2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

# WayIL: Image-based Indoor Localization with Wayfinding Maps

Obin Kwon[1,2*], Dongki Jung[2], Youngji Kim[2], Soohyun Ryu[2],
Suyong Yeon[2], Songhwai Oh[1], and Donghwan Lee[2]

***Abstract*— This paper tackles a localization problem in large-scale indoor environments with wayfinding maps. A wayfinding map abstractly portrays the environment, and humans can localize themselves based on the map. However, when it comes to using it for robot localization, large geometrical discrepancies between the wayfinding map and the real world make it hard to use conventional localization methods. Our objective is to estimate a robot pose within a wayfinding map, utilizing RGB images from perspective cameras. We introduce two different imagination modules which are inspired by how humans can comprehend and interpret their surroundings for localization purposes. These modules jointly learn how to effectively observe the first-person-view (FPV) world to interpret bird-eye-view (BEV) maps. Providing explicit guidance to the two imagination modules significantly improves the precision of the localization system. We demonstrate the effectiveness of the proposed approach using real-world datasets, which are collected from various large-scale crowded indoor environments. The experimental results show that, in 85% of scenarios, the proposed localization system can estimate its pose within 3m in large indoor spaces. Project Site: `https://rllab-snu.github.io/projects/WayIL/`**

## I. INTRODUCTION

Have you ever had difficulty finding your way in large shopping malls, train stations, or airports? These public areas usually provide a wayfinding map to help in such situations. Wayfinding maps are designed to assist visitors in locating themselves within the environment. These maps commonly illustrate various landmarks or shops in polygonal forms. The polygons might not always represent structural features (walls or doors) in the real world; they could outline semantic areas like a food court or a playground. Since the primary purpose of a wayfinding map is to convey information rather than accurately represent the environment, it often highlights specific areas in detail and simplifies less significant regions. Furthermore, while extra information like GPS or satellite image is possible in outdoor environments, it can be difficult to use such information indoors. However, humans can still locate themselves on a wayfinding map without expensive sensors or precise maps. This paper aims to develop a localization system inspired by these human capabilities.

Why can a human understand such an abstracted, simplified, and even inaccurate map? We believe two factors are related to such ability. First, humans inherently can grasp the layout and structures of pathways and align them with maps [1], [2]. We can identify features like corridors or junctions from visual signals, giving us a general idea of the location. Secondly, humans can spot and correlate landmarks shown on maps with those in the real-world view, aiding in precise localization [3], [4]. In this study, we focus on the first factor. Our objective is to create a localization system that recognizes and understands spatial structures and layouts.

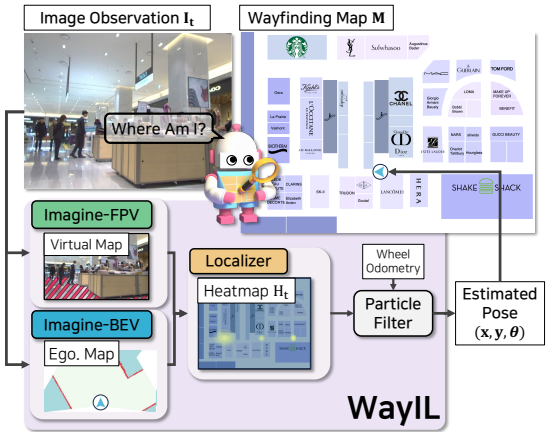



Fig. 1: **WayIL system overview.**

The overview of our proposed localization system, WayIL (pronounced 'whale'), is shown in Figure 1. To facilitate the understanding of surroundings, we introduce two types of imagination modules into the localization system, *Imagine-FPV* and *Imagine-BEV*. *Imagine-FPV* module imagines in a first-person-view (FPV), how the wayfinding map would appear in current RGB images. It learns which part of the FPV image would be drawn on the BEV map and which would not. *Imagine-BEV* module imagines in a bird-eye-view (BEV), how the current surroundings would be drawn as a wayfinding map. We attached the two imagine modules to a localizer module. The localizer module transforms the FPV image into an egocentric BEV representation and compares it with the BEV map. During this process, the localizer module learns useful latent features for localization with the assistance of two imagine modules. We found that attaching the two imagine modules significantly benefits localization accuracies. Additionally, indoor environments often consist of simple and repetitive structures, making it challenging to pinpoint a specific location from a single image. We extend our method with the particle filter algorithm, which effectively resolves such ambiguities by weighting and resampling potential positions based on their likelihood.

The contribution of this paper can be summarized as follows:

- We introduce a novel localization system with inaccurate wayfinding maps, relying on image and wheel

* Work done during the internship at NAVER LABS
[1] Department of Electrical and Computer Engineering and ASRI, Seoul National University, Seoul, 08826, South Korea. (email: obin.kwon@rllab.snu.ac.kr, songhwai@snu.ac.kr)
[2] NAVER LABS, Seongnam, 13561, South Korea. (email: {dongki.jung, youngji.b.kim, soohyun.ryu, suyong.yeon, donghwan.lee} @naverlabs.com) (Corresponding author: Donghwan Lee.)
This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-01190, [SW Star Lab] Robot Learning: Efficient, Safe, and Socially-Acceptable Machine Learning).

odometry data. The proposed robot localization system can accurately estimate its pose in large-scale indoor environments without additional knowledge, such as GPS signals and satellite images.
- We present two imagine modules designed to learn effective latent representation by imagining from both first-person-view (FPV) and bird-eye-view (BEV). These modules play an essential role in boosting localization accuracy.
- We demonstrate how the proposed method can be integrated with the particle filter algorithm. This integration efficiently handles ambiguities and enables consistent tracking in dynamic environments.

## II. Related Work

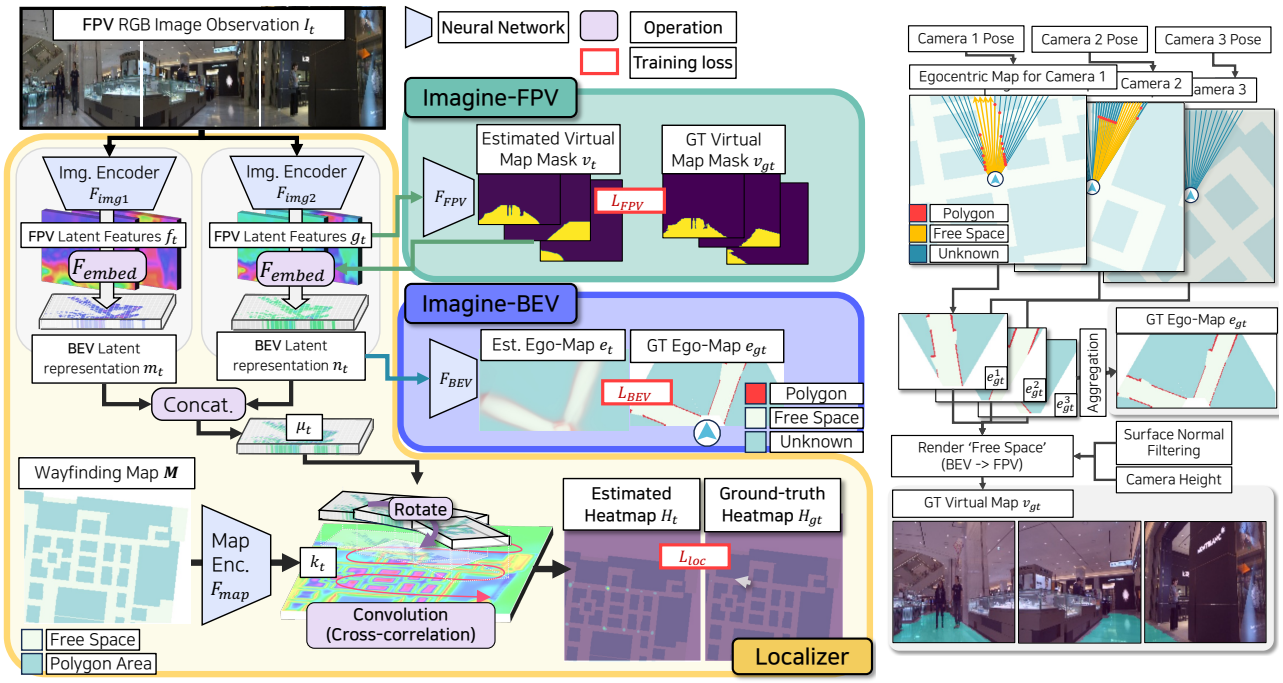
The localization problem with inaccurate maps has been studied consistently over time. **Architectural floorplans** provide information about the overall layout of an environment, without having to explore the actual environment in person. Although they only depict the structure of the environment and omit movable items, a floorplan provides clear geometric cues with a consistent scale (such as walls or doors). Several methods [5]–[9] concentrated on aligning observations with these structural hints from the floorplan. **Sketch maps** drawn by a human hand may contain both structural elements (like walls) and semantic elements (such as furniture, plants, or signage) in imprecise shapes. Hence, direct structural alignment with the real world is not guaranteed. Previous work [10]–[13] demonstrated successful localization and navigation using sketch maps. However, most experiments are limited to small, simple, and static environments like offices or laboratories. **Wayfinding map** considered in this paper has similar characteristics to a sketch map, but it handles much larger and more crowded environments, such as a department store or public station. Wayfinding maps usually portray a large number of stores or landmarks in polygonal shapes. Each polygon on the map does not always mean structural elements in the actual environment. Furthermore, some unnecessary elements can be omitted from the map, or some elements in the map can be drawn in detail. Despite substantial studies on other types of maps, there is limited research on localization using wayfinding maps. Wang *et al*. [14] explored visual localization in a shopping mall with a wayfinding map. They focused on utilizing clear semantic indicators like store signs, based on the assumption of a pre-established database containing the names and locations of each store. The robot estimates its pose based on the detected store signs and the estimated storefront exteriors. In contrast, we wanted to develop a more general localization system that operates even when the sign detector is ineffective or in the absence of a store database. Therefore, we focused on implicit geometric indicators such as identifying potential polygon areas or free spaces, and interpreting 3D surroundings into 2D map representation.

More recent studies, particularly with deep learning, have focused mostly on outdoor environments. **Satellite images** are actively investigated for localization. They are easily accessible from the internet and provide visual features of the landscape. The appearance of the area in a first-person-view can be roughly estimated using satellite images. Several prior studies [15]–[21] utilize such visual features to align first-person-view and bird-eye-view. However, since wayfinding maps do not contain these visual features, it is challenging to apply such approaches. On the other hand, **OpenStreetMap** [22] consists of polygons and provides semantic labels for each respective area, such as roads, buildings, lakes, parks, or forests. This closely resembles wayfinding maps which depict the environment in an abstract way. Earlier research [23]–[31] mainly aimed at identifying semantic cues from the first-person-view image and aligning them with map annotations. More recent approaches have investigated extracting useful latent features by end-to-end learning for localization. Samano *et al*. [32] and Zhou *et al*. [33] proposed to learn a common latent vector space shared by perspective RGB images and cropped map regions. Meanwhile, Sarlin *et al*. [34] proposed a localization system which transforms FPV image into a neural BEV representation, and matches with the map representation. Advancing beyond these initiatives, we build a localization system designed for large-scale indoor settings with a wayfinding map. As we consider a wayfinding map which has fewer semantic labels (namely three: blank, free space, and polygon area) than OpenStreetMap, we focus on enhancing the comprehension of the relationship between FPV and BEV. This is achieved by incorporating two imagine modules. We found that these modules significantly increase localization accuracies in indoor environments. Considering real-life usage, wayfinding maps may contain ambiguous scales and may have been partially distorted. In this paper, we also demonstrate the robustness of the proposed localization in such a situation.

## III. Problem Formulation

We consider a localization problem in unfamiliar large-scale indoor places. We assume that no prior knowledge is given except for a wayfinding map $M \in \mathbb{R}^{H_{\mathrm{map}} \times W_{\mathrm{map}}}$. $H_{\mathrm{map}}$ and $W_{\mathrm{map}}$ refer to the height and width of the wayfinding map, respectively. The wayfinding map consists of multiple polygons, which can be stores or arbitrary regions. We consider a wayfinding map without specific semantic labels other than free space or polygon area. This enables the general implementation on various environments without dependency on specific regions. Each pixel in the wayfinding map can be among the three categories: *blank* area not inside the map, *free space* area, and *polygon* area inside the polygons.

In this paper, we consider a robot navigating around the area, and it needs to localize its 3DOF pose $(x, y, \theta)$ in a wayfinding map $M$ based on the observed RGB images $I_t$ at the time step $t$. The robot has three perspective cameras to cover a $180^\circ$ field of view. This choice can vary depending on the user, and the proposed method is not limited to the number of cameras. The robot has access to wheel odometry information which can be inaccurate. The system consists of three modules: *Localizer*, *Imagine-FPV*, and *Imagine-BEV*. The details of each module are shown in Figure 2a, and we will explain each module in the following sections (III-A, III-B, III-C). Then, we describe how we adapted the particle filter algorithm to WayIL system in section III-E.




(a) **Components of WayIL system.** (b) **GT label generation process.**

Fig. 2: (a) **Components of WayIL system.** Three modules (Imagine-FPV, Imagine-BEV, Localizer) compose WayIL system. (b) **Ground-truth (GT) label generation process.** For each camera pose, an egocentric map is created. Using these egocentric maps, 180° ego map $e_{\text{gt}}$ is constructed. Also, the free space of each egocentric map is rendered onto the FPV image, using known camera height. (Best viewed in color.)

### A. *Localizer Module*

The *Localizer* module converts the FPV images into an egocentric BEV representation and compares it with the BEV wayfinding map $M$. The module utilizes a convolution-based localization method inspired by recent research [34]–[36]. The image encoder $F_{\text{img1}}$ processes the FPV images $I_t \in \mathbb{R}^{3\times H\times W\times C}$ into latent features $f_t \in \mathbb{R}^{3\times H\times W\times D}$, where $H$ and $W$ refer to the height and width of the image, $C$ and $D$ refer to the size of the channel. Then each pixel feature is embedded into BEV representation according to its estimated 3D position. The method for estimating the 3D position of each pixel can vary [34]–[38], and we adopted the embedding method from [36], which utilizes depth information and known camera parameters. As only RGB information is given, we estimate depth information from RGB using an off-the-shelf model (Omnidata [39], [40]). Since Omnidata outputs normalized depth and surface normal, a conversion process to metric scale is necessary for creating a consistent BEV representation. Inspired by [41], we estimate the floor area using the estimated surface normal, and adjusted the depth scale so that the height of the floor matches the known camera height. We can map each FPV pixel feature from $f_t$ into 3D position using estimated depth information and known camera parameters. The features corresponding to the same x-y position are averaged, and the averaged feature is recorded in the corresponding grid cell in BEV representation $m_t \in \mathbb{R}^{H_{\text{ego}}\times W_{\text{ego}}\times D}$. This FPV-to-BEV translation process is denoted as $F_{\text{embed}}$ in Figure 2a.

The wayfinding map $M$ is processed by a convolutional neural network $F_{\text{map}}$ to transfer the map into the same latent space as $m_t$. The processed wayfinding map becomes $k_t \in \mathbb{R}^{H_{\text{map}}\times W_{\text{map}}\times D}$. Then, the developed egocentric BEV representation $m_t$ is compared with $k_t$ by cross-correlation operation (or convolution) with rotating it. The output of the cross-correlation is a heatmap $H_t \in \mathbb{R}^{H_{\text{map}}\times W_{\text{map}}\times R}$, where $R$ represents the number of rotations, dividing 360° into $R$ discrete intervals. The localizer module is trained to generate $H_t$ which has a likelihood of camera pose in each grid cell. The wayfinding map is given after cropped into a fixed $(H_{\text{map}} \times W_{\text{map}})$ size, and the robot pose can be any pixel in the cropped map $M$. In this paper, the cropped map $M$ represents a real-world area of $64m \times 64m$. The loss term for localization is a cross-entropy loss: $L_{\text{loc}} = \text{CrossEntropy}(H_t, H_{\text{gt}})$, where $H_{\text{gt}}$ refers to the one-hot distribution indicating which grid cell the robot is present, similar formulation with [34]–[36]. We hypothesized that this localization approach entails the robot gradually learning where to look for localization and gaining insight into how the observed area would be represented on the BEV map. Moreover, we believe that having this comprehension is an important factor that allows localization from two different viewpoints. To enhance these abilities, we propose two *Imagine* modules designed to explicitly guide this process. In our WayIL system, we introduce a separate stream of building another BEV latent representation $n_t$. In contrast to $m_t$ which implicitly learns latent features solely through $F_{\text{loc}}$, $n_t$ is explicitly guided to learn useful features for localization by additional loss terms with $Imagine$ modules. The $n_t$ is generated from another FPV latent feature $g_t \in \mathbb{R}^{3\times H\times W\times D}$, which is encoded by $F_{\text{img2}}$. The $n_t$ is concatenated with $m_t$,

becoming $\mu_t$. The final BEV representation $\mu_t$ is used for convolution-based localization in WayIL. We present how the $Imagine$ modules affect $n_t$ in the following sections.

### B. Imagine-FPV Module

In *Imagine-FPV* module, U-Net Encoder $F_{\mathrm{FPV}}$ takes FPV latent features $g_t$ and output estimated map mask $v_t \in \mathbb{R}^{3\times H\times W}$ for each image. This module is trained to estimate which part of the observed image would be drawn on the wayfinding map. The process of making ground-truth virtual map mask $v_{\mathrm{gt}}$ is shown in Figure 2b. First, we crop a local egocentric map for each camera from the wayfinding map. Then we conduct ray casting on each map starting from the ground-truth camera pose, to obtain only observable parts. As these rays progress, areas before hitting any lines are labeled as 'Free space.' Upon encountering any line of polygons, the meeting points are labeled as 'Polygon.' The subsequent parts of the rays beyond these points are labeled as 'Unknown.' We mask out the rest of the not observable area as 'Unknown.' The labeled egocentric maps become $e_{\mathrm{gt}}^{1,2,3}$. Then we render the 'free space' in $e_{\mathrm{gt}}^{1,2,3}$ onto each FPV image using known camera height and intrinsic parameters, with the assumption that these are on the ground. These rendered parts become $v_{\mathrm{gt}}$, with a value of 1 in areas of the FPV image depicted on the map as free space, while the remainings are 0. We also utilize the estimated surface normal in III-A, to filter obstacles in rendered free space area. Pixels not sharing a similar orientation with the floor are marked as 0 in $v_{\mathrm{gt}}$. A convolutional neural network $F_{\mathrm{FPV}}$ is trained by $L1$ loss to make $v_t$ to be same with $v_{\mathrm{gt}}$: $L_{\mathrm{FPV}} = L_1(v_t, v_{\mathrm{gt}})$. Note that the area $v_{\mathrm{gt}}$ considering is different from simply classifying floor from an image. We can see that $v_{\mathrm{gt}}$ in Figure 2b does not highlight the inside area of the store. The estimated $v_t$ is utilized to mask out unimportant areas in building $n_t$. The FPV latent feature $g_t$ is multiplied with $v_t$, and only the estimated free space pixels are embedded into $n_t = F_{\mathrm{embed}}(g_t * v_t)$.

### C. Imagine-BEV Module

In *Imagine-BEV* module, a neural network $F_{\mathrm{BEV}}$ is trained to predict an egocentric local map $e_t$ from the developed BEV representation $n_t$. During this process, the network can comprehend the surroundings by imagining how the observed area will appear from the BEV perspective, especially within the context of the wayfinding map $M$. We combine the local egocentric maps $e_{\mathrm{gt}}^{1,2,3}$ to be a single $180°$ egocentric map $e_{\mathrm{gt}}$ for the center camera, according to each camera pose. This $e_{\mathrm{gt}}$ has three classes corresponding to 'Polygon', 'Free space', and 'Unknown'. The loss term for $e_t$ is formulated as a cross-entropy loss: $L_{\mathrm{BEV}} = \mathrm{CrossEntropy}(e_t, e_{\mathrm{gt}})$. The estimated egocentric map is not used during inference time, but training this module serves as an auxiliary task and encourages extracting better features for scene understanding and localization. The experiments show that attaching these two modules significantly increases the localization performances on the wayfinding map. We provide the experiment result and the ablation study in Section IV.

### D. Training

The proposed localization framework is trained using three loss functions correspond to each module in the previous sections. The total loss term is as follows:

$$L_{\mathrm{total}} = L_{\mathrm{loc}} + L_{\mathrm{FPV}} + L_{\mathrm{BEV}}. \tag{1}$$

Using this loss term, the neural networks in WayIL ($F_{\mathrm{img}}, F_{\mathrm{img2}}, F_{\mathrm{FPV}}, F_{\mathrm{BEV}}, F_{\mathrm{map}}$) are jointly trained. Each *Imagine* module contributes to learning the final BEV representation $\mu_t$ used in *Localizer* module. As a result, training $L_{\mathrm{FPV}}$ and $L_{\mathrm{BEV}}$ influences $L_{\mathrm{loc}}$, subsequently helping in performance enhancement.

### E. Particle Filter Algorithm

This paper focuses on large-scale indoor environments, which frequently have repetitive patterns and numerous dynamic obstacles. Maps for such environments might not always be accurate, leading the localization system to occasionally select an incorrect location that looks similar to the answer. To address this issue, we integrated the particle filter algorithm with the WayIL system to combine information over time and improve estimation.

At the start of the episode, the particles are scattered on the map, and the wheel odometry information is used for updating each particle pose. Each particle has a pose of $(h, w, r)$, which corresponds to a grid in $H_t \in \mathbb{R}^{H_{\mathrm{map}}\times W_{\mathrm{map}}\times R}$, and the value from $H_t$ becomes the weight of the particle. The weight is continuously multiplied as the particle moves, and particles are resampled according to the weight. In most situations, particles often group into clusters in similar areas, which is common for particle filter-based localization. These clusters maintain similar scores until they receive critical information that could clarify the pose. Therefore, selecting the particle based on the highest weight can lead to imprecise estimates due to possible outliers or inaccurate particle movement. To achieve consistent localization over time, we group the particles using K-means clustering, treating these clusters as potential pose candidates. In this paper, we use five clusters, and the sum of the particles' weights in each cluster determines its score. The centroid of the cluster with the highest score becomes the representative estimated pose.

## IV. Experiments

### A. Dataset

For evaluation, we use NAVER LABS localization dataset [42]. This dataset provides perspective RGB images and wheel odometry information in multiple types of large-scale indoor spaces. In this paper, we use RGB images ($128 \times 128 \times 3$) from three cameras which cover about $180°$ field of view. We used five different indoor spaces from a department store (*Dept. B1*, *Dept. 1F*), shopping center (*COEX B1*), and underground mall areas in metro stations (*Metro1*, *Metro2*). Wayfinding maps of each environment are shown in Figure 3.

The wayfinding maps are manually drawn for each region based on the reconstructed 3D map, where each grid corresponds to $25cm \times 25cm$. Hence, we can calculate the mapping between the pixels in the wayfinding map and the ground-truth poses. The polygons in the wayfinding maps we used correspond to those of actual wayfinding maps found in real environments. There can be some differences between shape details, but they outline the same features like stores, kiosks, counters, escalators, and so on. Considering

| Exp. Num. | Dataset | Method | Sequential Estimation | Mean ↓ | Std ↓ | Median ↓ | MAD ↓ | Recall@X ↑ | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 1m | 3m | 5m |
| 1 | NAVER LABS [42] | YouAreHere [32], [33] | X | 19.33 | 11.66 | 18.63 | 8.81 | 1.9 | 9.4 | 13.7 |
| 2 | | OrienterNet [34] | X | 11.65 | 9.22 | 10.28 | 7.24 | 10.2 | 24.2 | 31.4 |
| 3 | | WayIL (ours) | X | 8.61 | 9.42 | 4.16 | 3.81 | 27.0 | 46.4 | 52.6 |
| 4 | | YouAreHere [32], [33] | Particle Filter | 10.28 | 11.85 | 3.40 | 3.03 | 23.2 | 48.2 | 53.7 |
| 5 | | OrienterNet [34] | Weight Propagation | 5.40 | 7.17 | 1.60 | 1.25 | 39.9 | 60.5 | 66.6 |
| 6 | | OrienterNet [34] | Particle Filter | 4.58 | 6.70 | 1.43 | 0.78 | 31.4 | 68.9 | 75.2 |
| 7 | | WayIL (ours) | Particle Filter | **2.65** | **5.07** | **1.09** | **0.48** | **44.3** | **85.8** | **90.1** |
| 8 | MGL [34] | OrienterNet [34] | Weight Propagation | **2.39** | **3.04** | 1.51 | 0.82 | 29.2 | 73.0 | **90.9** |
| 9 | | WayIL (Ours) | Particle Filter | 2.45 | 4.98 | **1.45** | **0.79** | **31.1** | **74.5** | 90.1 |

TABLE I: **Localization Results on NAVER LABS Dataset [42] and MGL Dataset [34].**

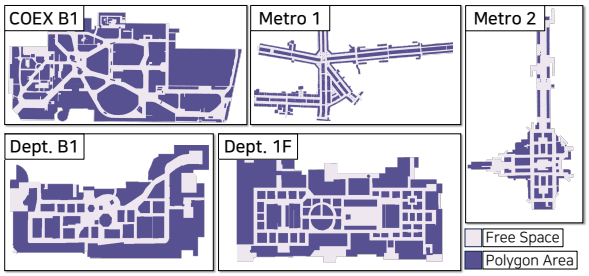



Fig. 3: **Wayfinding Maps for each environment.**

| Imagine Module | | Recall @ Xm ↑ | | |
|---|---|---|---|---|
| FPV | BEV | 1m | 3m | 5m |
| ✘ | ✘ | 31.4 | 66.0 | 71.0 |
| ✔ | ✘ | 39.4 | 81.0 | 86.2 |
| ✘ | ✔ | **45.4** | 81.2 | 86.0 |
| ✔ | ✔ | 44.3 | **85.8** | **90.1** |

TABLE II: **Ablation study on NAVER LABS Dataset.**

the real-life wayfinding maps, we also provide experiments with arbitrary global scales and partially distorted maps. We split each wayfinding map into *train* and *test* areas, and the images from the *train* area are only used for training. Each method is evaluated with images from unseen areas. We sampled 200 sequences from the unseen areas for each region, and a total of 1,000 sequences were evaluated. Each sequence consists of 100 frames of images, which is about 40 seconds in duration.

The final estimated pose at the end of the sequence is used for measuring the localization performance. The average and standard deviation of the distance errors are used for localization. In many cases of failed localization, the pose is predicted to be in a distant but similar space. So it is challenging to determine the localization accuracy by relying solely on the average distance error. Therefore, the localization performances are also measured using median distance error, median absolute deviation (MAD), and recall rate in $1m$, $3m$, and $5m$. We also report the localization results based on a single-step observation in Table I.1-3.

### B. Baseline methods

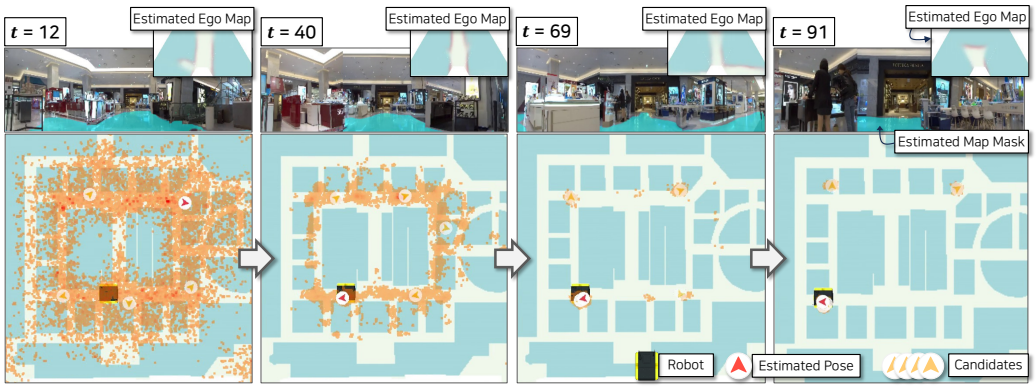
We compare the proposed method (WayIL) with two recent localization methods which address a similar kind of map. Both methods deal with the localization using OpenStreetMap [22] in outdoor environments. YouAreHere [32] learns to transform FPV image and BEV map pair into the same metric latent space. After training, the localization can be done using cosine similarity between the FPV image embedding vector and the BEV map embedding vector. OrienterNet [34] is analogous to WayIL without *Imagine* modules. For sequential estimation, OrienterNet moves the previous heatmap weights using precise positional shifts and multiplying them with a new heatmap. This does not consider resampling or noisy shift data. We denote this sequential estimation approach as 'Weight Propagation', and we also provide a comparison with the particle filter algorithm used in WayIL. Note that the robot has only access to noisy wheel odometry information. We changed the precise pose change information into wheel odometry in weight propagation.

### C. Comparison with Baselines

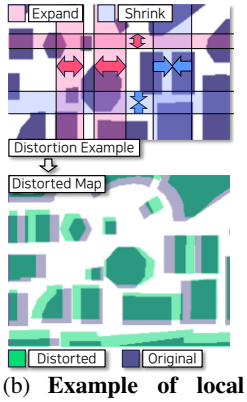
Table I shows the experiment results in the NAVER LABS dataset. First, we can see that localization using a single image observation (I.1-3) shows much lower accuracy than sequential estimation (I.4-7). Integrated with the particle filter algorithm, WayIL can localize the robot pose within 3m, in 85.8% of cases. We can see that WayIL more effectively estimates the robot pose than other localization methods. For YouAreHere (I.1,4), we observed that this method effectively aligns image embedding very close to the map embedding of ground-truth position. However, due to the prevalence of similar areas, incorrect selections were often made. Compared to OrienterNet (I.2,5,6), we can verify the effectiveness of the imagine modules. By attaching two imagine modules, the performance of localization is highly increased. We conducted an advanced ablation study on imagine modules in section IV-D. Further, we observed that particle filters can estimate the robot pose more accurately than weight propagation in indoor environments under noisy odometry (I.5 vs I.6).

### D. Ablation Study

Table II shows the results of the ablation study on two imagine modules. Note that the model without imagine modules is analogous to the OrienterNet method with a particle filter algorithm, except for the embedding method ($F_{\text{embed}}$) for BEV representation. While WayIL uses $\mu_t$ for matching, this model only uses $m_t$, without $n_t$ which is generated from *Imagine* modules. The channel size of $m_t$ was doubled for this model, equal to the channel size of $\mu_t$ in the original model. For models with a single imagine module, either $F_{\text{FPV}}$ or $F_{\text{BEV}}$ was removed, respectively. Both the Imagine-FPV module and the Imagine-BEV module increase




(a) **Example of localization episode.**




(b) **Example of local distortion.**

Fig. 4: **(a) Example visualization of a localization episode.** Robot observation and estimated $e_t$ and $v_t$ are shown above the map. The particles (orange) are shown on the wayfinding map. **(b) Example of local distortion and distorted map**. We randomly cut arbitrary patches from the wayfinding map, and distort them by downsizing or emphasizing.

localization performance. We observed that the Imagine-BEV module specifically increases the recall rate in fine-level distances (under $1m$). We believe that training to estimate an egocentric BEV map from latent representations enables the system to capture the spatial structure details of the environment. This comprehension subsequently facilitates more precise localization. When the Imagine-FPV module is combined, this enhances the understanding of surroundings by learning where to look in the FPV image to build effective BEV representation.

### *E. Qualitative Results*

We present an example of a localization episode in Figure 4a. The estimated virtual map mask $v_t$ and egocentric map $e_t$ are also shown in the figure. The *Imagine-FPV* module accurately identifies free space in the FPV image, and the *Imagine-BEV* module correctly detects pathway junctions. The convergence of particles is presented on the map according to time. We can see that the particles are clustered in multiple similar areas, and the cluster with the highest score correctly estimates the robot pose. Additional examples can be found in the accompanying video.

### *F. Distorted Map*

One of the difficulties in using a wayfinding map for localization is the ambiguity of the global scale. Moreover, it often highlights or downsizes specific regions for interpretability. Considering these factors, we conducted localization experiments using distorted wayfinding maps to assess their applicability in real-world scenarios. Figure 4b shows the examples of the distortions. First, we fixed the grid size and randomly emphasized or downsized specific patches of the wayfinding map. Table III shows the experiment results. When there are only local distortions, WayIL shows robust performances. When WayIL is trained with the distortions, it shows similar results to when there are no distortions. Second, we tested WayIL with various grid sizes, randomly changing it between 10cm and 40cm. When WayIL is not trained with the distortions, the performance significantly drops. However, when trained with the distortions, it can recover the localization performances to some extent. From the experiment results, we can see that WayIL can handle the wayfinding map with local distortions and arbitrary grid sizes, which is closer to the real-life situation. The examples of localization on distorted map are presented in the accompanying video.

| Trained w/ Distortion | Grid Size (cm) | Local Distortion | Recall @ Xm ↑ | | |
|---|---|---|---|---|---|
| | | | 1m | 3m | 5m |
| ✘ | 25 | - | 44.3 | 85.8 | 90.1 |
| ✘ | 25 | 80%∼120% | 38.2 | 80.1 | 85.5 |
| ✔ | 25 | 80%∼120% | 45.0 | 82.8 | 85.8 |
| ✘ | 10 ∼ 40 | 80%∼120% | 16.9 | 39.8 | 46.5 |
| ✔ | 10 ∼ 40 | 80%∼120% | 23.8 | 60.9 | 70.4 |

TABLE III: **Localization on Distorted maps.**

### *G. Outdoor: MGL dataset [34], [43]*

Additionally, to verify the usefulness of the proposed method in outdoor environments, we conducted experiments on the Mapillary GeoLocation (MGL) dataset [34]. MGL dataset consists of Monocular RGB images from various cities with paired OpenStreetMap. The results are shown in Table I. 8-9 Although the imagine modules of WayIL are designed for indoor localization, they can also be helpful in outdoor environments by increasing localization performances in sequential inference.

## V. Conclusion

We proposed WayIL, an indoor localization system with wayfinding maps. Compared to other maps previously used for localization, the wayfinding maps are geometrically inaccurate and highlight semantic regions. We introduced two imagination modules, which showed significant performance improvements in indoor environments. Integrated with the particle filter algorithm, the proposed localization system accurately estimates the robot pose in various types of large, crowded indoor environments. If a POI (Point of Interest) database is provided, and there is a detector capable of recognizing various types of shops, it can be combined with WayIL for more accurate localization. Additionally, the developed WayIL can serve as prior knowledge for accurate mapping by assisting in planning efficient exploration routes and aiding navigation.

## REFERENCES


[1] A. K. Lobben, "Navigational Map Reading: Predicting Performance and Identifying Relative Influence of Map-Related Abilities," *Annals of the association of American geographers*, vol. 97, no. 1, 2007.

[2] I. Hemmer, M. Hemmer, K. Kruschel, E. Neidhardt, G. Obermaier, and R. Uphues, "Which children can find a way through a strange town using a streetmap?–results of an empirical study on children's orientation competence," *International Research in Geographical and Environmental Education*, vol. 22, no. 1, 2013.

[3] D. Caduff and S. Timpf, "On the assessment of landmark salience for human navigation," *Cognitive Processing*, vol. 9, 2008.

[4] A. K. Deakin, "Landmarks as Navigational Aids on Street Maps," *Cartography and Geographic Information Systems*, vol. 23, no. 1, 1996.

[5] H. Blum, J. Stiefel, C. Cadena, R. Siegwart, and A. Gawel, "Precise Robot Localization in Architectural 3D Plans," in *Proceedings of the 38th International Symposium on Automation and Robotics in Construction (ISARC)*, 2021.

[6] O. Mendez, S. Hadfield, N. Pugeault, and R. Bowden, "Sedar: Reading floorplans like a human—using deep learning to enable human-inspired localisation," *International Journal of Computer Vision*, vol. 128, 2020.

[7] 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 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015.

[8] H. Chu, D. K. Kim, and T. Chen, "You are Here: Mimicking the Human Thinking Process in Reading Floor-Plans," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2015.

[9] H. Howard-Jenkins and V. A. Prisacariu, "Lalaloc++: Global floor plan comprehension for layout localisation in unvisited environments," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 693–709.

[10] V. Setalaphruk, A. Ueno, I. Kume, Y. Kono, and M. Kidode, "Robot navigation in corridor environments using a sketch floor map," in *Proceedings IEEE International Symposium on Computational Intelligence in Robotics and Automation. Computational Intelligence in Robotics and Automation for the New Millennium*, 2003.

[11] F. Boniardi, A. Valada, W. Burgard, and G. D. Tipaldi, "Autonomous indoor robot navigation using a sketch interface for drawing maps and routes," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.

[12] M. Mielle, M. Magnusson, and A. J. Lilienthal, "Using sketch-maps for robot navigation: Interpretation and matching," in *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2016.

[13] F. Boniardi, B. Behzadian, W. Burgard, and G. D. Tipaldi, "Robot Navigation in Hand-Drawn Sketched Maps," in *European conference on mobile robots (ECMR)*, 2015.

[14] S. Wang, S. Fidler, and R. Urtasun, "Lost Shopping! Monocular Localization in Large Indoor Spaces," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2015.

[15] T. Lentsch, Z. Xia, H. Caesar, and J. F. Kooij, "SliceMatch: Geometry-guided Aggregation for Cross-View Pose Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[16] S. Zhu, T. Yang, and C. Chen, "VIGOR: Cross-View Image Geo-localization beyond One-to-one Retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[17] Y. Shi and H. Li, "Beyond Cross-view Image Retrieval: Highly Accurate Vehicle Localization Using Satellite Image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[18] F. Fervers, S. Bullinger, C. Bodensteiner, M. Arens, and R. Stiefelhagen, "Uncertainty-Aware Vision-Based Metric Cross-View Geolocalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[19] S. Zhu, M. Shah, and C. Chen, "TransGeo: Transformer Is All You Need for Cross-view Image Geo-localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[20] X. Zhang, W. Sultani, and S. Wshah, "Cross-View Image Sequence Geo-localization," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023.

[21] Y. Shi, F. Wu, A. Perincherry, A. Vora, and H. Li, "Boosting 3-dof ground-to-satellite camera localization accuracy via geometry-guided cross-view transformer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 21 516–21 526.

[22] "OpenStreetMap." [Online]. Available: https://www.openstreetmap.org

[23] A. Armagan, M. Hirzer, P. M. Roth, and V. Lepetit, "Accurate Camera Registration in Urban Environments Using High-Level Feature Matching," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2017.

[24] ——, "Learning to Align Semantic Segmentation and 2.5D Maps for Geolocalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[25] T. Vojir, I. Budvytis, and R. Cipolla, "Efficient Large-Scale Semantic Visual Localization in 2D Maps," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2020.

[26] O. Vysotska and C. Stachniss, "Improving SLAM by Exploiting Building Information from Publicly Available Maps and Localization Priors," *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, vol. 85, 2017.

[27] C. Guo, M. Lin, H. Guo, P. Liang, and E. Cheng, "Coarse-to-fine Semantic Localization with HD Map for Autonomous Driving in Structural Scenes," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[28] W.-C. Ma, S. Wang, M. A. Brubaker, S. Fidler, and R. Urtasun, "Find your Way by Observing the Sun and Other Semantic Cues," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[29] P. Panphattarasap and A. Calway, "Automated Map Reading: Image Based Localisation in 2-D Maps Using Binary Semantic Descriptors," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[30] J.-H. Pauls, K. Petek, F. Poggenhans, and C. Stiller, "Monocular Localization in HD Maps by Combining Semantic Segmentation and Distance Transform," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[31] T.-J. Cham, A. Ciptadi, W.-C. Tan, M.-T. Pham, and L.-T. Chia, "Estimating camera pose from a single urban ground-view omnidirectional image and a 2d building outline map," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[32] N. Samano, M. Zhou, and A. Calway, "You Are Here: Geolocation by Embedding Maps and Images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[33] M. Zhou, X. Chen, N. Samano, C. Stachniss, and A. Calway, "Efficient Localisation Using Images and OpenStreetMaps," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[34] P.-E. Sarlin, D. DeTone, T.-Y. Yang, A. Avetisyan, J. Straub, T. Malisiewicz, S. R. Bulo, R. Newcombe, P. Kontschieder, and V. Balntas, "OrienterNet: Visual Localization in 2D Public Maps with Neural Matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[35] J. a. F. Henriques and A. Vedaldi, "MapNet: An Allocentric Spatial Memory for Mapping Environments," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[36] O. Kwon, J. Park, and S. Oh, "Renderable Neural Radiance Map for Visual Navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[37] J. Philion and S. Fidler, "Lift, Splat, Shoot: Encoding Images From Arbitrary Camera Rigs by Implicitly Unprojecting to 3D," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[38] A. W. Harley, Z. Fang, J. Li, R. Ambrus, and K. Fragkiadaki, "SimpleBEV: What Really Matters for Multi-Sensor BEV Perception?" in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[39] A. Eftekhar, A. Sax, J. Malik, and A. Zamir, "Omnidata: A Scalable Pipeline for Making Multi-Task Mid-Level Vision Datasets From 3D Scans," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[40] O. F. Kar, T. Yeo, A. Atanov, and A. Zamir, "3D Common Corruptions and Data Augmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[41] F. Xue, G. Zhuo, Z. Huang, W. Fu, Z. Wu, and M. H. Ang, "Toward Hierarchical Self-Supervised Monocular Absolute Depth Estimation for Autonomous Driving Applications," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[42] D. Lee, S. Ryu, S. Yeon, Y. Lee, D. Kim, C. Han, Y. Cabon, P. Weinzaepfel, N. Guerin, G. Csurka, and M. Humenberger, "Large-Scale Localization Datasets in Crowded Indoor Spaces," in *Proceed-*

*ings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
[43] "Mapillary. ." [Online]. Available: https://www.mapillary.com/app.