Structured3D: A Large Photo-realistic Dataset for Structured 3D Modeling

Jia Zheng[1,2*†], Junfei Zhang[1*], Jing Li[2], Rui Tang[1],
Shenghua Gao[2,3], and Zihan Zhou[4]

[1] KooLab, Kujiale.com
[2] ShanghaiTech University
[3] Shanghai Engineering Research Center of Intelligent Vision and Imaging
[4] The Pennsylvania State University
https://structured3d-dataset.org

Abstract. Recently, there has been growing interest in developing learning-based methods to detect and utilize salient semi-global or global structures, such as junctions, lines, planes, cuboids, smooth surfaces, and all types of symmetries, for 3D scene modeling and understanding. However, the ground truth annotations are often obtained via human labor, which is particularly challenging and inefficient for such tasks due to the large number of 3D structure instances (*e.g.*, line segments) and other factors such as viewpoints and occlusions. In this paper, we present a new synthetic dataset, Structured3D, with the aim of providing large-scale photo-realistic images with rich 3D structure annotations for a wide spectrum of structured 3D modeling tasks. We take advantage of the availability of professional interior designs and automatically extract 3D structures from them. We generate high-quality images with an industry-leading rendering engine. We use our synthetic dataset in combination with real images to train deep networks for room layout estimation and demonstrate improved performance on benchmark datasets.

Keywords: Dataset · 3D structure · Photo-realistic rendering

1 Introduction

Inferring 3D information from 2D sensory data such as images and videos has long been a central research topic in computer vision. Conventional approach to building 3D models typically relies on detecting, matching, and triangulating local image features (*e.g.*, patches, superpixels, edges, and SIFT features). Although significant progress has been made over the past decades, these methods still suffer from some fundamental problems. In particular, local feature detection is sensitive to a large number of factors such as scene appearance (*e.g.*, textureless areas and repetitive patterns), lighting conditions, and occlusions. Further,

*: Equal contribution.
†: The work was partially done when Jia Zheng interned at KooLab, Kujiale.com.

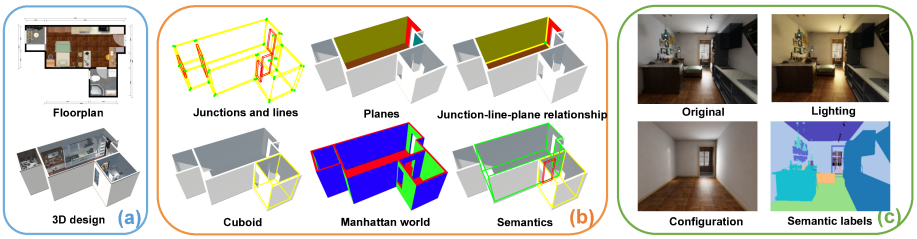


Fig. 1. The Structured3D dataset. From a large collection of house designs **(a)** created by professional designers, we automatically extract a variety of ground truth 3D structure annotations **(b)** and generate photo-realistic 2D images **(c)**.

the noisy, point cloud-based 3D model often fails to meet the increasing demand for high-level 3D understanding in real-world applications.

When perceiving 3D scenes, humans are remarkably effective in using salient global structures such as lines, contours, planes, smooth surfaces, symmetries, and repetitive patterns. Thus, if a reconstruction algorithm can take advantage of such global information, it is natural to expect the algorithm to obtain more accurate results. Traditionally, however, it has been computationally challenging to reliably detect such global structures from noisy local image features. Recently, deep learning-based methods have shown promising results in detecting various forms of structure directly from the images, including lines [12,40], planes [19,35,16,36], cuboids [10], floorplans [17,18], room layouts [14,41,26], abstracted 3D shapes [28,32], and smooth surfaces [11].

With the fast development of deep learning methods comes the need for large amounts of accurately annotated data. In order to train the proposed neural networks, most prior work collects their own sets of images and manually label the structure of interest in them. Such a strategy has several shortcomings. *First*, due to the tedious process of manually labeling and verifying all the structure instances (*e.g.*, line segments) in each image, existing datasets typically have limited sizes and scene diversity. And the annotations may also contain errors. *Second*, since each study primarily focuses on one type of structure, none of these datasets has multiple types of structure labeled. As a result, existing methods are unable to exploit relations between different types of structure (*e.g.*, lines and planes) as humans do for effective, efficient, and robust 3D reconstruction.

In this paper, we present a large synthetic dataset with rich annotations of 3D structure *and* photo-realistic 2D renderings of indoor man-made environments (Fig. 1). At the core of our dataset design is a unified representation of 3D structure which enables us to efficiently capture multiple types of 3D structure in the scene. Specifically, the proposed representation considers any structure as *relationship* among *geometric primitives*. For example, a "wireframe" structure encodes the incidence and intersection relationship between line segments, whereas a "cuboid" structure encodes the rotational and reflective symmetry relationship among its planar faces. With our "primitive + relationship" representation, one can easily derive the ground truth annotations for a wide variety of

Table 1. An overview of datasets with structure annotations. †: The actual numbers are not explicitly given and hard to estimate, because these datasets contain images from Internet (LSUN Room Layout, PanoContext), or multiple sources (LayoutNet). *: Dataset is unavailable online at the time of publication.

Datasets	#Scenes	#Rooms	#Frames	Annotated structure
PlaneRCNN [16]	-	-	100,000	planes
Wireframe [12]	-	-	5,462	wireframe (2D)
SceneCity 3D [40]	230	-	23,000	wireframe (3D)
SUN Primitive [34]	-	-	785	cuboids, other primitives
LSUN Room Layout [39]	-	n/a†	5,394	cuboid layout
PanoContext [37]	-	n/a†	500 (pano)	cuboid layout
LayoutNet [41]	-	n/a†	1,071 (pano)	cuboid layout
MatterportLayout* [42]	-	n/a†	2,295 (RGB-D pano)	Manhattan layout
Raster-to-Vector [17]	870	-	-	floorplan
Structured3D	3,500	21,835	196,515	"primitive + relationship"

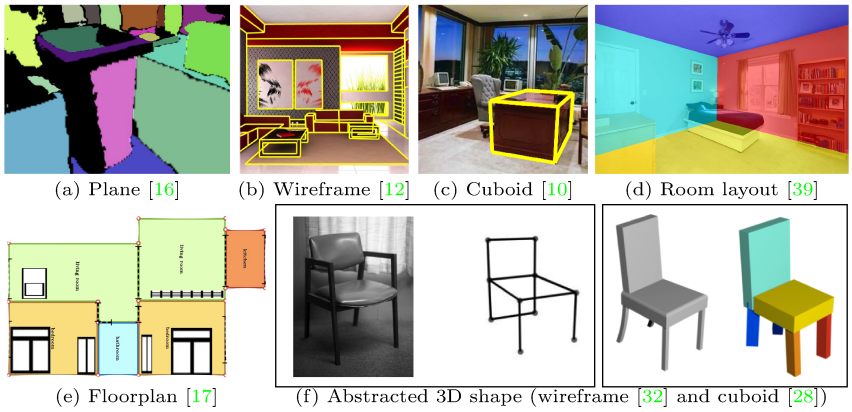
semi-global and global structures (*e.g.*, lines, wireframes, planes, regular shapes, floorplans, and room layouts), and also exploit their relations in future data-driven approaches (*e.g.*, the wireframe formed by intersecting planar surfaces in the scene).

To create a large-scale dataset with the aim of facilitating research on data-driven methods for structured 3D scene understanding, we leverage the availability of professional interior designs and millions of production-level 3D object models – all coming with fine geometric details and high-resolution textures (Fig. 1(a)). We first use computer programs to automatically extract information about 3D structure from the original house design files. As shown in Fig. 1(b), our dataset contains rich annotations of 3D room structure including a variety of geometric primitives and relationships. To further generate photo-realistic 2D images (Fig. 1(c)), we utilize industry-leading rendering engines to model the lighting conditions. Currently, our dataset consists of more than 196k images of 21,835 rooms in 3,500 scenes (*i.e.*, houses).

To showcase the usefulness and uniqueness of the proposed Structured3D dataset, we train deep networks for room layout estimation on a subset of the dataset. We show that the models trained on both synthetic and real data outperform the models trained on real data only. Further, following the spirit of [27,8], we show how multi-modal annotations in our dataset can benefit domain adaptation tasks.

In summary, the **main contributions** of this paper are:

- We create the Structured3D dataset, which contains rich ground truth 3D structure annotations of 21,835 rooms in 3,500 scenes, and more than 196k photo-realistic 2D renderings of the rooms.
- We introduce a unified "primitive + relationship" representation. This representation enables us to efficiently capture a wide variety of semi-global or global 3D structures and their mutual relationships.



(a) Plane [16] (b) Wireframe [12] (c) Cuboid [10] (d) Room layout [39]

(e) Floorplan [17] (f) Abstracted 3D shape (wireframe [32] and cuboid [28])

Fig. 2. Example annotations of structure in existing datasets. The reference number indicates the paper from which the illustration is originally from.

– We verify the usefulness of our dataset by using it to train deep networks for room layout estimation and demonstrating improved performance on public benchmarks.

2 Related Work

Datasets. Table 1 summarizes existing datasets for structured 3D scene modeling. Additionally, [28,32] provide datasets with structured representations of single objects. We show example annotations in these datasets in Fig. 2. Note that ground truth annotations in most datasets are manually labeled. This is one main reason why all these datasets have limited size, *i.e.*, contain no more than a few thousand images. One exception is [16], which employs a multi-model fitting algorithm to automatically extract planes from 3D scans in the ScanNet dataset [9]. But such algorithms are sensitive to data noises and outliers, thus introduce errors in the annotations (Fig. 2(a)). Similar to our work, SceneCity 3D [40] also contains synthetic images with ground truth automatically extracted from CAD models. But the number of scenes is limited to 230. Further, none of these datasets has more than one type of structure labeled, although different types of structure often have strong relations among them. For example, from the wireframe in Fig. 2(b) humans can easily identify other types of structure such as planes and cuboids. Our new dataset sets to bridge the gap between what is needed to train machine learning models to achieve human-level holistic 3D scene understanding and what is being offered by existing datasets.

Note that our dataset is very different from other popular large-scale 3D datasets, such as NYU v2 [23], SUN RGB-D [24], 2D-3D-S [4,3], ScanNet [9], and Matterport3D [6], in which the ground truth 3D information is stored in the format of point clouds or meshes. These datasets lack ground truth annotations of semi-global or global structures. While it is theoretically possible to extract

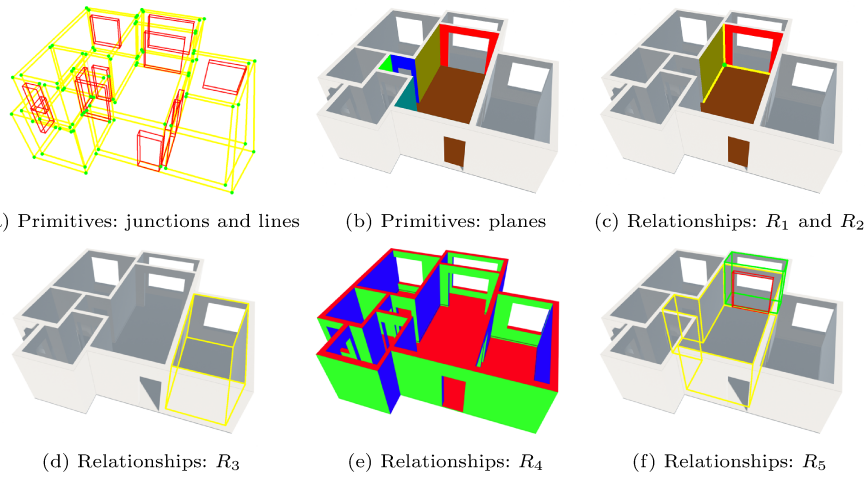
Table 2. Comparison of 3D scene datasets. †: Meshes are obtained by 3D reconstruction algorithm. Notations for applications: O (object detection), U (scene understanding), S (image synthesis), M (structured 3D modeling).

Datasets	Scene design type	3D annotation	2D rendering	Applications
NYU v2 [23]	Real	Raw RGB-D	Real images	O U
SUN RGB-D [24]	Real	Raw RGB-D	Real images	O U
2D-3D-S [4,3]	Real	Mesh†	Real images	O U
ScanNet [9]	Real	Mesh†	Real images	O U
Matterport3D [6]	Real	Mesh†	Real images	O U
SUNCG [25]	Amateur	Mesh	n/a	O U
SceneNet RGB-D [20]	Random	Mesh	Photo-realistic	O U
InteriorNet [15]	Professional	n/a	Photo-realistic	O U S
Structured3D	Professional	3D structures	Photo-realistic	O U S M

3D structure by applying structure detection algorithms to the point clouds or meshes (*e.g.*, extracting planes from ScanNet as did in [16]), the detection results are often noisy and even contain errors. In addition, for some types of structure like wireframes and room layouts, how to reliably detect them from raw sensor data remains an active research topic in computer vision.

In recent years, synthetic datasets have played an important role in the successful training of deep neural networks. Notable examples for indoor scene understanding include SUNCG [25], SceneNet RGB-D [20], and InteriorNet [15]. These datasets exceed real datasets in terms of scene diversity and frame numbers. But just like their real counterparts, these datasets lack ground truth structure annotations. Another issue with some synthetic datasets is the degree of realism in both the 3D models and the 2D renderings. [38] shows that physically-based rendering could boost the performance of various indoor scene understanding tasks. To ensure the quality of our dataset, we make use of 3D room models created by professional designers and the state-of-the-art industrial rendering engines. Table 2 summarizes the differences of 3D scene datasets.

Room layout estimation. Room layout estimation aims to reconstruct the enclosing structure of the indoor scene, consisting of walls, floor, and ceiling. Existing public datasets (*e.g.*, PanoContext [37] and LayoutNet [41]) assume a simple box-shaped layout. PanoContext [37] collects about 500 panoramas from the SUN360 dataset [33], LayoutNet [41] extends the layout annotations to include panoramas from 2D-3D-S [3]. Recently, MatterportLayout [42] collects 2,295 RGB-D panoramas from Matterport3D [6] and extends annotations to Manhattan layout. We note that all room layout in these real datasets is manually labeled by the human. Since the room structure may be occluded by furniture and other objects, the "ground truth" inferred by humans may not be consistent with the actual layout. In our dataset, all ground truth 3D annotations are automatically extracted from the original house design files.



(a) Primitives: junctions and lines (b) Primitives: planes (c) Relationships: R_1 and R_2

(d) Relationships: R_3 (e) Relationships: R_4 (f) Relationships: R_5

Fig. 3. The ground truth 3D structure annotations in our dataset are represented by primitives and relationships. **(a)**: Junctions and lines. **(b)**: Planes. We highlight the planes in a single room. **(c)**: Plane-line and line-junction relationships. We highlight a junction, the three lines intersecting at the junction, and the planes intersecting at each of the lines. **(d)**: Cuboids. We highlight one cuboid instance. **(e)**: Manhattan world. We use different colors to denote planes aligned with different directions. **(f)**: Semantic objects. We highlight a "room", a "balcony", and the "door" connecting them.

3 A Unified Representation of 3D Structure

The main goal of our dataset is to provide rich annotations of ground truth 3D structure. A naive way to do so is generating and storing different types of 3D annotations in the same format as existing works, like wireframes as in [12], planes as in [16], floorplans as in [17], and so on. But this leads to a lot of redundancy. For example, planes in man-made environments are often bounded by a number of line segments, which are part of the wireframe. Even worse, by representing wireframes and planes separately, the relationships between them are lost. In this paper, we present a unified representation in order to minimize redundancy while preserving mutual relationships. We show how the most common types of structure studied in the literature (*e.g.*, planes, cuboids, wireframes, room layouts, and floorplans) can be derived from our representation.

Our representation of the structure is largely inspired by the early work of Witkin and Tenenbaum [31], which characterizes structure as *"a shape, pattern, or configuration that replicates or continues with little or no change over an interval of space and time"*. Accordingly, to describe any structure, we need to specify: (i) what pattern is continuing or replicating (*e.g.*, a patch, an edge, or a texture descriptor), and (ii) the domain of its replication or continuation. In this paper, we call the former **primitives** and the latter **relationships**.

3.1 The "Primitive + Relationship" Representation

We now show how to describe a man-made environment using a unified representation. For ease of exposition, we assume all objects in the scene can be modeled by piece-wise planar surfaces. But our representation can be easily extended to more general surfaces. An illustration of our representation is shown in Fig. 3.
Primitives. Generally, a man-made scene has the following geometric primitives:

- **Planes P**: We model the scene as a collection of planes $\mathbf{P} = \{p_1, p_2, \ldots\}$. Each plane is described by its parameters $p = \{\mathbf{n}, d\}$, where $\mathbf{n}$ and d denote the surface normal and the distance to the origin, respectively.
- **Lines L**: When two planes intersect in the 3D space, a line is created. We use $\mathbf{L} = \{l_1, l_2, \ldots\}$ to represent the set of all 3D lines in the scene.
- **Junction points X**: When two lines meet in the 3D space, a junction point is formed. We use $\mathbf{X} = \{x_1, x_2, \ldots\}$ to represent the set of all junction points.

Relationships. Next, we define some common types of relationships between the geometric primitives:

- **Plane-line relationships (R_1)**: We use a matrix W_1 to record all incidence and intersection relationships between planes in $\mathbf{P}$ and lines in $\mathbf{L}$. Specifically, the ij-th entry of W_1 is 1 if l_i is on p_j, and 0 otherwise. Note that two planes are intersected at some line if and only if the corresponding entry in $W_1^T W_1$ is nonzero.
- **Line-point relationships (R_2)**: Similarly, we use a matrix W_2 to record all incidence and intersection relationships between lines in $\mathbf{L}$ and points in $\mathbf{X}$. Specifically, the mn-th entry of W_2 is 1 if x_m is on l_n, and 0 otherwise. Note that two lines are intersected at some junction if and only if the corresponding entry in $W_2^T W_2$ is nonzero.
- **Cuboids (R_3)**: A cuboid is a special arrangement of plane primitives with rotational and reflection symmetry along x-, y- and z-axes. The corresponding symmetry group is the dihedral group D_{2h}.
- **Manhattan world (R_4)**: This is a special type of 3D structure commonly used for indoor and outdoor scene modeling. It can be viewed as a *grouping* relationship, in which all the plane primitives can be grouped into three classes, $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$, $\mathbf{P} = \bigcup_{i=1}^{3} \mathbf{P}_i$. Further, each class is represented by a single normal vector $\mathbf{n}_i$, such that $\mathbf{n}_i^T \mathbf{n}_j = 0, i \neq j$.
- **Semantic objects (R_5)**: Semantic information is critical for many 3D computer vision tasks. It can be regarded as another type of *grouping* relationship, in which each semantic object instance corresponds to one or more primitives defined above. For example, each "wall", "ceiling", or "floor" instance is associated with one plane primitive; each "chair" instance is associated with a set of multiple plane primitives. Further, such a grouping is hierarchical. For example, we can further group one floor, one ceiling, and multiple walls to form a "living room" instance. And a "door" or a "window" is an opening which connects two rooms (or one room and the outer space).

Note that the relationships are not mutually exclusive, in the sense that a primitive can belong to multiple relationship instances of the same type or different types. For example, a plane primitive can be shared by two cuboids, and at the same time belong to one of the three classes in the Manhattan world model.

Discussion. The primitives and relationships we discussed above are just a few most common examples. They are by no means exhaustive. For example, our representation can be easily extended to include other primitives such as parametric surfaces. And besides cuboids, there are many other types of regular or symmetric shapes in man-made environments, where type corresponds to a different symmetry group.

Our representation of 3D structures is also related to the graph representations in semantic scene understanding [13,2,30]. As these graphs focus on semantics, geometry is represented in simplified manners by (i) 6D object poses and (ii) coarse, discrete spatial relations such as "supported by", "front", "back", and "adjacent". In contrast, our representation focuses on modeling the scene geometry using fine-grained primitives (*i.e.*, junctions, lines, and planes) and relationships (in terms of topology and regularities). Thus, it is highly complementary to the scene graphs in prior work. Intuitively, it can be used for geometric analysis and synthesis tasks, in a similar way as scene graphs are used for semantic scene understanding.

3.2 Relation to Existing Models

Given our representation which contains primitives $\mathcal{P} = \{\mathbf{P}, \mathbf{L}, \mathbf{X}\}$ and relationships $\mathcal{R} = \{R_1, R_2, \ldots\}$, we show how several types of 3D structure commonly studied in the literature can be derived from it. We again refer readers to Fig. 2 for illustrations of these structures.

Planes: A large volume of studies in the literature model the scene as a collection of 3D planes, where each plane is represented by its parameters and boundary. To generate such a model, we simply use the plane primitives $\mathbf{P}$. For each $p \in \mathbf{P}$, we further obtain its boundary by using matrix W_1 in R_1 to find all the lines in $\mathbf{L}$ that form an incidence relationship with p.

Wireframes: A wireframe consists of lines $\mathbf{L}$ and junction points $\mathbf{P}$, and their incidence and intersection relationships (R_2).

Cuboids: This model is same as R_3.

Manhattan layouts: A Manhattan room layout model includes a "room" as defined in R_5 which also satisfies the Manhattan world assumption (R_4).

Floorplans: A floorplan is a 2D vector representation that consists of a set of line segments and semantic labels (*e.g.*, room types). To obtain such a vector representation, we can identify all lines in $\mathbf{L}$ and junction points in $\mathbf{X}$ which lie on a "floor" (as defined in R_5). To further obtain the semantic room labels, we can project all "rooms", "doors", and "windows" (as defined in R_5) to this floor.

Abstracted 3D shapes: In addition to room structures, our representation can also be applied to individual 3D object models to create abstractions in the form of wireframes or cuboids, as described above.

(a) (b)

Fig. 4. Comparison of 3D house designs. **(a)**: The 3D models in our database are created by professional designers using high-quality furniture models from world-leading manufacturers. Most designs are being used in real-world production. **(b)**: The 3D models in SUNCG dataset [25] are created using Planner 5D [1], an online tool for amateur interior design.

4 The Structured3D Dataset

Our unified representation enables us to encode a rich set of geometric primitives and relationships for structured 3D modeling. With this representation, our ultimate goal is to build a dataset that can be used to train machines to achieve the human-level understanding of the 3D environment.

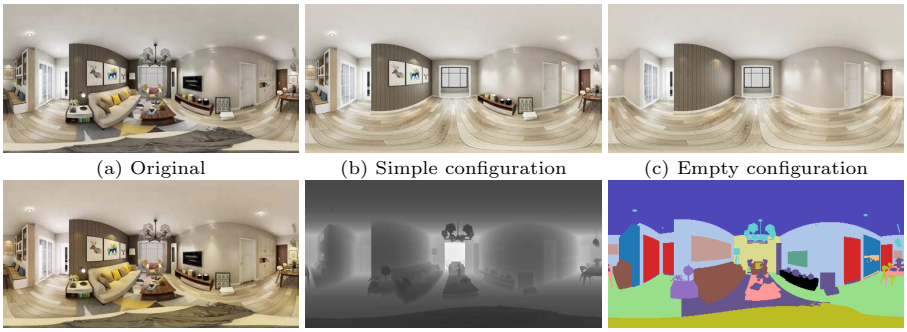
As a first step towards this goal, in this section, we describe our ongoing effort to create a large-scale dataset of indoor scenes which include (i) ground truth 3D structure annotations of the scene and (ii) realistic 2D renderings of the scene. Note that in this work we focus on extracting ground truth annotations on the room structure only. We plan to extend our dataset to include 3D structure annotations of individual furniture models in the future.

In the following, we describe our general procedure to create the dataset. We refer readers to the supplementary materials for additional details, including dataset statistics and example annotations.

4.1 Extraction of Structured 3D Models

To extract a "primitive + relationship" scene representation, we utilize a large database of house designs hand-crafted by professional designers. An example design is shown in Fig. 4(a). All information of the design is stored in an industry-standard format in the database so that specifications about the geometry (*e.g.*, the precise size of each wall), textures and materials, and functions (*e.g.*, which room the wall belongs to) of all objects can be easily retrieved.

From the database, we have selected 3,500 house designs with 21,835 rooms. We created a computer program to automatically extract all the geometric primitives associated with the room structure, which consists of the ceiling, floor, walls, and openings (doors and windows). Given the precise measurements and associated information of these entities, it is straightforward to generate all planes, lines, and junctions, as well as their relationships (R_1 and R_2).



(a) Original (b) Simple configuration (c) Empty configuration

(d) Lighting (e) Depth (f) Semantic labels

Fig. 5. Examples of our rendered panoramic images.

Since the measurements are highly accurate and noise-free, other types of relationship such a Manhattan world (R_3) and cuboids (R_4) can also be easily obtained by clustering the primitives, followed by a geometric verification process. Finally, to include semantic information (R_5) into our representation, we map the relevant labels provided by the professional designers to the geometric primitives in our representation. Fig. 3 shows examples of the extracted geometric primitives and relationships.

4.2 Photo-realistic 2D Rendering

To ensure the quality of our 2D renderings, our rendering engine is developed in collaboration with a company specialized in interior design rendering. Our engine uses a well-known ray-tracing method [21], a Monte Carlo approach to approximating realistic Global Illumination (GI), for RGB rendering. The other ground truth images are obtained by a customized path-tracer renderer on top of Intel Embree [29], an open-source collection of ray-tracing kernels for x86 CPUs.

Each room is manually created by professional designers with over one million CAD models of furniture from world-leading manufacturers. These high-resolution furniture models are measured in real-world dimensions and being used in real production. A default lighting setup is also provided. Fig. 4 compares the 3D models in our database with those in SUNCG [25], which are created using Planner 5D [1], an online tool for amateur interior design.

At the time of rendering, a panoramic or pin-hole camera is placed at random locations not occupied by objects in the room. We use 512×1024 resolution for panoramas and 720×1280 for perspective images. Fig. 5 shows example panoramas rendered by our engine. For each room, we generate different configurations (full, simple, and empty) by removing some or all the furniture. We also modify the lighting setup to generate images with different temperatures. For each image, our dataset also includes the depth map and semantic mask. Fig. 6 illustrates the degree of photo-realism of our dataset, where we compare the rendered images with photos of real decoration guided by the design.

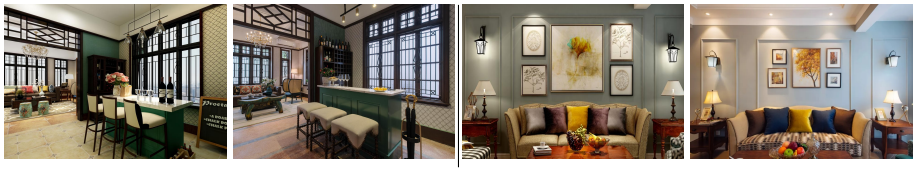


Fig. 6. Photo-realistic rendering vs. real-world decoration. The first and third columns are rendered images.

4.3 Use Cases

Due to the unique characteristics of our dataset, we envision it contributing to computer vision research in terms of both methodology and applications.

Methodology. As our dataset contains multiple types of 3D structure annotations as well as ground truth labels (*e.g.*, semantic maps, depth maps, and 3D object bounding boxes), it enables researchers to design novel multi-modal or multi-task approaches for a variety of vision tasks. As an example, we show in Section 5 that, by leveraging multi-modal annotations in our dataset, we can boost the performance of existing room layout estimation methods in the domain adaptation framework.

Applications. Our dataset also facilitates research on a number of problems and applications. For example, as shown in Table 1, all publicly available datasets for room layout estimation are limited to simple cuboid rooms. Our dataset is the first to provide the general (non-cuboid) room layout annotations. As another example, existing datasets for floorplan reconstruction [18,7] contain about 100-150 scenes, whereas our dataset includes 3,500 scenes.

Another major line of research that would benefit from our dataset is image synthesis. With a photo-realistic rendering engine, we are able to generate images given *any* scene configurations and viewpoints. These images may be used as ground truth for tasks including image inpainting (*e.g.*, completing an image when certain furniture is removed) and novel view synthesis.

Finally, we would like to emphasize the potential of our dataset in terms of extension capabilities. As we mentioned before, the unified representation enables us to include many other types of structure in the dataset. As for 2D rendering, depending on the application, we can easily simulate different effects such as lighting conditions, fisheye and novel camera designs, motion blur, and imaging noise. Furthermore, the dataset may be extended to include videos for applications such as visual SLAM [5].

5 Experiments

5.1 Experiment Setup

To demonstrate the benefits of our dataset, we use it to train deep neural networks for room layout estimation, an important task in structured 3D modeling.

Table 3. Room layout statistics. †: MatterportLayout is the only other dataset with non-cuboid layout annotations, but is unavailable at the time of publication.

#Corners	4	5	6	7	8	9	10+	Total
MatterportLayout†	1211	0	501	0	309	0	274	2295
Structured3D	13743	52	3727	30	1575	17	2691	21835

Real dataset. We use the same dataset as LayoutNet [41]. The dataset consists of images from PanoContext [37] and 2D-3D-S [3], including 818 training images, 79 validation images, and 166 test images. Note that both datasets only provide cuboid layout annotations.

Our Structured3D dataset. In this experiment, we use a subset of panoramas with the original lighting and full configuration. Each panorama corresponds to a different room in our dataset. We show statistics of different room layouts in our dataset in Table 3. Since the current real dataset only contains cuboid layout annotations (*i.e.*, 4 corners), we choose 12k panoramic images with the cuboid layout in our dataset. We split the images into 10k for training, 1k for validation, and 1k for testing.

Evaluation metrics. Following [41,26], we adopt three standard metrics: (i) 3D IoU: intersection over union between predicted 3D layout and the ground truth, (ii) Corner Error (CE): normalized ℓ_2 distance between predicted corner and ground truth, and (iii) Pixel Error (PE): pixel-wise error between predicted plane classes and ground truth.

Baselines. We choose two recent CNN-based approaches, LayoutNet [41,42][1] and HorizonNet [26][2], based on their performance and source code availability. LayoutNet uses a CNN to predict a corner probability map and a boundary map from the panorama and vanishing lines, then optimizes the layout parameters based on network predictions. HorizonNet represents room layout as three 1D vectors, *i.e.*, boundary positions of floor-wall, and ceiling-wall, and the existence of wall-wall boundary. It trains CNNs to directly predict the three 1D vectors. In this paper, we follow the default training setting of the respective methods. For specific training procedures, please refer to the supplementary materials.

5.2 Experiment Results

Augmenting real datasets. In this experiment, we train LayoutNet and HorizonNet in four different manners: (i) training only on our synthetic dataset ("**s**"), (ii) training only on the real dataset ("**r**"), (iii) training on the synthetic and real dataset with Balanced Gradient Contribution (BGC) [22] ("**s** + **r**"), and (iv) pre-training on our synthetic dataset, then fine-tuning on the real dataset ("**s** → **r**"). We adopt the training set of LayoutNet as the real dataset in this experiment. The results are shown in Table 4. As one can see, augmenting real

[1] https://github.com/zouchuhang/LayoutNetv2
[2] https://github.com/sunset1995/HorizonNet

Table 4. Quantitative evaluation under different training schemes. The best and the second best results are boldfaced and underlined, respectively.

Methods	Config.	PanoContext			2D-3D-S		
		3D IoU (%) ↑	CE (%) ↓	PE (%) ↓	3D IoU (%) ↑	CE (%) ↓	PE (%) ↓
LayoutNet [41,42]	s	75.64	1.31	4.10	57.18	2.28	7.55
	r	84.15	0.64	<u>1.80</u>	83.39	0.74	2.39
	s + r	**84.96**	**0.61**	**1.75**	<u>83.66</u>	<u>0.71</u>	<u>2.31</u>
	s → r	<u>84.77</u>	<u>0.63</u>	1.89	**84.04**	**0.66**	**2.08**
HorizonNet [26]	s	75.89	1.13	3.15	67.66	1.18	3.94
	r	83.42	0.73	2.09	84.33	0.64	2.04
	s + r	<u>84.45</u>	<u>0.70</u>	<u>1.89</u>	<u>84.36</u>	**0.59**	<u>1.90</u>
	s → r	**85.27**	**0.66**	**1.86**	**86.01**	<u>0.61</u>	**1.84**

Table 5. Quantitative evaluation using varying synthetic data size in pre-training. The best and the second best results are boldfaced and underlined, respectively.

Methods	Synthetic Data Size	PanoContext			2D-3D-S		
		3D IoU (%) ↑	CE (%) ↓	PE (%) ↓	3D IoU (%) ↑	CE (%) ↓	PE (%) ↓
LayoutNet [41,42]	1k	83.81	<u>0.66</u>	1.99	83.57	0.72	2.31
	5k	<u>84.47</u>	0.67	<u>1.97</u>	**84.55**	<u>0.69</u>	<u>2.21</u>
	10k	**84.77**	**0.63**	**1.89**	<u>84.04</u>	**0.66**	**2.08**
HorizonNet [26]	1k	83.77	0.74	2.11	85.19	<u>0.63</u>	2.01
	5k	<u>84.13</u>	<u>0.73</u>	<u>2.07</u>	**86.35**	**0.61**	<u>1.87</u>
	10k	**85.27**	**0.66**	**1.86**	<u>86.01</u>	**0.61**	**1.84**

datasets with our synthetic data boosts the performance of both networks. We refer readers to supplementary materials for more qualitative results.

Performance vs. synthetic data size. We further study the relationship between the number of synthetic images used in pre-training and the accuracy on the real dataset. We sample 1k, 5k and 10k synthetic images for pre-training, then fine-tune the model on the real dataset. The results are shown in Table 5. As expected, using more synthetic data generally improves the performance.

Domain adaptation. Domain adaptation techniques (*e.g.*, [27]) have been shown to be effective in bridging the performance gap when directly applying models learned on synthetic data to real environments. In this experiment, we do not assume access to ground truth layout labels in the real dataset. We adopt LayoutNet as the task network and use PanoContext and 2D-3D-S separately. We apply a discriminator network to align the output features of the LayoutNet for two domains. Inspired by [8], we further leverage multi-modal annotations in our dataset by adding another decoder branch to the LayoutNet for depth prediction. We concatenate the boundary, corner, and depth predictions as the input of the discriminator network. The results are shown in the Table 6. By incorporating additional information, *i.e.*, depth map, we further boost the performance on both datasets. This illustrates the advantage of including multiple types of ground truth in our dataset.

Limitation of real datasets. Due to human errors, the annotation in real datasets is not always consistent with the actual room layout. In the left image of Fig. 7, the room is a non-cuboid layout, but the ground truth layout is labeled

Table 6. Domain adaptation results. NA: non-adaptive baseline. +DA: align layout estimation output. +Depth: align both layout estimation and depth outputs. Real: train in the target domain.

Methods	PanoContext			2D-3D-S		
	3D IoU (%) ↑	CE (%) ↓	PE (%) ↓	3D IoU (%) ↑	CE (%) ↓	PE (%) ↓
NA	75.64	1.31	4.10	57.18	2.28	7.55
+DA	76.91	1.19	3.64	70.08	1.36	4.66
+Depth	78.34	1.03	2.99	72.99	1.24	3.60
Real	81.76	0.95	2.58	81.82	0.96	3.13

Fig. 7. Limitation of real datasets. **Left**: PanoContext dataset. **Right**: 2D-3D-S dataset. Blue lines are ground truth layout and green lines are predictions.

as cuboid shape. In the right image, the front wall is not labeled as ground truth. These examples illustrate the limitation of using real datasets as benchmarks. We avoid such errors in our dataset by automatically generating ground truth from the original design files.

6 Conclusion

In this paper, we present Structured3D, a large synthetic dataset with rich ground truth 3D structure annotations of 21,835 rooms and more than 196k photo-realistic 2D renderings. Among many potential use cases of our dataset, we further demonstrate its benefit in augmenting real data and facilitating domain adaptation for the room layout estimation task.

We view this work as an important and exciting step towards building intelligent machines which can achieve human-level holistic 3D scene understanding. In the future, we will continue to add more 3D structure annotations of the scenes and objects to the dataset, and explore novel ways to use the dataset to advance techniques for structured 3D modeling and understanding.

Acknowledgement. We would like to thank Kujiale.com for providing the database of house designs and the rendering engine. We especially thank Qing Ye and Qi Wu from Kujiale.com for the help on the data rendering. This work was partially supported by the National Key R&D Program of China (#2018AAA0100704) and the National Science Foundation of China (#61932020). Zihan Zhou was supported by NSF award #1815491.

References

1. Planner 5d. https://planner5d.com 9, 10
2. Armeni, I., He, Z.Y., Gwak, J., Zamir, A.R., Fischer, M., Malik, J., Savarese, S.: 3d scene graph: A structure for unified semantics, 3d space, and camera. In: ICCV. pp. 5664–5673 (2019) 8
3. Armeni, I., Sax, A., Zamir, A.R., Savarese, S.: Joint 2d-3d-semantic data for indoor scene understanding. CoRR **abs/1702.01105** (2017) 4, 5, 12
4. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I.K., Fischer, M., Savarese, S.: 3d semantic parsing of large-scale indoor spaces. In: CVPR. pp. 1534–1543 (2016) 4, 5
5. Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I.D., Leonard, J.J.: Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Trans. Robotics **32**(6), 1309–1332 (2016) 11
6. Chang, A.X., Dai, A., Funkhouser, T.A., Halber, M., Nießner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from RGB-D data in indoor environments. In: 3DV. pp. 667–676 (2017) 4, 5
7. Chen, J., Liu, C., Wu, J., Furukawa, Y.: Floor-sp: Inverse CAD for floorplans by sequential room-wise shortest path. In: ICCV. pp. 2661–2670 (2019) 11
8. Chen, Y., Li, W., Chen, X., Van Gool, L.: Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In: CVPR. pp. 1841–1850 (2019) 3, 13
9. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR. pp. 5828–5839 (2017) 4, 5
10. Dwibedi, D., Malisiewicz, T., Badrinarayanan, V., Rabinovich, A.: Deep cuboid detection: Beyond 2d bounding boxes. CoRR **abs/1611.10010** (2016) 2, 4
11. Groueix, T., Fisher, M., Kim, V.G., Russell, B., Aubry, M.: A papier-mâché approach to learning 3d surface generation. In: CVPR. pp. 216–224 (2018) 2
12. Huang, K., Wang, Y., Zhou, Z., Ding, T., Gao, S., Ma, Y.: Learning to parse wireframes in images of man-made environments. In: CVPR. pp. 626–635 (2018) 2, 3, 4, 6
13. Huang, S., Qi, S., Zhu, Y., Xiao, Y., Xu, Y., Zhu, S.C.: Holistic 3d scene parsing and reconstruction from a single rgb image. In: ECCV. pp. 194–211 (2018) 8
14. Lee, C., Badrinarayanan, V., Malisiewicz, T., Rabinovich, A.: Roomnet: End-to-end room layout estimation. In: ICCV. pp. 4875–4884 (2017) 2
15. Li, W., Saeedi, S., McCormac, J., Clark, R., Tzoumanikas, D., Ye, Q., Huang, Y., Tang, R., Leutenegger, S.: Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. In: BMVC. p. 77 (2018) 5
16. Liu, C., Kim, K., Gu, J., Furukawa, Y., Kautz, J.: Planercnn: 3d plane detection and reconstruction from a single image. In: CVPR. pp. 4450–4459 (2019) 2, 3, 4, 5, 6
17. Liu, C., Wu, J., Kohli, P., Furukawa, Y.: Raster-to-vector: Revisiting floorplan transformation. In: ICCV. pp. 2214–2222 (2017) 2, 3, 4, 6
18. Liu, C., Wu, J., Furukawa, Y.: Floornet: A unified framework for floorplan reconstruction from 3d scans. In: ECCV. pp. 203–219 (2018) 2, 11
19. Liu, C., Yang, J., Ceylan, D., Yumer, E., Furukawa, Y.: Planenet: Piece-wise planar reconstruction from a single rgb image. In: CVPR. pp. 2579–2588 (2018) 2

20. McCormac, J., Handa, A., Leutenegger, S., Davison, A.J.: Scenenet RGB-D: can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? In: ICCV. pp. 2697–2706 (2017) 5
21. Purcell, T.J., Buck, I., Mark, W.R., Hanrahan, P.: Ray tracing on programmable graphics hardware. ACM Trans. Graph. **21**(3), 703–712 (2002) 10
22. Ros, G., Stent, S., Alcantarilla, P.F., Watanabe, T.: Training constrained deconvolutional networks for road scene semantic segmentation. CoRR **abs/1604.01545** (2016) 12
23. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: ECCV. pp. 746–760 (2012) 4, 5
24. Song, S., Lichtenberg, S.P., Xiao, J.: SUN RGB-D: A RGB-D scene understanding benchmark suite. In: CVPR. pp. 567–576 (2015) 4, 5
25. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.A.: Semantic scene completion from a single depth image. In: CVPR. pp. 1746–1754 (2017) 5, 9, 10
26. Sun, C., Hsiao, C.W., Sun, M., Chen, H.T.: Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In: CVPR. pp. 1047–1056 (2019) 2, 12, 13
27. Tsai, Y.H., Hung, W.C., Schulter, S., Sohn, K., Yang, M.H., Chandraker, M.: Learning to adapt structured output space for semantic segmentation. In: CVPR. pp. 7472–7481 (2018) 3, 13
28. Tulsiani, S., Su, H., Guibas, L.J., Efros, A.A., Malik, J.: Learning shape abstractions by assembling volumetric primitives. In: CVPR. pp. 2635–2643 (2017) 2, 4
29. Wald, I., Woop, S., Benthin, C., Johnson, G.S., Ernst, M.: Embree: a kernel framework for efficient CPU ray tracing. ACM Trans. Graph. **33**(4), 143:1–143:8 (2014) 10
30. Wang, K., Lin, Y.A., Weissmann, B., Savva, M., Chang, A.X., Ritchie, D.: Planit: Planning and instantiating indoor scenes with relation graph and spatial prior networks. ACM Trans. Graph. **38**(4) (2019) 8
31. Witkin, A.P., Tenenbaum, J.M.: On the role of structure in vision. In: Beck, J., Hope, B., Rosenfeld, A. (eds.) Human and Machine Vision, pp. 481–543. Academic Press (1983) 6
32. Wu, J., Xue, T., Lim, J.J., Tian, Y., Tenenbaum, J.B., Torralba, A., Freeman, W.T.: 3d interpreter networks for viewer-centered wireframe modeling. IJCV **126**(9), 1009–1026 (2018) 2, 4
33. Xiao, J., Ehinger, K.A., Oliva, A., Torralba, A.: Recognizing scene viewpoint using panoramic place representation. In: CVPR. pp. 2695–2702 (2012) 5
34. Xiao, J., Russell, B., Torralba, A.: Localizing 3d cuboids in single-view images. In: NeurIPS. pp. 746–754 (2012) 3
35. Yang, F., Zhou, Z.: Recovering 3d planes from a single image via convolutional neural networks. In: ECCV. pp. 87–103 (2018) 2
36. Yu, Z., Zheng, J., Lian, D., Zhou, Z., Gao, S.: Single-image piece-wise planar 3d reconstruction via associative embedding. In: CVPR. pp. 1029–1037 (2019) 2
37. Zhang, Y., Song, S., Tan, P., Xiao, J.: Panocontext: A whole-room 3d context model for panoramic scene understanding. In: ECCV. pp. 668–686 (2014) 3, 5, 12
38. Zhang, Y., Song, S., Yumer, E., Savva, M., Lee, J.Y., Jin, H., Funkhouser, T.: Physically-based rendering for indoor scene understanding using convolutional neural networks. In: CVPR. pp. 5287–5295 (2017) 5
39. Zhang, Y., Yu, F., Song, S., Xu, P., Seff, A., Xiao, J.: Large-scale scene understanding challenge: Room layout estimation (2016) 3, 4

40. Zhou, Y., Qi, H., Zhai, S., Sun, Q., Chen, Z., Wei, L.Y., Ma, Y.: Learning to reconstruct 3d manhattan wireframes from a single image. In: ICCV. pp. 7698–7707 (2019) 2, 3, 4
41. Zou, C., Colburn, A., Shan, Q., Hoiem, D.: Layoutnet: Reconstructing the 3d room layout from a single RGB image. In: CVPR. pp. 2051–2059 (2018) 2, 3, 5, 12, 13
42. Zou, C., Su, J., Peng, C., Colburn, A., Shan, Q., Wonka, P., Chu, H., Hoiem, D.: 3d manhattan room layout reconstruction from a single 360 image. CoRR **abs/1910.04099** (2019) 3, 5, 12, 13