Supercharging Floorplan Localization with Semantic Rays

Yuval Grader
Tel Aviv University

Hadar Averbuch-Elor
Cornell University

https://tau-vailab.github.io/SemRayLoc/

Abstract

Floorplans provide a compact representation of the building's structure, revealing not only layout information but also detailed semantics such as the locations of windows and doors. However, contemporary floorplan localization techniques mostly focus on matching depth-based structural cues, ignoring the rich semantics communicated within floorplans. In this work, we introduce a semantic-aware localization framework that jointly estimates depth and semantic rays, consolidating over both for predicting a structural-semantic probability volume. Our probability volume is constructed in a coarse-to-fine manner: We first sample a small set of rays to obtain an initial low-resolution probability volume. We then refine these probabilities by performing a denser sampling only in high-probability regions and process the refined values for predicting a 2D location and orientation angle. We conduct an evaluation on two standard floorplan localization benchmarks. Our experiments demonstrate that our approach substantially outperforms state-of-the-art methods, achieving significant improvements in recall metrics compared to prior works. Moreover, we show that our framework can easily incorporate additional metadata such as room labels, enabling additional gains in both accuracy and efficiency.

1. Introduction

Camera localization is a longstanding problem in computer vision, with significant applications in 3D reconstruction [19, 25–28], augmented reality [11, 16, 23, 29, 31], and navigation [20, 24, 30, 32, 36]. Localization within indoor environments is especially challenging due to the absence of reliable GPS signals and the complexity of reasoning across multiple floors and layers. Hence, to bypass complicated 3D model-based localization techniques, prior work [7, 8, 12, 13, 15, 35] has explored the problem of localizing camera observations within a provided 2D floorplan map by matching depth-based structural cues.

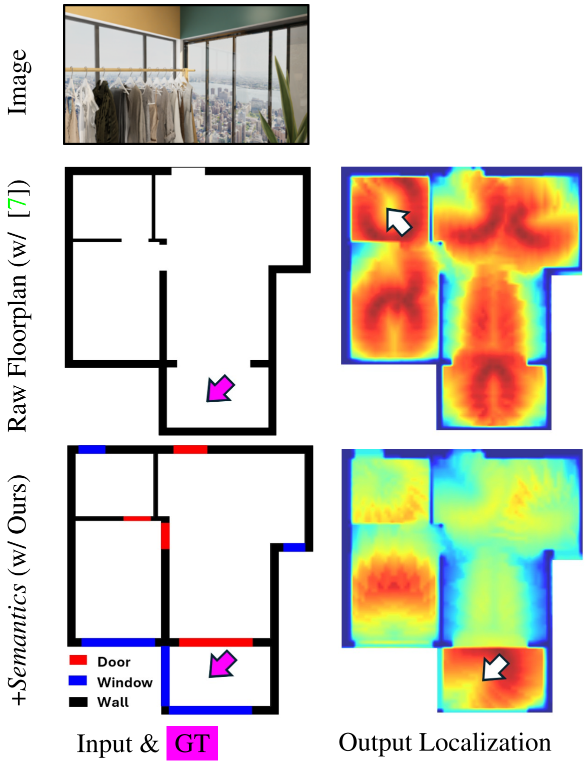

Figure 1. Floorplan localization using a raw binary floorplan (middle row) often yields ambiguous predictions. In this work, we utilize a richer, yet readily available, representation: a floorplan enhanced with semantic labels (bottom row). We present an approach that supercharges floorplan localization with semantic rays, enabling for resolving localization ambiguities, as illustrated by the comparison on the right.

However, while floorplans offer a compact and lightweight scene representation, structural cues within floorplans often correlate with multiple candidate locations, particularly for environments with repetitive or symmetric layouts. Consider the example in Figure 1. *Can you localize the input image within the floorplan?* Provided with just a *raw* (walls only) floorplan, room corners are indistinguishable and hence localization is highly ambiguous, as can also be observed by the probabilities predicted by the state-of-the-art F3Loc [7] technique (middle row, right). To resolve such ambiguities, we are interested in utilizing a slightly different representation: a *semantics*-aware floorplan, such as the one illustrated on the bottom row of Figure 1.

Accordingly, we introduce a semantic-aware ray-based localization framework that integrates semantic cues with depth-based predictions. In particular, we propose a semantic prediction network that predicts accurate semantic ray representations along with optional additional metadata (such as room labels) from a single RGB image with a limited field-of-view. We process these rays to compute a semantic probability volume, which is then fused with depth information for constructing a *structural–semantic* probability volume.

Our framework follows a coarse-to-fine localization strategy. We first operate over a low-resolution image for an efficient floorplan search. This initial search yields the Top-k candidate locations. Finally, we refine these candidates and select the best match using a high-resolution ray representation. By adopting this coarse-to-fine approach, our method effectively constrains the localization search to the most promising regions while keeping the computation cost feasible.

Our experiments demonstrate that our approach yields improvements by factors ranging from two to three across most metrics compared to the state-of-the-art technique [7], upon which our method is built. This substantial gain underscores the effectiveness of fusing semantic and depth ray predictions into a unified probability volume. We further show that our coarse-to-fine strategy offers a flexible tradeoff between accuracy and computational cost, with performance consistently improving as larger candidate sets (Top-k) are evaluated—making our method adaptable to diverse task requirements. Moreover, incorporating additional metadata further enhances precision, leading to significantly improved localization accuracies.

Explicitly stated, our contributions are:

- We introduce a semantic ray prediction network that receives a single RGB image as input.
- We propose an efficient and unified framework that fuses semantic and depth ray predictions into a structural–semantic probability volume, which effectively resolves localization ambiguities.
- Results that demonstrate that significant improvements over state-of-the-art methods.

2. Related Work

Visual Localization. The task of visual localization has received ongoing attention throughout the past several decades. Traditional approaches often rely on image retrieval or on a 3D Structure-from-Motion (SfM) model of the environment. In the *image retrieval* paradigm, methods such as NetVLAD [1] or RelocNet [2] compare a query image against a database of labeled images. Once the closest match is found, the query pose is approximated by the retrieved image's pose. Other methods explicitly construct a 3D SfM model of a scene to establish 2D–3D correspondences between the query image and the reconstructed 3D structure [19, 25–28]. After matching local image descriptors to 3D points, robust solvers estimate the 6-DoF pose.

Recent *learning-based* pipelines deviate from classical 2D–3D matching. Scene-coordinate regression methods predict dense 3D coordinates for every pixel in the query image [6, 29, 31], whereas pose regression methods directly estimate the 6-DoF camera pose via neural networks [16, 32, 36]. Although promising for single-scene scenarios, these methods must be retrained or fine-tuned to handle new environments.

Floorplan Localization. Prior work addressing the task of floorplan localization primarily focused on *depth-based cues*, leveraging image-derived depth predictions or sensors, to match depth obtained from floorplans. In particular, LiDAR-based methods [3, 4, 18, 34] utilize precise laser scans but restrict usability on most mobile devices. Alternative sources of geometric cues, including semi dense visual odometry (SDVO) [8], or point clouds from depth cameras [14], can circumvent heavy LiDAR hardware.

Earlier works compare extracted room edges directly to the 2D layout [5], often assuming knowledge of camera or room height [5, 8]. Other approaches, are embedding RGB images and floorplans into a shared metric space in order to do the localization. For instance, LaLaLoc [13] uses panoramic depth layout image which is rendered at known heights at different locations within the floorplan, where the localization is done by doing similarity in the embedded space. LaLaLoc++ [12] removes the hight assumption by embedding the entire floorplan into the feature space. However, these approaches often require an upright camera pose [12, 13], making them less flexible for hand-held or head-mounted devices. F3Loc [7] localizes by predicting depth rays from a given image and generating probability volumes that indicate the likelihood of each the depth prediction to a particular location on the floorplan. Although effectively leverages geometric cues from depth maps, it does not incorporate semantic information, which may reduce its robustness in environments with repetitive structures or occlusions.

Semantic-based cues are less commonly used for indoor localization, but several works have utilized them for this task. Wang et al.[33] extracts scene texts from images in large indoor spaces and performs localization by matching this text to the floorplan. SeDAR [21] uses a CNN to extract semantic labels from an image and perform Monte Carlo localization. By contrast to our single-image localization framework, it depends on sequential images and depth sensors. PF-Net [15] applies a differentiable particle filter with a learned observation model, relying on a computationally-intensive embedding process that yields limited performance for the single-image scenario. SPVLoc [10] matches captured images to panoramic renderings to estimate location, leveraging additional height information present in these renderings. Similarly, Kim et al. [17] perform localization by using a pre-captured 3D map of the environment. Our approach does not assume the availibity of these additional cues. LASER [22] treats the floorplan as a set of points and synthesizes view-dependent features, including the semantic label of each point, for matching purposes. They embed images into circular features, which are then compared to pose features in the same embedded space. By contrast, we model the semantics as fine-grained ray predictions. Furthermore, unlike these prior works, our approach can operate over images with non-zero pitch and roll.

3. Method

In this work, we propose a floorplan localization framework that jointly estimates semantic and depth rays to infer the 2D camera location and orientation relative to a given floorplan. Specifically, we assume that we are provided with an RGB Image $I \in \mathbb{R}^{h \times w \times 3}$, where h and w denote the height and width of the image, respectively, and a 2D floorplan map $F \in \{0, 1, 2, ..., C\}^{H \times W}$, represented as a matrix of dimensions $H \times W$, where each element is assigned a semantic label from C unique semantic categories, with zero denoting *empty space*. The semantic categories we consider in our work are *wall*, *window* and *door*, but our framework could easily incorporate additional categories (e.g, staircases, columns).

Our objective is to predict the camera's 2D location (x, y) and orientation angle θ at which the image I was captured. That is, given the observation $O_{I,F} = (I, F)$, our goal is to infer the location parameters $S_{I,F} = (x, y, \theta)$. To this end, we adopt a probabilistic framework by modeling the posterior distribution $p(S_{I,F} \mid O_{I,F})$. We discretize the camera pose space as $\mathcal{S} = \{S_i\}$ and define a probability volume $P \in \mathbb{R}^{\hat{H} \times \hat{W} \times O}$ where each element $P(S_i)$ represents the posterior probability $p(S_i \mid O_{I,F})$ for a candidate pose S_i. Here, $\hat{H}$ and $\hat{W}$ denote the number of discretized cells in the x and y dimensions, respectively, and O represents the number of orientation bins. The final predicted camera pose is then given by

$$\hat{S}_{I,F} = \arg\max_{S_i \in \mathcal{S}} p(S_i \mid O_{I,F}).$$

We proceed to provide background (Section 3.1), prior to introducing our semantic prediction network (Section 3.2), which constructs a semantic probability volume. We then describe how it is fused with depth cues to perform floorplan localization (Section 3.3). Finally, we provide training and implementation details (Section 3.4); additional details can be found in the supplementary material. An overview of our approach is provided in Figure 2.

3.1. Background: F3Loc

Our work builds upon F3Loc [7], a recent technique that estimates depth rays for performing floorplan localization given a single image or image sequence. We briefly outline several key components from their work that provide background for our framework. For additional details, we refer readers to their work.

Depth Rays Prediction. Given a query image, a depth prediction network estimates per-column depth values that capture the distance from the camera to the nearest wall along specific angles. These values are then linearly interpolated to produce a fixed set of equiangular depth rays $\hat{r}_d \in \mathbb{R}^l$ that represent the floorplan depth, with l denoting the number of predicted rays.

Estimating Depth Probability Volume. For every candidate location (x, y) on the floorplan and each discrete orientation θ, a corresponding set of reference rays is generated based on the floorplan's geometry. The predicted depth rays are compared with these reference rays to compute a likelihood score for each grid cell and orientation, resulting in a three-dimensional probability volume $P_d \in [0, 1]^{[\hat{H}, \hat{W}, O]}$. For instance, given a $10\,\text{m} \times 7\,\text{m}$ floorplan discretized at $0.1\,\text{m}$ with $10°$ increments in orientation yields a probability volume $P_d \in [0, 1]^{[100,\, 70,\, 36]}$.

The final camera 2D location and orientation are determined by selecting the grid cell with the highest likelihood in the probability volume. This process maximizes the posterior probability, thereby estimating the camera's x and y coordinates as well as its orientation.

3.2. Adding a Semantic Prediction Network

To utilize semantic cues for performing floorplan localization, we propose to add a semantic prediction network that first predicts semantic rays, and then processes these for constructing a semantic probability volume. We provide additional details in what follows, and then present an optional room type prediction component, which can be utilized if room labels are available.

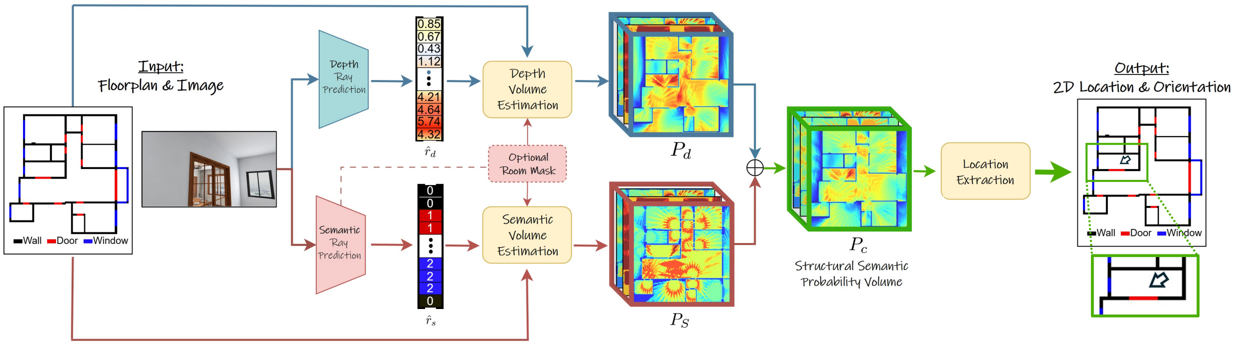

Figure 2. Overview of our pipeline. The input image is processed to generate depth rays, semantic rays, and optionally additional metadata (e.g., room type prediction). We interpolate the ray predictions to a low-resolution representation and generate the depth probability volume P_d and the semantic probability volume P_s (optionally masked according to the room type prediction). These probability volumes are then fused to form the structural-semantic probability volume P_c for efficient coarse localization. Finally, we refine the candidate poses using high-resolution ray predictions and predict the final 2D camera location and orientation, visualized with an arrow on the right.

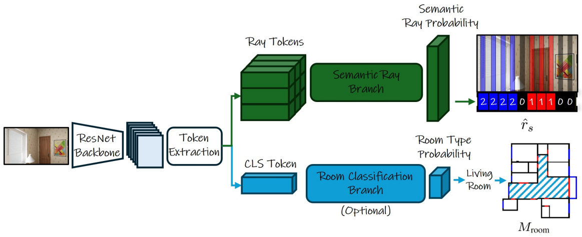

Figure 3. Overview of our semantic prediction network that predicts a set of semantic rays through the *Semantic Ray Branch* (top) and an optional room type value—*e.g.*, `Living Room`—through the *Room Classification Branch* (bottom). The room type is used for extracting the mask M_{room}, as visualized on the bottom right.

Semantic Rays Prediction. Unlike the continuous depth values estimated in prior work, the semantic rays should correspond to semantic categories, which are represented as a set of discrete classes. Therefore, we construct a network that produces a semantic ray representation $\hat{r}_s \in \{1, \dots, C\}^l$ from the image, where each ray is classified into one of C semantic categories. We provide an overview of our semantic ray prediction network in Figure 3.

As illustrated in the figure, our semantic network architecture leverages a pretrained ResNet50 backbone to extract robust, high-level features from an input RGB image I. After reducing the feature channels using a CNN and projecting them into a lower-dimensional subspace, positional encodings are computed to preserve spatial information. Two sets of learnable tokens are introduced: a set of l ray tokens responsible for predicting the semantic ray representation $\hat{r}_s$ and a single (optional) CLS token dedicated for representing *global* room classification information.

A single-head cross attention module integrates these tokens with the flattened spatial features, yielding refined tokens that capture both global context and local details. In the ray branch, the refined ray tokens are first processed by a self-attention block that enables each token to interact with all others, thereby aggregating complementary contextual information. The enriched tokens are then passed through an MLP to produce per-token semantic logits, which after normalization form the final semantic ray vector $\hat{r}_s$. If room labels are available in the dataset, a similar network processes the CLS token for room type prediction, as we further detail later.

Estimating Semantic Probability Volume. To obtain the semantic probability map, $P_s \in [0, 1]^{[\hat{H}, \hat{W}, O]}$, we first need to interpolate the l predicted semantic rays. Regular linear interpolation—which prior work used for depth estimation—is unsuitable in the context of discrete labeling since interpolating between class labels can produce non-valid or semantically meaningless results. Instead, we propose a voting-based interpolation scheme: We reduce the original equiangular rays to the desired count by applying a majority vote within a small neighborhood. We use a window of three rays, assigning the label that appears most frequently in that window to the center target ray; see the supplementary material for the full algorithm. Next, we compute the score for each set of rays by taking the L_1 difference between the predicted semantic labels and the reference labels. The score is then exponentiated and normalized to form the semantic probability volume P_s, which quantifies the likelihood of each candidate pose based on the alignment between the semantic rays and the candidate pose.

Room Type Prediction. In addition to predicting semantic rays, our semantic network can optionally also predict the room type, which corresponds to the room from which the input image was taken. This is achieved by processing the CLS token in the semantic network (see Figure 3). If the predicted room probability exceeds a threshold T_{room}, the predicted room type is then used to construct a mask M_{room} from the polygons associated with that room type. For example, if the model predicts a `Living Room` label with high confidence, the mask M_{room} retains only the regions corresponding to the living room by setting all other areas to zero. Let $\tilde{P}$ denote the original probability volume, then the masked probability volume P is computed as:

$$P = M_{\text{room}} \odot \tilde{P},$$

where $\odot$ denotes element-wise multiplication, thereby filtering out all the probabilities which are not in the living room and substantially narrowing down the search space. A detailed analysis of room type distributions and the model's prediction accuracy is provided in the supplementary material.

3.3. Floorplan Localization via a Structural–Semantic Probability Volume

We obtain the final probability map by generating the depth probability map P_d, following the procedure described in Section 3.1, and the semantic probability map P_s, according to the approach detailed in Section 3.2. To leverage both semantic and geometric cues, we fuse the two probability maps using a weighted combination:

$$P_c = w_s \cdot P_s + w_d \cdot P_d,$$

where w_s denotes the weight given to the semantic cues, while $w_d = 1 - w_s$ represents the weight of the depth cues. These weights are determined using a held-out validation set, as further detailed in Section 4. Note that as both P_s and P_d are generated from interpolated rays, which we refer to as low-resolution rays (see Figure 4, predicted), and hence the initial probability volume P_c is also in low resolution.

Location Extraction. Our approach follows a coarse-to-fine strategy to achieve precise localization while maintaining computational efficiency. Given P_c, we first extract the Top-k candidate poses from the structural-semantic probability volume based on their scores, ensuring that each candidate is separated by at least δ_{res} in translation. Next, for each candidate, we generate an augmented set of orientations by perturbing its original angle in increments of $\pm\delta_{\text{ang}}$ up to a maximum deviation of Δ_{max}, resulting in the final augmented set:

$$[0, \pm\delta_{\text{ang}}, \pm 2\delta_{\text{ang}}, \ldots, \pm\Delta_{\text{max}}].$$

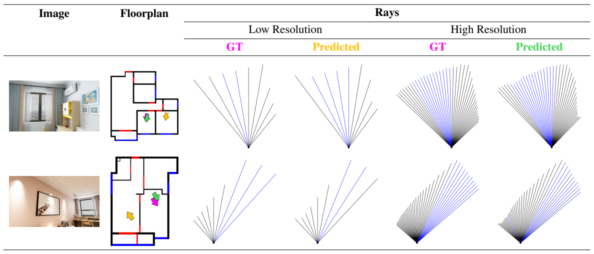

Figure 4. Comparing low-resolution and high-resolution rays with our coarse-to-fine approach. Given an input image (left), we construct a structural–semantic probability volume by comparing low resolution ground-truth and predicted rays (center). Location extraction from this coarse volume directly often yields significant errors, as illustrated with the yellow arrows. Results are refined only for the Top-k candidate poses by comparing the high resolution ground-truth and predicted rays (right). This allows for efficiently extracting more accurate predictions, as further detailed in Section 3.3.

At each candidate location, we compute the corresponding high-resolution ground-truth depth and semantic ray representations, yielding l rays per candidate location. These ground-truth rays are then compared against the original predicted l rays using a similarity metric (see supplementary material for metric details). For a visual comparison of the high-resolution ground-truth and predicted rays, please refer to Figure 4. The candidate with the highest similarity score is selected as the final predicted location.

This coarse-to-fine refinement process effectively leverages the full resolution of the predictions, which were initially interpolated for runtime efficiency, to achieve more precise localization. Note that there is a tradeoff between accuracy and computation time: finer resolutions yield improved precision at the expense of increased computational load. This tradeoff is illustrated in Figure 4. As can be observed, comparing low resolution rays often yields inaccurate predictions. For instance, environments with multiple semantic objects can suffer from loss of critical details (e.g., the left door is omitted in the middle row) in the low-resolution prediction, which can lead to misclassification, further motivating the refinement step. We present a quantitative analysis of this tradeoff in the supplementary material.

3.4. Training and Implementation Details

Both networks are trained in an end-to-end manner. For depth prediction, an L_1 loss supervises the predicted depths against ground-truth depth maps. For semantic prediction, a cross-entropy loss is applied to the predicted semantic labels. If room labels R are available in the dataset, an additional cross-entropy loss is used for the room label, and the network is trained to optimize both objectives jointly. As in prior work, data augmentation techniques—including virtual roll-pitch augmentation—are employed to improve robustness to non-upright camera poses. The networks are optimized using the Adam optimizer with an initial learning rate of 1×10^{-3}. We use a floorplan resolution of 0.1 m and an angular granularity of 10°. We set the number of predicted rays to $l = 40$ and interpolate these to 7 rays during the coarse stage of localization. For the *Location Extraction* module, we use $\delta_{\text{res}} = 0.1$ m, $\delta_{\text{ang}} = 5^{\circ}$, and $\Delta_{\text{max}} = 5^{\circ}$, and refine our results using Top-5 poses.

4. Results

In this section, we present a comprehensive evaluation of our localization method. We begin by introducing the datasets we use in our experiments, followed by a discussion of the baselines we compare against and the evaluation metrics. The main results are reported in Section 4.1. We conduct an ablation study to assess the contributions of various components of our approach in Section 4.2. We report results for our approach under two conditions: one in which room labels are not utilized (denoted as Ours$_s$) and another in which room labels are employed to further refine the predictions (denoted as Ours$_r$). Additional experiments and qualitative results are provided in the supplementary material.

Datasets. We conduct experiments on two popular datasets: Structured 3D (S3D) [37] and ZInD [9]. S3D is a synthetic dataset containing realistic 3D renders of 3,296 houses. We use the fully furnished version of S3D, as employed in previous works. ZInD consists of 1,575 unfurnished homes containing only panoramic images. We crop these panoramas to perspective views with an 80° field of view (as was done in S3D), and generate a fixed-size dataset from the resulting images. For both datasets, we follow their official train and test splits.

Baselines. We compare our approach against two baselines: F3Loc [7] (considering the single-image localization component) and LASER [22]. We also report performance using an ORACLE ray prediction. This oracle ray prediction simulates the best possible performance achievable by our pipeline using ground truth depth and semantic rays. Note that the Oracle ray prediction baseline does not incorporate room-aware features. For F3Loc, results on the ZIND dataset are obtained from our experiments by running their publicly available code on the dataset, as the original paper does not include results on this dataset. We also use the publicly available implementation of LASER. Additional details are provided in the supplementary material.

Metrics. Following prior work [7, 22], we report recall metrics computed at distance thresholds of 0.1 m, 0.5 m, and 1 m. We also report recall for predictions with an orientation error bounded to less than 30° (at the 1 m threshold). Recall is defined as the percentage of predictions that fall within these thresholds.

S3D R@				
Method	0.1m	0.5m	1m	1m 30°
LASER	0.7	6.4	10.4	8.7
F3Loc	1.5	14.6	22.4	21.3
Ours$_s$	5.42	41.87	53.52	52.61
Ours$_r$	**5.70**	**45.53**	**58.78**	**57.49**
ORACLE	61.00	93.84	94.87	94.57
ZInD R@				
Method	0.1m	0.5m	1m	1m 30°
LASER	1.38	11.06	17.55	13.64
F3Loc	0.67	7.90	15.07	11.46
Ours$_s$	2.98	24.00	33.96	29.30
Ours$_r$	**3.31**	**26.60**	**38.01**	**31.86**
ORACLE	26.42	60.85	67.69	65.13

Table 1. Recall performance on the S3D and ZInD datasets. The table reports recall at thresholds of 0.1 m, 0.5 m, 1 m, and 1 m with a 30° orientation tolerance for LASER, F3Loc, our approach without room labels (Ours$_s$), our approach with room labels (Ours$_r$), and an Oracle ray prediction.

4.1. Quantitative Evaluation

Results are reported in Table 1. On both S3D and ZinD, our method more than doubles F3Loc's and LASER performance across all thresholds. Notably, considering S3D over the R@1m30° metric—which reflects the quality of matches between the predicted and actual camera views—in comparison to F3Loc, our method improves by more than three times. Room type predictions yield improvements of 9.2% in R@1m30° on the S3D dataset and 8.7% on the ZInD dataset. We include an additional experiment in the supplementary material that evaluates F3Loc with our refinement module, demonstrating that both the semantic rays and our refinement module provide significant performance gains.

We observe that significant performance improvements are also achieved for a very fine-grained recall metric of 0.1 m, boosting performance from 1.5% (F3Loc) to 5.7% with our approach on S3D. We attribute this to our coarse-to-fine strategy, which effectively refines the coarse location estimates into precise predictions, as further validated in our ablation study.

Figure 5 presents qualitative examples that illustrate how the integration of floorplan semantics with precise depth cues enables our pipeline to effectively resolve localization ambiguities. For instance, in the third row, we can see that F3Loc, which does not use semantics, interprets this image as a blank wall, while LASER misinter-

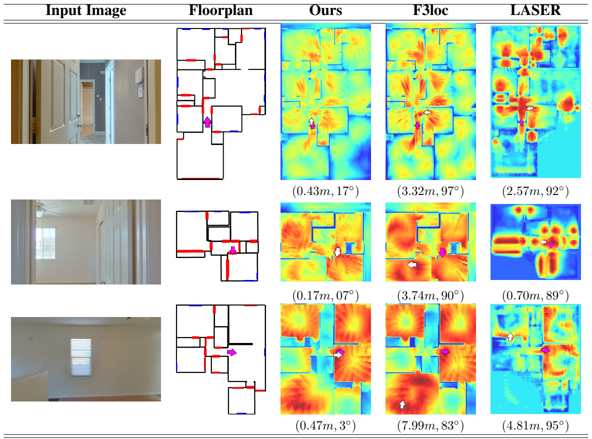

Figure 5. Qualitative comparison of our method with F3Loc and LASER. Warmer colors correspond to regions with higher probabilities. Below each map we report the localization error in meters and degrees. We use arrows to visualize the ground truth location (magenta) and the predicted location (white).

prets the window size and predicts an incorrect location.

4.2. Ablation Study

We demonstrate the effect of incorporating each component of our method on overall localization performance. Specifically, we conduct the following ablations: (1) *Base*, which corresponds to computing the structural-semantic probability volume and selecting the maximum probability without any additional refinement. (2) *Removing semantic interpolation* (denoted as –Interpolation), where we replace our majority voting interpolation with a simple linear interpolation followed by rounding. (3) *Adding room predictions* (denoted as +Room), where we assess the effect of integrating room type predictions into our localization pipeline. (4) *Adding refinement* (denoted as +Refine), which assesses our coarse-to-fine approach, which refines our localization extraction via the Top-K candidate poses. (5) *Adding room predictions and refinement* (denoted as +Room&Refine), which combines both components.

From the results in Table 2, we can see that on the 1m 30° metric, the addition of our refinement module improves performance by 8.6% relative to the baseline. This indicates that a substantial amount of information is lost during the initial interpolation process if not properly addressed, thereby strongly motivating the use of coarse-to-fine strategies in our Location Extraction module. We also observe a significant improvement of 11.5% from the room prediction component, which is discussed in further detail in the supplementary material and visualized in Figure 6. Finally, our majority voting interpolation approach contributes an additional gain of 0.9% compared to a simple interpolation strategy. When combining all these improvements, our method achieves an overall enhancement of 18.6% on the 1m 30° metric relative to our base model.

Method	**0.1m**	**0.5m**	**1m**	**1m 30°**
Base	4.65	38.35	49.40	48.44
– Interpolation	4.73	38.44	48.91	47.99
+ Room	5.12	42.92	55.57	54.04
+ Refine	5.42	41.87	53.52	52.61
+ Room&Refine	**5.70**	**45.53**	**58.78**	**57.49**

Table 2. Ablation study, evaluating the effect of incorporating the different components in our pipeline on the S3D dataset.

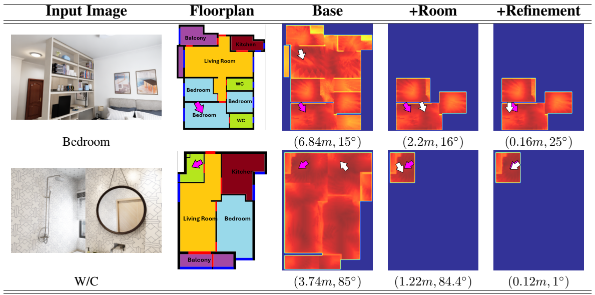

Figure 6. Qualitative comparison of using room predictions and our coarse-to-fine refinement approach. Below each map we report the localization error in meters and degrees. Warmer colors correspond to regions with higher probabilities. Overlaid on the estimated probabilities, we show the ground truth location (magenta) and the predicted location.

In Figure 6, we illustrate the impact of incorporating room type predictions alongside the location extraction module. The figure clearly demonstrates how these components refine the probability volume by narrowing down the candidate poses, which in turn improves overall localization accuracy.

We analyze the impact of different combinations of predicted depth and semantic features on the S3D dataset. Figure 7 summarizes the recall performance for various depth and semantic weight configurations used to compute the *structural-semantic probability volume*. In Figure 8 we visualize examples from two extreme scenarios (depth only and semantic only) and from the configuration adopted in our work ($[w_s, w_d] = [0.4, 0.6]$), selected according to the validation set.

As can be observed from Figure 8 (and also reflected in Figure 7), relying solely on semantic rays tends to produce a more diffused probability volume with multiple ambiguous candidate locations. This occurs because most images contain a single semantic object of a standard size, allowing an accurate ray pattern to be fitted in more than one location. Similarly, depth cues alone also suffer from ambiguity, particularly in repetitive environments or when the image captures three or fewer walls, which leads to uncertainty in the localization estimate. However, when these semantic cues are com-

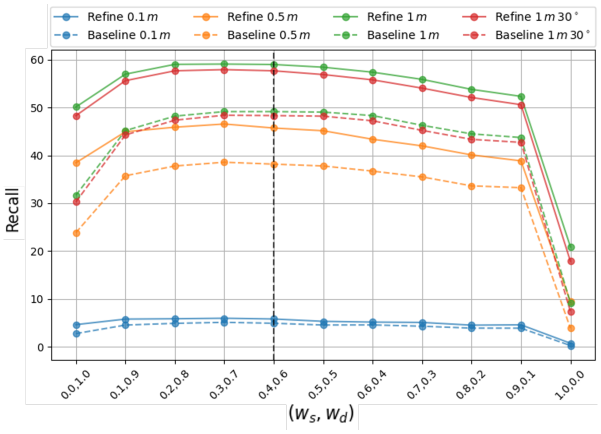

Figure 7. Recall vs. weight combinations on the S3D test set. The plot shows the recall metrics for four different thresholds: $0.1\,m$, $0.5\,m$, $1\,m$, and $1\,m\,30°$. The x-axis displays the weight combinations in the order (w_s, w_d). A vertical dashed line at $w_s = 0.4$, $w_d = 0.6$ highlights the weight combination selected using the validation set.

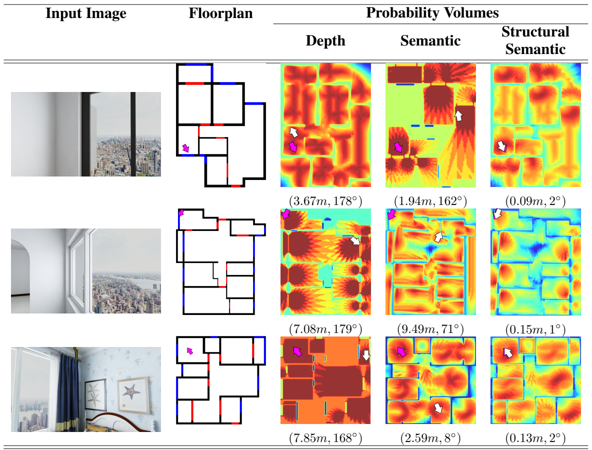

Figure 8. Qualitative Comparison of Depth-Only, Semantic-Only, and Fused Structural-Semantic Probability Volumes. Below each map we report the localization error in meters and degrees. Warmer colors correspond to regions with higher probabilities.

bined with depth rays, the resulting probability volume becomes significantly more concentrated. This integration effectively filters out spurious candidates and sharpens the localization estimate.

Additionally, the supplementary material offers a detailed analysis of how varying the Top-K candidates influences localization refinement, along with additional experiments that motivate various design choices, such as using hard thresholds in multiple steps of our pipeline.

K	**Prediction**	**Loc**	**Refin**	**Total**
1	0.038 ± 0.119	0.174 ± 0.033	0.141 ± 0.073	0.364 ± 0.142
3	0.033 ± 0.093	0.154 ± 0.026	0.356 ± 0.119	0.554 ± 0.157
5	0.034 ± 0.103	0.155 ± 0.029	0.577 ± 0.185	0.778 ± 0.218

Table 3. Performance breakdown over different Top-K values. Each entry is mean ± std (s). *Prediction* denotes the ray predictions, *Loc* refers to the localization process, and *Refin* represents the refinement stage, during which candidate locations are evaluated by computing the ground truth rays.

Runtime Analysis

We report runtime performance breakdown in Table 3, using the same parameters employed in Table 1 with varying K values. As shown in the table, the per-image inference time increases with the number of candidate refinements, K, reflecting the additional computations required during refinement. All experiments were conducted on a single CPU without multithreading to avoid introducing bias. Future work could explore parallelizing the refinement stage by computing all ground-truth rays simultaneously for further speed-up. Importantly, even at $K = 5$—the setting we adopt in our work, the inference time remains reasonable, striking a practical balance between computational cost and improvements in localization accuracy. We further observe that the prediction and localization steps require similar amounts of time, while the refinement step grows monotonically as K increases.

5. Conclusion

In this work, we presented a semantic-aware localization framework that extends floorplan-based camera localization by fusing semantic labels with geometric depth cues. Our approach leverages a novel semantic ray prediction network alongside an established depth estimation method to generate a semantic-structural probability volume, which significantly improves localization accuracy, especially in environments with repetitive or ambiguous structural patterns.

Our extensive experiments on the S3D and ZInD datasets demonstrate that integrating semantic cues effectively resolves depth-based ambiguities and consistently outperforms state-of-the-art methods such as F3Loc and LASER. Ablation studies confirm that a balanced combination of depth and semantic information, coupled with a coarse-to-fine localization strategy and the use of room labels, yields optimal performance.

Looking forward, extending our framework to incorporate additional semantic labels and other modalities, such as textual information, promises to further enhance localization robustness in challenging indoor settings. In general, our approach represents an important step towards accurate and reliable indoor localization systems by effectively leveraging semantic and geometric cues.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *CVPR*, pages 5297–5307, 2016. 2

[2] Vassileios Balntas, Shuda Li, and Victor Prisacariu. Relocnet: Continuous metric learning relocalisation using neural nets. In *ECCV*, pages 751–767, 2018. 2

[3] Federico Boniardi, Tim Caselitz, Rainer Kummerle, and Wolfram Burgard. Robust lidar-based localization in architectural floor plans. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3318–3324. IEEE, 2017. 2

[4] Federico Boniardi, Tim Caselitz, Rainer Kummerle, and Wolfram Burgard. A pose graph-based localization system for long-term navigation in cad floor plans. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 84–97. IEEE, 2019. 2

[5] Federico Boniardi, Abhinav Valada, Rohit Mohan, Tim Caselitz, and Wolfram Burgard. Robot localization in floor plans using a room layout edge extraction network. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5291–5297. IEEE, 2019. 2

[6] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. DSAC–differentiable RANSAC for camera localization. In *CVPR*, pages 6684–6692, 2017. 2

[7] Changan Chen, Rui Wang, Christoph Vogel, and Marc Pollefeys. F3Loc: Fusion and filtering for floorplan localization. *arXiv preprint arXiv:2403.03370*, 2024. 1, 2, 3, 6

[8] Hang Chu, Dong Ki Kim, and Tsuhan Chen. You are here: Mimicking the human thinking process in reading floorplans. In *ICCV*, pages 2210–2218, 2015. 1, 2

[9] Steve Cruz, Will Hutchcroft, Yuguang Li, Naji Khosravan, Ivaylo Boyadzhiev, and Sing Bing Kang. Zillow indoor dataset: Annotated floor plans with 360° panoramas and 3d room layouts. In *CVPR*, pages 2133–2143, 2021. 6

[10] Niklas Gard, Anna Hilsmann, and Peter Eisert. Spvloc: Semantic panoramic viewport matching for 6d camera localization in unseen environments. *arXiv preprint arXiv:2404.10527*, 2024. 3

[11] B. Glocker, S. Izadi, J. Shotton, and A. Criminisi. Real-time RGB-D camera relocalization. In *Proceedings of the Mixed and Augmented Reality (ISMAR) Conference*. IEEE, 2013. 1

[12] Henry Howard-Jenkins and Victor Adrian Prisacariu. LaLaLoc++: Global floor plan comprehension for layout localisation in unvisited environments. In *ECCV*, pages 693–709, 2022. 1, 2

[13] Henry Howard-Jenkins, Jose-Raul Ruiz-Sarmiento, and Victor Adrian Prisacariu. LaLaLoc: Latent layout localisation in dynamic, unvisited environments. arXiv preprint arXiv:2104.09169, 2021. 1, 2

[14] Seigo Ito, Felix Endres, Markus Kuderer, Gian Diego Tipaldi, Cyrill Stachniss, and Wolfram Burgard. W-RGBD: Floor-plan-based indoor global localization using a depth camera and wifi. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 417–422. IEEE, 2014. 2

[15] Peter Karkus, David Hsu, and Wee Sun Lee. Particle filter networks with application to visual localization. In *Proceedings of the 2nd Conference on Robot Learning (CoRL)*, pages 169–178. PMLR, 2018. 1, 3

[16] Alex Kendall, Matthew Grimes, and Roberto Cipolla. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *ICCV*, pages 2938–2946, 2015. 1, 2

[17] Junho Kim, Jiwon Jeong, and Young Min Kim. Fully geometric panoramic localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 3

[18] Z. Li, M. H. Ang, and D. Rus. Online localization with imprecise floor space maps using stochastic gradient descent. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8571–8578. IEEE, 2020. 2

[19] Liu Liu, Hongdong Li, and Yuchao Dai. Efficient global 2D-3D matching for camera localization in a large-scale 3D map. In *ICCV*, pages 2372–2381, 2017. 1, 2

[20] Pranay Mathur, Rajesh Kumar, and Sarthak Upadhyay. Sparse image-based navigation architecture to mitigate the need of precise localization in mobile robots. *arXiv preprint arXiv:2203.15272*, 2022. 1

[21] Oscar Mendez, Simon Hadfield, Nicolas Pugeault, and Richard Bowden. SeDAR: Reading floorplans like a human—using deep learning to enable human-inspired localisation. *IJCV*, 128:1286–1310, 2020. 3

[22] Zhixiang Min, Naji Khosravan, Zachary Bessinger, Manjunath Narayana, Sing Bing Kang, Enrique Dunn, and Ivaylo Boyadzhiev. LASER: Latent space rendering for 2D visual localization. In *CVPR*, pages 11122–11131, 2022. 3, 6

[23] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the Mixed and Augmented Reality (ISMAR) Conference*. IEEE, 2011. 1

[24] Takahiro Niwa, Shun Taguchi, and Noriaki Hirose. Spatio-temporal graph localization networks for image-based navigation. *arXiv preprint arXiv:2204.13237*, 2022. 1

[25] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: robust hierarchical localization at large scale. In *CVPR*, pages 12716–12725, 2019. 1, 2

[26] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Fast image-based localization using direct 2d-to-3d matching. In *ICCV*, pages 667–674, 2011.

[27] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Improving image-based localization by active correspondence search. In *ECCV*, pages 752–765, 2012.

[28] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE TPAMI*, 39(9):1744–1756, 2016. 1, 2

[29] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, pages 2930–2937, 2013. 1, 2

[30] Janine Thoma, Danda Pani Paudel, Ajad Chhatkuli, Thomas Probst, and Luc Van Gool. Mapping, localization and path planning for image-based navigation using visual features and map. In *CVPR*. IEEE, 2019. 1

[31] Julien Valentin, Matthias Nießner, Jamie Shotton, Andrew Fitzgibbon, Shahram Izadi, and Philip HS Torr. Exploiting uncertainty in regression forests for accurate camera relocalization. In *CVPR*, pages 4400–4408, 2015. 1, 2

[32] Florian Walch, Caner Hazirbas, Laura Leal-Taixé, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. Image-based localization using lstms for structured feature correlation. In *ICCV*, pages 627–637, 2017. 1, 2

[33] Shenlong Wang, Sanja Fidler, and Raquel Urtasun. Lost shopping! monocular localization in large indoor spaces. In *ICCV*, pages 2695–2703, 2015. 2

[34] Xipeng Wang, Ryan J Marcotte, and Edwin Olson. GLFP: Global localization from a floor plan. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1627–1632. IEEE, 2019. 2

[35] Wera Winterhalter, Freya Fleckenstein, Bastian Steder, Luciano Spinello, and Wolfram Burgard. Accurate indoor localization for RGB-D smartphones and tablets given 2d floor plans. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3138–3143. IEEE, 2015. 1

[36] Jian Wu, Liwei Ma, and Xiaolin Hu. Delving deeper into convolutional neural networks for camera relocalization. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5644–5651. IEEE, 2017. 1, 2

[37] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *ECCV*, 2020. 6