Self-training Room Layout Estimation via Geometry-aware Ray-casting

Bolivar Solarte[1,2], Chin-Hsuan Wu[1], Jin-Cheng Jhang[1], Jonathan Lee[1], Yi-Hsuan Tsai[3], and Min Sun[1]

[1] National Tsing Hua University, Taiwan
[2] Industrial Technology Research Institute ITRI, Taiwan
[3] Google

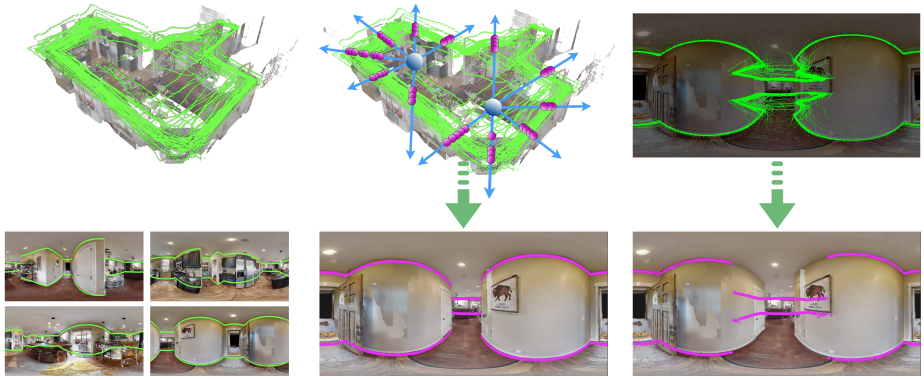
Abstract. In this paper, we introduce a novel geometry-aware self-training framework for room layout estimation models on unseen scenes with unlabeled data. Our approach utilizes a ray-casting formulation to aggregate multiple estimates from different viewing positions, enabling the computation of reliable pseudo-labels for self-training. In particular, our ray-casting approach enforces multi-view consistency along all ray directions and prioritizes spatial proximity to the camera view for geometry reasoning. As a result, our geometry-aware pseudo-labels effectively handle complex room geometries and occluded walls without relying on assumptions such as Manhattan World or planar room walls. Evaluation on publicly available datasets, including synthetic and real-world scenarios, demonstrates significant improvements in current state-of-the-art layout models without using any human annotation.

Keywords: Self-training · Room Layout Estimation · Multi-view Layout Consistency

1 Introduction

While significant progress has been made in room layout estimation, current state-of-the-art solutions predominantly rely on supervised frameworks, utilizing either monocular panoramic images [9, 21, 22, 27] or direct geometry sensors like depth cameras or LiDAR [2, 23]. However, this reliance presents a significant challenge for real-world applications due to variations in geometry complexity and scene conditions, thereby making data collection and manual labeling particularly cumbersome.

A practical solution for self-training a geometry-based model in unseen environments is by exploiting the multi-view consistency from multiple noisy estimations [7, 12]. However, applying multi-view consistency for room layout estimation has been poorly explored in the literature. For instance, recent approaches in multi-view layout estimation [8, 13, 19] particularly rely on ground truth annotations to define important concepts such as wall occlusion and wall match correspondences. Other solutions avoid partial dependency on label annotation by leveraging a semi-supervised approach [25]. To the best of our knowledge,



(a) Multi-view estimations (b) Ours Pseudo-labels (c) 360-MLC [18]

Fig. 1: By leveraging multiple estimates from a pre-trained model as presented in panel (a), Our solution leverages a ray-casting data aggregation process to estimate geometry-aware pseudo-labels for self-training, as depicted in panel (b), i.e., pseudo-labels that encompass a comprehensive representation of the room geometry. In comparison with previous solutions, as presented in (c), where multiple estimations are processed on the image domain without geometry reasoning, our approach excels in defining better pseudo-labels, especially for occluded geometries, highlighting the significance of our contribution.

only the recent self-training approach, 360-MLC [18], is capable of exploiting multi-view layout consistency (MLC) without human label annotations. Nevertheless, 360-MLC lacks any geometry reasoning and treats all layout estimates from every view equally, leading to noisy pseudo labels, especially for occluded regions. See Fig. 1-(c).

In this paper, we present a self-training framework for room layout estimation that leverages a pre-trained model to compute geometry-aware pseudo-labels for unseen environments. Our approach utilizes a ray-casting formulation to aggregate multiple noisy estimations along several ray directions for geometry reasoning. Our hypothesis is based on the idea that sampling layout estimates along a ray can locally approximate the probability distribution of the underlying geometry by considering their proximity to the camera view and mutual consistency between views. This simple yet effective approach yields remarkable room geometry definitions, including shapes with circular and non-planar walls, as well as effectively handling occluded geometries. See Fig. 1-(b).

To further exploit our proposed solution, we present a Weighted Distance Loss formulation that prioritizes the farthest geometry in the scene during self-training. This stems from the intuition that estimating distant geometries is typically challenging from a single view, suggesting that a multi-view setting may help overcome this issue by considering several complementary views along the scene.

To validate our proposed solution, we collect and label a new dataset (referred to as HM3D-MVL) from HM3D [15], particularly addressing occluded, complex, and ample room geometries. We validate the benefits of the proposed

self-training solution through an extensive evaluation in different settings and publicly available datasets [4,17], using synthetic and real-world data. Our contributions are as follows:

1. We propose a novel geometry-aware ray-casting formulation for pseudo-labeling unseen scenes directly from the multiple noisy estimations of a pre-trained model.
2. We propose a Weighted Distance Loss that exploits the benefits of a multi-view setting by prioritizing distant geometry during self-training.
3. We collect and label a new dataset (HM3D-MVL) from [15], particularly addressing occluded, complex, and ample room geometry for more diverse scenarios. The dataset and code will be released with this publication.

2 Related Work

Room Layout Estimation. Estimating the room layout geometry is a long-standing problem, where earlier works [3, 26, 28] mainly rely on key features, semantic cues, and prior geometries to reason about the underlying geometry. While deep learning solutions for this task have brought robustness in the estimation by leveraging supervision from labeled data [6, 10, 29, 31], most of these solutions define the problem as a regression map task. An outstanding solution that changes this paradigm is HorizonNet [21], which redefines the optimization as an 1D boundary regression problem, simplifying the definition for the layout geometry. Upon this solution, approaches like [22] have impressive results by leveraging a simple layout definition. Another advance is LED2Net [27] and LGTNet [9], which introduces a horizon-depth vector definition, constraining the layout geometry directly on Euclidean space. Upon this solution, recent approaches [5,30] present further constrains during training, none of them targeting multi-view consistency.

Multi-view Layout. Recent approaches in multi-view setting [8, 13, 19] define the multi-view layout estimation problem jointly with camera pose registration. In particular, [8] introduces important concepts for geometry reasoning, such as layout occlusion and layout match correspondences strictly relying on ground truth annotations. An outstanding solution in this manner is Graph-Covis [13], which is built upon [8] to define a multi-view setting capable of estimating layout and camera pose from multi-views using a graph neural network approach. Nevertheless, these solutions rely on ground truth annotations for reasoning the underlying geometry.

Semi-Supervised and Self-training Layout Estimation. Semi-supervision and self-training methods aim to define a reliable reference to constrain the learning optimization without ground truth annotations [11]. Along this line, SSLayout360 [25] utilizes a Mean Teacher framework [24] to train a layout estimation model using pseudo-labels from a exponential-moving-average operation. However, [25] treats each image in isolation, neglecting valuable geometric information from alternate camera views. Furthermore, the challenge arises from the inherent noise in pseudo labels. Existing approaches aim to mitigate this

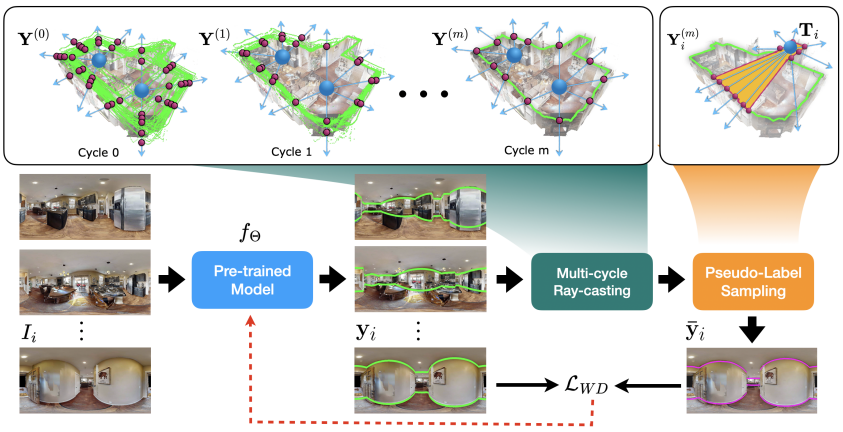


Fig. 2: Self-training Pipeline. We use a pre-trained model f_Θ to estimate multiple layouts $\mathbf{y}_i$ from multiple views I_i in an unseen scene. We aggregate all noisy estimates $\mathbf{Y}^{(0)} = \text{concat}(\{\mathbf{y}_i\}_{i:n})$ using our proposed Multi-cycle ray-casting process. Then, we sample our pseudo-label $\bar{\mathbf{y}}_i$ at the camera position $\mathbf{T}_i$ from the filtered set of layouts $\mathbf{Y}_i^{(m)}$. Finally, we constraint our self-training optimization using our proposed Weighted-distance loss $\mathcal{L}_{WD}$.

noise through techniques such as assembling predictions across diverse augmentations [1,14] or by selectively retaining only those pseudo-labels with high confidence [16].

On the other hand, a practical solution for self-training models is to leverage information from a pre-trained model. In 360-MLC [18], multiple estimations of a pre-trained model [21] are re-projected into a camera view from which pseudo labels are sampled. However, this formulation does not consider any geometry prior and treats every geometry estimation equally, which yields noisy labels, particularly for occluded geometry. To the best of our knowledge, a self-training formulation that handles geometry in a multi-view setting without relying on label annotation has not been studied.

3 Proposed Method

The following outlines our proposed self-training framework for room layout estimation. In Sec. 3.1, we describe the multi-view layout consistency problem (MLC) as well as the preliminaries for self-training room layout models. In Sec. 3.2, we present our ray-casting data aggregation process to create geometry-aware pseudo-labels solely from estimated data. Lastly, in Sec. 3.3, we present our weighted loss formulation towards leveraging the farthest distant geometry in a scene. For illustration purposes, an overview of our self-training framework is depicted in Fig. 2.

3.1 Self-training Room Layout with Multi-view Layout Consistency

In general, self-training a room layout model by multi-view layout consistency (MLC) aims to fine-tune a pre-trained model with reliable pseudo-labels com-

puted from multiple estimations along an unseen scene [18]. This scene with n views can be defined as follows:

$$\mathcal{S} = \{(I_i, \mathbf{T}_i)\}_{i=1:n} \ , \quad I_i \in \mathbb{R}^{H \times W} \ , \ \mathbf{T}_i \in SE(3) \ , \tag{1}$$

where $\mathcal{S}$ is the set of inputs views, I_i represents a panoramic image of size $H \times W$ pixels, and $\mathbf{T}_i$ is the corresponding camera pose with rotation $\mathbf{R}_i \in SO(3)$ and translation $\mathbf{t}_i \in \mathbb{R}^3$ defined in world coordinates. For any view in the set $\mathcal{S}$, we can define an estimated layout geometry as follows:

$$\mathbf{y}_i = \pi(f_\Theta(I_i), \mathbf{T}_i), \quad \mathbf{y}_i \in \mathbb{R}^{3 \times W}, \tag{2}$$

where f_Θ is a layout model parameterized by Θ, $\pi(\cdot)$ is a projection function that transforms the model's prediction into the Euclidean space, and $\mathbf{y}_i$ is the estimated layout geometry registered in world coordinates. For simplicity, we refer to $\mathbf{y}_i$ as the floor boundary only. For layout models such as [21, 22], $\pi(\cdot)$ processes a 1D boundary vector defined in spherical coordinates, while models [9, 27] handle a 1D horizon-depth estimation. A closed-form definition for both is described in our supplementary material.

By estimating multiple layouts from every view in the scene, we can define the pseudo labeling process as follows:

$$\begin{aligned} &\mathbf{Y} = \text{concat}(\{\mathbf{y}_0, \cdots, \mathbf{y}_n\}) \ , \quad \mathbf{Y} \in \mathbb{R}^{3 \times nW} \ , \\ &\mathbf{Y}_i = \mathbf{R}_i \mathbf{Y} + \mathbf{t}_i, \quad \bar{\mathbf{y}}_i = \Phi(\mathbf{Y}_i), \quad \bar{\mathbf{y}}_i \in \mathbb{R}^{3 \times W}, \end{aligned} \tag{3}$$

where $\mathbf{Y}$ is the concatenation of n layout geometries estimated by Eq. (2), $\mathbf{Y}_i$ stands as the rigid transformation of $\mathbf{Y}$ into the $i-$th camera reference, and $\Phi(\cdot)$ is the aggregating function that estimates a pseudo-label $\bar{\mathbf{y}}_i$ for the $i-$th view in the scene.

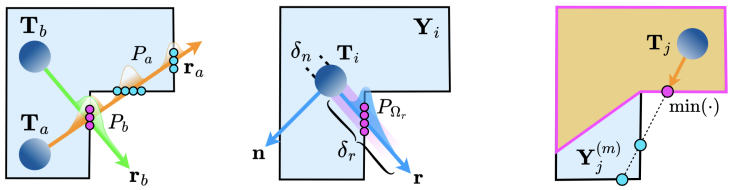
Note that, in the case of 360-MLC [18], $\Phi(\cdot)$ is the function that samples the median values of re-projected points in the image domain without any geometry reasoning, see Fig. 1-(c). In Sec. 3.2, we redefine $\Phi(\cdot)$ as a ray-casting function for computing geometry-aware pseudo-labels.

The self-training optimization of f_Θ with multiple pseudo-labels $\bar{\mathbf{y}}_i$ can be defined as follows:

$$\min_{\Theta} \frac{1}{n} \sum_{i=1}^{n} \omega_i \cdot \mathcal{L} \left(\ f_\Theta(I_i), \ \pi^{-1}(\bar{\mathbf{y}}_i) \right), \tag{4}$$

where $\pi^{-1}(\cdot)$ is the inverse function presented in Eq. (2), $\omega_i \in \mathbb{R}^W$ is a weighted vector associated to the uncertainty in each pseudo-label $\bar{\mathbf{y}}_i$, and $\mathcal{L}(\cdot)$ is the loss function that constraints the self-training optimization.

Note that, in the case of 360-MLC [18], The self-training constraint is defined as a weighted L1 loss with $\omega_i = \sigma_i^{-2}$, where σ_i is the standard deviation of re-protected points in the image domain. In Sec. 3.3, we redefine ω_i into our weighted-distance function that prioritizes distance geometries from the camera view during self-training.



(a) (b) (c)

Fig. 3: Ray-Casting: In panel (a), different ray directions from different camera views are shown. Note that due to occluded geometries and different camera positions, the probability distribution along a ray may vary significantly. In panel (b), one of our constraints to handle occluded geometries is depicted, i.e., sampling a nearby region along the ray to define P_{Ω_r}. In Panel (c), we sample a pseudo-label (magnet contour) from a filtered layout boundary $\mathbf{Y}_j^{(m)}$ at the camera $\mathbf{T}_j$ by using $\min(\cdot)$ function to sample the non-occluded points on the rays (see Sec. 3.2).

3.2 Pseudo-labeling by Ray-casting

Probability distribution on a ray. We hypothesize that the projection of multiple layout estimates onto a ray can describe a probability distribution of the underlying geometry. This distribution can then serve as the basis for sampling reliable pseudo-labels. To this end, we propose a ray-casting formulation that projects multiple estimates of a pre-trained model into a set of ray directions defined in the bird-eye-view (BEV), i.e., ray vectors defined in the xz Euclidean plane. This is motivated by previous works [9, 27] to represent a room layout geometry directly in the Euclidean space, avoiding distortion and discrete issues presented in the image domain.

We define a set of ray directions in world coordinates as follows:

$$\begin{aligned} \mathcal{R} &= \{\mathbf{r}_j\}_{j=1:W} \;, \quad \mathbf{r}_j \in \mathbb{R}^3 \;, \quad |\mathbf{r}_j| = 1, \\ \mathcal{V} &= \{\mathbf{n}_j\}_{j=1:W} \;, \quad \mathbf{n}_j \in \mathbb{R}^3 \;, \quad \mathbf{n}_j \cdot \mathbf{r}_j^\top = \mathbf{0}, \end{aligned} \tag{5}$$

where $\mathbf{r}_j$ is a ray direction constrained by $\mathbf{r}_j \cdot [0, 1, 0]^\top = \mathbf{0}$ (i.e., on the xz Euclidean plane), and $\mathbf{n}_j$ is its corresponding normal vector. Then, a pseudo-label from a probability function defined on a ray vector can be defined as follows:

$$\bar{\mathbf{y}}_{i,r} = \mathbb{E}[P_r(\mathbf{Y}_i)]\mathbf{r}, \quad \mathbf{r} \in \mathcal{R}, \tag{6}$$

where $\mathbf{r}$ is a ray vector introduced by Eq. (5), $\mathbf{Y}_i$ is the concatenation of all estimated layouts in the $i-$th camera reference as presented in Eq. (3), $\bar{\mathbf{y}}_{i,r}$ stands for the $i-$th pseudo label defined on the ray $\mathbf{r}$, and $P_r(\cdot)$ is the unknown probability function along a ray direction $\mathbf{r}$. For simplicity, we refer to this probability function as P_r.

Regardless of the noise within the estimated layout geometries, the density function P_r may vary significantly for every camera view and ray direction, in particular for occluded geometry. This phenomenon is illustrated in Fig. 3-(a), where two density functions P_a and P_b for the same underlying geometry (magenta dots) are presented. Note that P_a defines a multi-modal density function

due to multiple occluded geometries (cyan dots), which may lead to a different expectation value compared to P_b.

Multi-cycle ray-casting for pseudo-labeling. To tackle occlusions, we condition P_r, presented by Eq. (6), in three ways. First, we increase the sample count near each ray direction and camera view based on the intuition that a higher sample count may enhance the representation of non-occluded geometries. Second, similar to 360-MLC [18], we approximate the expectation of projected samples to $\texttt{median}(\cdot)$ for filtering out noisy estimates, i.e., the median value of points on the ray. However, instead of sampling from a unique view (in the image domain), we sample them from multiple camera locations and ray directions in an iterative process named multi-cycle ray-casting (see Fig. 2). This stems from the fact that sampling over P_r from multiple camera locations and directions must yield the same underlying room geometry. Finally, following the noise reduction, we approximate the expectation of P_r to the closest sample on the ray. This is based on the understanding that non-occluded geometries must lie at the closest point along the ray direction. This is illustrated in Fig. 3-(c), where the pseudo-label for the camera view $\mathbf{T}_j$ (magenta contour) is computed by sampling points on the rays by using the $\texttt{min}(\cdot)$ function.

With a slight notation abuse, the projection of nearby estimates onto a ray direction can be defined as follows:

$$\begin{aligned} &\Omega_r(\mathbf{Y}_i) = \{\mathbf{r}\cdot\mathbf{x}^\top \mid \forall\, \mathbf{x}\in\mathbf{Y}_i\} \quad st. \\ &0 < \mathbf{r}\cdot\mathbf{x}^\top \le \delta_r, \quad \text{and} \quad |\mathbf{n}\cdot\mathbf{x}^\top| \le \delta_n, \end{aligned} \tag{7}$$

where $\mathbf{x}$ is a 3D-point $\in \mathbb{R}^3$ defined in $\mathbf{Y}_i$, $\mathbf{r}$ and $\mathbf{n}$ are ray-vectors define by Eq. (5), and $\{\delta_r, \delta_n\}$ is a set of hyper-parameters that allows us to filter out non-local points. This projection is illustrated in Fig. 3-(b), where the subset of points Ω_r (magenta dots) is defined along the ray vector $\mathbf{r}$. For simplicity, we refer to the probability of these projected samples as P_{Ω_r}.

The multi-cycle ray-casting process to filter out noisy estimates can be described as follows:

$$\mathbf{Y}^{(k+1)} = \{\texttt{median}(\Omega_{r_j}(\mathbf{Y}_i^{(k)}))\mathbf{r}_j\}_{i=1:n \quad j=1:W}\ , \tag{8}$$

where $\mathbf{Y}_i^{(k)}$ stands for the layout estimates in the i-th camera reference at the k-th cycle. Note that this filtering process is evaluated from all camera views i and all ray directions $\mathbf{r}_j$.

Finally, a pseudo label and its uncertainty from a filtered set of layout estimations can be evaluated as follows:

$$\begin{aligned} \bar{\mathbf{y}}_i &= \{\texttt{min}(\Omega_{r_j}(\mathbf{Y}_i^{(m)}))\mathbf{r}_j\}_{j=1:W}, \\ \sigma_i &= \{\text{std}(\Omega_{r_j}(\mathbf{Y}_i^{(0)}))\}_{j=1:W}, \end{aligned} \tag{9}$$

where $\mathbf{Y}_i^{(m)}$ stands for the filtered layout estimates after applying Eq. (8) in $m-$th cycles, and $\mathbf{Y}_i^{(0)}$ is the layout estimates before noise reduction. This is because σ_i aims to describe the underlying noise of the initial layout estimates along the ray directions.

(a) (b) (c)

Fig. 4: Weighted-distance function: In panel (a), we illustrate our proposed weighted-distance function ω_i that prioritizes the farthest geometries in the scene for self-training. In panel (b), under the same scale as (a), we show the $L1$ loss between our proposed pseudo-label and the model estimation. Note that the $L1$ loss evaluation presents a small range w.r.t ω_i and does not aim at any particular region in the scene. In Panel (c), we present our pseudo-label (magenta line) and the model estimation (green line).

3.3 Weighted Distance Loss

To complement our proposed ray-casting pseudo-labels resented in Sec. 3.2, we introduce a weighted loss formulation that particularly focuses on the farthest geometries within a room. This stems from the empirical evidence that pre-trained layout models tend to estimate more accurately the geometries closer to the camera view than those farther away. This limitation can be attributed, in part, to the datasets used for training, e.g., [2, 4], where room scenes are predominantly captured from the room center, and larger-sized rooms are less represented. Another contributing factor to this limitation is the difficulty in capturing accurate details for the farthest regions from a single view [9]. Therefore, we hypothesize that our pseudo-labels may present the most significant impact during self-training when targeting the farthest geometries in a scene.

Our weighted formulation can be described as follows:

$$\mathcal{L}_{WD} = \omega_i ||\mathbf{y}_i - \bar{\mathbf{y}}_i||_1 \quad \omega_i = \frac{e^{\kappa(||\bar{\mathbf{y}}_i|| - d_{min})}}{\sigma_i^2} \tag{10}$$

where $||\bar{\mathbf{y}}_i||$ is the Euclidean norm of the pseudo labels computed by Eq. (9), d_{min} is the distance from which we want to prioritize the self-training, κ is a hyper-parameter that allows us to control the weighting priority to the farthest geometries, and σ_i represent the standard deviation computed in Eq. (9). In Fig. 4, we compare our proposed weighted-distance function with traditional L_1 loss [18, 21, 22, 27]. Note that a L_1 evaluation does not aim at any particular geometry in the scene, while our proposed ω_i aims at the farthest walls from the camera view.

4 Experiments

4.1 Experimental Setup

Baseline and Model Backbones. The baseline used in the following experiments is the recent 360-MLC [18] taken from the official implementation provided by the authors. For a fair comparison with 360-MLC, we use the same layout model backbone by default, i.e., HorizonNet [21] pre-trained in [2]. To further compare our proposed solution, we present results using LGTNet [9] pre-trained on [2] as an additional layout model backbone.

Table 1: Datasets used in this paper with their statistics, i.e., total frames and average number of frames per room.

Dataset	Training set	Testing set	Occlusion Subset	Avg. frames per room
HM3D-MVL	24344	2491	119	56
MP3D-FPE [17]	20126	5254	157	46
ZInD [4]	9514	1157	191	6

Table 2: Quantitative results using the HorizonNet [21] backbone. The symbol ‡ represents that the model is trained with the available labels in the training set, which represents the upper-bound performance.

	Testing set				Occlusion Subset			
	2D IoU (%) ↑		3D IoU (%) ↑		2D IoU (%) ↑		3D IoU (%) ↑	
Method	10%	100%	10%	100%	10%	100%	10%	100%
Our HM3D-MVL dataset								
Pre-trained [21]	76.71		71.79		78.74		75.72	
360-MLC [18]	81.69	82.71	77.67	78.71	81.66	79.19	80.08	77.72
Ours	**81.74**	**82.99**	**77.99**	**78.95**	**82.05**	**83.01**	**80.45**	**81.38**
MP3D-FPE dataset [17]								
Pre-trained	77.33		74.07		75.09		73.36	
360-MLC	80.84	80.93	77.71	77.69	84.15	84.27	82.27	82.04
Ours	**81.25**	**81.65**	**78.15**	**78.21**	**85.21**	**85.71**	**83.16**	**83.58**
ZInD dataset [4]								
Pre-trained	68.63		65.54		59.98		53.95	
360-MLC	74.09	75.44	71.21	72.28	62.04	63.33	59.29	60.47
Ours	**74.51**	**75.71**	**72.01**	**73.04**	**62.72**	**64.01**	**60.12**	**61.37**
Supervised‡ [21]	84.87		81.55		79.44		75.56	

Datasets. Similar to 360-MLC [18], we show evaluations in the MP3D-FPE dataset [17]. We also show results on the real-world ZInD dataset [4]. In addition, we show results in our newly collected dataset rendered from Habitat-v2 [15], referred to as HM3D-MVL. In the case of the ZInD dataset, we use the layout category "*visible layout*" provided by the authors and select the scenes that contain at least five frames per room. For all the mentioned datasets, we compute pseudo labels from the training splits, self-train the pre-trained model, and evaluate results on the testing splits using ground truth annotations provided by the authors. To further corroborate our claim of handling occluded geometries, we also present evaluations on a manually selected subset of the testing split that contains samples with geometry occlusions only. We refer to this subset as *Occlusion subset*. Details of these datasets are present in Tab. 1.

Evaluation Metrics. Following [9, 18, 21, 32], we evaluate results using standard metrics defined for room layout estimation. For room boundary prediction,

we evaluate the 2D and 3D intersection-over-union (IoU). For evaluating the smoothness and consistency of layout depth maps, we evaluate root-mean-square (RMS) and δ_1 errors as defined in [9, 21, 27]. All experiments show the median results of 10 self-training runs, each consisting of 15 training epochs.

Implementation Details. The layout models' backbones and their pre-trained weights used in our experiments are taken from their official implementation provided by the authors [9, 21]. To train the models, we use common data augmentation techniques for the room layout task, i.e., left-right flipping, panoramic rotation, and luminance augmentation. We use the Adam optimizer with a batch size of 4 and a learning rate 1×10^{-4} with a decay ratio of 90%. All models are trained on a single Nvidia RTX 2080Ti GPU with 12 GB of memory. For constructing our ray-casting pseudo-labels, we use 15 cycles per room scene, $\delta_r = 20$ and $\delta_n = 0.01$. For our weighted distance loss function, we use $\kappa = 0.5$ and $d_{min} = 2$.

4.2 Quantitative Results

Evaluation using HorizonNet Backbone. In these experiments, we compare our proposed ray-casting self-training frameworks with the baseline 360-MLC [18], utilizing the HorizonNet layout model [21] pre-trained in [2]. The results are presented in Tab. 2 under two main settings: using 10% and 100% of the training set. Results in the 10% setting show that our proposed solution outperforms 360-MLC, even with a limited number of samples for self-training. Results in the 100% setting further demonstrate the improved performance of our proposed self-training framework.

By comparing results in the occlusion subset, we find evidence that our solution significantly outperforms 360-MLC. Particularly, while our proposed ray-casting self-training consistently improves performance with increased data, 360-MLC shows only marginal improvement and in some settings, presents a decline in performance. For instance, consider the evaluation of the occlusion subset of the HM3D-MVL dataset. When using only 10% of the data, 360-MLC achieves 81.66% 2D IoU. However, the result on the 100% setting shows a drop in performance to 79.19%. This suggests that 360-MLC contains a large amount of noisy pseudo labels such that increasing the amount of data significantly hurts the performance. We argue that the general benefit of our ray-casting pseudo-labels is mainly due to their strong reasoning capability on occluded geometries. Additionally, we present a comparison against the fully-supervised HorizonNet [21] on ZInD [4] as an upper-bound references. Although our proposed ray-casting framework effectively self-train a pre-trained model into a new domain, we still found a gap when using manual labels, showing potential direction for future works.

Evaluation using LGTNet Backbone. In this experiment, we aim to validate the performance of our proposed solution compared to 360-MLC when utilizing a state-of-the-art solution for room layout estimation, i.e., LGTNet [9]. The

Table 3: Quantitative results using the LGTNet [9] backbone. The symbol ‡ represents that the model is trained with the available labels in the training set, which represents the upper-bound performance.

Method	Testing set				Occlusion Subset			
	2D IoU ↑	3D IoU ↑	RMS ↓	δ_1 ↑	2D IoU ↑	3D IoU ↑	RMS ↓	δ_1 ↑
Our HM3D-MVL dataset								
pre-trained [9]	78.90	74.04	0.409	0.864	80.22	78.10	0.2784	0.931
360-MLC [18]	84.07	78.85	0.394	0.897	71.29	68.54	0.573	0.884
Ours	**86.49**	**81.90**	**0.293**	**0.913**	**83.75**	**82.06**	**0.264**	**0.950**
MP3D-FPE Dataset [17]								
pre-trained	79.66	76.32	0.324	0.892	78.22	76.39	0.243	0.949
360-MLC	82.99	77.22	0.358	0.883	79.16	75.07	0.378	0.907
Ours	**85.69**	**81.80**	**0.242**	**0.931**	**86.33**	**84.27**	**0.168**	**0.963**
ZInD dataset [4]								
pre-trained	72.59	69.67	0.445	0.897	60.30	57.51	0.645	0.846
Ours	**76.77**	**74.42**	**0.406**	**0.905**	**64.76**	**62.38**	**0.593**	**0.857**
Supervised‡ [9]	87.64	84.61	0.286	0.931	80.51	77.87	0.393	0.873

(a) 360-MLC [18] (b) Ours (c) Point Cloud reference

Fig. 5: Qualitative comparisons of estimated pseudo-labels. We show a BEV projection of all pseudo-labels for the scene: (a) pseudo-labels from 360-MLC [18], (b) pseudo-labels from our proposed multi-cycle ray-casting, and (c) Point cloud for reference purposes.

results are depicted in Tab. 3. Although a robust backbone model benefits both models, our self-training framework significantly outperforms 360-MLC across all evaluations. Hence corroborating the versatility of our solution by leveraging new room layout formulations. Results of 360-MLC in the ZInD dataset were omitted due to several failures during self-training, we argue that this is due to the limitation of 360-MLC to handle a setting with a few number frames and horizon-depth constrain. Similar to the experiment presented in Tab. 2, We present upper-bound results that provide evidence of a gap between training on manual annotations and pseudo-labels, indicating a potential direction for future work.

Fig. 6: Qualitative results in real-world scenes. We show layout boundaries estimated in real-world data using a hand-handled camera (Insta360). In the first row, we illustrate all layouts estimated from a pre-trained model [21]. In the second row, we show the results of our ray-casting pseudo labeling process presented in Sec. 3.2.

4.3 Qualitative Results

Qualitative Results on Panoramic Images. For illustration purposes, we present in Fig. 7 several qualitative results of our proposed self-training framework compared with 360-MLC. Based on these results, we find that our solution shows a significant improvement in handling occluded geometries in all datasets. In addition, we observe that our self-training formulation consistently provides more accurate estimations of geometries near entrances and gates. We argue that this is due to the effectiveness of our ray-casting pseudo-labels in defining reliable room geometry, even for those challenging view locations.

Qualitative Pseudo-labels Results. In this section, we present qualitative results for our proposed ray-casting pseudo-labeling framework. These results are presented in Fig. 8 and Fig. 5, where the former presents pseudo-labels projected on panoramic images and the latter presents pseudo-labels projected in BEV. Based on the results in Fig. 8, we corroborate our hypothesis that our ray-casting pseudo-labels can handle occluded geometries better than 360-MLC. Furthermore, we find evidence that challenging views such as entrance and gates are better defined by our proposed pseudo-labels. This evidence aligns with our findings in Fig. 7, where results of a self-trained model using our proposed framework show better estimation for such challenging view locations. Furthermore, based on the results presented in Fig. 5, we can assert that our ray-casting pseudo-labels yield a less noisy geometry compared to 360-MLC, as well as it is capable of defining circular walls directly from multiple estimations.

Qualitative Results on Real-world Data. In Fig. 6, we present two qualitative results in two real-world scenes, demonstrating the versatility of our ray-casting pseudo-labeling in real-world scenarios. For these experiments, we collect

Table 4: Ablation study for our weighted-distance loss using 10% of data.

Loss	Testing set 2D IoU ↑	Testing set 3D IoU ↑	Occlusion Subset 2D Io ↑	Occlusion Subset 3D IoU ↑
(a) Pre-trained [21]	76.71	71.79	78.74	75.72
(b) Pseudo-labels	81.65	76.99	80.85	78.98
(c) $\omega{=}\sigma^{-2}$	81.02	76.58	81.28	79.53
(d) ω = Eq. (10)	**81.74**	**77.99**	**82.05**	**80.45**

several panoramic images using a commercial camera, Insta360[4], and estimate their camera poses using OpenVSLAM [20]. Subsequently, we register each image with its corresponding layout estimation (utilizing HorizonNet [21] pre-trained in [2]) by using the layout registration method outlined in [17]. In the first row, we present evidence of the domain gap in the pre-trained model showing a significant level of noise in the boundary layout estimations for both depicted scenes. In the second row, we present the results of our proposed ray-casting pseudo-labeling framework presented in Sec. 3.2. Note that our solution is capable of aggregating multiple noisy estimates to define a reliable underlying geometry for self-training remarkably.

4.4 Ablation Study for Weighted Distance Loss Formulation

We present an ablation study that validates our weighted distance loss formulation presented in Sec. 3.3. The results of this ablation are shown in Tab. 4. By comparing rows (a) and (b), we validate the gain in performance of self-training directly using our proposed ray-casting pseudo-labels without any weighting formulation. By comparing (c) and (b), we verify a weighted formulation based only on the uncertainty σ computed by Eq. (9). We can appreciate that this weighting formulation yields better performance on the occlusion subset but not for the whole testing set. We argue that a weighting formulation based on uncertainty σ does not consider any geometry information. In contrast, in row (d), we show the results of our weighted formulation as presented in Eq. (10). Thus we can assert that a weighting formulation that prioritizes the farthest geometries with respect to the camera view yields better performance.

5 Conclusions

In this paper, we present a geometry-aware self-training framework for multi-view room layout estimation that requires only unlabeled images as input. Our approach utilizes a ray-casting formulation capable of handling occluded geometries directly from noisy estimations. To further exploit the benefit of the multi-view setting, we propose a weighted distance loss function that focuses on the farthest geometries in the scene. Through a comprehensive evaluation using different datasets, room layout models, and settings, we demonstrate the state-of-the-art performance of our solution.

[4] https://www.insta360.com/

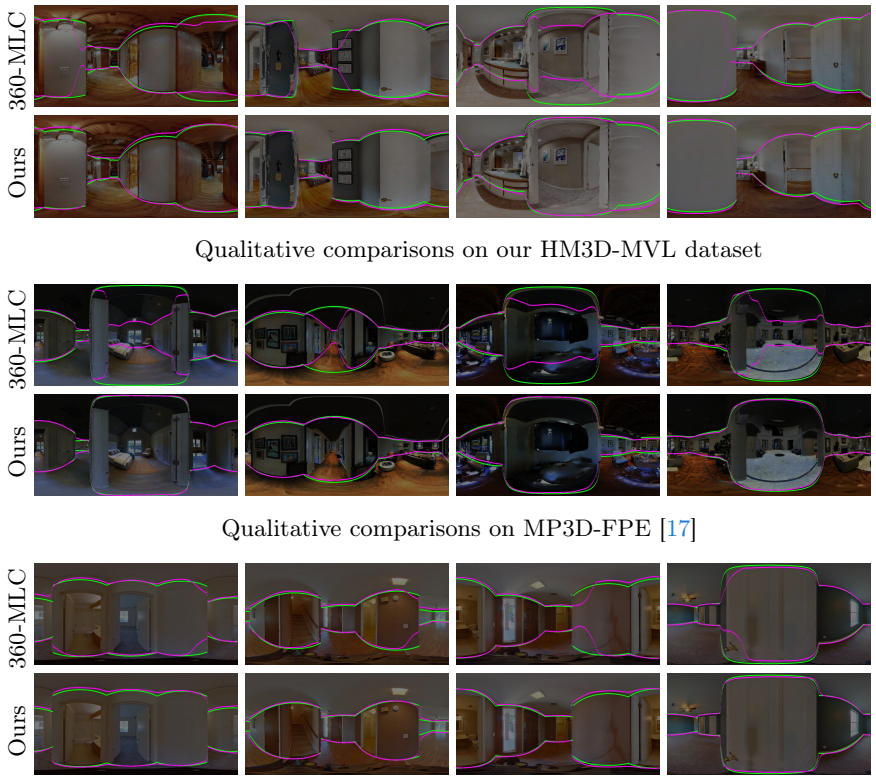


Fig. 7: Qualitative comparisons on panoramic images.. We present the results of room layout estimation after self-training using 360-MLC [18] and our proposed framework. Results are evaluated in three different datasets: 1) at the top on our proposed HM3D-MVL, 2) in the middle on MP3D-FPE [17], and 3) at the bottom on the real-world dataset ZInD [4]. The green lines represent the ground truth reference and the magenta lines represent the layout estimations.

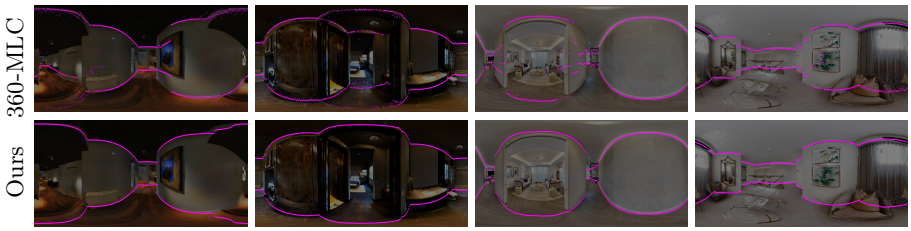


Fig. 8: Qualitative comparisons of pseudo labels on panoramic images. We present the qualitative results of estimated pseudo labels (magenta lines) on the panoramic images: 1) the first row, 360-MLC [18]; 2) the second row, our ray-casting pseudo labels.

Acknowledgements

This project is supported by The National Science and Technology Council NSTC and The Taiwan Computing Cloud TWCC under the project NSTC 112-2634-F-002-006.

References

1. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: Mixmatch: A holistic approach to semi-supervised learning. Advances in Neural Information Processing Systems **32** (2019) 4
2. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3D: Learning from RGB-D data in indoor environments. International Virtual Conference on 3D Vision (3DV) (2017) 1, 8, 10, 13
3. Chao, Y.W., Choi, W., Pantofaru, C., Savarese, S.: Layout estimation of highly cluttered indoor scenes using geometric and semantic cues. In: International Conference on Image Analysis and Processing. pp. 489–499. Springer (2013) 3
4. Cruz, S., Hutchcroft, W., Li, Y., Khosravan, N., Boyadzhiev, I., Kang, S.B.: Zillow indoor dataset: Annotated floor plans with 360deg panoramas and 3d room layouts. In: CVPR (2021) 3, 8, 9, 10, 11, 14
5. Fayyazsanavi, P., Wan, Z., Hutchcroft, W., Boyadzhiev, I., Li, Y., Kosecka, J., Kang, S.B.: U2rle: Uncertainty-guided 2-stage room layout estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3562–3570 (2023) 3
6. Fernandez-Labrador, C., Facil, J.M., Perez-Yus, A., Demonceaux, C., Civera, J., Guerrero, J.J.: Corners for layout: End-to-end layout recovery from 360 images. IEEE Robotics and Automation Letters **5**(2), 1255–1262 (2020) 3
7. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth prediction. In: The International Conference on Computer Vision (ICCV) (October 2019) 1
8. Hutchcroft, W., Li, Y., Boyadzhiev, I., Wan, Z., Wang, H., Kang, S.B.: Covispose: Co-visibility pose transformer for wide-baseline relative pose estimation in 360 indoor panoramas. In: European Conference on Computer Vision. pp. 615–633. Springer (2022) 1, 3
9. Jiang, Z., Xiang, Z., Xu, J., Zhao, M.: Lgt-net: Indoor panoramic room layout estimation with geometry-aware transformer network. In: CVPR (2022) 1, 3, 5, 6, 8, 9, 10, 11
10. Lee, C.Y., Badrinarayanan, V., Malisiewicz, T., Rabinovich, A.: Roomnet: End-to-end room layout estimation. In: Proceedings of the IEEE international conference on computer vision. pp. 4865–4874 (2017) 3
11. Lee, D.H., et al.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on challenges in representation learning, ICML. p. 896. Atlanta (2013) 3
12. Li, J., Dai, H., Han, H., Ding, Y.: Mseg3d: Multi-modal 3d semantic segmentation for autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 21694–21704 (June 2023) 1

13. Nejatishahidin, N., Hutchcroft, W., Narayana, M., Boyadzhiev, I., Li, Y., Khosravan, N., Košecká, J., Kang, S.B.: Graph-covis: Gnn-based multi-view panorama global pose estimation. In: CVPR Workshop on Omnidirectional Computer Vision (2023) 1, 3
14. Radosavovic, I., Dollár, P., Girshick, R., Gkioxari, G., He, K.: Data distillation: Towards omni-supervised learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4119–4128 (2018) 4
15. Ramakrishnan, S.K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J.M., Undersander, E., Galuba, W., Westbury, A., Chang, A.X., Savva, M., Zhao, Y., Batra, D.: Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In: NeurIPS, Datasets and Benchmarks Track (2021) 2, 3, 9
16. Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C.A., Cubuk, E.D., Kurakin, A., Li, C.L.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. Advances in Neural Information Processing Systems **33**, 596–608 (2020) 4
17. Solarte, B., Liu, Y.C., Wu, C.H., Tsai, Y.H., Sun, M.: 360-dfpe: Leveraging monocular 360-layouts for direct floor plan estimation. IEEE Robotics and Automation Letters **7**(3), 6503–6510 (2022) 3, 9, 11, 13, 14
18. Solarte, B., Wu, C.H., Liu, Y.C., Tsai, Y.H., Sun, M.: 360-mlc: Multi-view layout consistency for self-training and hyper-parameter tuning. In: NeurIPS (2022) 2, 4, 5, 7, 8, 9, 10, 11, 14
19. Su, J.W., Peng, C.H., Wonka, P., Chu, H.K.: Gpr-net: Multi-view layout estimation via a geometry-aware panorama registration network. In: CVPR (2023) 1, 3
20. Sumikura, S., Shibuya, M., Sakurada, K.: OpenVSLAM: A Versatile Visual SLAM Framework. In: Proceedings of the 27th ACM International Conference on Multimedia. pp. 2292–2295. MM '19, ACM, New York, NY, USA (2019). https://doi.org/10.1145/3343031.3350539, http://doi.acm.org/10.1145/3343031.3350539 13
21. Sun, C., Hsiao, C., Sun, M., Chen, H.: Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In: CVPR (2019) 1, 3, 4, 5, 8, 9, 10, 12, 13
22. Sun, C., Sun, M., Chen, H.: Hohonet: 360 indoor holistic understanding with latent horizontal features. In: CVPR (2021) 1, 3, 5, 8
23. Tang, S., Zhang, F., Chen, J., Wang, P., Furukawa, Y.: Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. arXiv preprint arXiv:2307.01097 (2023) 1
24. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. Advances in neural information processing systems **30** (2017) 3
25. Tran, P.V.: SSLayout360: Semi-Supervised Indoor Layout Estimation from 360-Degree Panorama. In: CVPR (2021) 1, 3
26. Tsai, G., Xu, C., Liu, J., Kuipers, B.: Real-time indoor scene understanding using bayesian filtering with motion cues. In: 2011 International Conference on Computer Vision. pp. 121–128. IEEE (2011) 3
27. Wang, F.E., Yeh, Y.H., Sun, M., Chiu, W.C., Tsai, Y.H.: Led2-net: Monocular 360deg layout estimation via differentiable depth rendering. In: CVPR (2021) 1, 3, 5, 6, 8, 10
28. Xu, J., Stenger, B., Kerola, T., Tung, T.: Pano2cad: Room layout from a single panorama image. In: 2017 IEEE winter conference on applications of computer vision (WACV). pp. 354–362. IEEE (2017) 3

29. Yang, S.T., Wang, F.E., Peng, C.H., Wonka, P., Sun, M., Chu, H.K.: Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3363–3372 (2019) 3
30. Zhao, Y., Wen, C., Xue, Z., Gao, Y.: 3d room layout estimation from a cubemap of panorama image via deep manhattan hough transform. In: European conference on computer vision. pp. 637–654. Springer (2022) 3
31. Zou, C., Colburn, A., Shan, Q., Hoiem, D.: Layoutnet: Reconstructing the 3d room layout from a single rgb image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2051–2059 (2018) 3
32. Zou, C., Su, J.W., Peng, C.H., Colburn, A., Shan, Q., Wonka, P., Chu, H.K., Hoiem, D.: Manhattan room layout reconstruction from a single 360 image: A comparative study of state-of-the-art methods (2020) 9