# SSLayout360: Semi-Supervised Indoor Layout Estimation from 360° Panorama

Phi Vu Tran
Flyreel AI Research
vuptran@flyreel.co

## Abstract

*Recent years have seen flourishing research on both semi-supervised learning and 3D room layout reconstruction. In this work, we explore the intersection of these two fields to advance the research objective of enabling more accurate 3D indoor scene modeling with less labeled data. We propose the first approach to learn representations of room corners and boundaries by using a combination of labeled and unlabeled data for improved layout estimation in a 360° panoramic scene. Through extensive comparative experiments, we demonstrate that our approach can advance layout estimation of complex indoor scenes using as few as 20 labeled examples. When coupled with a layout predictor pre-trained on synthetic data, our semi-supervised method matches the fully supervised counterpart using only 12% of the labels. Our work takes an important first step towards robust semi-supervised layout estimation that can enable many applications in 3D perception with limited labeled data.*

## 1. Introduction

The task of inferring room layout from a single view 360° panoramic image has been gaining much attention from the computer vision community in the past several years. The problem addresses an important step towards holistic indoor scene understanding that can enable structured 3D modeling of the physical environment. Recent state-of-the-art methods [33, 36, 40] have made substantial progress towards accurate 3D room layout reconstruction by adopting large and powerful neural network architectures for representation learning. However, there are several challenges associated with acquiring vast quantities of high-quality room layout annotations to supervise deep neural networks. For one, it is difficult to consistently annotate cluttered scenes with ambiguous wall boundaries, especially in rooms with complex layouts that contain many corners. The lack of large-scale labeled data with precise layout annotations for panoramic scenes further hinders the progress of this important problem that has many pertinent applications in 3D computer vision.

At the same time, recent years have seen flourishing research on deep semi-supervised learning (SSL) [3, 34, 37] that can leverage abundant unlabeled data for enhanced learning and generalization in the limited labeled data setting. The success of deep SSL has been mostly demonstrated on the relatively simple task of image classification, where the labeling procedure is the binary indication for the presence or absence of an object class. To our knowledge, the principles of SSL have not been studied in conjunction with room layout estimation, a complex and challenging task that depends on fine-grained human annotations, which presents an effective and efficient opportunity to improve learning with few annotated examples, as illustrated in Figure 1. In this work, we propose to explore and evaluate the potential contribution of unlabeled data for 3D room layout estimation, with the

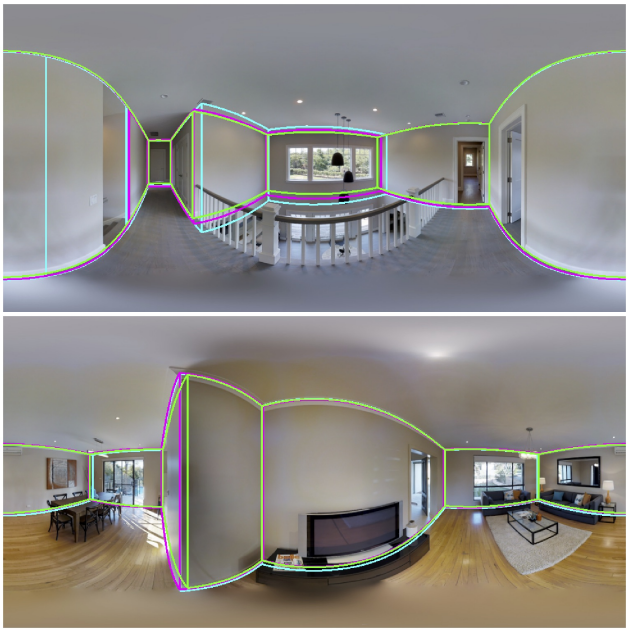

Figure 1. The effective use of unlabeled data improves complex 3D layout estimation with limited labels. We compare predicted layout boundary lines from a state-of-the-art supervised model [33] trained on 1,650 labels (cyan) with our proposed semi-supervised model (magenta) and show that our model's predictions follow more closely to the ground truth (green lines) using just 100 labels.

goal of promoting directed research towards the intersection of semi-supervised learning and 3D perception.

**Summary of Contributions** We present SSLayout360, a neural architecture capable of learning representations of floor-wall, ceiling-wall, and wall-wall boundaries from a combination of labeled and unlabeled data for improved room layout estimation in a 360° panoramic scene. Our work is the first substantive attempt at semi-supervised 3D layout reconstruction of complex indoor scenes using as few as 20 labeled examples.

Through extensive comparative experiments, we show SSLayout360 achieves new state-of-the-art results on a number of benchmarks for both simple and complex room layouts. Coupled with a layout predictor pre-trained on synthetic data, SSLayout360 matches the best-in-class fully supervised baseline using only 12% of the required labels.

We establish the first comprehensive set of semi-supervised benchmarks to measure the contribution of unlabeled data for indoor layout estimation. As part of this contribution, we propose a rigorous evaluation protocol to encourage the use of error bounds as standard practice and demonstrate the utility of unlabeled data across many experimental settings. We hope that our results serve as a strong baseline to inspire future research towards even more robust semi-supervised 3D layout reconstruction.

## 2. Related Work

**Deep Semi-Supervised Learning** The overarching goal of SSL is to make effective use of unlabeled data, without relying on any human supervision, to augment conventional supervised learning where labeled training data is scarce [7]. One set of approaches involves pre-training neural models on large-scale unlabeled data, by way of unsupervised [15, 19] or self-supervised [12, 17] representation learning, followed by supervised fine-tuning on downstream tasks with limited ground truth information (*e.g.*, detection, segmentation).

Another set of methods produces proxy targets for unlabeled data to be jointly trained end-to-end with ground truth labels. The training protocol for this class of SSL algorithms imposes an additional loss term to *regularize* the objective function of the supervised algorithm. Recent examples of self-supervised regularization [35, 37] improve supervised image classification performance by jointly training with an auxiliary self-supervised loss component based on the pretext task of image rotation recognition.

A third set of SSL methods based on consistency regularization [2, 30] largely follows the student-teacher framework [14]. As the student, the model learns from labeled data in the conventional supervised manner. As the teacher, it generates soft unsupervised targets by enforcing consistent ensembles of predictions on unlabeled training samples under random perturbations. The consistency constraint encourages the student to learn representations from unlabeled data for enhanced SSL. Existing research on student-teacher SSL formulates clever ways to generate good unsupervised targets. Rasmus *et al*. [28] showed the effectiveness of random noise in regularizing the targets. Miyato *et al*. [21, 22] further explored this idea and adopted adversarial noise as an implicit teacher to improve the quality of the targets. Laine and Aila [18] reduced teacher prediction variance by using an exponential moving average (EMA) to accumulate the predictions over training epochs. Tarvainen and Valpola [34] used an EMA of model weights to obtain an explicit "mean teacher", a simple but effective approach that was shown to achieve among the best SSL performances for image classification [24]. More recent extensions of the student-teacher framework have been demonstrated to surpass state-of-the-art fully supervised baselines using a fraction of the required labeled examples [3, 4].

**3D Room Layout Estimation** Room layout reconstruction has been an active research topic for over a decade [27], dating back to Delage *et al*. [9] fitting floor-wall boundaries in a perspective image under "Manhattan world" assumptions [8]. In this paper, we focus our review on modern, state-of-the-art approaches that recover room layout from a single RGB panorama represented in equirectangular projection covering a 360° horizontal and 180° vertical field-of-view.

Existing methods for layout estimation include PanoContext [38], LayoutNet [40], DuLa-Net [36], CFL [11], and HorizonNet [33]. These methods were tested on datasets with strong Manhattan assumptions, *i.e.*, the wall-wall boundaries form right angles and are orthogonal to the horizontal floor plane. The recent work of AtlantaNet [26] is not constrained to Manhattan scenes, and can recover room layout with walls that do not form right angles or are curved. All of these methods utilize a deep encoder-decoder neural network to predict layout elements, such as floor-wall and ceiling-wall boundaries and corner positions (LayoutNet, CFL, HorizonNet) or a semantic 2D floor plan in the ceiling view (DuLa-Net and AtlantaNet), and then fit the predicted elements to a 3D layout via a post-processing step [41].

Although the aforementioned methods have made substantial contributions towards accurate layout estimation, these models have all been trained in a fully supervised manner, on relatively small sets of annotated layout examples, while leaving an abundant amount of unlabeled panoramic indoor images completely untapped. While effective data augmentation strategies, such as panoramic horizontal rotation, horizontal flipping, and the recently introduced Pano Stretch [33], have been used to improve supervised learning, their utilization in conventional supervised training cannot exploit unlabeled data in a principled way.

Recent work in [20, 39] proposed the use of well-crafted, photo-realistic synthetic data with detailed ground truth structure annotations as a measure to alleviate costly hand-

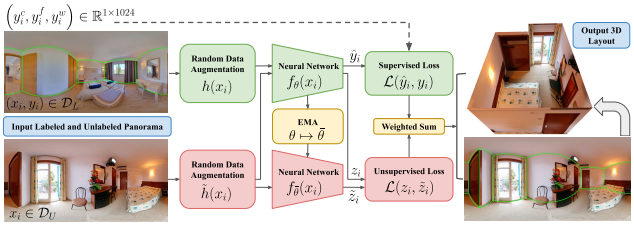


Figure 2. An illustration of the SSLayout360 architecture for semi-supervised indoor layout estimation from a 360° panoramic scene.

labeling efforts. The physically-based rendering process requires data modeling with extensive domain expertise, but generating synthetic data at scale is cheaper than the traditional approach of collecting, curating, and annotating panoramas from the real world. While synthetic data offers many advantages, there is still a challenge to learn transferable representations between real and synthetic domains in order to overcome the dataset bias [29].

In this work, we bridge the gap by combining unlabeled data with available labeled data from both real and synthetic contexts in a semi-supervised setting to further push the performance envelope of 3D room layout reconstruction.

## 3. Approach

The goal of this work is to learn 3D room layout from a single view 360° RGB panorama in the semi-supervised setting. The design and algorithmic overview of SSLayout360 are depicted in Figure 2 and Algorithm 1, with more details given in Sections A and B.4 of the supplementary material. The input is a set of labeled input-target pairs $(x_l, y_l) \in \mathcal{D}_L$ and a set of unlabeled examples $x_u \in \mathcal{D}_U$. As is common in prior work, we assume $\mathcal{D}_L$ and $\mathcal{D}_U$ are sampled from the same underlying data distribution (*e.g.*, indoor scenes), in which case $\mathcal{D}_L$ is a labeled subset of $\mathcal{D}_U$. In real-world applications, however, $\mathcal{D}_L$ and $\mathcal{D}_U$ often come from different, but somewhat related, data distributions (*e.g.*, indoor + outdoor panoramic scenes), and it is desirable for the SSL algorithm to appropriately learn from such distribution mismatch.

We train a neural network $f_\theta(x)$, a stochastic prediction function parametrized by $\theta$, to learn room layout boundaries by using a combination of $\mathcal{D}_L$ and $\mathcal{D}_U$. Our work builds on a successful, state-of-the-art variant of the student-teacher model, Mean Teacher [34], originally formulated for image classification and extends it to the more challenging task of layout estimation from complex panoramic indoor scenes. We consider the following compound objective for SSL:

$$\min_\theta \mathcal{L}_l(\mathcal{D}_L, \theta) + \lambda \mathcal{L}_u(\mathcal{D}_U, \theta), \tag{1}$$

where $\mathcal{L}_l$ is the supervised loss over labeled examples and $\mathcal{L}_u$ is the unsupervised loss defined for unlabeled data. The learning objective treats $\mathcal{L}_u$ as a regularizer, and $\lambda > 0$ is a hyper-parameter controlling the strength of regularization.

**Algorithm 1:** SSLayout360 training procedure.

**Input:** Training set of labeled inputs $(x_l, y_l) \in \mathcal{D}_L$.
Training set of unlabeled inputs $x_u \in \mathcal{D}_U$.
Data augmentation functions $h(x)$ and $\tilde{h}(x)$.
Student network $f_\theta(x)$ with trainable parameters $\theta$.
Teacher network $f_{\bar{\theta}}(x)$ with parameters $\bar{\theta} = \theta$.
Distance function $d$ (*e.g.*, $L_1$ and $L_2$).
**for** *each epoch over* $\mathcal{D}_U$ **do**
  $b_l \leftarrow h(x_l)$ ▷ Mini-batches of labeled input.
  $b_u \leftarrow \tilde{h}(x_u)$ ▷ Mini-batches of unlabeled input.
  **for** *each mini-batch* **do**
    $\hat{y}_l \leftarrow f_\theta(b_l)$ ▷ Forward pass on labeled input.
    $z_u \leftarrow f_\theta(b_u)$ ▷ Forward pass on unlabeled input.
    $\tilde{z}_u \leftarrow f_{\bar{\theta}}(b_u)$ ▷ Again using $\bar{\theta}$.
    $\mathcal{L} \leftarrow \frac{1}{|b_l|} \sum_{i \in b_l} d(\hat{y}_{il}, y_{il})$ ▷ Supervised loss.
      $+ \frac{\lambda}{|b_u|} \sum_{i \in b_u} d(z_{iu}, \tilde{z}_{iu})$ ▷ Unsupervised loss.
    $\lambda \leftarrow e^{-5(1-T)^2}$ ▷ Ramp up $\lambda$ for $T \in [0, 1]$.
    $\theta \leftarrow \theta - \nabla_\theta \mathcal{L}$ ▷ Update $\theta$ via gradient descent.
    $\bar{\theta} \leftarrow \alpha\bar{\theta} + (1-\alpha)\theta$ ▷ Update $\bar{\theta}$ via EMA.
  **end**
**end**
**return** $\theta, \bar{\theta}$

### 3.1. Semi-Supervised Layout Estimation

The task of room layout estimation essentially boils down to inferring the floor-wall and ceiling-wall boundaries and wall-wall (or corner) positions. In this work, we derive insight from HorizonNet [33] to regress layout boundaries and corners to the ground truth for each column of the input image in the semi-supervised setting. In principle, other layout prediction methods based on pixel-wise classification (*e.g.*, LayoutNet, AtlantaNet) could be extended to the semi-supervised setting. But a comparative investigation of alternative layout prediction methods under the semi-supervised setting is beyond the scope of this paper, and would be an interesting research direction for future work.

**HorizonNet** We choose HorizonNet as our prediction function $f_\theta(x)$ for its simplicity, efficient computation, and state-of-the-art performance on room layout estimation. The input to HorizonNet is an RGB panorama with shape $3 \times 512 \times 1024$ (for channel, height, width) along with a 3-channel target vector of size $3 \times 1 \times 1024$ representing the ceiling-wall ($y_c$), floor-wall ($y_f$), and wall-wall ($y_w$) boundary position of each image column. The values of $y_c$ and $y_f$ are normalized in $[-\pi/2, \pi/2]$, and $y_w$ is scaled to $[0, 1]$.

HorizonNet follows an encoder-decoder approach to learn whole-room layout from a panoramic scene, similar to other

competing methods. The encoder is the ResNet-50 architecture [13] pre-trained on ImageNet [10], combined with a sequence of convolution layers followed by ReLU [23] activation, to compute an abstract $1024 \times 1 \times 256$ dimensional feature representation from the input image. The decoder is a bidirectional recurrent neural network [31] that predicts $(\hat{y}_c, \hat{y}_f, \hat{y}_w) \in \mathbb{R}^{1\times1024}$ column by column. Next, we formulate HorizonNet as a student-teacher model, and describe the resulting SSLayout360 architecture as a semi-supervised learner for 3D layout reconstruction.

**SSLayout360** Our approach treats HorizonNet as a stochastic predictor with the dual role of being both the student and teacher. Given a batch $b_l$ of labeled examples, and their real-valued target vectors $y_l \in \mathbb{R}^{3\times1\times1024}$, and a batch $b_u$ of unlabeled examples at each training step, we forward propagate HorizonNet three times: (1) on the batch of labeled examples as the student $f_\theta(x)$ using parameters $\theta$ to produce real-valued prediction vectors $\hat{y}_l \in \mathbb{R}^{3\times1\times1024}$, (2) on the unlabeled batch using the same parameters $\theta$ to compute $z_u \in \mathbb{R}^{3\times1\times1024}$, and (3) on the unlabeled batch as the teacher $f_{\bar{\theta}}(x)$ using parameters $\bar{\theta}$ to output $\tilde{z}_u \in \mathbb{R}^{3\times1\times1024}$. Here, $\bar{\theta}$ is an exponential moving average (EMA) of the student's parameters $\theta$ after each training step $t$:

$$\bar{\theta}_t = \alpha\bar{\theta}_{t-1} + (1-\alpha)\theta_t, \tag{2}$$

where $\alpha \in [0, 1]$ is a decay hyper-parameter. The intuition for setting $\bar{\theta} = \text{EMA}(\theta)$ is to obtain a good teacher that provides stable unsupervised targets for the student to imitate, and was the main result of Mean Teacher. As is common practice, we do not back-propagate gradients through the teacher and keep its prediction fixed at each training step [4, 34]. The alternative case is to set $\bar{\theta} = \theta$ with $\alpha = 0$ and back-propagate gradients through both student and teacher models, which was the formulation of $\Pi$ model [18], and has been shown to produce less stable unsupervised targets and overall inferior SSL performance to Mean Teacher [34]. Section 5.3 provides an ablation experiment where we evaluate both settings for comparative semi-supervised layout estimation.

### 3.2. Loss Function

The real-valued prediction vectors $\hat{y}_l$ are regressed to target vectors $y_l$ using $L_1$ distance for $(y_c, y_f)$ and squared $L_2$ distance for $y_w$. The supervised loss component, evaluated over a mini-batch of labeled examples, is computed as:

$$\mathcal{L}_l = \frac{1}{|b_l|}\sum_{i\in b_l} \|\hat{y}_{ic} - y_{ic}\|_1 + \|\hat{y}_{if} - y_{if}\|_1 + \|\hat{y}_{iw} - y_{iw}\|_2^2. \tag{3}$$

Our supervised objective is different from HorizonNet, in that we use the squared $L_2$ loss, or the Brier score [5], instead of binary cross-entropy loss for the wall-wall corner $y_w$. The Brier score is commonly used in the SSL literature because it is bounded and does not heavily penalize predicted probabilities far away from the ground truth [4, 18, 34]. Our initial experiments showed that the squared $L_2$ loss gave slightly better accuracy performance than cross-entropy loss.

For the unsupervised loss component, we constrain $z_u$ and $\tilde{z}_u$ to be close by computing their $L_1$ and squared $L_2$ distances over a mini-batch of unlabeled examples, similar to Equation (3):

$$\mathcal{L}_u = \frac{1}{|b_u|}\sum_{i\in b_u} \|z_{ic} - \tilde{z}_{ic}\|_1 + \|z_{if} - \tilde{z}_{if}\|_1 + \|z_{iw} - \tilde{z}_{iw}\|_2^2. \tag{4}$$

It is a reasonable objective to enforce consistency on $z_u$ and $\tilde{z}_u$ because $f_\theta(x)$ and $f_{\bar{\theta}}(x)$ are stochastic predictors with random dropout [32] and input data augmentation at each forward pass. By minimizing the discrepancy between student $z_u$ and teacher $\tilde{z}_u$ predictions on unlabeled instances, we encourage the student to learn additional layout representations from unlabeled data via the unsupervised targets provided by the teacher. The compound objective for training SSLayout360 on both labeled and unlabeled data is the weighted sum of the supervised and unsupervised losses:

$$\mathcal{L} = \mathcal{L}_l + \lambda\mathcal{L}_u. \tag{5}$$

Our formulation of the SSLayout360 objective works well for layout estimation and stands out from previous SSL methods in that we *maintain compatibility* between the supervised and unsupervised terms by applying $L_1$ and $L_2$ losses to both. This has the intended benefit of removing the need to tune the weight hyper-parameter $\lambda$ during training; we simply set $\lambda = 1$ to obtain reliable results in all experiments across all datasets under consideration. By contrast, Mean Teacher and other SSL methods used different loss functions for the supervised and unsupervised terms (*e.g.*, cross entropy + $L_2$), resulting in the need to carefully tune $\lambda$ to manage the balance between the two objectives.

Our approach to SSLayout360 assumes that the teacher provides good unsupervised targets for the student to imitate. As illustrated in Figure 3, at the beginning of model training, both student and teacher models are likely to produce incorrect and inconsistent predictions, especially when few labels are available. Similar to SSL for image classification [18, 34], we mitigate a potentially degenerate solution by ramping up the unsupervised loss weight from 0 to 1 according to the sigmoid-shaped function: $\lambda(t) = e^{-5(1-t/T)^2}$, where $t$ is the current training iteration and $T$ is the number of iterations at which to stop the ramp-up. We define $T$ to be a percentage of the maximum number of iterations, which is the product of training epochs and the cardinality of mini-batches in the unlabeled dataset. For example, $T@30\%$ means that if we train SSLayout360 for 100 epochs over an unlabeled set of 2,000 images using a mini-batch of 4, then $T@30\% = 0.30 \times 100 \times 2000/4 = 15,000$ iterations. The ramp-up period ensures that student learning progresses


Figure 3. The evolution of HorizonNet as both the student (red lines) and teacher (green lines) over the course of training on 100 labeled and 1,009 unlabeled images. We encourage the student to learn layout representations of floor, ceiling, and wall boundaries from unlabeled data by enforcing consistency (or minimizing discrepancy) between student and teacher predictions perturbed by random model and data noise.

predominantly via the supervised loss on labeled data at the beginning of training up to $T$ iterations, after which the teacher becomes reliable to provide good stable targets.

# 4. Experimental Setup

## 4.1. Datasets

We train and evaluate our SSLayout360 algorithm on three challenging benchmark datasets covering both simple cuboid and complex non-cuboid indoor layouts. For cuboid layout estimation, we use PanoContext [38] and Stanford-2D3D [1], which consist of 512 and 550 RGB panoramic images, respectively. For non-cuboid layout estimation, we train and evaluate on MatterportLayout [41], which comprise 2,295 RGB panoramic images of indoor scenes having up to 22 corners. We use the standard training, validation, and test splits provided by Zou *et al*. [40, 41] for all three datasets. See Section B.1 in the supplementary material for details.

MatterportLayout is a labeled subset of the larger Matterport3D [6] dataset comprising 10,912 panoramas of general indoor and outdoor environments. We use the auxiliary Matterport3D dataset (minus 458 test instances) of 10,454 panoramas as a source of extra unlabeled data to augment our SSL experiments on MatterportLayout, and to evaluate SSLayout360 under the condition of data distribution mismatch that includes a mixture of indoor and outdoor scenes.

We also experiment with Structured3D [39], a large photo-realistic *synthetic* dataset comprising 21,835 panoramas of rooms in 3,500 diverse indoor scenes with ground truth cuboid and non-cuboid layout annotations. We pre-train HorizonNet on 18,362 synthetic images and perform transfer learning on the MatterportLayout dataset via fine-tuning.

## 4.2. Evaluation Protocol

Existing evaluation procedures for layout estimation suffer from statistical unreliability, given the relatively small sample sizes of training, validation, and test instances. Case in point, our findings in Table 1 show that supervised results can vary as much as 4% points. Moreover, prior work only reported point estimates without standard error bounds for performance evaluations, making a comparison of published methods difficult when considering for statistical significance. We adopt the rigorous evaluation protocol previously used for semi-supervised image classification [24] and extend it to this work for semi-supervised layout estimation.

We conduct our supervised and semi-supervised experiments over four runs using different random seeds, and report the mean and standard deviation to assess statistical significance. This helps to evaluate the contributions of unlabeled data instead of confounding statistical noise inherent in deep neural networks (*e.g*., dropout, weight initialization). For SSL, we follow the standard practice of randomly sampling varying amounts of the training data as labeled examples while treating the combined training and validation sets, discarding all label information, as the source of unlabeled data [24, 34, 35]. We take extra care not to include any test instances as unlabeled data. We train SSLayout360 with both labeled and unlabeled data according to Algorithm 1 and compare its performance to that of HorizonNet trained using only the labeled portion in the traditional supervised manner.

We extend our rigorous evaluation protocol to include experiments with extra unlabeled and synthetic data, which have not been explored for layout estimation. In experiments using synthetic data, we ask the question: given a strong predictor pre-trained on synthetic data, can our model bridge the performance gap between synthetic and real data by using a combination of both in the semi-supervised setting?

## 4.3. Implementation Details

We implement SSLayout360 using PyTorch [25] and train on 2 NVIDIA TITAN X GPUs each with 12GB of video memory. Our SSL experiments take between 6 and 65 hours to complete, depending on the number of training epochs and how much unlabeled data is used in combination with labeled data. The supervised HorizonNet baselines are trained using the original authors' publicly available source code for direct comparison. See Section B.2 in the supplementary material for a detailed breakdown of our training protocol.

**Data Processing and Augmentation** We separate the input data source into labeled and unlabeled branches. All images are pre-processed by the panorama alignment algorithm described in [40] to enforce the Manhattan constraint, owing to the use of HorizonNet as the prediction function. We follow

| **PanoContext** | 3D IoU (%) ↑ | | | | |
|---|---|---|---|---|---|
| Method | 20 labels 1,009 images | 50 labels 1,009 images | 100 labels 1,009 images | 200 labels 1,009 images | 963 labels 1,009 images |
| HorizonNet | 61.48 ± 2.07 | 63.84 ± 2.87 | 65.43 ± 1.30 | 75.76 ± 0.62 | **83.55 ± 0.31** |
| SSLayout360 | **63.05 ± 0.65** | **68.41 ± 1.02** | **72.86 ± 1.07** | **78.64 ± 0.72** | 83.30 ± 0.53 |
| | Corner Error (%) ↓ | | | | |
| Method | 20 labels 1,009 images | 50 labels 1,009 images | 100 labels 1,009 images | 200 labels 1,009 images | 963 labels 1,009 images |
| HorizonNet | 3.51 ± 0.79 | 2.78 ± 0.90 | 3.17 ± 0.27 | **1.07 ± 0.15** | **0.70 ± 0.02** |
| SSLayout360 | **2.57 ± 0.58** | **1.76 ± 0.51** | **1.42 ± 0.31** | **0.96 ± 0.06** | **0.69 ± 0.01** |
| | Pixel Error (%) ↓ | | | | |
| Method | 20 labels 1,009 images | 50 labels 1,009 images | 100 labels 1,009 images | 200 labels 1,009 images | 963 labels 1,009 images |
| HorizonNet | 5.68 ± 0.52 | 5.03 ± 0.45 | 4.75 ± 0.06 | 3.17 ± 0.17 | 1.97 ± 0.03 |
| SSLayout360 | **4.84 ± 0.32** | **3.73 ± 0.26** | **3.47 ± 0.24** | **2.72 ± 0.08** | **1.90 ± 0.02** |

| **Stanford-2D3D** | 3D IoU (%) ↑ | | | | |
|---|---|---|---|---|---|
| Method | 20 labels 949 images | 50 labels 949 images | 100 labels 949 images | 200 labels 949 images | 916 labels 949 images |
| HorizonNet | 62.20 ± 3.98 | 68.27 ± 1.45 | 69.94 ± 3.64 | 74.95 ± 3.69 | 82.79 ± 0.90 |
| SSLayout360 | **71.60 ± 2.04** | **73.86 ± 1.65** | **76.96 ± 1.20** | **79.78 ± 0.83** | **84.66 ± 0.57** |
| | Corner Error (%) ↓ | | | | |
| Method | 20 labels 949 images | 50 labels 949 images | 100 labels 949 images | 200 labels 949 images | 916 labels 949 images |
| HorizonNet | 2.70 ± 0.60 | 1.64 ± 0.12 | 1.66 ± 0.20 | 1.50 ± 0.18 | 0.64 ± 0.02 |
| SSLayout360 | **1.69 ± 0.22** | **1.32 ± 0.04** | **1.15 ± 0.24** | **1.04 ± 0.13** | **0.60 ± 0.01** |
| | Pixel Error (%) ↓ | | | | |
| Method | 20 labels 949 images | 50 labels 949 images | 100 labels 949 images | 200 labels 949 images | 916 labels 949 images |
| HorizonNet | 5.03 ± 0.51 | 3.95 ± 0.22 | 3.77 ± 0.39 | 3.69 ± 0.26 | 2.13 ± 0.05 |
| SSLayout360 | **3.50 ± 0.17** | **3.24 ± 0.18** | **3.01 ± 0.26** | **2.91 ± 0.27** | **1.97 ± 0.06** |

Table 1. Quantitative cuboid layout results evaluated on the **PanoContext (left)** and **Stanford-2D3D (right)** test sets over four randomized trials. The semi-supervised SSLayout360 settings outperform the supervised HorizonNet baselines across most metrics under consideration.

| **Mixed Corners** | 3D IoU (%) ↑ | | | | |
|---|---|---|---|---|---|
| Method | 50 labels 1,837 images | 100 labels 1,837 images | 200 labels 1,837 images | 400 labels 1,837 images | 1,650 labels 1,837 images |
| HorizonNet | 63.44 ± 0.56 | 68.79 ± 0.49 | 72.25 ± 0.50 | 74.46 ± 0.35 | 79.12 ± 0.37 |
| SSLayout360 | **67.42 ± 0.24** | **72.37 ± 0.35** | **75.31 ± 0.37** | **77.09 ± 0.41** | **80.33 ± 0.48** |
| | 2D IoU (%) ↑ | | | | |
| Method | 50 labels 1,837 images | 100 labels 1,837 images | 200 labels 1,837 images | 400 labels 1,837 images | 1,650 labels 1,837 images |
| HorizonNet | 67.17 ± 0.65 | 72.06 ± 0.49 | 75.16 ± 0.53 | 77.15 ± 0.36 | 81.54 ± 0.31 |
| SSLayout360 | **71.03 ± 0.28** | **75.46 ± 0.36** | **78.05 ± 0.33** | **79.67 ± 0.40** | **82.54 ± 0.51** |
| | $\delta_1$ ↑ | | | | |
| Method | 50 labels 1,837 images | 100 labels 1,837 images | 200 labels 1,837 images | 400 labels 1,837 images | 1,650 labels 1,837 images |
| HorizonNet | 0.76 ± 0.01 | 0.84 ± 0.01 | 0.89 ± 0.01 | 0.91 ± 0.01 | **0.94 ± 0.01** |
| SSLayout360 | **0.81 ± 0.01** | **0.89 ± 0.01** | **0.91 ± 0.01** | **0.93 ± 0.01** | **0.95 ± 0.01** |
| | RMSE ↓ | | | | |
| Method | 50 labels 1,837 images | 100 labels 1,837 images | 200 labels 1,837 images | 400 labels 1,837 images | 1,650 labels 1,837 images |
| HorizonNet | 0.41 ± 0.01 | 0.34 ± 0.01 | 0.30 ± 0.01 | 0.28 ± 0.01 | **0.23 ± 0.01** |
| SSLayout360 | **0.35 ± 0.01** | **0.29 ± 0.01** | **0.27 ± 0.01** | **0.25 ± 0.01** | **0.22 ± 0.01** |

Table 2. Quantitative non-cuboid layout results evaluated on the MatterportLayout test set over four runs. SSLayout360 surpasses the supervised HorizonNet baselines across all settings and metrics.

standard panorama augmentation techniques [33, 36, 40], and apply random stretching in both $(k_x, k_z) \in [0.5, 1.5]$ directions, horizontal rotation $r \in (0°, 360°]$, left-right flipping with probability $0.5$, and gamma correction with $\gamma \in [0.5, 2.0]$ to each branch independently.

**Hyper-parameters** We start with the hyper-parameter configuration for the HorizonNet baseline, and only tune hyper-parameters specific to SSLayout360 which include: the consistency loss weight $\lambda$, the ramp-up period $T$, and EMA decay $\alpha$. In our implementation, we tune hyper-parameters on the PanoContext validation set and fix them constant for experiments on Stanford-2D3D and MatterportLayout. We make a conscientious effort to keep our hyper-parameter configuration general to avoid overfitting on a per-dataset or per-experiment basis, which can limit the real-world applicability of our method. Section 5.3 discusses our hyper-parameter choices with detailed ablation experiments.

**Model Selection** We use the same underlying architecture and training protocol for both supervised and SSL experiments, with the exception of tuning hyper-parameters specific to SSLayout360. This is to ensure that any performance boost in the SSL setting is directly attributed to unlabeled data and not to changes in model configuration.

We employ the Adam optimizer [16] to train HorizonNet and SSLayout360. Following HorizonNet's training protocol, we use learning rate 0.0003 and batch size 8 for PanoContext and Stanford-2D3D experiments; for MatterportLayout experiments, we use learning rate 0.0001 and batch size 4 to achieve the best results on both HorizonNet and SSLayout360. Similar to Tran [35], we anneal the learning rate hyper-parameter after each training step $t$ according to the polynomial schedule: $\text{lr} \times (1 - {}^{t}/_{t_{\max}})^{0.5}$.

We check-point the best models for testing based on their best performances on the validation sets. At test time, SSLayout360 produces two sets of model parameters $\theta$ and $\bar{\theta} = \text{EMA}(\theta)$, both of which are expected to have comparable predictive accuracy. For simplicity, we take the average of both predictions on each test instance and report results, but otherwise do not perform any test-time augmentation.

**Performance Metrics** We assess layout estimation performance using six standard metrics to maintain parity with previous work [33, 40, 41]. For evaluation between predicted layout and the ground truth, we use 3D intersection over union (IoU), 2D IoU, corner error, and pixel error. For evaluation between predicted and ground truth layout depth, we use root mean squared error (RMSE) and $\delta_1$, defined by Zou *et al.* [41] as "the percentage of pixels where the ratio (or its reciprocal) between the prediction and the label is within a threshold of 1.25." We evaluate cuboid layout estimation using 3D IoU, corner error, pixel error, and non-cuboid layout using 3D IoU, 2D IoU, RMSE, and $\delta_1$.

## 5. Results and Analysis

### 5.1. Quantitative Evaluation

**Cuboid Layout Estimation** Quantitative evaluations on the PanoContext and Stanford-2D3D test sets are presented

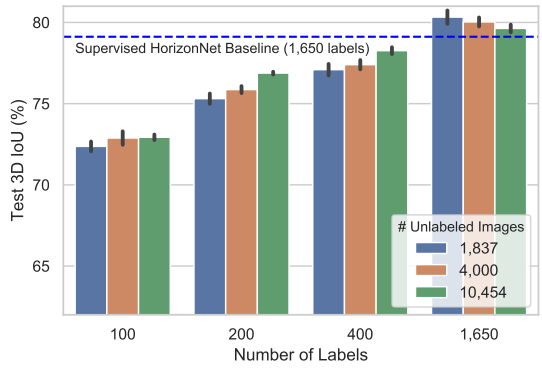


Figure 4. SSLayout360 results averaged over four runs on MatterportLayout with increasing amount of unlabeled data, evaluated for rooms with mixed corners. More unlabeled data improves semi-supervised layout estimation when the number of labels is 400 or less. The effect of unlabeled data diminishes when all labels are used, but performance still remains above the supervised baseline.

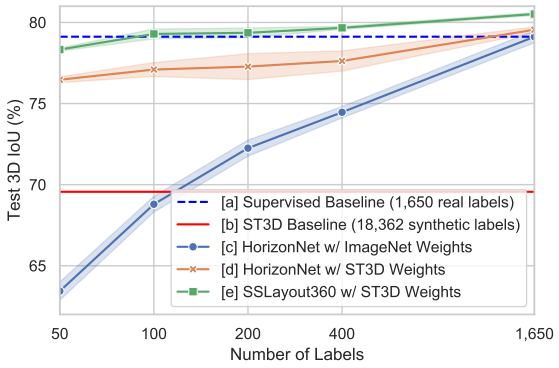


Figure 5. Supervised and semi-supervised fine-tuning experiments on MatterportLayout for rooms with mixed corners using Structured3D (ST3D) synthetic data. Shaded regions denote standard deviation over four runs. The $x$-axis is shown on the log scale. Coupled with HorizonNet pre-trained on ST3D **[d]**, SSLayout360 matches the fully supervised counterpart with only 200 labels **[e]**.

in Table 1. We show SSLayout360 surpasses the supervised HorizonNet baselines across most settings under consideration. For the fully supervised setting with all labeled images, SSLayout360 achieves similar performance to HorizonNet on the 3D IoU metric, and outperforms HorizonNet on the corner and pixel error metrics, indicating the benefit of learning with consistency regularization for layout estimation.

**Non-Cuboid Layout Estimation** We present quantitative non-cuboid results on the challenging MatterportLayout test set in Table 2. For space considerations, we report the overall performances of rooms having "mixed corners". Detailed results for rooms having 4, 6, 8, 10 or more corners are reported in Section D of the supplementary material. The trend is clear: the effective use of unlabeled data, in combination with labeled data, improves layout estimation across all settings and metrics under investigation, most notably in the scenarios with only 50 and 100 labeled images.

**SSLayout360 without Unlabeled Data** In light of the results observed in Tables 1 – 2 for the fully supervised setting, we perform experiments to clarify the utility of SSLayout360 without unlabeled data, and report them in Section C of the supplementary material due to space limitation. Our findings show that SSLayout360 without unlabeled data produces slightly better results than the HorizonNet counterpart across most settings and metrics. Our results corroborate previous SSL literature that regularization can slightly improve supervised learning without unlabeled data [21, 35, 37]. In scenarios with additional unlabeled data, SSLayout360 gives a significant boost in accuracy over the supervised baselines.

**SSLayout360 with Extra Unlabeled Data** We randomly sample 2,163 and 8,617 extra unlabeled images from the Matterport3D dataset and combine them with the MatterportLayout training and validation sets (1,837 images) for the total of 4,000 and 10,454 unlabeled images, respectively. Figure 4 shows that extra unlabeled data helps improve semi-supervised layout estimation by an average of about 1% point when the number of labels is 400 or less. When the full labeled set is used with 4,000 and 10,454 unlabeled images, test performance dips a bit but remains above the fully supervised HorizonNet baseline. Our encouraging results suggest that SSLayout360 is capable of learning additional supervisory signals from extra unlabeled data, even when there is a data distribution mismatch.

**SSLayout360 with Synthetic Data** Figure 5 summarizes the following findings when utilizing Structured3D synthetic data to evaluate on MatterportLayout: there is a large performance gap between HorizonNet models trained on real and synthetic data **[a] vs. [b]**; HorizonNet pre-trained on Structured3D and fine-tuned on MatterportLayout **[d]** always outperforms HorizonNet initialized with ImageNet weights **[c]**; and SSLayout360 with pre-trained HorizonNet matches the fully supervised results using only 200 labels **[e]**, effectively bridging the performance gap between real and synthetic domains in the semi-supervised setting.

## 5.2. Qualitative Evaluation

Figure 6 compares qualitative test results between HorizonNet and SSLayout360 trained on 100 labels for PanoContext, Stanford-2D3D, and MatterportLayout under equirectangular view. We report additional qualitative results for 3D layout reconstruction utilizing the post-processing algorithm of HorizonNet in Section E of the supplementary material.

## 5.3. Ablation Study

Figures 7(a) – 7(c) show ablation experiments where we systematically search for good values of the three hyper-parameters essential to the SSLayout360 algorithm. In each

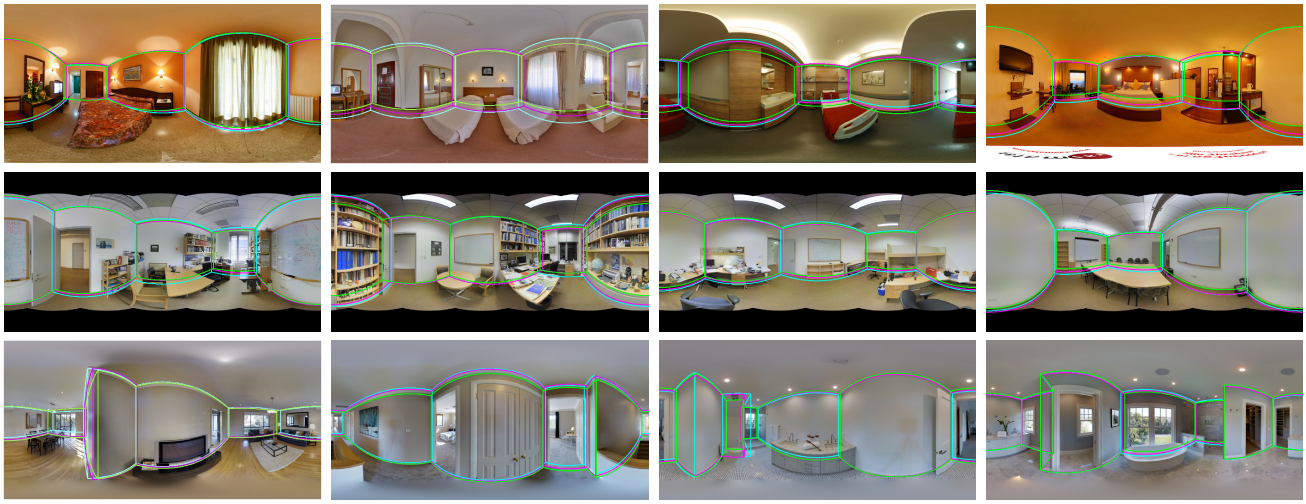

Figure 6. Exemplar qualitative test results of cuboid and non-cuboid layout estimation under equirectangular view. Best viewed electronically. We compare supervised HorizonNet trained on 100 labels with our SSLayout360 model trained on the same 100 labels along with unlabeled images for **PanoContext (top)**, **Stanford-2D3D (middle)**, and **MatterportLayout (bottom)**. Layout boundary lines for HorizonNet are shown in cyan, SSLayout360 in magenta, and ground truth in green. We observe that SSLayout360 predicts layout boundary lines following more closely to the ground truth than HorizonNet, which explains the performance gap between the supervised and semi-supervised models.

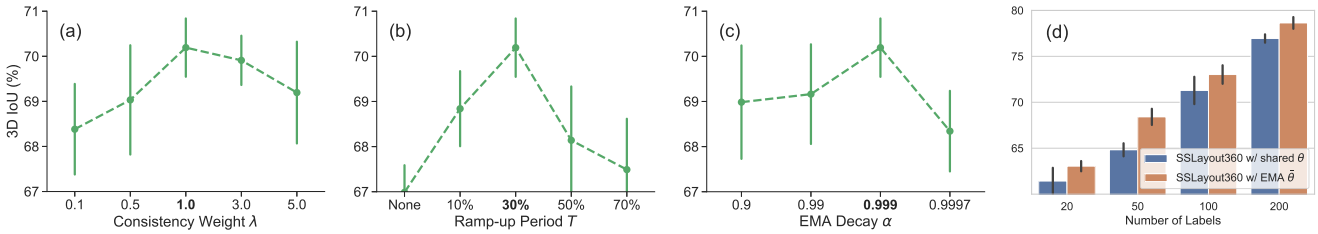


Figure 7. Ablation experiments on PanoContext using 100 labeled and 1,009 unlabeled images over four randomized runs. For each hyper-parameter **(a) – (c)**, we find the optimal value (in boldface) via the "knee in the curve" that produces the best 3D IoU performance on the validation set. For the experiment in **(d)**, we show SSLayout360 achieves uniformly better performance with EMA $\bar{\theta}$ as the teacher.

experiment, we vary one hyper-parameter while keeping the other two constant. We confirm our hypothesis in the formulation of Equation (5) that $\lambda = 1$ is the principled choice for the consistency weight. We observe that the ramp-up period $T@30\%$ provides a good balance between student learning and teacher soft supervision that results in the highest validation accuracy. The intuition for the ramp-up is that if it is too short, then the teacher provides unstable targets; and if ramp-up takes too long, then the teacher's supervisory contributions to performance are delayed. We also find the EMA decay coefficient $\alpha = 0.999$ gives the best validation accuracy with the lowest standard deviation. Lastly, Figure 7(d) shows that SSLayout360 with teacher parameters $\bar{\theta}$, which is the main driver for all experiments in this paper, uniformly outperforms SSLayout360 with shared $\bar{\theta} = \theta$.

## 6. Conclusion

We presented SSLayout360, an approach that combines the strengths of HorizonNet and Mean Teacher, and extends them both to enable semi-supervised layout estimation from complex 360° panoramic indoor scenes. A distinct modification that allows our algorithm to work well is the loss formulation to regress real-valued prediction vectors to ground truth via $L_1$ and $L_2$ distances. We evaluated our approach on three challenging benchmarks across six metrics and reported steady gains in semi-supervised layout estimation from the state-of-the-art supervised baseline, utilizing only unlabeled data as the additional source of soft "supervisory" information. Our work takes an important first step towards robust semi-supervised layout estimation with exciting applications related to 3D scene modeling and understanding.

## Acknowledgments

The author thanks Cole Winans and Brian Keller for their continued support, Victor Palmer for technical assistance with Matlab and fruitful discussions, Mike Procopio and anonymous reviewers for their thoughtful and constructive feedback on this paper.

## References


[1] Iro Armeni, Sasha Sax, Amir R. Zamir, and Silvio Savarese. Joint 2D-3D-Semantic Data for Indoor Scene Understanding. arXiv:1702.01105, 2017. 5

[2] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with Pseudo-Ensembles. In *Advances in Neural Information Processing Systems*, 2014. 2

[3] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. ReMixMatch: Semi-Supervised Learning with Distribution Alignment and Augmentation Anchoring. In *International Conference on Learning Representations*, 2020. 1, 2

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. MixMatch: A Holistic Approach to Semi-Supervised Learning. In *Advances in Neural Information Processing Systems*, 2019. 2, 4

[5] Glenn W. Brier. Verification of Forecasts Expressed in Terms of Probability. *Monthly Weather Review*, 78:1–3, 1950. 4

[6] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D Data in Indoor Environments. In *International Conference on 3D Vision*, 2017. 5

[7] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 1st edition, 2010. 2

[8] James M. Coughlan and Alan L. Yuille. Manhattan World: Compass Direction from a Single Image by Bayesian Inference. In *International Conference on Computer Vision*, volume 2, pages 941–947, 1999. 2

[9] Erick Delage, Honglak Lee, and Andrew Y. Ng. A Dynamic Bayesian Network Model for Autonomous 3D Reconstruction from a Single Indoor Image. In *Computer Vision and Pattern Recognition*, pages 2418–2428, 2006. 2

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Computer Vision and Pattern Recognition*, pages 248–255, 2009. 4

[11] Clara Fernandez-Labrador, José M. Fácil, Alejandro Pérez-Yus, Cédric Demonceaux, Javier Civera, and J. J. Guerrero. Corners for Layout: End-to-End Layout Recovery From 360 Images. *IEEE Robotics and Automation Letters*, 5:1255–1262, 2020. 2

[12] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised Representation Learning by Predicting Image Rotations. In *International Conference on Learning Representations*, 2018. 2

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Computer Vision and Pattern Recognition*, pages 770–778, 2016. 4

[14] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the Knowledge in a Neural Network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015. 2

[15] Kevin Jarrett, Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann LeCun. What is the best multi-stage architecture for object recognition? In *International Conference on Computer Vision*, pages 2146–2153, 2009. 2

[16] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*, 2015. 6

[17] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting Self-Supervised Visual Representation Learning. In *Computer Vision and Pattern Recognition*, 2019. 2

[18] Samuli Laine and Timo Aila. Temporal Ensembling for Semi-Supervised Learning. In *International Conference on Learning Representations*, 2017. 2, 4

[19] Quoc V. Le, Marc'Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg S. Corrado, Jeff Dean, and Andrew Y. Ng. Building High-Level Features Using Large Scale Unsupervised Learning. In *International Conference on Machine Learning*, 2012. 2

[20] Wenbin Li, Sajad Saeedi, John McCormac, Ronald Clark, Dimos Tzoumanikas, Qing Ye, Yuzhong Huang, Rui Tang, and Stefan Leutenegger. InteriorNet: Mega-Scale Multi-Sensor Photo-Realistic Indoor Scenes Dataset. In *British Machine Vision Conference*, 2018. 2

[21] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. *Transactions on Pattern Analysis and Machine Intelligence*, 41:1979–1993, 2017. 2, 7

[22] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, Ken Nakae, and Shin Ishii. Distributional Smoothing with Virtual Adversarial Training. In *International Conference on Learning Representations*, 2016. 2

[23] Vinod Nair and Geoffrey Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. In *International Conference on Machine Learning*, 2010. 4

[24] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian J. Goodfellow. Realistic Evaluation of Semi-Supervised Learning Algorithms. In *Advances in Neural Information Processing Systems*, 2018. 2, 5

[25] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, pages 8024–8035. Curran Associates, Inc., 2019. 5

[26] Giovanni Pintore, Marco Agus, and Enrico Gobbetti. AtlantaNet: Inferring the 3D Indoor Layout from a Single 360 Image beyond the Manhattan World Assumption. In *European Conference on Computer Vision*, 2020. 2

[27] Giovanni Pintore, Claudio Mura, Fabio Ganovelli, Lizeth Fuentes-Perez, Renato Pajarola, and Enrico Gobbetti. State-of-the-art in Automatic 3D Reconstruction of Structured Indoor Environments. *Computer Graphics Forum*, 39:667–699, 2020. 2

[28] Antti Rasmus, Harri Valpola, Mikko Honkala, Mathias Berglund, and Tapani Raiko. Semi-Supervised Learning with Ladder Networks. In *Advances in Neural Information Processing Systems*, 2015. 2

[29] Zhongzheng Ren and Yong Jae Lee. Cross-Domain Self-Supervised Multi-Task Feature Learning using Synthetic Imagery. In *Computer Vision and Pattern Recognition*, 2018. 3

[30] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with Stochastic Perturbations for Deep Semi-Supervised Learning. In *Advances in Neural Information Processing Systems*, 2016. 2

[31] Mike Schuster and Kuldip K. Paliwal. Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997. 4

[32] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014. 4

[33] Cheng Sun, Chi-Wei Hsiao, Min Sun, and Hwann-Tzong Chen. HorizonNet: Learning Room Layout With 1D Representation and Pano Stretch Data Augmentation. In *Computer Vision and Pattern Recognition*, pages 1047–1056, 2019. 1, 2, 3, 6

[34] Antti Tarvainen and Harri Valpola. Mean Teachers are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Deep Learning Results. In *Advances in Neural Information Processing Systems*, 2017. 1, 2, 3, 4, 5

[35] Phi Vu Tran. Exploring Self-Supervised Regularization for Supervised and Semi-Supervised Learning. In *NeurIPS Workshop on Learning with Rich Experience: Integration of Learning Paradigms*, 2019. 2, 5, 6, 7

[36] Shang-Ta Yang, Fu-En Wang, Chi-Han Peng, Peter Wonka, Min Sun, and Hung-Kuo Chu. DuLa-Net: A Dual-Projection Network for Estimating Room Layouts From a Single RGB Panorama. In *Computer Vision and Pattern Recognition*, pages 3358–3367, 2019. 1, 2, 6

[37] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4L: Self-Supervised Semi-Supervised Learning. In *International Conference on Computer Vision*, 2019. 1, 2, 7

[38] Yinda Zhang, Shuran Song, Ping Tan, and Jianxiong Xiao. PanoContext: A Whole-Room 3D Context Model for Panoramic Scene Understanding. In *European Conference on Computer Vision*, 2014. 2, 5

[39] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3D: A Large Photo-realistic Dataset for Structured 3D Modeling. In *European Conference on Computer Vision*, 2020. 2, 5

[40] Chuhang Zou, Alex Colburn, Qi Shan, and Derek Hoiem. LayoutNet: Reconstructing the 3D Room Layout from a Single RGB Image. In *Computer Vision and Pattern Recognition*, pages 2051–2059, 2018. 1, 2, 5, 6

[41] Chuhang Zou, Jheng-Wei Su, Chi-Han Peng, Alex Colburn, Qi Shan, Peter Wonka, Hung-Kuo Chu, and Derek Hoiem. 3D Manhattan Room Layout Reconstruction from a Single 360 Image. arXiv:1910.04099, 2019. 2, 5, 6