IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2023, London, UK. Cite as RA-L paper.

Sigma-FP: Robot Mapping of 3D Floor Plans with an RGB-D Camera under Uncertainty

Jose-Luis Matez-Bandera[1], Javier Monroy[1] and Javier Gonzalez-Jimenez[1]

***Abstract*—This work presents Sigma-FP, a novel 3D reconstruction method to obtain the floor plan of a multi-room environment from a sequence of RGB-D images captured by a wheeled mobile robot. For each input image, the planar patches of visible walls are extracted and subsequently characterized by a multivariate Gaussian distribution in the convenient Plane Parameter Space. Then, accounting for the probabilistic nature of the robot localization, we transform and combine the planar patches from the camera frame into a 3D global model, where the planar patches include both the plane estimation uncertainty and the propagation of the robot pose uncertainty. Additionally, processing depth data, we detect openings (doors and windows) in the wall, which are also incorporated in the 3D global model to provide a more realistic representation. Experimental results, in both real-world and synthetic environments, demonstrate that our method outperforms state-of-the-art methods, both in time and accuracy, while just relying on Atlanta world assumption.**

***Index Terms*—Mapping, RGB-D Perception, 3D Floor Plan Reconstruction, Probability and Statistical Methods**

I. Introduction

HIGH-LEVEL scene understanding is essential for the operation of mobile robots in human-centered environments. In this context, a complete world representation involves not only capturing the geometry and semantics of objects [1], but also identifying the structural elements of the scene (*i.e.* walls, floor, ceiling and even doors and windows) [2]. Building a model of such structural elements, commonly referred in the literature as 3D floor plan, is of great value for the robot navigation and exploration [3], [4] as well as for enhancing object positioning in semantic mapping [1], [5], among other robotic tasks.

The generation of 3D floor plans is usually performed by extracting primitive shapes such as cuboids or planes from data acquired with on board cameras [6], [7] and/or range sensors [8], [9]. The limitations of current techniques involve (i) the requirement of a large amount of input data (*e.g.* a dense point cloud of the entire environment) which hinders the online building of the floor plan [9], [10], (ii) the lack of detail in the reconstruction, disregarding wall thickness or the presence of doors and windows [11], [12], or (iii) the assumption of orthogonal planes [13], [14]. Moreover, a major challenge when transferring these techniques into the real-world is to account for the unavoidable uncertainty in the robot localization during the data acquisition process. This represents one of main source of errors and failures in current 3D floor plan reconstruction methods [6].

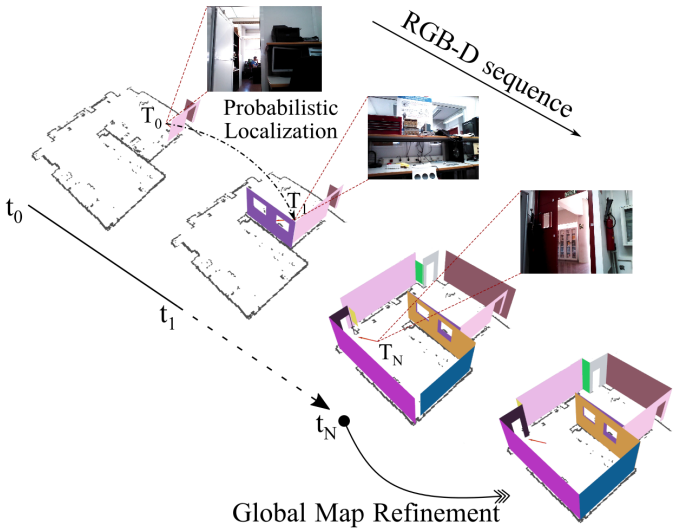


Fig. 1: Incremental reconstruction of a 3D floor plan from a sequence of RGB-D images using Sigma-FP. The extracted planes and their openings are integrated image-by-image in a global model by considering both the uncertainty in the robot localization and the uncertainty in the plane extraction. Finally, the optional step *Global Map Refinement* is carried out to enhance the floor plan.

In this work, we propose Sigma-FP, an incremental plane-based method for the 3D reconstruction of multi-room floor plans that delimits openings (*i.e.* doors and windows) as shown in Figure 1. Our proposal takes a sequence of RGB-D images captured by a wheeled mobile robot, whose localization is given with some Gaussian uncertainty. Following an image-by-image basis, and exploiting the convenient Plane Parameter Space (PPS) [15], we extract a set of planar patches from the visible walls, and their respective openings. Planar patches are characterized by a multivariate Gaussian distribution in the PPS, which are then conveniently transformed from the camera frame into the world frame –where they are fused into a 3D global model–, propagating the uncertainty in both the plane extraction and the robot localization. This enables a sound integration between the robot localization and the floor plan reconstruction, also providing a coherent framework to account for the error and drift in the localization. Moreover, our approach is able to work under relaxed constraints in the

Manuscript received: July, 14, 2022; Revised September, 29, 2022; Accepted November, 2, 2022.

This paper was recommended for publication by Editor Javier Civera upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the research projects HOUNDBOT (P20-01302) and ARPEGGIO (PID2020-117057), and the Spanish grant program FPU19/00704.

[1]Jose-Luis Matez-Bandera, Javier Monroy and Javier Gonzalez-Jimenez are with the Machine Perception and Intelligent Robotics (MAPIR) Group, Malaga Institute for Mechatronics Engineering and Cyber-Physical Systems (IMECH.UMA). University of Malaga. Spain. josematez@uma.es, jgmonroy@uma.es, javiergonzalez@uma.es

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

room geometry, just considering the Atlanta world assumption (*i.e.* walls are orthogonal to the vertical axis) [16].

For evaluation, we carry out a set of experiments in both real-world and synthetic environments with different robots, comparing our proposal with state-of-the-art methods. The results demonstrate that our method generalizes properly in the reconstruction of 3D floor plans while reducing the error and enabling an incremental online reconstruction. In summary, our work provides the following contributions:

1) The inclusion of the probabilistic nature of the robot localization in the generation of 3D floor plans, propagating the robot uncertainty to the extracted planes.
2) A functional method that increasingly reconstructs the 3D floor plan of a multi-room scenario from a sequence of RGB-D images, which features:
   - A significant level of details in the reconstruction, including windows and doors in the scene.
   - A relaxation of the room geometry constraints, considering only the Atlanta world assumption.
3) The code of Sigma-FP, which is available as a ROS package, as well as a demonstration video, can be found at https://MAPIRlab.github.io/Sigma-FP.

II. Related Work

In the context of mobile robotics, contributions to the reconstruction of the structural elements of an indoor environment can be divided into two main groups according to their scope: layout estimation, focusing on single-room contexts, and floor plan reconstruction, covering multi-room environments. Next, we review the most important works of each group, while for an in-depth overview of the state-of-the-art, the reader is referred to [17], [18].

A. Layout Estimation

Layout estimation refers to the problem of extracting the enclosing structure of a single room, usually from a single RGB/RGB-D image. For example, Lee *et al.* [14] and Yan *et al.* [13] presented deep learning networks to estimate the room layout from a single monocular RGB image under the Manhattan world (MW) assumption [19]. Zhang *et al.* [20] presented a similar approach but considering also depth information from an RGB-D camera, reducing considerably the error in estimated layout, while Howard-Jenkins *et al.* [21] focused on relaxing the constraints imposed on the room shape by reformulating the problem as an instance detection task. The latter consists in extracting 3D planes using a Region-based Convolutional Neural Network frame-by-frame from a sequence of posed RGB-D images and later combining the planes in a single 3D model.

B. Floor Plan Reconstruction

The reconstruction of floor plans aims to generate a 2D/3D global model of a multi-room environment based on the extraction of primitives from a sequence of RGB-D images or even, a curated point cloud of the whole environment. Works under this category include these of Chen *et al.* [11], based on the initial generation a complete point cloud of the environment from a sequence of RGB-D scans, which is latter processed by a Deep Neural Network to obtain a 2D floor plan only of the walls, or the work from Liu *et al.* [10] presenting a similar approach which also includes 2D openings (doors and windows). Also noticeable are the contributions that require an uncluttered point cloud as input, from which a 2D vector-graphics floor plan [9], [12], a 3D floor plan [22], [9], or even BIM models [8], [12] are generated. However, these works share a common drawback in terms of usability due to the consideration of strong assumptions such as boxy world (*i.e.* each room is composed by just four orthogonal walls) [23] or Manhattan world (walls lie only along the two perpendicular directions) [12]. Furthermore, the generation of the global model is usually performed offline over the complete point cloud and not frame-by-frame, which precludes its use in applications such as semantic mapping. To the best of the authors' knowledge, the recent work from Solarte *et al.* [6] is the first one addressing the sequential reconstruction of multi-room environments. Yet, as opposed to our proposal, they rely on 360-images, lack the detection of openings, and do not handle uncertainty, which limits its applicability in real-world.

Our approach is placed in a middle point between both scopes, *i.e.* from layout estimation works, we seize the concept of working image-by-image while from floor plan reconstruction, our method is suitable for multi-room environments. However, Sigma-FP distances from previous works in that we jointly consider the following aspects: (i) the fact that we seek to obtain the floor plan of environments without the strong assumption of MW or boxy world, (ii) being able to achieve it sequentially image-by-image instead of needing of a complete point cloud of the environment by registering high-level features of the world, *i.e.* planes [24], (iii) dealing with both the uncertainty in the robot localization and in the plane extraction and (iv) accounting to 3D openings such as doors and windows.

III. Method Overview

Given a wheeled mobile robot equipped with an RGB-D camera, whose localization over time is known with some uncertainty, we aim to incrementally generate a plane-based 3D floor plan of the environment. To do so, for each input image, we propose to extract planar patches of the visible walls and their respective openings (*i.e.* doors and windows). Then, the recognized patches are transformed from the camera frame to a global frame, where they are fused with previously detected patches to build a global 3D model (see Figure 2). Next, we provide a detailed description of the different stages of our proposal.

A. Plane Extraction and Characterization

For each input RGB-D image, we first carry out a per-pixel semantic segmentation of the RGB image with an off-the-shelf deep network that provides candidate pixels to belong to *walls* that eventually will define the floor plan of the building. Since this segmentation is not error-free, we treat the output of this network as observations that will be further

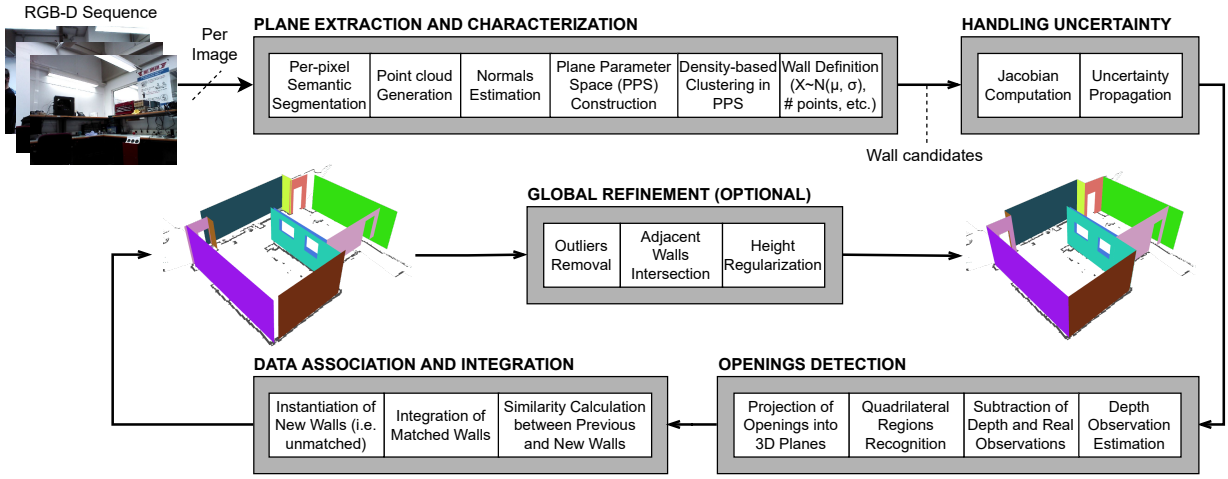


Fig. 2: From a sequence of RGB-D images and relying on an off-the-shelf robot localization method, for each image, Sigma-FP robustly extracts a set of wall candidates which are characterized by a Gaussian distribution, its openings and a set of features. The extracted planes are transformed into a global frame, where their uncertainty are also propagated. Finally, the transformed planes are integrated into a global 3D model. At the end of the floor plan reconstruction, an optional step can be performed to refine the result.

filtered. In a second step, we exploit the spatial organization of the depth image by considering that each point p_i of the point cloud defines a planar patch (π_{p_i}) composed of the point itself and its neighbors within a specific radius. Each point is then annotated with both the normal vector ($\mathbf{n}_{p_i}$) (pointing towards the interior of the environment, *i.e.* towards the camera location) and the distance-to-origin (d_{p_i}). Finally, to perform the plane segmentation over the candidate planar patches, we transform the point cloud from the Cartesian space into the Plane Parameter Space (PPS) [15], a more suitable space where the segmentation of planes can be performed with higher robustness to noise.

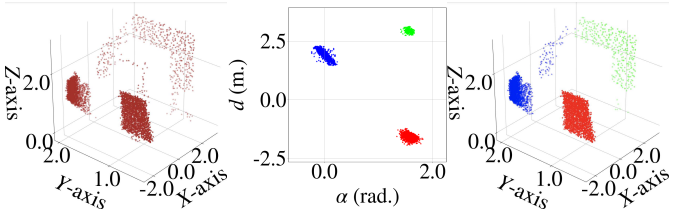
Given a plane in the Cartesian space defined by $\pi = [\mathbf{n}_\pi, d_\pi]^T$ with $\mathbf{n}_\pi = [n_x, n_y, n_z]^T$, its representation in the PPS is a point $\mathbf{p}_\pi$ computed as follows:

$$\mathbf{p}_\pi = \begin{bmatrix} \alpha \\ \beta \\ d \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{n_y}{n_x}\right) \\ \cos^{-1}(n_z) \\ d_\pi \end{bmatrix}, \quad (1)$$

where α and β are the azimuth and elevation angles of the normal vector, respectively, and d is the distance-to-origin of the plane.

Exploiting that planes in the Cartesian space are represented by single points in the PPS, and that different planar patches belonging to the same plane should satisfy that their associated points in the PPS are close to each other, the fusion and segmentation of wall planes is done in this convenient space (see Figure 3). Note that ideally, two planar patches from the same plane are represented by the same point in the PPS. In practice, due to noise and other errors, the resulting points are not equal but similar.

Walls' segmentation is then performed by applying a spatial density clustering algorithm (DBSCAN [25]) in the PPS. This plane extraction approach is more robust to noise, while less computationally expensive for multi-plane scenarios (as it is our case) than other widely employed approaches such as RANSAC [9].



(a) *wall*-filtered input point cloud. (b) Plane segmentation in the PPS. (c) Point cloud of clustered wall planes.

Fig. 3: Overview of the plane segmentation process. The *wall*-filtered input point cloud refers to the point cloud generated after the per-pixel semantic segmentation of the RGB image. Note that the clusterized point cloud shows fewer points than the input because points belonging to a cluster with few points are considered as outliers and then removed.

Based on the fact that in indoor environments, the vast majority of walls are orthogonal to the floor and ceiling, we adopt the Atlanta world (AW) assumption. This means that a plane representing a wall must meet $\beta \triangleq 90°$ while $\alpha \in [-\pi, \pi]$. This assumption is less restrictive than the MW, considered in recent works [12], [22].

Finally, provided that the planar patch segmentation results in K clusters, for each cluster we define a wall ${}^C\Omega_k$ composed of a set of features. Concretely, a multivariate Gaussian distribution fitted over its parameters in the PPS referenced w.r.t. the robot frame, *i.e.* ${}^R\pi_k \sim \mathcal{N}({}^R\alpha_k, {}^Rd_k; \mu_{R\pi_k}, \Sigma_{R\pi_k})$, its dimensions (maximum and minimum bounds) in the Cartesian space, the number of planar patch candidates in the cluster, and the openings detected in the plane (explained in Section III-C). Note that the wall segmentation is expressed w.r.t. the robot frame because the camera-robot transformation is known and uncertainty-free, hence previously to the segmentation, we transform the point cloud from the camera frame to the robot frame.

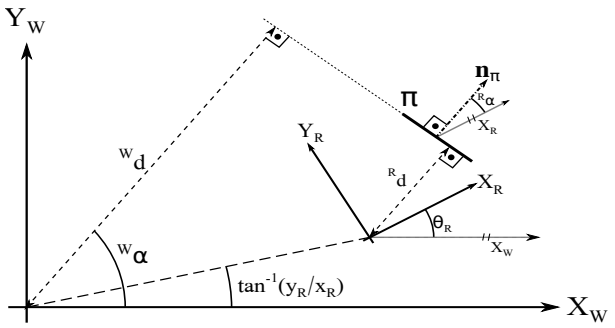


Fig. 4: Transformation of the parameters of a plane π from the robot frame to the world frame.

B. Handling Uncertainty

In this work, we consider the uncertainty in the camera pose when acquiring the images, which derives from the uncertainty in the robot localization. We assume that the camera-robot relative pose (${}^{R}T_{C}$) is fixed and exactly known (error-free), leading to ${}^{W}T_{C} = {}^{W}T_{R}\,{}^{R}T_{C}$, where W, R and C stand for the world, robot and camera frames, respectively. Moreover, the robot-world transform is assumed to be given by a generic localization method (*e.g.* [26], [27]) or a SLAM algorithm (*e.g.* [28], [29]). Concretely, we assume that the estimated robot pose (*i.e.* position (x_R, y_R) and orientation θ_R) is represented by a Gaussian distribution $T_R \sim \mathcal{N}(x_R, y_R, \theta_R; \mu_{T_R}, \Sigma_{T_R})$, which is a standard representation for the robot pose. Note that the robot pose is referred w.r.t. the world frame, although the superscript W is omitted to simplify the notation.

To propagate the uncertainty from the camera pose to the planes detected in the images, we must transform the Gaussian distributions representing planes in the PPS w.r.t the robot frame, to the global world frame. The conversion between both spaces (see Figure 4) is given by:

$$\begin{bmatrix} {}^{W}\alpha \\ {}^{W}d \end{bmatrix} = f(T_R, {}^{R}\pi) = \begin{bmatrix} {}^{R}\alpha + \theta_R \\ {}^{R}d + \delta\cos\left({}^{W}\alpha - \tan^{-1}\left(\frac{y_R}{x_R}\right)\right) \end{bmatrix} \tag{2}$$

where $\cos(\cdot)$ is the cosine function and $\delta = \sqrt{x_R^2 + y_R^2}$.

Accounting for their Gaussian nature, the mean value of the plane coordinates in PPS w.r.t the global frame ($\mu_{W\pi}$) can be computed by applying Eq. (2) directly, while for the case of the covariance matrix, it must include the propagation of the robot pose uncertainty, computed as:

$$\Sigma_{W\pi} = \mathrm{J}_{T_R}\,\Sigma_{T_R}\,\mathrm{J}_{T_R}^{T} + \mathrm{J}_{R\pi}\,\Sigma_{R\pi}\,\mathrm{J}_{R\pi}^{T}, \tag{3}$$

where J_{T_R} and $\mathrm{J}_{R\pi}$ are Jacobians of $f(\cdot,\cdot)$ evaluated at $(\mu_{T_R}, \mu_{R\pi})$, respectively. The resulting Jacobians are:

$$\mathrm{J}_{T_R} = \begin{bmatrix} 0 & 0 & 1 \\ \frac{x_R\cos(\gamma) - y_R\sin(\gamma)}{\delta} & \frac{x_R\sin(\gamma) + y_R\cos(\gamma)}{\delta} & -\delta\sin(\gamma) \end{bmatrix}, \tag{4}$$

$$\mathrm{J}_{R\pi} = \begin{bmatrix} 1 & 0 \\ -\delta\sin(\gamma) & 1 \end{bmatrix}, \tag{5}$$

where $\sin(\cdot)$ and $\cos(\cdot)$ refer to the sine and cosine functions, respectively, $\gamma = {}^{W}\alpha - \tan^{-1}\left(\frac{y_R}{x_R}\right)$ and $\delta = \sqrt{x_R^2 + y_R^2}$.

C. Opening Detection

The detection of openings, such as doors or windows, is of paramount importance to obtain realistic floor plans. Yet, it is a challenging task because openings may not necessarily be seen from a convenient perspective and their observation on depth maps is prone to noise. To address this problem, for each image we first obtain the visible wall planes in it (see Section III-A), and then compute their projection onto the depth image ($\widetilde{\mathcal{I}}_{\pi}^{D}$). Openings are finally detected by comparing the projections with the real depth image masked to the respective walls ($\mathcal{I}_{\pi}^{D}$).

Since, in general, wall planes are not parallel to the image plane, in order to estimate $\widetilde{\mathcal{I}}_{\pi}^{D}$ it is required to apply an image rectification over the wall projection. The latter is carried out by computing the 2D homography between the projection of the wall and the physical wall using the Direct Linear Transformation (DLT) method [30]. Then, since the correspondences between the boundaries of the wall and their projection in the image plane are known, and also the wall pose w.r.t. the camera frame, we can estimate how the wall should be projected on the depth sensor by performing a linear interpolation from the depth value of the wall boundaries in the rectified image. Next, to obtain $\widetilde{\mathcal{I}}_{\pi}^{D}$, we undo the image rectification with the inverse homography matrix. By subtracting $\widetilde{\mathcal{I}}_{\pi}^{D}$ from $\mathcal{I}_{\pi}^{D}$, both occlusions and openings are highlighted as shown in Figure 5. Since we are only interested in openings, we impose that $\mathcal{I}_{\pi}^{D} - \widetilde{\mathcal{I}}_{\pi}^{D} > 0$, hence $\mathcal{I}_{open(\pi)}^{D} = \max(0, \mathcal{I}_{\pi}^{D} - \widetilde{\mathcal{I}}_{\pi}^{D})$.

The openings we are looking for are doors and windows, which are mostly rectangular in real-world. For this reason, we impose a constraint to just search for quadrilateral regions in $\mathcal{I}_{open(\pi)}^{D}$. Each extracted region ${}^{\mathcal{I}}r$ is annotated with the coordinates, in pixels, of its four corners $({}^{\mathcal{I}}\mathbf{c}_{r,1}, {}^{\mathcal{I}}\mathbf{c}_{r,2}, {}^{\mathcal{I}}\mathbf{c}_{r,3}, {}^{\mathcal{I}}\mathbf{c}_{r,4})$ in the image plane. Finally, their positions in the 3D world are determined to incorporate them into the global floor plan. To do so, using the inverse intrinsic matrix of the camera we compute the 3D projection line ${}^{C}\ell_{c_r}$ w.r.t. the camera frame for each corner ${}^{\mathcal{I}}\mathbf{c}_r$:

$$\begin{aligned} {}^{C}\ell_{c_r} &= \lambda P^{T}(PP^{T})^{-1}\,{}^{\mathcal{I}}\mathbf{c}_r \\ &= \lambda \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ x_0 & y_0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \underbrace{\frac{1}{f^2}\begin{bmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ -x_0 & -y_0 & f^2 + x_0^2 + y_0^2 \end{bmatrix}}_{(PP^T)^{-1}} {}^{\mathcal{I}}\mathbf{c}_r \\ &= \frac{\lambda}{f}\begin{bmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & f \\ 0 & 0 & 0 \end{bmatrix} {}^{\mathcal{I}}\mathbf{c}_r = \lambda \begin{bmatrix} \frac{{}^{\mathcal{I}}c_{r,x} - x_0}{f} \\ \frac{{}^{\mathcal{I}}c_{r,y} - y_0}{f} \\ 1 \\ 0 \end{bmatrix}, \end{aligned} \tag{6}$$

where P is the projection matrix, f is the focal length of the camera, (x_0, y_0) is the camera center and λ is the parameter of the parametric line. As ${}^{C}\ell_{c_r}$ is given in homogeneous coordinates, a zero in its fourth element means a 3D line.

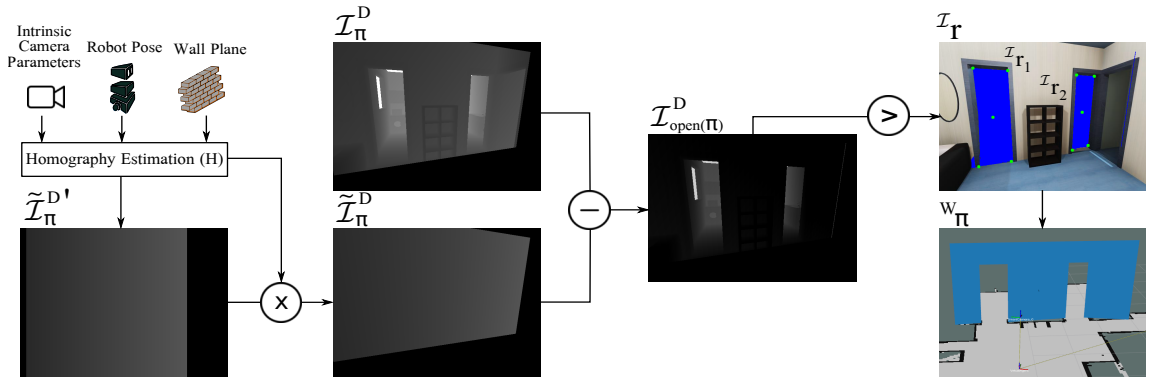


Fig. 5: Overview of the openings' detection process. From each extracted plane, we estimate its rectified projection in the depth sensor ($\widetilde{\mathcal{I}}_{\pi}^{D'}$) based on the knowledge of the camera parameters, the robot pose and the wall plane representation. Through the estimation of the homography between the wall and its projection, we recover the projection of the wall with perspective ($\widetilde{\mathcal{I}}_{\pi}^{D}$). Subtracting the real depth observation ($\mathcal{I}_{\pi}^{D}$) and the estimation ($\widetilde{\mathcal{I}}_{\pi}^{D}$), we obtain regions with occlusions or openings. Then, considering just openings, we extract quadrilateral regions representing doors and windows. Next, openings are projected into the 3D world and integrated in the global model.

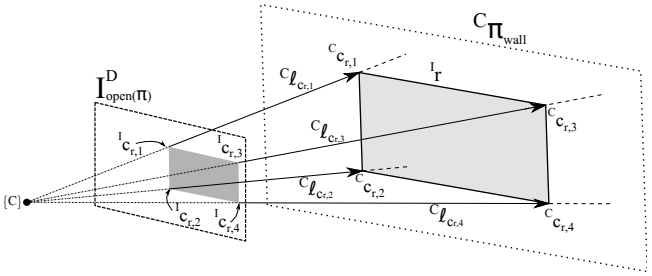


Fig. 6: Illustration of the projection of the openings corners in the image plane into the 3D space. The corners in the 3D world corresponds to the intersection between each projection line ${}^{C}\ell_{c_r}$ and the plane ${}^{C}\pi$.

Since the opening must belong to the plane ${}^{C}\pi$, we can obtain the 3D back-projected point ${}^{C}\mathbf{c}_r$ by computing the intersection between ${}^{C}\ell_{c_r}$ and ${}^{C}\pi$: ${}^{C}\mathbf{c}_r = {}^{C}\ell_{c_r} \cap {}^{C}\pi$ (see Figure 6). Then, this 3D point is transformed to the world frame.

D. Data Association and Integration

An incremental reconstruction of the 3D floor plan requires to sequentially integrate new extracted planar patches into the global model. Thus, for every set of walls ${}^{W}\mathbf{\Omega}^{t}$ segmented at time instant t, we must verify whether the walls already exists in the set of observed walls ${}^{W}\mathbf{\Omega}^{1:t-1}$ or not. If a wall matches an existing one, we merge their features, otherwise the wall is initialized in the global representation.

For the sake of clarity, from now on we omit the superscript W since all the variables are referred to the world frame. To discern whether two walls Ω_j and Ω_k represent the same physical wall, we carry out a twofold assessment: plane representation similarity ($s_{j,k}$) and minimum euclidean distance ($d_{j,k}$) between their planar patches (π_j and π_k). Measuring the similarity enables to determine whether both planar patches belong to the same infinite plane. However, given the fact that two different physical walls can be defined by the same infinite plane (see Figure 7), we additionally measure the minimum distance between planar patches to avoid merging planar patches with the same support plane but that represent different physical walls. Concretely, we consider that two walls

Fig. 7: Sample scenario where two different physical walls (Ω_1 and Ω_2) are represented by the same infinite plane π_∞.

match when $s_{j,k} < \tau_s$ and $d_{j,k} < \tau_d$, where τ_s and τ_d are the similarity and minimum distance thresholds, respectively.

The similarity is measured by computing the Bhattacharyya[1] distance between the Gaussian distributions of their plane representation in the PPS (π_j and π_k) as follows:

$$s_{i,j} = \frac{1}{8}(\mu_j - \mu_k)^T \Sigma^{-1} (\mu_j - \mu_k) + \frac{1}{2}\ln\left(\frac{\det\Sigma}{\sqrt{\det\Sigma_j \det\Sigma_k}}\right), \tag{7}$$

where $\Sigma = \frac{\Sigma_j + \Sigma_k}{2}$.

The association step is performed following an all-vs-all approach, thus N walls can be matched together. For each set of matched planes, we obtain a representative wall through a linear combination of their multivariate Gaussian distributions $\pi_n \sim \mathcal{N}(\mu_{\pi_n}, \Sigma_{\pi_n}), n = 1, \ldots, N$, which are considered independent of each other. Then, the resulting Gaussian distribution is computed as a weighted sum and is defined by:

$$\mu_\pi = \sum_{n=1}^{N} a_n \mu_{\pi_n}, \;\; \Sigma_\pi = \sum_{n=1}^{N} a_n^2 \Sigma_{\pi_n}. \tag{8}$$

where $a_n = \frac{\rho_n}{\sum_{l=1}^{N}\rho_l}$, being ρ_n the number of points in the cluster, i.e. that give rise to the wall, and satisfying that $\sum_{n=1}^{N} a_n = 1$. Note that as the first variable of the Gaussian distribution is angular, its mean cannot be computed through the arithmetical average, but is determined as follows:

$$\overline{\alpha} = \text{atan2}\left(\frac{1}{N}\sum_{n=1}^{N}\sin(\alpha_n), \frac{1}{N}\sum_{n=1}^{N}\cos(\alpha_n)\right), \tag{9}$$

[1]Note that the use of the Bhattacharyya distance is just a choice of the authors, but other statistical distances such as Mahalanobis distance or Kullback–Leibler divergence are also valid.

TABLE I: Comparison of plane's and opening's estimation errors for the simulated environments. Best results are marked in bold. Note that this evaluation is not feasible in real-world environments because of the lack of a ground-truth.

		Walls		Openings
Method	Scene	α-error (rad.)	d-error (m.)	IoU
Sigma-FP	Small	0.031 ± 0.039	0.054 ± 0.045	0.842 ± 0.049
	Non-MW	$\mathbf{0.030 \pm 0.012}$	$\mathbf{0.131 \pm 0.089}$	0.710 ± 0.177
	Large	0.019 ± 0.012	$\mathbf{0.069 \pm 0.066}$	0.763 ± 0.074
Sigma-FP + GR	Small	0.019 ± 0.005	$\mathbf{0.046 \pm 0.039}$	$\mathbf{0.856 \pm 0.060}$
	Non-MW	$\mathbf{0.030 \pm 0.012}$	$\mathbf{0.131 \pm 0.089}$	$\mathbf{0.725 \pm 0.187}$
	Large	0.019 ± 0.012	$\mathbf{0.069 \pm 0.066}$	$\mathbf{0.781 \pm 0.072}$
Gankhuyag *et al.* [12]	Small	$\mathbf{0.0 \pm 0.0}$	0.160 ± 0.050	–
	Non-MW	0.090 ± 0.125	0.630 ± 0.760	–
	Large	$\mathbf{0.0 \pm 0.0}$	0.320 ± 0.211	–
Floor-SP [11]	Small	$\mathbf{0.0 \pm 0.0}$	0.076 ± 0.063	–
	Non-MW	0.176 ± 0.231	0.302 ± 0.421	–
	Large	$\mathbf{0.0 \pm 0.0}$	0.184 ± 0.098	–

where atan2$(\cdot)$ denotes the 2-argument arctangent function.

Concerning the remaining features of the wall, the number of points in the cluster that votes for the same wall is updated by $\rho = \sum_{n=1}^{N} \rho_n$, and the dimensions of the fused wall are determined as the convex hull of the all integrated planar patches. Finally, the integration of openings is carried out in a two-step process. First, we determine the openings from the different walls which refers to the same physical opening by measuring its similarity using the Intersection over Union (IoU) function. Then, matched openings are integrated by computing their global convex hull, and unmatched openings are included unmodified.

E. Global Map Refinement

Once the complete floor plan is generated, we perform an optional refinement stage which comprises: i) outliers removal, ii) adjacent walls intersection and iii) height regularization. Concretely, given the fact that during the inspection each extracted planar patch votes for a wall in the scene, we consider that a represented wall is an outlier when it is poorly voted compared to adjacent walls.

Depending on the intended use of the floor plan reconstruction, it could be necessary to refine the representation of the walls, for example when exploited as a BIM model [8]. In this sense, we include a step that computes the intersection between adjacent walls to enhance the walls' extent. Also, assuming that the ceiling is represented by a single plane, we extend the height of the walls to the maximum height detected (an example of the result in Figure 1).

IV. Experimental Validation

A. Setup, Datasets and Baseline

To evaluate the performance of our proposal, we carry out a set of experiments comparing Sigma-FP with two state-of-the-art methods, Floor-SP [11] and Gankhuyag *et al.* [12]. It should be noted that both methods require the entire point cloud of the environment in advance, and that comparison with [6] has been discarded because of the requirement of 360-images, which are not easily available.

Comparison is performed over synthetic and real-world data to account for quantitative and qualitative results. For the former, we employ the synthetic dataset Robot@VirtualHome [31], analyzing three representative scenes (*i.e.* a small, a non-MW and a large environment). For the evaluation with real-world data, we consider the household scene from OpenLORIS [32], and a set of data collected by teleoperating a mobile robot in our lab.

TABLE II: Performance of the evaluated methods for floor plan reconstruction and opening detection. First column refers to the dataset (R@VH: Robot@VirtualHome, MAPIR: MAPIR-Lab and OL-S: OpenLORIS-Scene). Best results are marked in bold.

		Walls			Openings		
	Method	Precision	Recall	F1-score	Precision	Recall	F1-score
R@VH	Sigma-FP	88.50%	**93.33%**	**90.39%**	**95.76%**	**72.73%**	**82.55%**
	Sigma-FP + GR	90.22%	91.11%	89.99%	**95.76%**	**72.73%**	**82.55%**
	Gankhuyag *et al.* [12]	86.05%	60.00%	68.62%	–	–	–
	Floor-SP [11]	**94.53%**	76.33%	82.53%	–	–	–
MAPIR	Sigma-FP	91.67%	**100.00%**	95.65%	**100.00%**	**77.78%**	**87.50%**
	Sigma-FP + GR	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**77.78%**	**87.50%**
	Gankhuyag *et al.* [12]	**100.00%**	72.73%	84.21%	–	–	–
	Floor-SP [11]	53.33%	72.73%	61.54%	–	–	–
OL-S	Sigma-FP	**93.75%**	**88.24%**	**90.91%**	**100.00%**	**62.50%**	**76.92%**
	Sigma-FP + GR	**93.75%**	**88.24%**	**90.91%**	**100.00%**	**62.50%**	**76.92%**
	Gankhuyag *et al.* [12]	81.82%	52.94%	64.29%	–	–	–
	Floor-SP [11]	46.43%	76.47%	57.78%	–	–	–

To complete the setup, we rely on the Panoptic FPN [33] implemented in Detectron2 [34] to preprocess the RGB images and to obtain a per-pixel segmentation of walls candidates, and the well-known Adaptive Monte Carlo Localization (AMCL) [27] method to obtain a probabilistic localization of the robot.

B. Quantitative Results

Table I shows the results for the compared methods on the three selected environments from Robot@VirtualHome [31], depicting the errors in the plane parameters (α, d), as well as the IoU for the openings. It can be seen that our proposal generalizes properly for all three environments, keeping the error values relatively low in comparison with Floor-SP [11] and Gankhuyag *et al.* [12]. An exception to the latter is the α-error for both MW scenes (*i.e.* Small and Large), as they meet the MW assumption imposed by [12] and the prior assumption of MW from Floor-SP. In contrast, when this assumption is not met (*i.e.* Non-MW scene), the α-error is particularly significant. Thus, generally speaking, it can be said that our method reduces considerably the error in the plane representation of the walls compared to both state-of-the-art methods. Moreover, Table I illustrates through the intersection over union values that our method is able to recognize properly the openings in the environment.

Extending the comparison to also account for the real-world scenarios, we analyze the precision results in Table II, noticing that the three evaluated methods show an overall good performance. Since precision does not account for the number of detected walls, we also compute the recall. In this sense, Sigma-FP outperforms significantly the other methods, overcoming the second-best method by a 17%. A combined measure of precision and recall is the F1-score, where our method demonstrates a strong overall performance. Related to the openings' detection, our proposal exhibits a high value of F1-score, which means that is able to identify and represent properly most of the openings in the scene. However, since our proposed opening detection phase is carried out in the image plane, it is required that both the wall and the opening are

TABLE III: Processing time of the main stages of the evaluated methods. Note that depicted times for both Floor-SP and [12] do not account for the point cloud registration.

		Sigma-FP	Gankhuyag *et al.* [12]	Floor-SP [11]
Online	Semantic Segmentation	126.11 ms	–	–
	Plane Extraction	115.09 ms	–	–
	Opening Detection	12.27 ms	–	–
	Plane Matching	11.21 ms	–	–
Offline Processing		29.49 ms	29.48 s	237.72 s

simultaneously visible in the image. In this sense, openings that are not properly observed cannot be detected, which is reflected directly in the recall results.

Table III completes the comparison by illustrating the time performances of the three evaluated methods. As long as the semantic segmentation network does not act as a bottleneck, our method is able to work in parallel to the neural network (except for the first image), achieving an online operation at $\sim$7 Hz, a sufficient frequency for a robot inspecting an indoor environment. Operating online allows the robot to generate the floor plan incrementally, which can be used at any moment. In contrast, the point cloud-based methods require finishing both the inspection of the environment and the data generation. Hence, given the fact that the methods work with large amount of data, the processing time is considerable higher while the map is not available during the inspection.

C. Qualitative Results

Figure 8 illustrates the 3D floor plans obtained by the evaluated methods for a set of representative environments, including non-MW scenes. An important aspect that can be observed is that Sigma-FP is able to represent both sides of a wall, which together with the 3D openings enhance the quality of the representation in comparison to the state-of-the-art methods. Another aspect to highlight from Sigma-FP is that it is able to represent properly non-MW scenarios, contrary to Floor-SP, which tends to over-segment a single wall into multiple sections or the method described in [12], which directly is unable to reconstruct non-MW scenes. From the results obtained by Floor-SP, it should be mentioned that this method generates closed-loop rooms and hence, it generates walls that does not exist. Referring to the method from [12], as it detects walls based on a 2D density map, in cases where walls are significantly occluded it fails to detect the walls.

D. Case Study: Uncertainty in Sequential Reconstruction

To demonstrate the importance of considering the uncertainty in the robot pose for sequential 3D floor plan reconstruction, we carried out a set of experiments in the MAPIR-UMA Lab using Sigma-FP working with three different localization methods: AMCL [27], Scan Matching [26] and ORB-SLAM2 [29]. For the latter, we fixed the covariance matrix of the estimated pose since in the available implementation such matrix is not provided. As it can be seen in Figure 9, when the uncertainty is available, Sigma-FP weights incoming observations accordingly, relying more on those with less uncertainty and obtaining a more accurate reconstruction. In contrast, using ORB-SLAM2, all observations are integrated equally, resulting in a floor plan with a higher level of error.

V. Conclusions and Future Work

In this work, we propose a method to generate a 3D floor plan of a multi-room environment from a sequence of RGB-D images captured by a wheeled mobile robot. Our method operates in an image-by-image basis, extracting for each image the visible planar patches of walls, which are characterized by multivariate Gaussian distributions. The integration of the planar patches into a 3D global model is achieved by considering the probabilistic localization of the robot with uncertainty, and the corresponding propagation to the planar patches' representation. Moreover, our method accounts for openings like doors and windows, and relaxes the common geometry assumptions of Manhattan world or boxy world, to just Atlanta world, *i.e.* walls are orthogonal to the vertical axis, broading the application range.

Results demonstrate that our method successfully reconstruct the 3D floor plan of scenes with different settings, achieving low error and a relatively low computational load in comparison with other approaches. For future work, we plan to incorporate the robot localization problem in the method formulation and not using the localization estimation just as a service, hence making information to flow bidirectionally to improve the localization. Furthermore, we plan to integrate Sigma-FP with place categorization algorithms to extend the semantic information of the 3D floor plan, using this information to segment the floor plan into rooms.

References

[1] D. Fernandez-Chaves, J. Ruiz-Sarmiento, N. Petkov, and J. Gonzalez-Jimenez, "Vimantic, a distributed robotic architecture for semantic mapping in indoor environments," *Knowledge-Based Systems*, vol. 232, p. 107440, 2021.

[2] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2020, pp. 1689–1696.

[3] Y. Wang, S. James, E. K. Stathopoulou, C. Beltrán-González, Y. Konishi, and A. Del Bue, "Autonomous 3-d reconstruction, mapping, and exploration of indoor environments with a robotic arm," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3340–3347, 2019.

[4] J. L. Matez-Bandera, J. Monroy, and J. Gonzalez-Jimenez, "Efficient semantic place categorization by a robot through active line-of-sight selection," *Knowledge-Based Systems*, vol. 240, p. 108022, 2022.

[5] J.-R. Ruiz-Sarmiento, C. Galindo, and J. Gonzalez-Jimenez, "Building multiversal semantic maps for mobile robot operation," *Knowledge-Based Systems*, vol. 119, pp. 257–272, 2017.

[6] B. Solarte, Y.-C. Liu, C.-H. Wu, Y.-H. Tsai, and M. Sun, "360-dfpe: Leveraging monocular 360-layouts for direct floor plan estimation," *IEEE Robotics and Automation Letters*, 2022.

[7] S. Yang, D. Maturana, and S. Scherer, "Real-time 3d scene layout from a single image using convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2016, pp. 2183–2189.

[8] S. Murali, P. Speciale, M. R. Oswald, and M. Pollefeys, "Indoor scan2bim: Building information models of house interiors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2017, pp. 6126–6133.

[9] A. Phalak, V. Badrinarayanan, and A. Rabinovich, "Scan2plan: Efficient floorplan generation from 3d scans of indoor scenes," *arXiv preprint arXiv:2003.07356*, 2020.

[10] C. Liu, J. Wu, and Y. Furukawa, "Floornet: A unified framework for floorplan reconstruction from 3d scans," in *ECCV*, 2018, pp. 201–217.

[11] J. Chen, C. Liu, J. Wu, and Y. Furukawa, "Floor-SP: Inverse cad for floorplans by sequential room-wise shortest path," in *IEEE ICCV*, 2019, pp. 2661–2670.

[12] U. Gankhuyag and J.-H. Han, "Automatic 2d floorplan cad generation from 3d point clouds," *Applied Sciences*, vol. 10, no. 8, p. 2817, 2020.

[13] C. Yan, B. Shao, H. Zhao, R. Ning, Y. Zhang, and F. Xu, "3d room layout estimation from a single rgb image," *IEEE Trans. Multimedia*, vol. 22, no. 11, pp. 3014–3024, 2020.

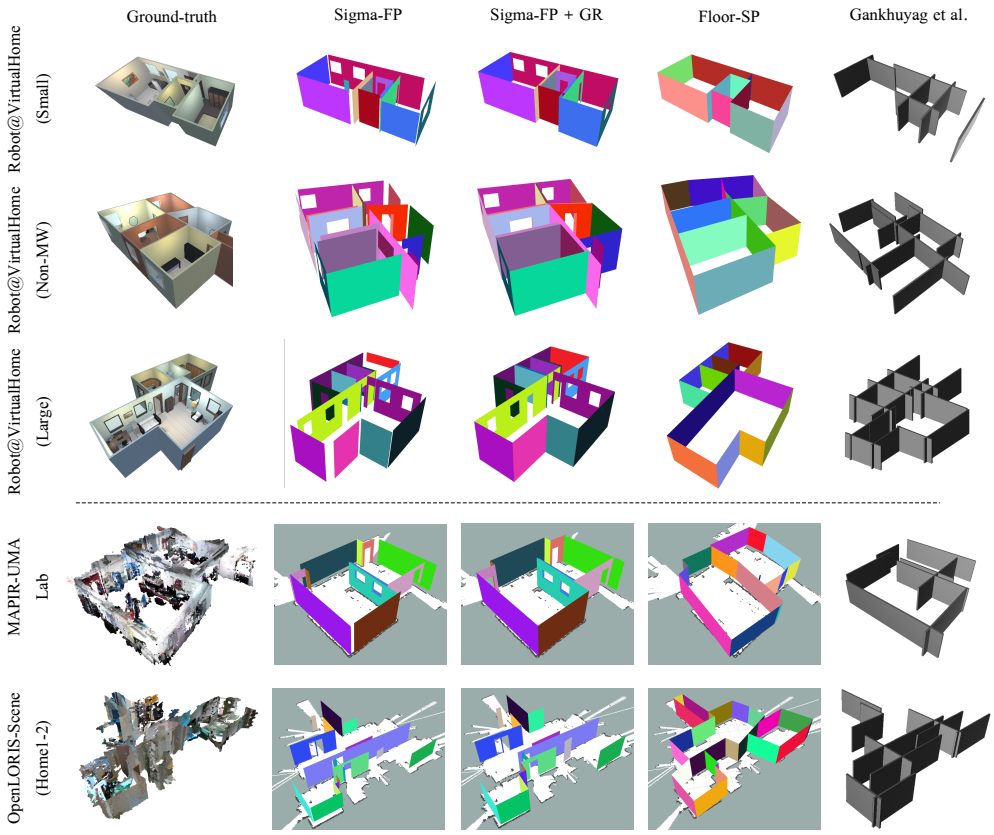
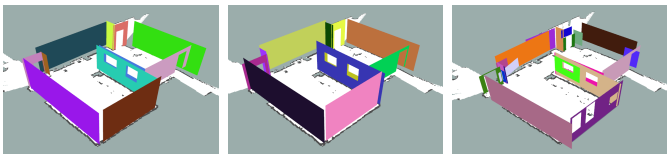


Fig. 8: 3D floor plan reconstructions for simulated and real-word environments. The walls' height have been set to a fixed value for visualization, while their colors are selected randomly to facilitate the identification of the different walls. For real-world scenes, a 2D occupancy grid-map is included as reference.



(a) AMCL [27] (b) Scan Matching [26] (c) ORB-SLAM2 [29]

Fig. 9: Comparison of Sigma-FP working in the MAPIR-UMA Lab under different localization methods.

[14] C.-Y. Lee, V. Badrinarayanan, T. Malisiewicz, and A. Rabinovich, "Roomnet: End-to-end room layout estimation," in *IEEE ICCV*, 2017, pp. 4865–4874.

[15] Q. Sun, J. Yuan, X. Zhang, and F. Sun, "Rgb-d slam in indoor environments with sting-based plane feature extraction," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 3, pp. 1071–1082, 2017.

[16] G. Schindler and F. Dellaert, "Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments," in *IEEE CVPR*, vol. 1, 2004, pp. I–I.

[17] Z. Kang, J. Yang, Z. Yang, and S. Cheng, "A review of techniques for 3d reconstruction of indoor environments," *ISPRS Int. J. Geo-Inf.*, vol. 9, no. 5, p. 330, 2020.

[18] G. Pintore, C. Mura, F. Ganovelli, L. Fuentes-Perez, R. Pajarola, and E. Gobbetti, "State-of-the-art in automatic 3d reconstruction of structured indoor environments," in *Comput. Graph. Forum*, vol. 39, no. 2, 2020, pp. 667–699.

[19] J. M. Coughlan and A. L. Yuille, "Manhattan world: Compass direction from a single image by bayesian inference," in *IEEE ICCV*, vol. 2, 1999, pp. 941–947.

[20] J. Zhang, C. Kan, A. G. Schwing, and R. Urtasun, "Estimating the 3d layout of indoor scenes and its clutter from depth sensors," in *IEEE ICCV*, 2013, pp. 1273–1280.

[21] H. Howard-Jenkins, S. Li, and V. Prisacariu, "Thinking outside the box: generation of unconstrained 3d room layouts," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 432–448.

[22] A. Phalak, Z. Chen, D. Yi, K. Gupta, V. Badrinarayanan, and A. Rabinovich, "Deepperimeter: Indoor boundary estimation from posed monocular sequences," *arXiv preprint arXiv:1904.11595*, 2019.

[23] H. Izadinia, Q. Shan, and S. M. Seitz, "Im2cad," in *IEEE CVPR*, 2017, pp. 5134–5143.

[24] M. Kaess, "Simultaneous localization and mapping with infinite planes," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2015, pp. 4605–4611.

[25] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[26] E. Pedrosa, A. Pereira, and N. Lau, "Efficient localization based on scan matching with a continuous likelihood field," in *IEEE Int. Conf. on Auton. Robot Syst. and Compet.*, 2017, pp. 61–66.

[27] D. Fox, "Kld-sampling: Adaptive particle filters," *Advances in neural information processing systems*, vol. 14, 2001.

[28] R. Gomez-Ojeda, F.-A. Moreno, D. Zuniga-Noël, D. Scaramuzza, and J. Gonzalez-Jimenez, "Pl-slam: A stereo slam system through the combination of points and line segments," *IEEE Trans. Robot.*, vol. 35, no. 3, pp. 734–746, 2019.

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

[30] R. Szeliski, *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[31] D. Fernandez-Chaves, J.-R. Ruiz-Sarmiento, A. Jaenal, N. Petkov, and J. Gonzalez-Jimenez, "Robot@virtualhome, an ecosystem of virtual environments and tools for realistic indoor robotic simulation," *Expert Systems with Applications*, p. 117970, 2022.

[32] X. Shi, D. Li, P. Zhao, Q. Tian, Y. Tian, Q. Long, C. Zhu, J. Song, F. Qiao, L. Song *et al.*, "Are we ready for service robots? the openloris-scene datasets for lifelong slam," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2020, pp. 3139–3145.

[33] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *IEEE/CVF CVPR*, 2019, pp. 6399–6408.

[34] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," 2019.