# SeDAR – Semantic Detection and Ranging: Humans can localise without LiDAR, can robots?

Oscar Mendez
University of Surrey
Guildford GU2
o.mendez@surrey.ac.uk

Simon Hadfield
University of Surrey
Guildford GU2
s.hadfield@surrey.ac.uk

Nicolas Pugeault
University of Exeter
Exeter EX4
n.pugeault@exeter.ac.uk

Richard Bowden
University of Surrey
Guildford GU2
r.bowden@surrey.ac.uk

***Abstract*— How does a person work out their location using a floorplan? It is probably safe to say that we do not explicitly measure depths to every visible surface and try to match them against different pose estimates in the floorplan. And yet, this is exactly how most robotic scan-matching algorithms operate. Similarly, we do not extrude the 2D geometry present in the floorplan into 3D and try to align it to the real-world. And yet, this is how most vision-based approaches localise.**

**Humans do the exact opposite. Instead of depth, we use high level semantic cues. Instead of extruding the floorplan up into the third dimension, we collapse the 3D world into a 2D representation. Evidence of this is that many of the floorplans we use in everyday life are not accurate, opting instead for high levels of discriminative landmarks.**

**In this work, we use this insight to present a global localisation approach that relies solely on the semantic labels present in the floorplan and extracted from RGB images. While our approach is able to use range measurements if available, we demonstrate that they are unnecessary as we can achieve results comparable to state-of-the-art without them.**

## I. Introduction

Indoor localisation is perhaps one of the most crucial aspects for any robotic system. It allows robots to interact with the world and provides a representation and understanding that can be shared with humans and other agents. Traditional Vision-Based Simultaneous Localization and Mapping (VSLAM) systems can provide localisation within a map that is built on-the-fly. However, VSLAM systems are liable to drift in terms of both pose and scale. They can also become globally inconsistent in the case of failed loop closures. Finally, even in the case of no scale drift and correct loop closures, a VSLAM system can only ever guarantee global consistency *internally*. This means that while pose estimates are globally consistent, they are only valid within the context of the VSLAM system. There are no guarantees, at least in vision-only systems, that we can directly map the reconstruction to the real world (or between agents).

This problem is normally addressed by having a localisation system that can relate the pose of the robot to a pre-existing map. Examples of global localisation frameworks include the Global Positioning System (GPS) and traditional Monte-Carlo Localisation (MCL). MCL has the ability to localise within an existing floorplan (which can be safely assumed to be available for most indoor scenarios). This is a highly desirable trait, as it implicitly eliminates drift, is globally consistent and provides a way for the created 3D reconstructions to be related to the real world without having to perform expensive post-hoc optimizations. Traditionally, the range-based scans required by MCL have been produced by expensive sensors such as Light Detection And Ranging (LiDAR). More recently, modern robotic platforms have used RGB-D cameras as a cheap and low-footprint alternative. This has made vision-based floorplan localisation an active topic in the literature.

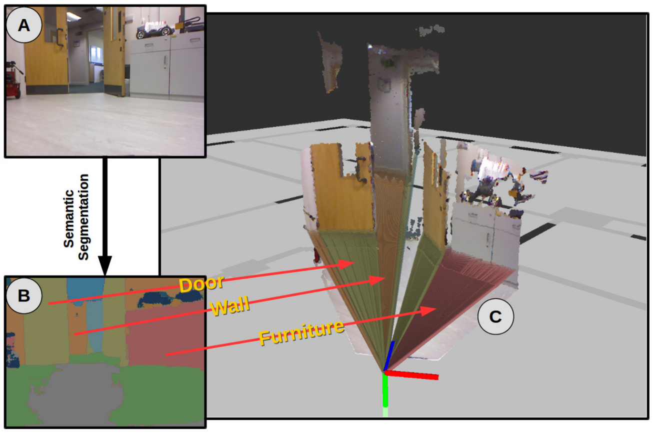


Fig. 1: A) RGB Image, B) CNN-Based Semantic Labelling and C) Sample SeDAR Scan within floorplan.

However, while several vision-based approaches have been proposed, they normally use heuristics to lift the 2D plan into the 3D coordinate system of VSLAM. Examples include Liu *et al.* [17], who use visual cues such as Vanishing Points (VPs) or Chu *et al.* [5] who perform piecemeal 3D reconstructions that can then be fitted back to an extruded floorplan. A common problem with these approaches is that the 3D data extracted from the image is normally orthogonal to the floorplan that it is meant to localise in. This means that assumptions must be made about dimensions not present in the floorplan. These approaches also do not fully exploit the floorplan, ignoring the semantic information.

We propose a fundamentally different approach that is inspired by how humans perform the task. Instead of discarding valuable semantic information, we use a Convolutional Neural Network (CNN)-based encoder-decoder to extract high-level semantic information. We then collapse all semantic information into 2D in order to reduce the assumptions about the environment. We then use these labels, image geometry and (optionally) depth along with a semantically labelled floorplan to create a state-of-the-art sensing and localisation framework.

Semantic Detection and Ranging (SeDAR) is an innovative human-inspired framework that combines new semantic sensing capabilities with a novel semantic Monte-Carlo Localisation (MCL) approach. As an example, figure 1 shows a sample SeDAR scan localised in the floorplan. We show that SeDAR has the ability to surpass LiDAR-based MCL approaches. SeDAR also has the ability to perform drift-free local, as well as global, localisation. Furthermore, experimental results show that the semantic labels are

The Titan X used for this research was donated by the NVIDIA Corporation.

sufficiently strong visual cues such that depth estimates are no longer needed. Not only does this vision-only approach perform comparably to depth-based methods, it is also capable of coping with map inaccuracies more gracefully.

This paper describes the process by which SeDAR is used as a novel human-inspired sensing and localisation framework. In section III-A, semantically salient elements are extracted from a floorplan. Section III-B describes how these semantic elements are identified in the robot's camera by using a state-of-the-art CNN-based semantic segmentation algorithm and presented as a novel sensing modality. We then present the three main contributions of this paper. First, section III-C introduces a novel motion model that includes a "ghost factor" that uses semantic information to influence how particles move through occupied space. Second, section III-D introduces a novel sensor model that estimates observation likelihoods using semantic information, range and bearing information. Third, section III-E introduces a second novel motion model that uses semantic and bearing information to allow observation likelihoods to be estimated from an RGB image only. Finally, in section IV we present the results obtained by using our approach in multiple sensing modalities.

## II. Literature Review

Monte-Carlo Localisation (MCL) was made possible by the arrival of accurate range-based sensors such as SOund Navigation And Ranging (SONAR) and Light Detection And Ranging (LiDAR). These approaches, which we call Range-Based Monte-Carlo Localisation (RMCL), are robust and reliable and still considered state-of-the-art in many robotic applications. Recent advances in computer vision have made it possible for us to imagine new types of perceptual sensors which are capable of semantic understanding of a scene. Semantic sensing modalities, such as SeDAR, have the ability to revolutionize MCL.

RMCL was first introduced by Fox *et al.* [11] and Dellaert *et al.* [9]. RMCL improved the Kalman Filter based state-of-the-art by allowing multi-modal distributions to be represented. It also solved the computational complexity of grid-based Markov approaches. However, these approaches require expensive LiDAR and/or SONAR sensors to operate reliably. Instead, Dellaert *et al.* [8] extended their approach to use vision-based sensor models. Vision-based MCL allowed the use of rich visual features and cheap sensors, but had limited performance compared to the more robust LiDAR-based systems.

With the rising popularity of RGB-D sensors, more robust vision-based MCL approaches became possible. Paton and Kosecka [19] use a combination of feature matching and Iterative Closest Point (ICP) to perform pose estimation and localisation. Brubaker *et al.* [4] used visual odometry and pre-existing roadmaps in a joint MCL/closed-form approach in order to localise a moving car. Fallon *et al.* [10] presented a robust MCL approach that used a low fidelity *a priori* map to localise in, but required the space to be traversed by a depth sensor beforehand. Winterhalter *et al.* [26] performed MCL, but based the likelihood of the sensor model on the normals of an extruded floorplan. Chu *et al.* [5] is the closest to us, they attempted to mimic the human thinking process by creating piecemeal reconstructions of an extruded floorplan, the MCL sensor model was then based on matches against these reconstructions. These MCL-based approaches tend to be robust, but they operate entirely on the *geometric* information present in the floorplan and therefore require depth images either from sensors and/or reconstructions. By contrast our approach aims to use *non-geometric* semantic information present in the floorplan in order to perform the localisation.

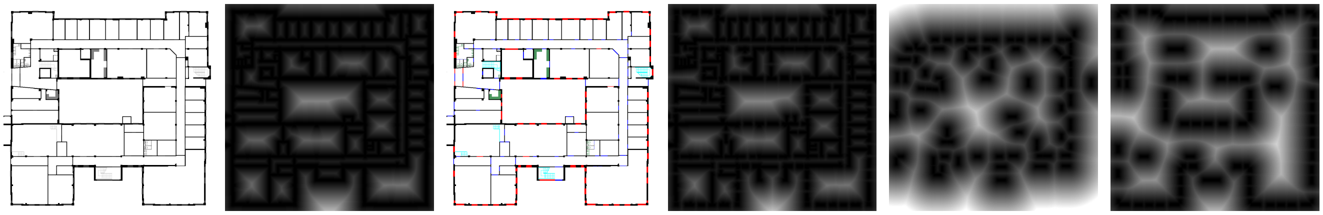
Our approach is most similar to bearing-only [3], [24] approaches, where the angular distrubtion of known landmarks can be used to deduce the location of a robotic agent. However, our approach is fundamentally different from these methods, as it does not require active landmarks with known positions. Instead, we rely on the semantic information already present in the world: we use the angular distribution of detected semantic labels to localise a robot.

While the field of MCL evolved in the robotics community, in vision, the non-MCL-based field of floorplan localisation became more popular. Melbouci *et al.* [18] used extruded floorplans, but performed local bundle adjustments instead of MCL. Shotton *et al.* [21] used regression forests to predict the correspondences of every pixel in the image to a known 3D scene, they then combined this in a RANdom Sample And Consensus (RANSAC) approach in order to solve the camera pose. Chu *et al.* [6] use information from the floorplans and Google StreetView in order to reason about the geometry of the building and perform a robust reconstruction. The most similar work to our approach is Wang *et al.* [25] who use text detection from shop fronts as semantic cues to localise in the floorplan of a shopping centre and Liu *et al.* [17] who use floorplans as a source of geometric and semantic information, combined with vanishing points, to localise monocular cameras. These vision-based approaches tend to use more of the non-geometric information present in the floorplan. However, a common trend is that assumptions must be made about geometry not present in the floorplan (*e.g.* ceiling height). The floorplan is then extruded out into the 3$^{rd}$ dimension to allow approaches to use the information present in the image. By contrast, our approach aims to extract the information from the image and collapse the 3D world down into the 2D floorplan where localisation can be performed. This provides a 3-Degrees of Freedom (DoF) localisation requiring less assumptions about the environment.

Recently, advances in Deep Learning have made robust semantic segmentation models widely available. Approaches like that of Badrinarayanan *et al.* [1], Kendal *et al.* [13] and Long *et al.* [20] have made semantically informed approaches possible. One such approach is Tateno *et al.* [23] who use the CNN-based depth and semantic label predictions of Laina *et al.* [15] to aid in their Simultaneous Localization and Mapping (SLAM) pipeline. Lee *et al.* [16] extend the approach of Badrinarayanan *et al.* [1] to directly estimate room layout keypoints. While many such approaches exist, they mainly focus on extracting the room layout based on Manhattan world assumptions. Instead, this work proposes to use CNN-based semantic segmentation (that is understandable to humans) in order to extract labels that are inherently present in human-readable floorplans. This allows us to take all that information and collapse it into a 3-DoF problem, making our approach more tractable than competing 6-DoF approaches while avoiding additional assumptions.

## III. Methodology

The problem with state-of-the-art approaches is that they are limited to range information. Instead, we present a novel semantic sensing and localisation framework called SeDAR



(a) Original floorplan (b) Occupancy (c) Semantic Floorplan (d) Wall Likelihood (e) Door Likelihood (f) Window Likelihood

Fig. 2: Left: Original floorplan and occupancy likelihood field. Right: semantic floorplan and label likelihood fields.

that leverages semantic and, optionally, range information. We will show that we can use our novel SeDAR sensing and localisation framework to outperform traditional RMCL.

### A. Semantic Floorplans

RMCL requires a floorplan and/or previously created range-scan map that is accurate in scale and globally consistent. Use of human-readable floorplans makes a system much more broadly applicable than relying on prior exploration and mapping. However, differences between the floorplan and the robot's observations (*e.g.* inaccuracies, scale variation and furniture) can reduce the reliability.

To overcome this, we augment the localisation with semantic labels extracted from an existing floorplan. In our experiments we limit these labels to walls, doors and windows (see figure 2), which are easy to automatically extract from a floorplan, and are also salient for human localisation.

In order to make a labelled floorplan readable by the robot, it must first be converted into an occupancy grid. An occupancy grid is a 2D representation of the world, in which each cell in the grid has an occupancy probability, determined by it's normalized greyscale value.

If $\mathbb{M}$ is a set of 2D positions, the map can then be defined as $\mathbb{V} = \left\{ v_{\mathbf{m}}; \mathbf{m} \in \mathbb{M} \subset \mathbb{Z}^{+2} \right\}$. Then, assuming $\mathcal{L} = \{a, d, w\}$ is the set of possible cell labels (wall, door, window), each cell is defined as $v_{\mathbf{m}} = \left\langle v^o_{\mathbf{m}}, v^w_{\mathbf{m}}, v^d_{\mathbf{m}}, v^a_{\mathbf{m}} \right\rangle$ where $v^o_{\mathbf{m}}$ is the occupancy likelihood and $\ell \in \mathcal{L}$ denotes the label likelihood.

### B. SeDAR Sensor

Modern low-cost robotics systems turn the RGB-D image received at time $t$ into a set of range ($r^k_t$) and bearing ($\theta^k_t$) tuples. SeDAR adds a semantic label ($\ell^k_t$) to this tuple. Instead of using the whole image simultaneously (which would be intractible), tuples are arranged along horizontal scanlines ($z_t = \left\{ \left\langle \theta^k_t, r^k_t, \ell^k_t \right\rangle; k = 1..K \right\}$), where $k$ is the horizontal pixel location. In this work, the centre scanline is assumed to be parallel with the ground plane and is therefore used to collapse the 3D information of the RGB-D image into the 2D floorplan.

While range and bearing values can be extracted using simple geometry, their corresponding labels must be estimated using a state-of-the-art semantic segmentation algorithm. Any semantic segmentation approach can be used, however, Deep Learning based approaches currently dominate the benchmarks [7] in this field. Therefore, a CNN-based encoder-decoder network [13] is used. This is trained on the SUN3D [27] dataset, and can reliably detect doors, walls, floors, ceilings, furniture and windows. This state-of-the-art semantic segmentation runs in real-time, which allows images to be parsed into a SeDAR-scan with negligible latency. The label $\ell^k_t$ is then simply the label at pixel $k$ along the horizontal scanline.

It is important to note that we extract the labels from the RGB image only. This is by design, as it allows the use of cameras that cannot sense depth. In the following sections we will use this novel sensing modality in a novel MCL formulation with and without the range-based measurements.

### C. Motion Model

MCL motion models are normally represented by the distribution $\Pr\left(s^{i\prime}_t \middle| u_t, s^i_{t-1}\right)$, where the previous set of particles $s^i_{t-1}$ is propagated using the odometry measurements $u_t$ into the current set of particles $s^i_t$. However, it is well understood in the literature that the actual distribution being approximated is $\Pr\left(s^{i\prime}_t \middle| u_t, s^i_{t-1}, \mathbb{V}\right)$. This encodes the idea that certain motions are more or less likely depending on the map (*e.g.* through walls). Under the assumption that the motion of the robot is small, it can be shown that

$$\Pr\left(s^{i\prime}_t \middle| u_t, s^i_{t-1}, \mathbb{V}\right) = \kappa \Pr\left(s^{i\prime}_t \middle| u_t, s^i_{t-1}\right) \Pr\left(s^i_{t-1} \middle| \mathbb{V}\right) \qquad (1)$$

(see *e.g.* [24]) where $\kappa$ is a normalising factor and $\mathbb{V}$ is the set containing every cell in the map. This allows the two likelihoods to be treated independently. The motion $\Pr\left(s^{i\prime}_t \middle| u_t, s^i_{t-1}\right)$ is defined as in RCML [24]. The prior is the occupancy likelihood of the cell that contains $s^i_t$, that is $\Pr\left(s^i_{t-1} \middle| \mathbb{V}\right) = 1 - \Pr\left(v^o_{s_{t-1}}\right)$

However, this prior estimation approach becomes problematic when using human-made floorplans, as these typically have image artefacts introduced during the scanning process. Therefore, most approaches threshold the occupancy

$$\Pr\left(v^o_{s_{t-1}}\right) = \begin{cases} 1 & \text{if } v^o_{s_{t-1}} \geq \tau_o \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

where $\tau_o$ is a user defined threshold. This exacerbates problems with floorplan accuracy and occlusions. For instance, most humans would not even notice if a door is a few centimetres away from where it should be. However, this presents real problems when particles propagate though doors, as many valid particles will be discarded upon contact with the expected edge of the door frame. Instead, we propose to augment this with a *ghost factor* ($\epsilon_{\text{G}}$) that allows particles more leeway in these scenarios. Therefore the proposed prior is

$$\Pr\left(s^i_{t-1} \middle| \mathbb{V}\right) = \left(1 - \Pr\left(v^o_{s_{t-1}}\right)\right) e^{-\epsilon_{\text{G}}\,\delta_a} \qquad (3)$$

where $\delta_a$ is the distance to the nearest door. While other labels such as windows can be used, in the case of a ground-based robot doors are sufficient. The distance, $\delta_a$, can be efficiently estimated using a lookup table as defined in section III-D.

More importantly, $\epsilon_{\text{G}}$ is a user defined factor that determines how harshly this penalty is applied. Setting $\epsilon_{\text{G}} = 0$ allows particles to navigate through walls with no penalty, while very high values approximate equation 2. We will explore the effects of $\epsilon_{\text{G}}$ in section IV-D. This motion model is more probabilistically accurate than the occupancy model used in most RMCL approaches, and has the added advantage of leveraging the high-level semantic information present in the map.

### D. Sensor Model

The naïve way of incorporating semantic measurements into the sensor model would be to use the beam model. In this modality, the raycasting operation would provide not only the distance travelled by the ray, but also the label of the cell the ray hit. If the label of the cell and the observation match, the likelihood of that particle being correct is increased. However, this approach suffers from the same limitations as the traditional beam model: it has a distinct lack of smoothness. On the other hand, the likelihood field model is significantly smoother, as it provides a gradient between each of the cells. By contrast, the approach presented here uses a joint method that can use likelihood fields to incorporate semantic information in the presence of semantic labels. More importantly, it can also use raycasting within a likelihood field in order to operate without range measurements.

The likelihood field model calculates a distance map. For each cell $v_{\mathbf{m}}$, the distance to the nearest occupied cell

$$\delta_o\left(\mathbf{m}\right) = \min_{\mathbf{m}'} \left\| \mathbf{m} - \mathbf{m}' \right\|, \quad v^o_{\mathbf{m}'} > \tau_o \tag{4}$$

is calculated and stored. For clarity, we omit the parameter $\mathbf{m}$ for the remainder of the paper. When a measurement $z_t^k = \left\langle \theta_t^k, r_t^k \right\rangle$ is received, the endpoint is estimated and used as an index to the distance map. Assuming a Gaussian error distribution, the weight of each particle $s_t^{i\prime}$ can then estimated as

$$\Pr_{\text{RNG}}\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) = e^{-\delta_o^2 / 2\,\sigma_o^2} \tag{5}$$

where $\delta_o$ is the value obtained from the distance map and $\sigma_o$ is dictated by the noise characteristics of the sensor. However, this model has three main limitations. First, it makes no use of the semantic information present in the map. Second, the parameter $\sigma_o$ must be estimated by the user and assumes all measurements within a scan have the same noise parameters. Third, it is incapable of operating in the absence of range measurements.

Instead, this work uses the semantic labels present in the map to create multiple likelihood fields. For each label present in the floorplan, we can calculate a distance map that stores the shortest distance to a cell with the same label. Formally, for each map cell $v_{\mathbf{m}}$ we can estimate the distance to the nearest cell of each label as

$$\delta_\ell\left(\mathbf{m}\right) = \min_{\mathbf{m}'} \left\| \mathbf{m} - \mathbf{m}' \right\|, \quad v^\ell_{\mathbf{m}'} > \tau_o \tag{6}$$

where $\delta_\ell = \left\{\delta_a, \delta_d, \delta_w\right\}$ are distances to the nearest wall, door and window, respectively. Figure 2 shows the distance maps for each label. This approach overcomes the three limitations of the state-of-the-art, which we will now discuss.

*1) Semantic Information:* First, SeDAR uses the semantic information present in the map. When we receive an observation $z_t^k = \left\langle \theta_t^k, r_t^k, \ell_t^k \right\rangle$, we use the bearing $\theta_t^k$ and range $r_t^k$ information to estimate the endpoint of the scan. We then use the label $\ell_t^k$ to decide which semantic likelihood field to use. Using the endpoint from the previous step, the label-likelihood can be estimated similarly to equation 5,

$$\Pr_{\text{LBL}}\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) = e^{-\delta_\ell^2 / 2\,\sigma_\ell^2} \tag{7}$$

where $\delta_\ell$ is the distance to the nearest cell of the relevant label and $\sigma_\ell$ is the standard deviation (which we will define using the label prior). The probability of an observation given the map and pose can then be estimated as

$$\Pr\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) = \epsilon_o \Pr_{\text{RNG}}\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) + \epsilon_\ell \Pr_{\text{LBL}}\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) \tag{8}$$

where $\epsilon_o$ and $\epsilon_\ell$ are user defined weights. When $\epsilon_\ell = 0$ the likelihood is the same as standard RMCL. On the other hand, when $\epsilon_o = 0$ the approach is using only the semantic information present in the floorplan. These weights are properly explored and defined in section IV-C. Unlike range scanners, $\sigma_\ell$ cannot be related to the physical properties of the sensor. Instead, this standard deviation is estimated directly from the prior of each label on the map. Defining $\sigma_\ell$ this way has the benefit of not requiring tuning. However, there is a much more important effect that must be discussed.

*2) Semantically Adaptive Standard Deviation:* When a human reads a floorplan, unique landmarks are the most discriminative features: it is easier to localise on a floorplan from the configuration of doors and windows than it is from the configuration of walls. This translates into the a simple insight: *lower priors are more discriminative*. Therefore, $\sigma_\ell$ is tied to the prior of each label not only because it is one less parameter to tune, but because it implicitly makes observing rare landmarks more beneficial than common landmarks.

Relating $\sigma_\ell$ to the label prior $\Pr\left(\ell\right)$ controls how smoothly the distribution decays w.r.t. distance from the cell. The smaller $\Pr\left(\ell\right)$ is, the smoother the decay. In essence, the localisation algorithm should be more *lenient* on sparser labels.

### E. Range-less Semantic Scan-Matching

The final, and most important, strength of this approach is the ability to perform all of the previously described methodology in the complete absence of range measurements. So far, we have formalised this approach on the assumption that we received either $\left\langle \theta_t^k, r_t^k \right\rangle$ tuples (existing approaches) or $\left\langle \theta_t^k, r_t^k, \ell_t^k \right\rangle$ tuples (SeDAR-based approach). However, this approach is capable of operating directly on $\left\langle \theta_t^k, \ell_t^k \right\rangle$ tuples. In other words, depth measurments are *explicitly* not added to this approach.

Incorporating range-less measurements is simple. The beam and likelihood field models are combined in a novel approach that avoids the degeneracies that would happen in traditional RMCL approaches. In the standard approach, the raycasting operation terminates when an occupied cell is reached and the likelihood is estimated as

$$\Pr\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) = e^{-\left(r_t^k - r_t^{k*}\right)^2 / 2\,\sigma_o^2} \tag{9}$$

where $r_t^k$ is the range obtained from the sensor and $r_t^{k*}$ is the distance travelled by the ray. Unfortunately, in the absence of a range-based measurement $r_t^k$ this is impossible. Using the standard distance map is also impossible, since we can not estimate the endpoint of the ray. Using raycasting in the distance map fails similarly. The raycasting terminates on an occupied cell, implying $\delta_o = 0$ for every ray cast.

On the other hand, the semantic likelihood fields can still be used as $\delta_\ell$ will still have a meaningful and discriminative value. We call this operation semantic raycasting. For every $z_t^k = \left\langle \theta_t^k, \ell_t^k \right\rangle$, the raycasting is performed. However, instead of comparing $r_t^k$ and $r_t^{k*}$ or using $\delta_o$, the label $\ell_t^k$ determines which likelihood field to use. The cost is then

$$\Pr\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) = \Pr_{\text{LBL}}\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right) \tag{10}$$

where $\Pr_{\text{LBL}}\left(z_t^k \middle| s_t^{i\prime}, \mathbb{V}\right)$ is defined in equation 7. This method is essentially a combination of the beam-model and the likelihood field model. In the absence of range-measurements to estimate an endpoint from, this hybrid approach uses semantic raycasting to find the nearest occupied cell. The distances are then used to provide smoothness to equation 10, which implies that the observation likelihood is directly proportional to the angular distribution of labels. The net effect is that this approach is invariant to scale changes, as long as the aspect ratio of the map is respected.

To summarise, this section presented several important concepts. We introduced the idea of a semantic floorplan that contains information that is salient to humans. We also introduced a new sensing modality, SeDAR, that adds semantic labels to the traditional LiDAR information. We then incorporated these two ideas into a novel MCL-based approach. This approach is capable of using the semantic information present in the map to define a novel motion model. It is also capable of using the labels from a CNN-based segmentation to localise within the map. Our approach can do all of the above both in the presence, and absence, of range measurements. In the following section, we show that our approach is capable of outperforming standard RMCL approaches when using depth, and that it provides comparable performance in its absence.

## IV. Results

This section will demonstrate that SeDAR-based MCL is capable of reliably out-performing the state-of-the-art when using range measurements. It will also show that our approach it is capable of comparable performance even in the absence of range. First, the experimental setup is described. This consists of creating a dataset of a trajectory within a floorplan, as well as establishing error metrics. Then a comparison of several approaches is performed. The comparison is done in terms of room-level and global localisation, both quantitative and qualitative. Finally, we show the effects of our parameters.

### A. Experimental Setup

In order to evaluate this approach, we require a dataset that has several important characteristics. The dataset should consist of a robot navigating within a human-readable floorplan. Human-readability is required to ensure semantic information is present.The trajectory should be captured with an RGB-D camera in order to extract all the possible tuple combinations (range, bearing and label). Finally, we expect the trajectory of the robot to happen on the same plane as the floorplan. Unfortunately, most of the MCL datasets in the literature do not contain a floorplan, opting instead for laser-scans. RGB-D SLAM datasets are more appropriate, but they either do not move on the floorplan plane or simply do not contain ground-truth trajectory estimation.

Therefore, we are forced to use our own dataset - which we will make publicly available. We use the floorplan in figure 2a because it is large enough to provide multiple trajectories with no overlap. The dataset was collected using the popular TurtleBot platform, as it has a front-facing Kinect that can be used for emulating both LiDAR and SeDAR.

Normally, the ground-truth trajectory for floorplan localisation is either manually estimated (as in [26]) or estimated using Motion Capture (MoCap) systems (as in [22]). However, both of these approaches are limited in scope. Manual ground-truth estimation is time-consuming and impractical. MoCap is expensive, difficult to calibrate, and normally cannot remain in the public areas required for floorplan localisation. In order to overcome these limitations, well established RGB-D SLAM systems are used instead. The excellent approach by Labbe *et al.* [14] provides very accurate pose estimation in complex environments. While it does not localise within a floorplan, it does provide an accurate reconstruction and trajectory for the robot, which can then be registered into the floorplan.

| **Average Trajectory Error (m)** | | | | | | |
|---|---|---|---|---|---|---|
| **Approach** | **RMSE** | **Mean** | **Median** | **Std. Dev.** | **Min** | **Max** |
| AMCL | 0.24 | 0.21 | 0.20 | 0.11 | 0.04 | 0.95 |
| **Range** (Label Only) | **0.19** | **0.16** | **0.14** | **0.10** | **0.02** | **0.55** |
| **Range** (Combined) | 0.22 | 0.19 | 0.17 | 0.11 | 0.04 | 0.62 |
| **Rays** ($\epsilon_{\text{G}} = 3.0$) | 0.40 | 0.34 | 0.27 | 0.22 | 0.07 | 1.51 |
| **Rays** ($\epsilon_{\text{G}} = 7.0$) | 0.58 | 0.45 | 0.38 | 0.37 | 0.02 | 2.23 |

TABLE I: Room-Level Initialisation

To quantitatively evaluate the presented approach against ground truth, the Absolute Trajectory Error (A) metric presented by Sturm *et al.* [22] is used. A is estimated by first registering the two trajectories using the closed form solution of Horn [12], who finds a rigid transformation ${}^{\mathbb{G}}\mathbf{T}_{\mathbb{X}}$ that registers the trajectory $\mathbb{X}_t$ to the ground truth $\mathbb{G}_t$. At every time step $t$, the A can then be estimated as

$$e_{\mathbf{g}} = \mathbf{g}_t^{-1}\,{}^{\mathbb{G}}\mathbf{T}_{\mathbb{X}}\,\mathbf{x}_t \tag{11}$$

where $\mathbf{g}_t \in \mathbb{G}_t$ and $\mathbf{x}_t \in \mathbb{X}_t$ are the current time-aligned poses of the ground truth and estimated trajectory, respectively. The Root Mean Square Error (RMSE), mean and median values of this error metric are reported, as these are indicative of performance over room-level initialisation. In order to visualise the global localisation process, the error of each successive pose is shown (error as it varies with time). These metrics are sufficient to objectively demonstrate the systems ability to globally localise in a floorplan, while also being able to measure room-level initialisation performance.

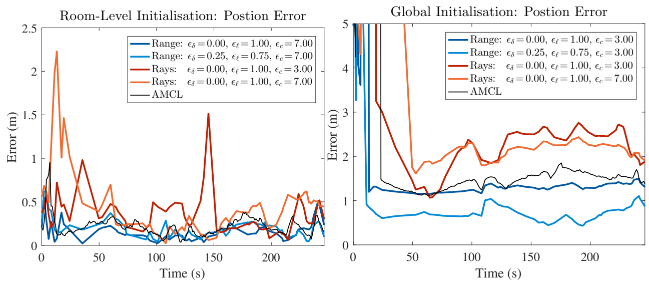
We compare the work presented here against the extremely popular MCL approach present in Robot Operating System (ROS), called Adaptive Monte Carlo Localisation (AMCL) [9]. While more modern approaches [2] exist, they are based on the same principles as AMCL and simply change the particle sampling strategy. More importantly, AMCL is the standard MCL approach in the robotics community. Any improvements over this approach are therefore extremely valuable. In all experiments, any overlapping parameters (such as $\sigma_o$) are kept the same. The only parameters varied are $\epsilon_\ell$, $\epsilon_o$ and $\epsilon_{\text{G}}$.

### B. Room-Level Initialisation

For this evaluation, a room-level initialisation with standard deviations of $2.0m$ in $(x, y)$ and $2.0rad$ in $\theta$ is given to both AMCL and the proposed approach. The systems then ran with a maximum of 1000 particles (minimum 250) placed around the covariance ellipse. We record the error as each new image in the dataset is added.

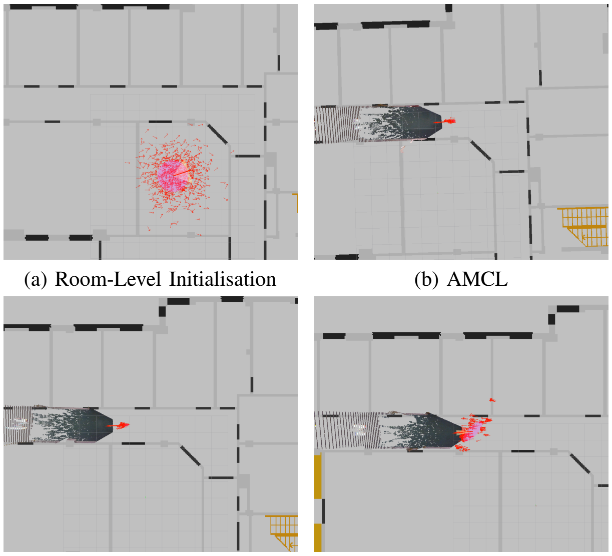
*1) Quantitative Results:* Figure 3a compares four distinct scenarios against AMCL. Of these four scenarios, two use the range measurements from the Microsoft Kinect (blue lines) and two only use the RGB image (red lines).

The first range-enabled scenario uses the range measurements to estimate the endpoint of the measurement (and therefore the index in the distance map) and then sets ($\epsilon_o = 0.0, \epsilon_\ell = 1.0$). This means that while the range information is used to inform the lookup in the distance map, the costs are always directly related to the labels. The second range-enabled



(a) Room-level (b) Global

Fig. 3: Semantic localisation with different initialisations.



(a) Room-Level Initialisation (b) AMCL

(c) SeDAR (Range-Based) (d) SeDAR (Ray-Based)

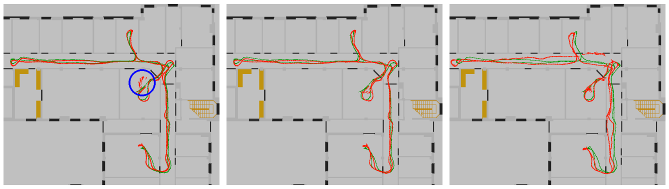
Fig. 4: Qualitative view of Localisation in different modalities.

scenario performs a weighted combination ($\epsilon_o = 0.25, \epsilon_\ell = 0.75$) of both the semantic and traditional approaches.

In terms of the ray-based version of our approach, we use equation 10. This means there are no parameters to set. Instead, a mild ghost factor ($\epsilon_{\mathrm{G}} = 3.0$) and a harsh one ($\epsilon_{\mathrm{G}} = 7.0$) are shown.

Since room-level initialisation is an easier problem than global initialisation, the advantages of the range-enabled version of our approach are harder to see compared to state-of-the-art. However, it is important to notice how closely the ray-based version of the approach performs to the rest of the scenarios, despite using no depth data. Apart from a couple of peaks, we essentially perform at the same level as AMCL. This becomes even more noticeable in table I, where it is clear that range-based semantic MCL (using only the labels) outperforms state of the art, while the ray-based $\epsilon_{\mathrm{G}} = 3.0$ version lags closely behind. The reason $\epsilon_{\mathrm{G}} = 3.0$ performs better than $\epsilon_{\mathrm{G}} = 7.0$ is because small errors in the pose can cause the robot to "clip" a wall as it goes through the door. Since $\epsilon_{\mathrm{G}} = 3.0$ is more lenient on these scenarios, it is able to outperform the harsher ghost factors. We will explore this relationship further in section IV-D.

*2) Qualitative Results:* In terms of qualitative evaluation, we show the convergence behaviour and the estimated path. The convergence behaviour can be seen in figure 4. Here, figure 4a shows how the filter is initialised to roughly correspond to the room the robot is in. As the robot starts moving, we can see how AMCL (4b), the range-based version of SeDAR (4c) and the ray-based version (4d) converge. Notice that while the ray-based approach has a predictably larger variance on the particles, the filter has successfully localised. This can be seen from the fact that the reconstructed Kinect pointcloud is properly aligned with the floorplan. It is important to note that although the Kinect pointcloud is present for visualisation in the ray-based method, it is *not* used.



(a) AMCL (b) SeDAR (Range) (c) SeDAR (Ray)

Fig. 5: Estimated path from room-level initialisations.

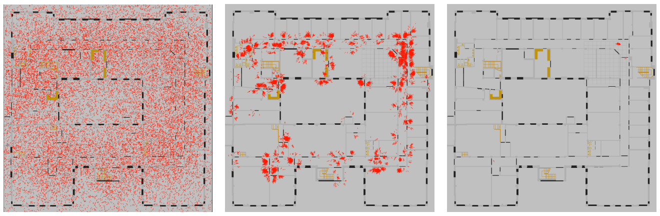
The estimated paths can be seen in figure 5, where the red path is the estimated path and green is the ground truth. Figure 5a shows the state-of-the-art, which struggles to converge at the beginning of the sequence (marked by a blue circle). It can be seen that the range-based approach in figure 5b (combined label and range), converges more quickly and maintains a similar performance to AMCL. It only slightly deviates from the path at the end of the ambiguous corridor on the left, which also happens to AMCL. It can also be seen that the ray-based approach performs very well. While it takes longer to converge, as can be seen by the estimated trajectory in figure 5c, it corrects itself and only deviates from the path in areas of large uncertainty (like long corridors).

These experiments show that SeDAR-based MCL is capable of operating in a room-level initialised scenario. It is now important to discuss how discriminative SeDAR is when there is no initial pose estimate provided to the system.

### C. Global Initialisation

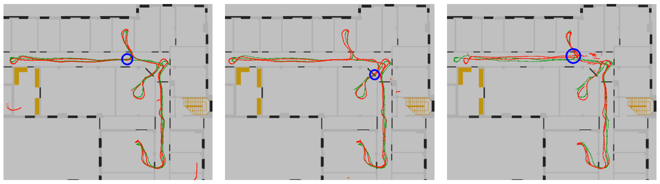
We now focus on SeDAR-based MCL's ability to perform global localisation. In these experiments, the system is given no indication of where in the map the robot is. Instead, a maximum $50k$ particles (min. $15k$) is placed over the floorplan.

*1) Quantitative Results:* Figure 3b shows the same four scenarios as in the previous section. For the range-based scenarios (blue lines) it can be seen that using only the label information ($\epsilon_o = 0.0, \epsilon_\ell = 1.00$) consistently outperforms the state of the art, both in terms of how quickly the values converge to a final result and the actual error on convergence. This shows that SeDAR used in an MCL context is more discriminative than standard occupancy maps in RMCL. The second range-based measurement ($\epsilon_o = 0.25, \epsilon_\ell = 0.75$) significantly outperforms all other approaches. This is probably because, in principle, the occupancy maps can be considered another "label" in the semantic floorplan. This makes sense because setting $\epsilon_o = 0.25$ is equivalent to weighting all labels equally, as it is a third of $\epsilon_\ell = 0.75$ which is the weight of 3 labels. In terms of the ray-based version of our approach (red lines), we compare two scenarios. A mild ghost factor ($\epsilon_{\mathrm{G}} = 3.0$) and a harsh one ($\epsilon_{\mathrm{G}} = 7.0$). These versions of the approach both provide comparable performance to the state-of-the-art. It is important to emphasise that this approach uses absolutely no range and/or depth measurements. As such, comparing against depth-based systems is inherently unfair. Still, SeDAR ray-based



(a) Global Initialisation (b) Looking at Doors (c) Converged

Fig. 6: Qualitative view of Localisation in different modalities.



(a) AMCL Path (b) SeDAR (Range) (c) SeDAR (Ray)

Fig. 7: Estimated path from global initialisations.

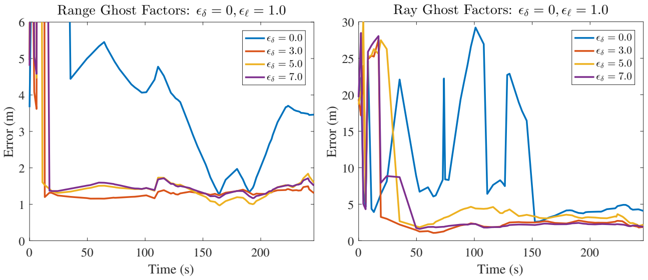
approaches compare favourably to AMCL. In terms of convergence, the mild ghost factor $\epsilon_{\text{G}} = 3.0$ gets to within several meters even quicker than AMCL, at which point the convergence rate slows down and is overtaken by AMCL. The steady state performance is also comparable. While the performance temporarily degrades, it manages to recover and keep a steady error rate throughout the whole run. On the other hand, the harsher ghost factor $\epsilon_{\text{G}} = 7.0$ takes longer to converge, but remains steady and eventually outperforms the milder ghost factor. Table II shows the RMSE, error along with other statistics.

*2) Qualitative Results:* Similar to the previous section, we can provide qualitative analysis by looking at the convergence behaviour and the estimated paths.

In order to visualise the convergence behaviour, figure 6a shows a series of time steps during the filters' initialisation. On the first image, the particles have been spread over the ground floor of a $(49 \text{ x } 49)$m office area. In this dataset, the robot is looking directly at a door during the beginning of the sequence. Therefore, in figure 6b the filter converges with particles looking at doors that are a similar distance away. The robot then proceeds to move through the doors. Going through the door makes the filter converge significantly faster as it implicitly uses the ghost factor in the motion model. It also gives the robot a more unique distribution of doors (on a corner), which makes the filter converge quickly. This is shown in figure 6c, where the filter converges.

The estimated paths can be seen in figure 7, where the blue circle denotes the point of convergence. It can be seen that AMCL takes longer to converge (further away from the corner room) than the range-based approach. More importantly, it can be seen that the range-based approach suffers no noticeable degradation in the estimated trajectory over the room-level initialisation. On the other hand, the ray-based method's performance degrades more noticeably. This is because the filter converges in a long corridor with ambiguous label distributions (doors left and right are similarly spaced). However, once the robot turns around the system recovers and performs comparably to the range-based approach.

As mentioned previously, entering or exiting rooms helps the filter converge because it can use the ghost factor in the motion model. The following experiments, evaluate how the ghost factor affects the performance of the approach.



(a) Range-Based (b) Ray-Based

Fig. 8: Different ghost factors ($\epsilon_{\text{G}}$), global initialisation.

| Average Trajectory Error (m) | | | | | | |
|---|---|---|---|---|---|---|
| **Approach** | **RMSE** | **Mean** | **Median** | **Std. Dev.** | **Min** | **Max** |
| AMCL | 7.31 | 2.26 | **0.20** | 6.95 | **0.028** | 35.45 |
| **Range** (Label Only) | 6.71 | 2.59 | 1.31 | 6.20 | 1.15 | 38.60 |
| **Range** (Combined) | **4.78** | **1.69** | 0.69 | **4.47** | 0.43 | **31.19** |
| **Rays** ($\epsilon_{\text{G}} = 3.0$) | 7.74 | 4.36 | 2.46 | 6.40 | 1.07 | 27.55 |
| **Rays** ($\epsilon_{\text{G}} = 7.0$) | 8.09 | 4.49 | 2.22 | 6.73 | 1.61 | 28.47 |

TABLE II: Global Initialisation

### *D. Ghost Factor*

The effect of the ghost factor can be measured in a similar way to the overall filter performance. We show that the ghost factor provides more discriminative information when it is *not* defined in a binary fashion. This is shown in the label-only scenario for both the range-based and ray-based approaches, in both the global and room-level initialisation.

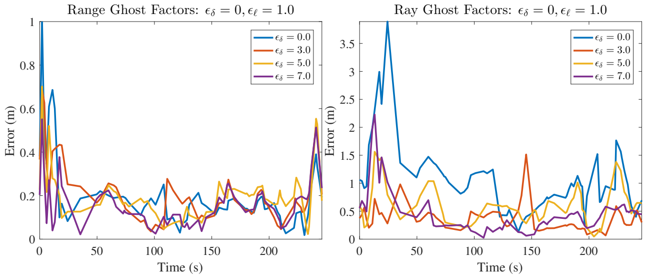
*1) Global Initialisation:* Figure 8 shows the effect of varying the ghost factor during global initialisation. It can be seen that not penalising particles going through walls, ($\epsilon_{\text{G}} = 0$), is not a good choice. This makes sense, as there is very little to be gained from allowing particles to traverse occupied cells without any consequence. It follows that we should set the ghost factor as high as possible. However, setting the ghost factor to a large value ($\epsilon_{\text{G}} = 7.0$), which corresponds to reducing the probability by $95\%$ at $0.43$m, does not provide the best results.

While it might seem intuitive to assume that a higher ($\epsilon_{\text{G}}$) will always be better, this is not the case. High values of the ghost factor correspond to a binary interpretation of occupancy which makes MCL systems unstable in the presence of discrepancies between the map and the environment. This happens because otherwise correct particles can clip door edges and be completely eliminated from the system. A harsh ghost factor also exacerbates problems with limited number of particles. In fact, $\epsilon_{\text{G}} = 3.0$, corresponding to a $95\%$ reduction at $1.0$m, consistently showed the best results in all of the global initialisation experiments, as can be seen in table III.

*2) Room-Level Initialisation:* In terms of room-level initialisation, having an aggressive ghost factor is more in line with our initial intuition. Table IV shows that for both of the range-based scenarios, $\epsilon_{\text{G}} = 7.0$ provides the best results. This is because room-level initialisation in the presence of range-based measurements is a much easier problem to solve. On the other hand, the ray-based scenario still prefers a milder

| Average Trajectory Error (RMSE) | | | |
|---|---|---|---|
| **Ghost Factor** ($\epsilon_{\text{G}}$) | **Range (Labels)** | **Range (Weighted)** | **Rays** |
| 0.0 | 10.88 | 10.13 | 11.71 |
| 3.0 | **6.71** | **4.78** | **7.74** |
| 5.0 | 6.97 | 6.30 | 9.54 |
| 7.0 | 7.19 | 6.10 | 8.09 |

TABLE III: Global A for Different Ghost Factors



(a) Range-Based

(b) Ray-Based

Fig. 9: Different ghost factors ($\epsilon_{\text{G}}$), room-level initialisation.

ghost factor of $\epsilon_{\text{G}} = 3.0$. In this scenario, inaccuracies in both the map and the sensing modalities allow for otherwise correct particles to be heavily penalised by an aggressive ghost factor. Both of these results are reflected in figures 9a and 9b.

These results allow us to come to a single conclusion. The ghost factor must be tuned to the expected amount of noise in the map and sensing modality. Aggressive ghost factors can be used in cases where the pre-existing map is accurate and densely sampled, such as the case where the map was collected by the same sensor being used to localise (*i.e.* SLAM). On the other hand, where there are expected differences between the map and what the robot observes (*e.g.* furniture, scale errors, etc.), it is beneficial to provide a milder ghost factor and to be more lenient to small pose errors.

### E. Timing

The speed of our approach was evaluated on a machine equipped with an Intel Xeon X5550 (2.67GHz) and an NVidia Titan X (Maxwell). During room-level initialisation, or once the system has converged, our approach can run with 250 particles in 10ms, leaving us more than enough time to process the images from the Kinect into a SeDAR scan. Transforming the RGB images into semantic labels is the most expensive operation, taking on average 120ms. This means that a converged filter can run at $8-10$ fps. When performing global localisation, we can integrate a new sensor update, using $50,000$ particles, in 2.25 seconds. As MCL-based approaches require motion between each sensor integration, this is still effectively near real-time, and orders of magnitude faster than competing vision approaches.

## V. Conclusion

In conclusion, this work has demonstrated that human-inspired localisation based on distinctive landmarks, is an effective alternative to traditional scan-matching. We demonstrated how the semantic information provided by SeDAR could be utilised in both the motion model and the sensor model (with and without range data). Our experiments show that this new information is highly complementary to state-of-the-art techniques, providing a 35% reduction in errors over either technique alone. Based on this compelling evidence, we can conclude the application of SeDAR (and semantic information in general) should be explored further within the wider field of robotics.

More generally, this work reinforces the conclusions of other recent research: machine learning has now reached the point where the subjective aspects of biological perception (such as semantic scene understanding) can be reliably emulated. As such, the biologically-inspired paradigm which has long been a staple of robot hardware design, is now also feasible (and essential) for robot software design.

| **Average Trajectory Error (RMSE)** | | | |
|---|---|---|---|
| **Ghost Factor ($\epsilon_{\text{G}}$)** | **Range (Labels)** | **Range (Weighted)** | **Rays** |
| 0.0 | 0.25 | 0.27 | 1.20 |
| 3.0 | 0.24 | 0.25 | **0.40** |
| 5.0 | 0.22 | 0.24 | 0.70 |
| 7.0 | **0.19** | **0.22** | 0.58 |

TABLE IV: Room-Level A for Different Ghost Factors

To this end an interesting avenue for future work would be to follow recent research in visual odometry, and utilise single-image depth and/or surface normal estimation techniques for localisation. This could implicitly detect scene elements of known sizes, which is another vital component of biological perception.

## References

[1] V Badrinarayanan, A Kendall, and R Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *arXiv*, 2015.

[2] JL Blanco, J Gonzalez, and JA Fernandez-Madrigal. Optimal Filtering for Non-parametric Observation Models: Applications to Localization and SLAM. *IJRR*, 2008.

[3] K Briechle and UD Hanebeck. Localization of a mobile robot using relative bearing measurements. *T-RO*, 20(1):36–44, 2004.

[4] MA Brubaker, A Geiger, and R Urtasun. Lost! leveraging the crowd for probabilistic visual self-localization. In *CVPR*, 2013.

[5] H Chu, DK Kim, and T Chen. You are here: Mimicking the Human Thinking Process in Reading Floor-Plans. In *ICCV*, 2015.

[6] H Chu, S Wang, R Urtasun, and S Fidler. Housecraft: Building houses from rental Ads and street views. In *ECCV*, 2016.

[7] M Cordts, M Omran, S Ramos, T Rehfeld, M Enzweiler, R Benenson, U Franke, S Roth, and B Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *CVPR*, 2016.

[8] F Dellaert, W Burgard, D Fox, and S Thrun. Using the Condensation algorithm for robust, vision-based mobile robot localization. *CVPR*, 1999.

[9] F Dellaert, D Fox, W Burgard, and S Thrun. Monte Carlo localization for mobile robots. In *ICRA*, 1999.

[10] MF Fallon, H Johannsson, and JJ Leonard. Efficient scene simulation for robust monte carlo localization using an RGB-D camera. In *ICRA*, 2012.

[11] D Fox, W Burgard, F Dellaert, and S Thrun. Monte Carlo Localization: Efficient Position Estimation for Mobile Robots. In *AAAI*, 1999.

[12] BKP Horn. Closed-form solution of absolute orientation using unit quaternions. *JOSA A*, 1987.

[13] A Kendall, V Badrinarayanan, and R Cipolla. Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding. *arXiv*, 2015.

[14] M Labbe and F Michaud. Online Global Loop Closure Detection for Large-Scale Multi-Session Graph-Based SLAM. In *IROS*, 2014.

[15] I Laina, C Rupprecht, V Belagiannis, F Tombari, and N Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*, 2016.

[16] C Lee, V Badrinarayanan, T Malisiewicz, and A Rabinovich. RoomNet: End-to-End Room Layout Estimation. *arXiv*, 2017.

[17] C Liu, AG Schwing, K Kundu, R Urtasun, and S Fidler. Rent3D: Floor-plan priors for monocular layout estimation. In *CVPR*, 2015.

[18] K Melbouci, S Naudet Collette, V Gay-Bellile, O Ait-Aider, and M Dhome. Model based RGBD SLAM. *ICIP*, 2016.

[19] M Paton and J Kosecka. Adaptive RGB-D localization. *CRV*, 2012.

[20] E Shelhamer, J Long, and T Darrell. Fully Convolutional Networks for Semantic Segmentation. *PAMI*, 2017.

[21] J Shotton, B Glocker, C Zach, S Izadi, A Criminisi, and A Fitzgibbon. Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images. In *CVPR*, 2013.

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

[23] K Tateno, F Tombari, I Laina, and N Navab. CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction. *arXiv*, 2017.

[24] S Thrun. Probabilistic Robotics. *Comms. of the ACM*, 2002.

[25] S Wang, S Fidler, and R Urtasun. Lost Shopping! Monocular Localization in Large Indoor Spaces. *ICCV*, 2015.

[26] W Winterhalter, F Fleckenstein, B Steder, L Spinello, and W Burgard. Accurate indoor localization for RGB-D smartphones and tablets given 2D floor plans. In *IROS*, 2015.

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