Constructing Metric-Semantic Maps using Floor Plan Priors for Long-Term Indoor Localization

Nicky Zimmerman⋆   Matteo Sodano⋆   Elias Marks   Jens Behley   Cyrill Stachniss

Abstract— Object-based maps are relevant for scene under-standing since they integrate geometric and semantic informa-tion of the environment, allowing autonomous robots to robustly localize and interact with on objects. In this paper, we address the task of constructing a metric-semantic map for the purpose of long-term object-based localization. We exploit 3D object de-tections from monocular RGB frames for both, the object-based map construction, and for globally localizing in the constructed map. To tailor the approach to a target environment, we propose an efficient way of generating 3D annotations to finetune the 3D object detection model. We evaluate our map construction in an office building, and test our long-term localization approach on challenging sequences recorded in the same environment over nine months. The experiments suggest that our approach is suitable for constructing metric-semantic maps, and that our localization approach is robust to long-term changes. Both, the mapping algorithm and the localization pipeline can run online on an onboard computer. We release an open-source C++/ROS implementation of our approach.

I. INTRODUCTION

Localization and map construction are essential capabil-ities for mobile autonomous systems. Object-based maps, coupled with semantically-augmented localization are the foundation for more complex robotics tasks such as nav-igation and manipulation, as well as AR/VR applications. They enable to estimate the 3D geometry of the environment, and enrich it with semantic information. We focus on the metric-semantic map construction coupled with long-term object-based localization, using only monocular RGB frames and a floor plan prior. We use RGB cameras instead of RGB-D due to their lower power consumption and bandwidth requirements. For both tasks, we are interested in approaches that operate online on a mobile platform.

Previous works on 3D mapping leverage 3D reconstruction techniques to create a geometric description of the environ-ment [20] [21]. In recent years, the progress in semantic segmentation and object detection enabled the integration of semantic information into 3D reconstruction [8] [12] [26]. Some works have put emphasis on highly-accurate and detailed reconstruction of the environment [15] [28]. Other approaches use 3D bounding boxes as a valuable and com-pact abstraction for objects [14] [35]. Here, we leverage 3D object detection for enriching a floor plan map with metric-semantic information suitable for long-term object-based
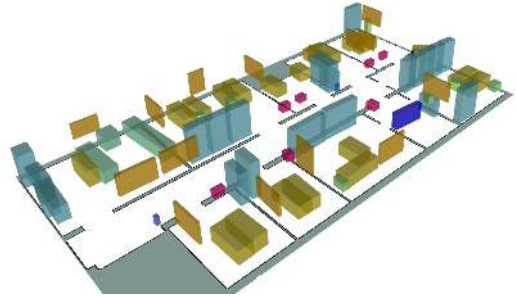
Fig. 1: A 3D semantic metric map combining a floor plan with 3D object bounding boxes built using our approach. This map is used for long-term localization in dynamic indoor environments. Different box colors indicate different object classes.

localization. Floor plan maps are commonly available and store information about unchanging structures, such as walls. Localization in floor plan maps is challenging due to the sparsity of geometric information; its advantage, however, is that information stored in floor plans rarely change. Often, additional sources of information are used to support robust localization, such as textual cues [38] and WiFi signals [10]. 3D object-based maps can be used to improve long-term localization in floor plan maps. Works such as the ones by Li et al. [14] [15] allow the construction of object-based maps from monocular RGB frames, but they do not include static structural elements such as walls, which have a critical importance for localization and planning tasks.

The main contribution of this paper is a global localiza-tion and object-based mapping system using 3D semantic information suitable for long-term operations in dynamic environments. We address the difficulty of creating 2D and 3D labels by proposing an efficient method for label gener-ation from RGB frames. These labels can then be exploited to achieve accurate performance on the target environment by finetuning off-the-shelf detection models. We analyze the performance of our detector, creating a probabilistic detection model that benefits both, map creation and object-based global localization. We utilize 3D object detection to construct object-centric maps, as seen in Fig. 1, augmenting readily-available floor plan maps with semantic information. We provide a global localization system for the pre-built object-based map with an uncertainty-aware sensor model for 3D object information, relying solely on monocular cameras. In sum, our approach is able to (i) generate 3D labels for fine-tuning of 3D object detection models, (ii) enrich floor plans with object information, (iii) and localize in such maps in an online fashion, using onboard computers. These claims are backed up by the paper and our experimental evaluation.

II. Related Work

A. 3D Object Detection

Scene understanding is the ability of recognizing objects and obtaining a semantic interpretation of the surrounding environment. In the last years, deep learning enabled tremendous advancements in image-based object detection [7] [25], semantic segmentation [29], and panoptic segmentation [3] [11] [31]. In the 3D domain, object detection approaches aimed to reproduce the efforts of 2D object detection in 3D [30] [33]. Most approaches, however, took alternative paths. Qi et al. [22] propose an end-to-end 3D object detection network inspired by the generalized Hough voting. The authors also propose an extension fusing 2D and 3D voting for boosting 3D object detection [23]. Recent efforts in 3D object detection tackle the problem by processing more than one frame at a time [14] [34], since performances of single-view approaches [15] are degraded by the depth-scale ambiguity [14]. Recently, Brazil et al. [2] proposed a single-view approach, called Cube R-CNN, that achieves state-of-the-art results for 3D object detection and solves the depth-scale ambiguity by introducing a training objective that incorporates a virtual depth.

B. Semantic Mapping

Map construction is a crucial elements of most robotics applications, and object-based maps are a step further towards higher level scene understanding. In recent years, learning-based approaches were applied to 3D object-based map construction. Many works were aimed at object-aware mapping [8] [12] or simultaneous localization and mapping (SLAM) [16] [32] with RGB-D cameras. In this paper, we focus on object-based mapping with known poses using monocular RGB images. For this task, Li et al. [14] use a graph neural network for data association and a super-quadratic formulation for their multi-view optimization. Li et al. [15] use Bayesian filtering to track objects and the Hungarian algorithm for data association. They perform a coarse-to-fine reconstruction, first representing the objects only as localized bounding boxes, and using shape codes for a detailed reconstruction. A similar approach is the one introduced by Runz et al. [28], that uses instead ray clustering for data association.

C. Semantic Localization

Long-term global localization in changing environments is a challenging task. Researchers investigated different sources of information such as textual cues [4] [38], wifi signal strength [18], and semantic cues to support pose estimation. Rottmann et al. [27] use a semantically-labeled occupancy grid map to localize in a Monte Carlo Localization (MCL) [5] framework, preforming place classification from RGB frames. Mendez et al. [17] extract semantic information about structural elements (walls, windows, doors) from floor plans, classify the pixels of an RGB-D image and match it against the semantic floor plan to localize. Both approaches do not use an explicit object-based map. Atanasov et al. [1] use objects as landmarks, defined by their 3D pose, semantic
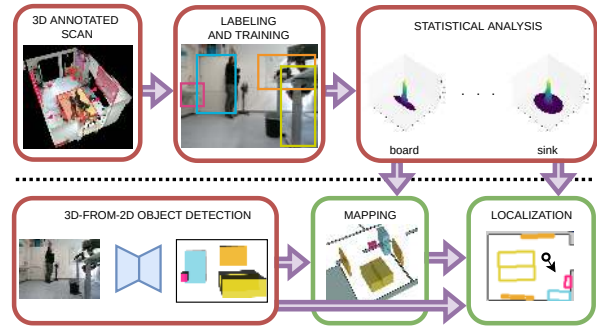

Fig. 2: An overview of our approach. Top row: offline pre-computation to adapt the approach to a specific environment. Bottom row: 3D-from-2D object detection and mapping, that can be executed on demand when the environment undergoes structural changes and a map update is necessary.

class and possible shape priors. The object are detected using a deformable part model [6], and the semantic information is integrated into a MCL framework. Ranganathan et al. [24] suggest a different representation for a semantic map, by using a constellation model. They detect objects using hand-crafted features, and rely on depth information provided by stereo cameras. Yi et al. [36] localize using a topological graph, where each node is characterized by the semantic objects in its vicinity. Similarly to these approaches, we incorporate the prediction uncertainty into our localization approach and use sparse object-based maps. We provide a pipeline for aquiring the metric semantic map. Most other approaches use 2D object detection, while we utilize 3D object information. Zimmerman et al. [37] incorporate high and low-level semantic cues from 2D object detection and geometric information from 2D LiDAR scanners, but use manual map creation. In contrast to this, our approach does not require LiDAR input.

III. Our Approach

We aim to enhance sparse floor plans with semantic cues and globally localize in these object-based maps using monocular cameras. In Sec. III-A, we present a way of creating 3D labels for fine-tuning 3D object detection models. Using such labels, we show how to fine-tune an object detection model and learn a noise model, see Sec. III-B, and use it to build a probabilistic object-based map. We construct a standard metric-semantic map using 3D object detections on posed RGB images, detailed in Sec. III-C. We exploit the object-based maps for object-based global localization, as described in Sec. III-D. An overview of our approach is visualized in Fig. 2.

A. Label Generation for 3D Object Detection

The performance of off-the-shelf 3D object detection models are often not accurate enough for the purpose of map building and they possibly focus on classes that are not beneficial for the purpose of indoor localization. For this reason, mapping and localization systems might need to finetune an existing model to the environment of interest. 3D object detection from RGB images requires 3D bounding

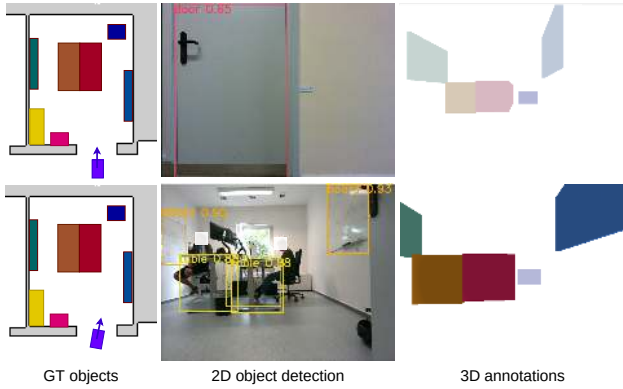GT objects   2D object detection   3D annotations

Fig. 3: 2D object detection to create better 3D annotations. Top: when rendering the ground truth objects from the camera, we have no information about dynamic objects like closed doors, which results in wrong annotations. The 2D object detection detects a door, and none of the objects in the 3D map. Therefore, no 3D annotations are generated (faded colors). Bottom: the 2D object detection detects tables and boards, but not the drawers due to occlusion. Therefore 3D annotations are generated only for the boards and the tables.

boxes annotations that include the object dimension, rotation and translation relative to the camera frame, in addition to 2D bounding box annotation for the RGB image with semantic category. Often, truncation and visibility of objects are also needed. Truncation refers to the percentage of the object in the camera frustum, while visibility is a measure of occlusion of an object by other object in the scene. Providing accurate labels in a real-world environment is challenging. In our approach, we construct the 3D labels based on a 3D model of the environment extracted from a 3D scan, 2D object detector and posed RGB images. As a pre-processing step, we manually annotate the 3D model of the environment with 3D bounding boxes of objects of interest. We project the 3D bounding boxes onto the posed camera frame using

$$\mathbf{x} = \mathsf{K} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ 0 & 1 \end{bmatrix} \mathbf{X}, \tag{1}$$

where $\mathbf{X} = (x, y, z, 1)^\top$ is a 3D point in the world coordinate system in homogenous coordinates, $\mathsf{K} \in \mathbb{R}^{3 \times 3}$ is the camera calibration matrix, $\boldsymbol{R}$ and $\boldsymbol{t}$ are the camera rotation matrix and translation vector, and $\mathbf{x}$ is a point in the image plane in homogenous coordinates. We then determine the visibility and truncation for every annotated object. To determine if an object is occluded by a dynamic obstacle, such as a person or a closed door, or by a static obstacle that is not annotated, we use a 2D object detector, as seen in Fig. 3. The detector is trained on classes of interest including dynamics and is made available by Zimmerman et al. [37]. For every object detected using the 2D object detector, we match a previously-annotated 3D bounding box based on the semantic class and the IoU between the projected and the detected 2D bounding box. In this way, for every posed RGB frame, the system annotates the detectable 3D objects, including their relative pose in the camera frame, dimensions, semantic class, 2D bounding boxes, visibility and truncation. Using these labels, we fine-tune a 3D object detector. We choose Cube R-
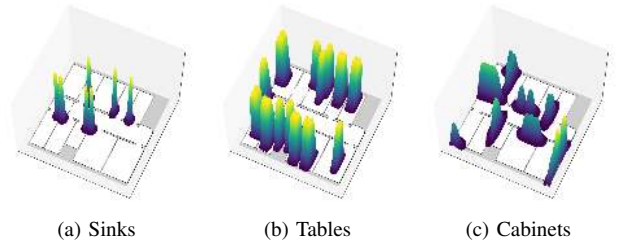

(a) Sinks   (b) Tables   (c) Cabinets

Fig. 4: Object probability map m_p which contains the per-object distribution $p_o(c \mid l)$, for specific classes of interest.

CNN [2], but our implementation allows integration of any other 3D object detector with the same output structure. Further details about the architecture and training procedure can be found in Sec. IV.

B. Statistical Analysis of 3D Object Detections

Given posed RGB frames, capturing objects belonging to classes of interest, we run inference on the images using Cube R-CNN, fine-tuned with our labels. Given two non-overlapping object predictions of the same class, a key issue is determining whether they belong to the same instance suffering from noisy detection or they are two separate instances in close vicinity. This is particularly common with semantic classes whose bounding boxes have one dimension which is substantially smaller than the other, such as white-boards. To address this problem, we analyze the per-class characteristic noise for detections. For each class, we build a 2D probability distribution by matching the 3D predictions to map objects $\mathbb{O}$, and projecting them in 2D on the ground plane. For matching, a predicted object can be assigned to a ground truth object only if the euclidean distance between the center of the predicted object $\boldsymbol{c}_o$ and the center of the ground truth object $\boldsymbol{c}_{GT}$ is smaller than a threshold δ.

In the case a predicted object can be assigned to multiple ground truth objects, we select the one with the highest 3D IoU. In case the prediction has no overlap with any ground truth, we match based on center distance only. Additionally, if no ground truth center is within distance δ from the center of the predicted object, we discard the prediction. Then, we aggregate predictions of different objects of the same class by transforming them into an object-centric coordinate system where the center of the associated ground truth object is the origin. For every class, we take the matched predictions and project their center on the 2D plane. Additionally, we discretize the 2D plane into cells of $5\,cm$ creating a histogram of occurrences. Then, we fit a Gaussian distribution

$$p(\boldsymbol{c} \mid l) = \frac{1}{\sqrt{2\pi|\boldsymbol{\Sigma}|}} \exp\left\{ -\frac{(\boldsymbol{c} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{c} - \boldsymbol{\mu})}{2} \right\}, \tag{2}$$

where $\boldsymbol{c}$ indicates the 2D center coordinate of the predicted object, and l is its semantic class.

Then, we transform the per-class distribution $p(\boldsymbol{c} \mid l)$ to be centered around each map object o of class l. We shift the mean $\boldsymbol{\mu}$ and rotate the covariance $\boldsymbol{\Sigma}$ of the class-specific distribution in local frame, according to the projected 2D

center point c_o and the rotation matrix $\boldsymbol{R}_o$ of an object:

$$\boldsymbol{\mu}_o = \boldsymbol{\mu} + \boldsymbol{c}_o, \qquad \boldsymbol{\Sigma}_o = \boldsymbol{R}_o \, \boldsymbol{\Sigma} \, \boldsymbol{R}_o^\top. \qquad (3)$$

Thus, the per-object distribution $p_o(\boldsymbol{c} \mid l)$ is given by the parameters $\boldsymbol{\mu}_o$ and $\boldsymbol{\Sigma}_o$ of Eq. (3).

We build the object probability map m_p by composing the individual Gaussians of the map objects $p_o(\boldsymbol{c} \mid l), \forall o \in \mathbb{O}$. Our object probability map m_p can be seen in Fig. 4.

C. 3D Semantic Map Construction

Given posed RGB images and a 3D object detector, we construct a metric semantic map to enhance a floor plan map. We define an object as

$$o = \{\boldsymbol{c}, \boldsymbol{D}, \boldsymbol{R}, \mathrm{I}_{\mathrm{active}}, n_{\mathrm{skip}}, n_{\mathrm{match}}\}, \qquad (4)$$

where $\boldsymbol{c}$ is the center of the object, $\boldsymbol{D} = (W, H, L)$ is the object's bounding box dimensions, $\boldsymbol{R}$ is the orientation of the bounding box, $\mathrm{I}_{\mathrm{active}} \in \{0, 1\}$ is a state that indicates whether an object is active or not, n_{skip} and n_{match} are two object-specific counters that will be explained in the following. An object is in the active state if it is in the camera frustum and is not occluded by other objects. We test for the visibility of an object by rendering the 3D scene into the camera frame, constraining the visibility with walls extracted from the floor plan map. We aggregate consecutive detections into a short-term, local map $\hat{m}_g$ by associating detected objects across different frames. Active objects are associated by means of the Hungarian algorithm [13] and a cost function defined by

$$\mathcal{C}(o_1, o_2) = \frac{1}{2} \left(\mathcal{C}_{\mathrm{IoU}} + \mathcal{C}_{\mathrm{cen}}\right) \qquad (5)$$

$$\mathcal{C}_{\mathrm{IoU}} = 1 - \mathrm{IoU}(o_1, o_2) \qquad (6)$$

$$\mathcal{C}_{\mathrm{cen}} = 1 - p_{o_1}(\boldsymbol{c}_{o_2} \mid l_{o_2}), \qquad (7)$$

where o_1 and o_2 are detected objects, $p_{o_1}(\boldsymbol{c}_{o_2} \mid l_{o_2})$ represents the goodness score of a center prediction based on the statistical analysis in Sec. III-B, $\boldsymbol{c}_{o_2}$ and l_{o_2} are the center and the semantic class of object o_2, respectively. After the robot has moved more than d_{xy} or rotated more than d_θ, we integrate the objects of $\hat{m}_g$ into the global object map m_g using the matching strategy described above. Given the associations computed by the Hungarian algorithm, we merge matched objects if the cost in Eq. (7) is below a threshold τ_{cost}. Otherwise, we initialize a new map object. When objects are merged, we increase the n_{match} count. If an active map object was not associated with a prediction, we increase the n_{skip}. When merging a prediction o_1 to a map object o_2, we use a weighted average to update the center and the dimensions of the bounding box:

$$\begin{aligned} \boldsymbol{c}_{o_2} &= \frac{n_{\mathrm{match}}^{o_2} \, \boldsymbol{c}_{o_2} + n_{\mathrm{match}}^{o_1} \, \boldsymbol{c}_{o_1}}{n_{\mathrm{match}}^{o_1} + n_{\mathrm{match}}^{o_2}} \\ \boldsymbol{D}_{o_2} &= \frac{n_{\mathrm{match}}^{o_2} \, \boldsymbol{D}_{o_2} + n_{\mathrm{match}}^{o_1} \, \boldsymbol{D}_{o_1}}{n_{\mathrm{match}}^{o_1} + n_{\mathrm{match}}^{o_2}}. \end{aligned} \qquad (8)$$

We update the rotation matrix by computing the weighted average and extracting the rotation matrix via SVD as proposed by Moakher et al. [19]:

$$\boldsymbol{U} \, \boldsymbol{\Sigma} \, \boldsymbol{V}^\top = \mathrm{SVD}\left(\frac{n_{\mathrm{match}}^{o_2} \, \boldsymbol{R}_{o_2} + n_{\mathrm{match}}^{o_1} \, \boldsymbol{R}_{o_1}}{n_{\mathrm{match}}^{o_1} + n_{\mathrm{match}}^{o_2}}\right) \qquad (9)$$

$$\boldsymbol{R}_{o_2} = \boldsymbol{U} \, \boldsymbol{V}^\top. \qquad (10)$$

Additionally, we update the weight of the object by summing up n_{match}. We purge objects from the map when

$$\frac{n_{\mathrm{match}}}{n_{\mathrm{skip}}} < \tau_{\mathrm{purge}},$$

where τ_{purge} is an empirically-chosen threshold.

We obtain room segmentation by applying morphological operations and connected-component analysis on the floor plan. This allows us to associate objects to rooms, so we update only objects located in the room we are currently in.

D. 3D Semantic Localization

We globally localize using an MCL [5] framework. MCL is a particle-based approach for estimating the robot state $\boldsymbol{x}_t$, given the map m, odometry input $\boldsymbol{u}_t$ and the observations $\boldsymbol{z}_t$. Each particle $\boldsymbol{s}_t$ is represented by a state $\boldsymbol{x}_t$ and a weight w_t, and the purpose of the sensor model is to assign a weight w_t to a particle $\boldsymbol{s}_t$ given an observation $\boldsymbol{z}_t$ and the map m. In our case, we localize in a 2D occupancy grid map, and therefore the robot state is $\boldsymbol{x}_t = (x, y, \theta)^\top$.

Our sensor model is based on the probabilistic analysis of the accuracy of the 3D object detection model, see Sec. III-B. Our observation $\boldsymbol{z}$ includes the object class l, the confidence score f and the object 3D bounding box $\boldsymbol{b}_{3\mathrm{D}}$ in the coordinate frame of the camera. For every particle $\boldsymbol{s}_t$, we transform the prediction into world frame using the particle state $\boldsymbol{x}_t$, and we project it on the ground, obtaining the corresponding 2D bounding box $\hat{\boldsymbol{b}}_{2\mathrm{D}}$.

To compute the weight of every particle we consider two measures – the object-based likelihood $p_o(\boldsymbol{z} \mid m_p, \boldsymbol{x}_t)$ and the shape similarity score p_s. Using the object probability map m_p computed in Sec. III-B, we sample p_o at the location indicated by the center $\boldsymbol{c}$ of the transformed 2D bounding box. To compute the likelihood of predicting a center $\boldsymbol{c}$ of class l, we consider the corresponding class distribution for every object given by the object probability map m_p:

$$p_o(\boldsymbol{z} \mid m_p, \boldsymbol{x}_t) = \max_{\{o \in \mathbb{O} \mid l_o = l\}} p_o(\boldsymbol{c} \mid l), \qquad (11)$$

where l_o is the semantic class of object o. For data association, we match the prediction to map object with the highest likelihood, referred to as o_{max}.

The metric semantic map m_s, constructed in Sec. III-C, stores the 3D bounding box of o_{max}, which we project to 2D bounding box $\hat{\boldsymbol{b}}_{2\mathrm{D}}^{\mathrm{max}}$. The shape similarity score is computed from the IoU of the 2D bounding boxes $\hat{\boldsymbol{b}}_{2\mathrm{D}}$ and $\hat{\boldsymbol{b}}_{2\mathrm{D}}^{\mathrm{max}}$:

$$p_g(\boldsymbol{z} \mid m_g, \boldsymbol{x}_t) = \exp\left(-\left(1 - \mathrm{IoU}\left(\hat{\boldsymbol{b}}_{2\mathrm{D}}, \hat{\boldsymbol{b}}_{2\mathrm{D}}^{\mathrm{max}}\right)\right)\right). \qquad (12)$$

The probability of detecting an object from particle state $\boldsymbol{x}_t$ is given by

$$p(\boldsymbol{z} \mid m, \boldsymbol{x}_t) = p_o \, p_g + (1 - p_o) \, \eta, \qquad (13)$$

Method	σ_{odom}	σ_{obs}	$r_{\max}$	τ_s	d_{xy}	d_θ
Ours	$(0.15\,\mathrm{m},\ 0.15\,\mathrm{m},\ 0.15\,\mathrm{rad})$	-	-	-	$0.1\,\mathrm{m}$	$0.03\,\mathrm{rad}$
HSMCL	$(0.15\,\mathrm{m},\ 0.15\,\mathrm{m},\ 0.15\,\mathrm{rad})$	6.0	$15.0\,\mathrm{m}$	0.6	$0.1\,\mathrm{m}$	$0.03\,\mathrm{rad}$

where η is an empirically computed constant, representing the weight of a false data association, and p_o and p_g are defined in Eqs. 11 and 12, respectively. When K objects are detected in a single frame, we compute the overall particle weight as a geometric average

$$p(\boldsymbol{z}_t \mid m,\ \boldsymbol{x}_t) = \prod_{k=1}^{K} p(\boldsymbol{z}_t^k \mid m,\ \boldsymbol{x}_t)^{\frac{1}{K}}. \qquad (14)$$

IV. EXPERIMENTAL EVALUATION

We present our experiments to show the capabilities of our method. The results of our experiments also support our key claims, which are: our approach is able to (i) generate 3D labels for fine-tuning of 3D object detection models, (ii) enrich floor plans with object information, (iii) and localize in such maps in an online fashion, using onboard computers.

A. Experimental Setup

To assess the performance of our approach, we made multiple recordings in our building. Our data collection platform was a Kuka YouBot with 2 Hokuyo UTM-30LX LiDARs, wheel encoders, 4 cameras with a joint coverage of $360°$ field-of-view, and an up-looking camera used strictly for evaluation purposes. The recordings span across 9 months, capturing changes to the lab furniture, varying amount of clutter, human movement and opening and closing of doors.

To extract precise ground truth information about the robot's pose, we use an external localization infrastructure based on densely placed (approx. $1\,\mathrm{tag/m}^2$) AprilTags, covering the ceiling of each room and corridor of our lab. In every frame captured with the up-looking camera, we detect multiple AprilTags computing the pose estimation in a least-squares fashion and achieving accuracy of under $3\,\mathrm{cm}$. A high resolution point cloud of the lab, generated with a terrestrial laser scanner, was also used to produce the 3D labels used in III-A, and the localization infrastructure was used to generate the poses for the RGB frames used the metric-semantic mapping in III-C.

For Cube R-CNN [2], we fine-tuned the ResNet34-based model [9] the authors provide for indoor perception. For doing this, we created a list of objects of interest suitable for long-term localization. The success of our approach depends on the appropriate choice of the classes of interest. It is important to consider the stability of the object classes, as discussed in et al. [37], and their observability and map coverage. We optimize our model with stochastic gradient descent for 100 epochs, with an initial learning rate of 0.0015 and a batch size of 12. For training, we recorded sequences T1-T7 on the second floor of our building.

As baseline, we compare against HSMCL [37], another semantic MCL framework. All experiments were executed
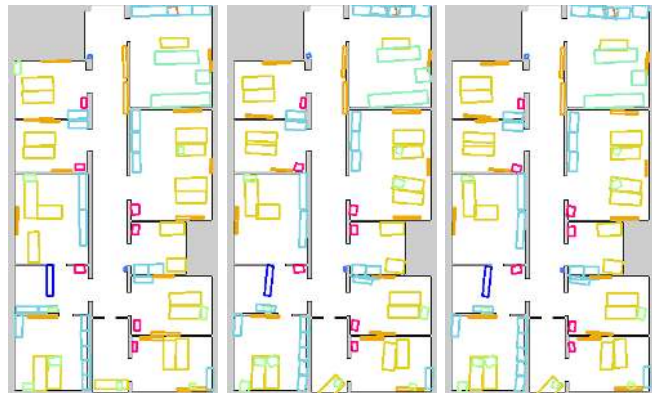

Fig. 5: 2D projection for the 3 maps. Left: ground truth map obtained with terrestrial laser scanner. Middle: KP map. Right: map built using scan matching (ICP).

with 5000 particles. The parameters for the different approaches are reported in Tab. I. We consider three metrics, the success rate, absolute trajectory error (ATE) after convergence and convergence time. We define convergence as the time when the estimate pose is within $0.3\,m$ radius of the ground truth pose, and the orientation is within $\frac{\pi}{4}$. After globally localizing, the tracked pose must not diverge for an accumulated $1.5\,s$. A localization run is successful if convergence is achieved in the first 95% of the sequence time and the tracked pose does not diverge. Each sequence was evaluated multiple times to account for the inherent stochasticity of the MCL framework, and the success rate is computed over multiple runs.

B. Mapping

To support our second claim, we evaluate the quality of our mapping pipeline. We use sequence M1, which include ground truth poses, RGB stream, and 2D LiDAR scans, while the robot was traversing the entire second floor. We evaluate our constructed maps by matching them against a ground truth (GT) map, extracted with a highly accurate 3D terrestrial laser scanner, and annotated manually. The creation of the ground truth map is labour-intense offline procedure that requires specialized sensors and equipment, providing sub-centimeters accuracy. It has been carried out for evaluation purposes only. Our constructed maps are built using the metric-semantic mapping pipeline describe in Sec. III-C, using data for sequence M1. For the first constructed map, we used precise poses extracted from our localization infrastructure, and we refer to it as known poses (KP) map. We also generated a map using sensor-based poses, estimated via ICP on 2D LiDAR scans. We aligned the estimated poses to an existing grid map, such as floor plan, by initializing the ICP with a known pose. We believe the poses could also be extracted with low-drift visual-inertial odometry pipelines [26]. We evaluated the accuracy of the generated maps directly by comparing it to the ground truth map, considering the IoU between objects in the object-based map, and precision and recall. As can be seen in Tab. II and Fig. 5, the constructed maps are well-matched to the ground truth map: the IoU scores between the ground truth map and

TABLE II: Computed metrics for the two constructed maps compared to the map obtained with a FARO 3D scan. KP was constructed based on known poses from infrastructure, and ICP was constructed with poses extracted from 2D LiDAR ICP.

	map	board	cabinet	desk	drawers	fire ext.	oven	plant	sink	sofa	table	AVG
IoU	KP	0.54	0.75	0.79	0.39	0.55	0.85	0.61	0.77	0.68	0.76	0.67
Pr	KP	1.00	1.00	1.00	0.56	1.00	1.00	1.00	1.00	1.00	1.00	0.96
Rc	KP	0.94	0.92	1.00	0.62	1.00	1.00	1.00	1.00	1.00	0.95	0.94
IoU	ICP	0.51	0.70	0.77	0.40	0.66	0.68	0.60	0.65	0.77	0.75	0.65
Pr	ICP	1.00	1.00	1.00	0.56	1.00	1.00	1.00	1.00	1.00	1.00	0.96
Rc	ICP	1.00	0.92	1.00	0.62	1.00	1.00	1.00	1.00	1.00	0.95	0.95

our generated map are high, while precision and recall are close to 1, indicating a low number of false positives and false negatives. We also evaluate the performance of robot localization based on the constructed maps. The localization accuracy for the KP map and the ICP map is on par with the accuracy for the ground truth (GT) map, see Tab. III, suggesting that our maps are well-suited for localization.

C. Long-Term Localization in CAD Floor Plans

To assess the performance of our localization approach, we recorded sequences R1-R10, through our building, spanning across nine months. These sequences include the addition and removal of furniture, dynamic obstacles, and quasi-static changes such as the closing and opening of doors. The starting points for the localization sequences were spread between different rooms and the corridor. Accuracy and success rate are reported in Tab. III for the three different maps. Each map was generated once, and has not been updated during the evaluation period. Convergence time averaged over sequences R1-R10 is $10.9\,s$ for ground truth map and $12.1\,s$ for KP map.

D. Baseline Comparisons for Semantic Localization

We compare our approach to three strategies for integrating 3D object information into MCL. The first baseline, called EDT-MCL, extends the commonly-used beam-end point model. Based on the semantic metric map m_s (see Sec. III-C), we create an Euclidean distance transform (EDT) for each semantic class.

The second approach, D-MCL, computes the particle weight based on the object probability map m_p and the likelihood p_o defined in Eq. (11). The third approach, O-MCL, is based on the overlap between a predicted bounding box and the semantic-metric map m_s. The weight is computed based on the overlap score described in Eq. (12).

As can be seen in Tab. IV, our approach outperforms the baselines. Both EDT-MCL and O-MCL do not incorporate information about the model noise. D-MCL makes use of the learned statistical information about the model performance, but only considers the center of the prediction, discarding valuable information about the object dimensions and rotation. Our approach leverages both the geometric information from the 3D bounding box and takes into account the characteristic model noise, which results in improved performance. Unlike HSMCL, our approach does not use LiDAR information, yet the localization is more robust.

E. Runtime

Our mapping and localization approaches both run online, onboard of a mobile platform. On our robots, we use an Intel NUC10i7FNK and a NVidia Jetson Xavier AGX. The 3D object detection runs at $\sim$9 FPS on the NVidia Jetson, and the sensor model executes at 60 FPS on the NUC10i7FNK. This performance is sufficient to construct a 3D metric semantic map and globally localize on our mobile platform. A longer video, including live demoes for both mapping and localization, and the code can be found on our GitHub repository at `https://github.com/PRBonn/SIMP`.

V. CONCLUSION

In this paper, we presented a novel approach for semantic global localization with a complementary 3D mapping procedure to build the object-based maps used for localization. The augmentation of floor plan maps with the 3D metric semantic maps assists navigation in cluttered and dynamic indoor environments. We show that our semantically-guided localization is reliable and accurate on both, the ground truth map and the map acquired with our proposed pipeline, benchmarking it over a dataset of challenging scenarios spanning over nine months. We compared our approach to similar approaches, and the experiments show how long-term localization can benefit from 3D metric semantic maps.

REFERENCES

[1] N. Atanasov, M. Zhu, K. Daniilidis, and G.J. Pappas. Localization from semantic observations via the matrix permanent. *Intl. Journal of Robotics Research (IJRR)*, 35(1-3):73–99, 2016.

[2] G. Brazil, J. Straub, N. Ravi, J. Johnson, and G. Gkioxari. Omni3D: A Large Benchmark and Model for 3D Object Detection in the Wild. *arXiv preprint:2207.10660*, 2022.

[3] B. Cheng, M.D. Collins, Y. Zhu, T. Liu, T.S. Huang, H. Adam, and L.C. Chen. Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[4] L. Cui, C. Rong, J. Huang, A. Rosendo, and L. Kneip. Monte-Carlo Localization in Underground Parking Lots Using Parking Slot Numbers. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[5] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 1999.

[6] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Visual Object Detection with Deformable Part Models. *Communications of the ACM*, 56(9):97–105, 2013.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

TABLE III: Evaluation of the map construction quality through long-term localization performance. The success rate for all maps on all reported sequences is 100%. We report ATE in [rad/m] format for 10 sequences recorded all across our lab in the span of nine months.

Method	Map	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	AVG
Ours	GT	0.089/0.14	0.065/0.11	0.070/0.14	0.043/0.17	0.065/0.16	0.075/0.15	0.073/0.17	0.077/0.20	0.060/0.17	0.047/0.12	0.066/0.15
Ours	KP	0.082/0.14	0.098/0.16	0.082/0.15	0.041/0.14	0.088/0.22	0.075/0.14	0.073/0.16	0.085/0.23	0.065/0.15	0.042/0.14	0.073/0.16
Ours	ICP	0.078/0.14	0.115/0.11	0.085/0.16	0.039/0.15	0.077/0.18	0.081/0.14	0.074/0.18	0.083/0.27	0.057/0.12	0.045/0.11	0.073/0.16

TABLE IV: Baseline comparison for long-term localization on the ground truth map. We report success rate and ATE in [rad/m] format for 10 sequences recorded all across our lab in the span of nine months.

Method	Map	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	AVG
HSMCL	GT	0%	60%	100%	100%	100%	0%	100%	100%	100%	20%	68%
EDT-MCL	GT	100%	80%	100%	100%	100%	100%	100%	100%	100%	0%	88%
D-MCL	GT	100%	60%	100%	100%	100%	100%	100%	100%	80%	0%	84%
O-MCL	GT	100%	80%	100%	100%	40%	0%	100%	100%	100%	0%	72%
Ours	GT	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
HSMCL	GT	-/-	-/-	0.101/0.30	0.147/0.32	0.084/0.25	-/-	0.052/0.19	0.063/0.24	0.078/0.30	-/-	0.088/0.27
EDT-MCL	GT	0.094/0.17	-/-	0.075/0.18	0.043/0.20	0.122/0.16	0.076/0.20	0.073/0.25	0.058/0.23	0.060/0.16	-/-	0.075/0.19
D-MCL	GT	0.068/0.13	-/-	0.055/0.18	0.036/0.17	0.075/0.16	0.037/0.21	0.046/0.24	0.039/0.22	-/-	-/-	0.051/0.19
O-MCL	GT	0.082/0.14	-/-	0.054/0.15	0.045/0.16	-/-	-/-	0.056/0.18	0.049/0.19	0.061/0.14	-/-	0.058/0.16
Ours	GT	0.089/0.14	0.065/0.11	0.070/0.14	0.043/0.17	0.065/0.16	0.075/0.15	0.073/0.17	0.077/0.20	0.060/0.17	0.047/0.12	0.066/0.15

[8] M. Grinvald, F. Furrer, T. Novkovic, J.J. Chung, C. Cadena, R. Siegwart, and J. Nieto. Volumetric Instance-Aware Semantic Mapping and 3D Object Discovery. *IEEE Robotics and Automation Letters (RA-L)*, 4(3):3037–3044, 2019.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] S. Ito, F. Endres, M. Kuderer, G. Tipaldi, C. Stachniss, and W. Burgard. W-RGB-D: Floor-Plan-Based Indoor Global Localization Using a Depth Camera and WiFi. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2014.

[11] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár. Panoptic Segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[12] I. Kostavelis and A. Gasteratos. Semantic mapping for mobile robotics tasks: A survey. *Journal on Robotics and Autonomous Systems (RAS)*, 66:86–103, 2015.

[13] H. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[14] K. Li, D. DeTone, Y.F.S. Chen, M. Vo, I. Reid, H. Rezatofighi, C. Sweeney, J. Straub, and R. Newcombe. ODAM: Object Detection, Association, and Mapping using Posed RGB Video. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[15] K. Li, H. Rezatofighi, and I. Reid. MOLTR: Multiple Object Localization, Tracking and Reconstruction From Monocular RGB Videos. *IEEE Robotics and Automation Letters (RA-L)*, 6(2):3341–3348, 2021.

[16] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger. Fusion++: Volumetric Object-Level SLAM. In *Proc. of the Intl. Conf. on 3D Vision*, 2018.

[17] O. Mendez, S. Hadfield, N. Pugeault, and R. Bowden. Sedar-semantic detection and ranging: Humans can localise without lidar, can robots? In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2018.

[18] R. Miyagusuku, A. Yamashita, and H. Asama. Data information fusion from multiple access points for wifi-based self-localization. *IEEE Robotics and Automation Letters (RA-L)*, 4(2):269–276, 2018.

[19] M. Moakher. Means and averaging in the group of rotations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):1–16, 2002.

[20] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto. Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[21] E. Palazzolo, J. Behley, P. Lottes, P. Giguere, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[22] C.R. Qi, O. Litany, K. He, and L.J. Guibas. Deep Hough Voting for 3D Object Detection in Point Clouds. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2019.

[23] C.R. Qi, X. Chen, O. Litany, and L.J. Guibas. Imvotenet: Boosting 3d object detection in point clouds with image votes. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[24] A. Ranganathan and F. Dellaert. Semantic modeling of places using objects. In *Proc. of Robotics: Science and Systems (RSS)*, 2007.

[25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once: Unified, Real-Time Object Detection. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[26] A. Rosinol, M. Abate, Y. Chang, and L. Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[27] A. Rottmann, O. Martínez-Mozos, C. Stachniss, and W. Burgard. Place Classification of Indoor Environments with Mobile Robots using Boosting. In *Proc. of the National Conf. on Artificial Intelligence (AAAI)*, 2005.

[28] M. Runz, K. Li, M. Tang, L. Ma, C. Kong, T. Schmidt, I. Reid, L. Agapito, J. Straub, S. Lovegrove, and R. Newcombe. FroDO: From Detections to 3D Objects. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[29] D. Seichter, M. Köhler, B. Lewandowski, T. Wengefeld, and H.M. Gross. Efficient RGB-D Semantic Segmentation for Indoor Scene Analysis. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[30] S. Shi, X. Wang, and H. Li. PointRCNN: 3D Object Proposal Generation and Detection From Point Cloud. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[31] M. Sodano, F. Magistri, T. Guadagnino, J. Behley, and C. Stachniss. Robust Double-Encoder Network for RGB-D Panoptic Segmentation. *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[32] J. Stückler, N. Biresev, and S. Behnke. Semantic mapping using object-class segmentation of RGB-D images. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[33] T. Wang, X. Zhu, J. Pang, and D. Lin. FCOS3D: Fully convolutional one-stage monocular 3d object detection. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[34] Y. Wang, V.C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon. DETR3D: 3D object detection from multi-view images via 3D-to-2D queries. In *Proc. of the Conf. on Robot Learning (CoRL)*, 2022.

[35] S. Yang and S. Scherer. Cubeslam: Monocular 3-D object SLAM. *IEEE Trans. on Robotics (TRO)*, 35(4):925–938, 2019.

[36] C. Yi, I.H. Suh, G.H. Lim, and B.U. Choi. Bayesian robot localization using spatial object contexts. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2009.

[37] N. Zimmerman, T. Guadagnino, X. Chen, J. Behley, and C. Stachniss. Long-Term Localization Using Semantic Cues in Floor Plan Maps. *IEEE Robotics and Automation Letters (RA-L)*, 8(1):176–183, 2023.

[38] N. Zimmerman, L. Wiesmann, T. Guadagnino, T. Läbe, J. Behley, and C. Stachniss. Robust Onboard Localization in Changing Environments Exploiting Text Spotting. *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.