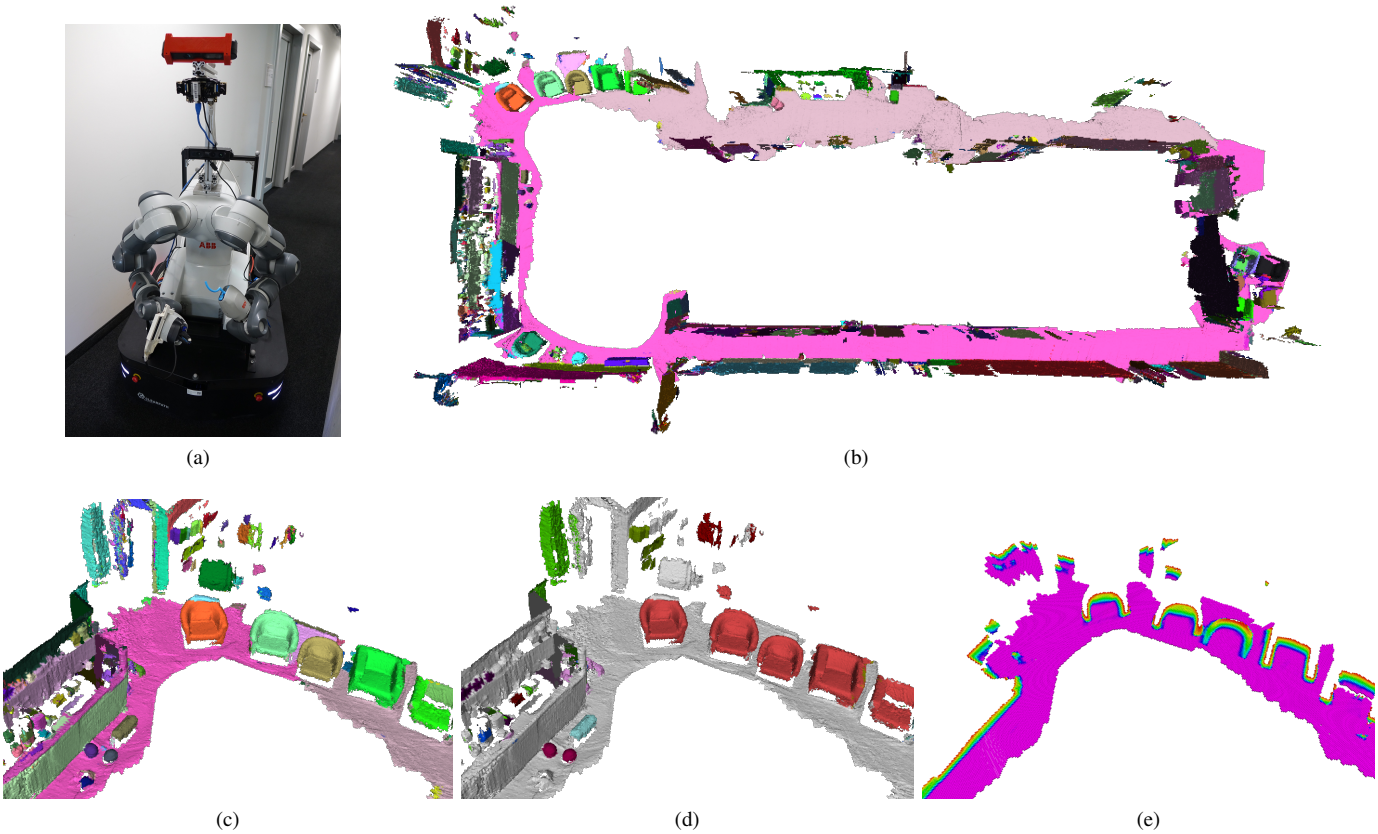
A. *Instance-aware semantic segmentation*

Several recent works explore the task of semantic instance segmentation of 3D scenes. The majority of these, however, take as input the full reconstructed scene, either processing it in chunks or directly as a whole. Because such methods are not constrained to progressively fusing predictions from partial observations into a global map but can learn from the entire 3D layout of the scene, these are not directly comparable to the approach presented in this work. Among the frameworks that instead explore online, incremental instance-aware semantic mapping, the work in [5] is, to the best of our knowledge, the only one to present quantitative results in terms of the achieved 3D segmentation accuracy. While a comparison with [5] does not provide any insight into the performance of the proposed unsupervised object discovery strategy, it can help to assess the efficacy of the semantic instance-aware segmentation component of our system.

In their work, Pham *et al.* [5] report instance-level 3D segmentation accuracy results for the NYUDv2 40 class task, which includes commonly-encountered indoor object classes, as well as structural, non-object categories, such as *wall*, *window*, *door*, *floor*, and *ceiling*. This set of classes is well-suited for semantic segmentation tasks in which the goal is to classify and label every single element, either voxel of surfel, of the 3D scene. Indeed, the approach in [5] initially employs a purely semantic segmentation strategy, and later clusters the semantically annotated scene into individual instances. However, a set of classes which includes non-object categories does not apply to the object-based segmentation approach proposed in this work. Therefore, rather than training on a class-set that does not meet the requirements and goals of the proposed framework, we relied on a Mask R-CNN model trained on the 80 Microsoft COCO object classes [13]. We then evaluated the segmentation accuracy on the 9 object categories in common between the NYUDv2 40 COCO class tasks. Specifically, we picked the 9 categories that have an unambiguous one-to-one mapping between the two sets.

The proposed approach is evaluated on the 10 indoor sequences from the SceneNN [8] dataset for which [5] reports instance-level segmentation results. For each scene, the per-class Average Precision (AP) is computed using an Intersection over Union (IoU) threshold of 0.5 over the predicted 3D segmentation masks. As [5] only provides class-averaged mean Average Precision (mAP) values, these are compared with mAP averaged over the 9 evaluated categories. The results in Table I show that the proposed approach outperforms the baseline on 7 of the 10 evaluated sequences, however it is worth noting again that the reported mAP values are computed over a smaller set of classes.

Besides evaluating the semantic instance-aware segmentation, Figure 3 additionally shows a sample inventory of selected object instances detected and densely reconstructed across the 10 sequences. Along with recognized, semantically annotated objects, the shown collection includes newly discovered scene elements, highlighting the benefits of the proposed unsupervised object discovery strategy.



(a) (b) (c) (d) (e)

Fig. 4: Figure (a) shows the robotic platform used for the online mapping experiment of an office floor. The map is reconstructed from RGB-D data recorded with two Primesense cameras mounted on an ABB YuMi robot attached to a Clearpath Ridgeback mobile base. The final map shown as a mesh in Figure (b) is reconstructed at a voxel size of 2 cm. Figure (c) shows a detail of the map where individual objects identified in the scene are represented with different colors. The corresponding semantic categories of the recognized instances are shown in Figure (d) using the same color coding as in Figure 1. Figure (e) shows a single horizontal slice at 1 m height of the reconstructed TSDF grid with magenta indicating observed free space, knowledge about which can directly benefit safe planning for navigation and interaction tasks.

Table II shows the running times of the individual components of the framework averaged over the 10 evaluated sequences. The numbers indicate that the system is capable of running at approximately 1 Hz on 640x480 input.

Component	Time (ms)
Mask R-CNN *	407
Depth segmentation *	753
Data association	136
Map integration	276

TABLE II: Measured execution times of each stage of the proposed incremental object-level mapping framework, averaged over the 10 evaluated sequences from the SceneNN [8] dataset with RGB-D input of 640x480 resolution. Inference through Mask R-CNN runs on GPU, while the remaining stages are implemented on CPU. The map resolution is set here to 1 cm voxels. Note that the components with * can be processed in parallel.

B. Online reconstruction and object mapping

The proposed system is evaluated in a real-life online mapping scenario. The robotic setup used for evaluation consists of a collaborative dual arm ABB YuMi robot mounted on an omnidirectional Clearpath Ridgeback mobile base. The platform is equipped with the custom-built visual-inertial sensor described in [25], used only for online localization. Two PrimeSense RGB-D cameras are mounted facing forwards and downwards at 45 degrees, respectively, to capture dense depth maps and color images at an increased effective field of view. The complete setup is shown in Figure 4a.

Within the course of 5 minutes, the mobile base was manually steered along a trajectory through an entire office floor. Real-time poses were estimated through a combination of visual-inertial and wheel odometry and online feature-based localization in an existing map built and optimized with Maplab [26]. During scanning, the RGB-D stream of the two depth cameras is recorded to be later fed through our mapping framework at a frame rate of 1 Hz, emulating real-time on-board operation. That is, any frames that exceed the processing abilities of the system are discarded and not used to reconstruct the object-level map of the scene. The accompanying video illustrates the progressive output of the incremental reconstruction and segmentation of the scene.

Qualitative results for the final object-centric map are shown in Figure 4. Despite only a subset of the incoming RGB-D frames being integrated into the map volume, the resulting reconstruction of the environment densely describes the observed surface geometry. The system is further able to detect recognized objects of known class, and to discover novel, previously unseen object-like elements in the scene. Reconstructed over a trajectory length of over 80 m with a voxel resolution of 2 cm, the entire map fits into 605 MB of memory, which is comparable with the memory usage of the bare

Voxblox framework. The final volumetric map additionally provides free space information, relevant for safe planning for robotic navigation and interaction tasks. Such tasks can be carried out in parallel, as the total computational load of the individual components of the framework corresponds to using only 5 out of the 8 CPU cores.

It is worth noting that the quality of the reconstruction in Figure 4 has been in part affected by empirically measured pose estimation errors accumulating up to 0.5 m. Because this work focuses entirely on mapping and assumes localization to be given, we leave the task of quantifying the impact of inaccurate localization on the map quality to future work.

V. CONCLUSIONS

We presented a framework for online volumetric instance-aware semantic mapping from RGB-D data. By reasoning jointly over geometric and semantic cues, a frame-wise segmentation approach is able to infer high-level category information about detected and recognized elements, and to discover novel objects in the scene, for which no previous knowledge about their exact appearance is available. The partial segmentation information is incrementally fused into a global map and the resulting object-level semantically annotated volumetric maps are expected to directly benefit both navigation and manipulation planning tasks.

Real-world experiments validate the online nature of the proposed incremental framework. However, to achieve real-time capabilities, the runtime performance of the individual components requires further optimization. A future research direction involves investigating the optimal way to fuse RGB and depth information within a unified per-frame object detection, discovery and segmentation framework.

ACKNOWLEDGMENT

The authors would like to thank T. Aebi for his help in collecting data for the office floor mapping experiment.

REFERENCES

[1] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 2980–2988.

[2] T. Pham, B. G. Vijay Kumar, T.-T. Do, G. Carneiro, and I. Reid, "Bayesian Semantic Instance Segmentation in Open Set World," in *Computer Vision – ECCV 2018*. Springer International Publishing, 2018, pp. 3–18.

[3] N. Sünderhauf, T. T. Pham, Y. Latif, M. Milford, and I. Reid, "Meaningful Maps With Object-Oriented Semantic Mapping," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 5079–5085.

[4] M. Rünz, M. Buffier, and L. Agapito, "MaskFusion: Real-Time Recognition, Tracking and Reconstruction of Multiple Moving Objects," in *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, Oct 2018, pp. 10–20.

[5] Q. Pham, B. Hua, T. Nguyen, and S. Yeung, "Real-Time Progressive 3D Semantic Segmentation for Indoor Scenes," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Jan 2019, pp. 1089–1098.

[6] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger, "Fusion++: Volumetric Object-Level SLAM," in *2018 International Conference on 3D Vision (3DV)*, Sep. 2018, pp. 32–41.

[7] F. Furrer, T. Novkovic, M. Fehr, A. Gawel, M. Grinvald, T. Sattler, R. Siegwart, and J. Nieto, "Incremental Object Database: Building 3D Models from Multiple Partial Observations," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 6835–6842.

[8] B. Hua, Q. Pham, D. T. Nguyen, M. Tran, L. Yu, and S. Yeung, "SceneNN: A Scene Meshes Dataset with aNNotations," in *2016 International Conference on 3D Vision (3DV)*, Oct 2016, pp. 92–101.

[9] K. Tateno, F. Tombari, and N. Navab, "Real-Time and Scalable Incremental Segmentation on Dense SLAM," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2015, pp. 4465–4472.

[10] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 1366–1373.

[11] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 6517–6525.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Neural Information Processing Systems (NIPS)*, 2015.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*. Springer International Publishing, 2014, pp. 740–755.

[14] R. Hu, P. Dollár, K. He, T. Darrell, and R. Girshick, "Learning to Segment Every Thing," in *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018, pp. 4233–4241.

[15] T. T. Pham, T. Do, N. Sünderhauf, and I. Reid, "SceneCut: Joint Geometric and Object Segmentation for Indoor Scenes," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1–9.

[16] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "SemanticFusion: Dense 3D Semantic Mapping with Convolutional Neural Networks," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 4628–4635.

[17] Y. Nakajima, K. Tateno, F. Tombari, and H. Saito, "Fast and Accurate Semantic Mapping through Geometric-based Incremental Segmentation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 385–392.

[18] 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 *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013, pp. 1352–1359.

[19] K. Tateno, F. Tombari, and N. Navab, "When 2.5D is not enough: Simultaneous Reconstruction, Segmentation and Recognition on dense SLAM," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 2295–2302.

[20] Y. Nakajima and H. Saito, "Efficient Object-Oriented Semantic Mapping With Object Detector," *IEEE Access*, vol. 7, pp. 3206–3213, 2019.

[21] V. A. Prisacariu, O. Kähler, S. Golodetz, M. Sapienza, T. Cavallari, P. H. Torr, and D. W. Murray, "InfiniTAM v3: A Framework for Large-Scale 3D Reconstruction with Loop Closure," *arXiv:1708.00783*, Aug. 2017.

[22] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb, "Real-Time 3D Reconstruction in Dynamic Scenes Using Point-Based Fusion," in *2013 International Conference on 3D Vision (3DV)*, June 2013, pp. 1–8.

[23] K. M. Wurm, D. Hennes, D. Holz, R. B. Rusu, C. Stachniss, K. Konolige, and W. Burgard, "Hierarchies of Octrees for Efficient 3D Mapping," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2011, pp. 4249–4255.

[24] E. Vespa, N. Nikolov, M. Grimm, L. Nardi, P. H. J. Kelly, and S. Leutenegger, "Efficient Octree-Based Volumetric SLAM Supporting Signed-Distance and Occupancy Mapping," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1144–1151, April 2018.

[25] J. Nikolic, J. Rehder, M. Burri, P. Gohl, S. Leutenegger, P. T. Furgale, and R. Siegwart, "A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 431–437.

[26] T. Schneider, M. Dymczyk, M. Fehr, K. Egger, S. Lynen, I. Gilitschenski, and R. Siegwart, "Maplab: An Open Framework for Research in Visual-Inertial Mapping and Localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1418–1425, July 2018.