Room Envelopes: A Synthetic Dataset for Indoor Layout Reconstruction from Images

Sam Bahrami and Dylan Campbell

The Australian National University, Canberra, Australia
{sam.bahrami,dylan.campbell}@anu.edu.au

Abstract. Modern scene reconstruction methods are able to accurately recover 3D surfaces that are visible in one or more images. However, this leads to incomplete reconstructions, missing all occluded surfaces. While much progress has been made on reconstructing entire objects given partial observations using generative models, the structural elements of a scene, like the walls, floors and ceilings, have received less attention. We argue that these scene elements should be relatively easy to predict, since they are typically planar, repetitive and simple, and so less costly approaches may be suitable. In this work, we present a synthetic dataset—Room Envelopes[1]—that facilitates progress on this task by providing a set of RGB images and two associated pointmaps for each image: one capturing the visible surface and one capturing the first surface once fittings and fixtures are removed, that is, the structural layout. As we show, this enables direct supervision for feed-forward monocular geometry estimators that predict both the first visible surface and the first layout surface. This confers an understanding of the scene's extent, as well as the shape and location of its objects.

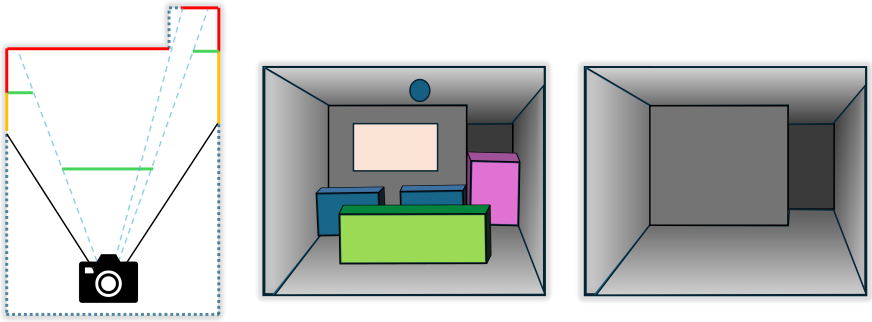
Keywords: 3D Scene Reconstruction · Indoor Scene Understanding · Layout Estimation · Synthetic Data

1 Introduction

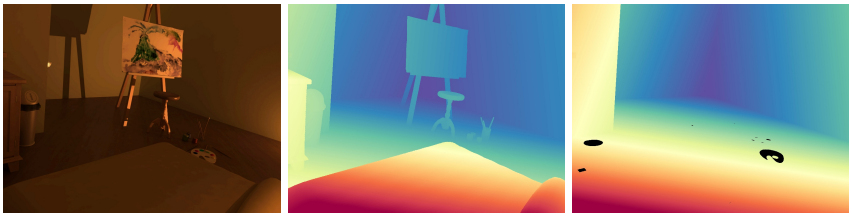
Indoor scene reconstruction remains a challenging problem, particularly when attempting to understand the complete spatial structure of indoor environments from limited visual information. While recent advances in monocular depth estimation and 3D reconstruction have shown promising results, existing approaches often struggle with occluded regions and fail to provide comprehensive geometric understanding of indoor scenes outside of the first visible surface.

Current models used for indoor scene reconstruction typically use depth images [25,10] or layered depth representations [11] for training, but these formats have inherent limitations. Depth maps only capture the first visible surface, losing information about the underlying room structure that might be occluded.

[1] Project website: https://sambahrami.github.io/room_envelopes



(a) Floor Plan View (b) Visible Surface (c) Layout Surface



(d) Original RGB Image (e) Visible Surface Depth (f) Layout Surface Depth

Fig. 1. Room Envelopes dataset overview. Our synthetic dataset provides dual pointmap representations for indoor scene reconstruction. (a) Overhead floor plan showing the view from a camera capturing layout areas in red, the first visible surface in green, and parts shared in both in orange. (b) The visible surface capturing all directly visible geometry including furniture and objects. (c) The layout surface showing structural elements (walls, floors, ceilings, windows, doors) as they would appear without occlusion. This dual representation enables direct supervision for layout reconstruction in occluded regions. (d–f) Example data from our dataset. (d) The original RGB image from Hypersim. (e) The visible surface depth capturing all visible surfaces including furniture and objects (f) The first layout surface depth showing only structural elements (walls, floors, ceiling, windows, doors).

Layered depth approaches attempt to address occlusion by representing multiple depth layers per pixel, but surfaces that are spatially continuous in 3D space often become fragmented across different depth layers, creating artificial discontinuities and requiring methods that perform layer assignment as well as geometry estimation.

For a given viewpoint, we refer to the union of the first visible surface and the first structural layout surface as the *envelope* of the view, as illustrated in Fig. 1. The structural layout comprises structural elements like walls, floors, ceilings, windows and doors. This is essentially the room with fittings (removable items like furniture, appliances, curtains, carpets) and fixtures (attached items like sinks, baths, lights) removed, which largely coincides with the concept of layout in existing layout estimation works [7,14,15]. Understanding this structural layout is useful for many applications including robotic navigation, augmented

reality, and architectural analysis. However, in typical indoor scenes, this layout is often partially occluded by furniture, making it difficult for models to learn robust representations of the underlying room geometry.

Feed-forward 3D reconstruction models face particular challenges at occlusion edges, where they tend to predict averaged depth values rather than making decisive geometric predictions [21,25,10]. This occurs because these models optimise a reconstruction loss by predicting the statistical mean in ambiguous regions, resulting in blurry or noisy outputs at object boundaries. Room layouts however are predominantly continuous and planar, making them well-suited for feed-forward prediction. By focusing on layout estimation, models can leverage the geometric regularity of architectural structures to produce sharp, accurate reconstructions even in partially occluded regions.

To address these limitations, we introduce Room Envelopes, a synthetic dataset that explicitly provides two complementary pointmaps for each RGB image: (1) one representing the first visible surface, capturing all directly visible surfaces including fittings and fixtures, and (2) one representing the first layout surface, capturing the scene's structural elements as they would appear from the same viewpoint if the fittings and fixtures were removed. This dual representation provides several advantages over existing dataset formats. First, the first layout surface provides structural clarity by offering unoccluded views of room boundaries, enabling models to learn robust representations of indoor spatial structure. Second, it facilitates the development of scene reconstructors that have less ambiguity compared to layered depth approaches, eliminating uncertainty about layer assignment. Finally, it allows models to be trained with direct supervision on room layout estimation, rather than relying on indirect multi-view photometric losses.

Our dataset is constructed by combining multiple viewpoints from high-quality synthetic indoor scenes, similar to how real-world datasets are collected through multi-view reconstruction approaches. This multi-view aggregation allows us to build a complete point cloud representation of indoor environments, from which we isolate structural elements and re-render layout surfaces unoccluded. We process these pointclouds into pointmap representations which maintain spatial correspondence with the original RGB images, enable pixel-aligned supervision, and preserve the natural image structure that existing vision neural network architectures are optimised for. This approach produces a dataset that is fundamentally equivalent to what could be achieved with the best possible real-world data collection tools for this task, with the advantage of perfect semantic labelling, perfect camera information, and controlled lighting conditions.

We demonstrate the utility of this dataset by training a model for room layout estimation that can predict the surface envelope of an indoor scene from a single RGB image. The main contributions of this work are

1. Room Envelopes, an indoor synthetic dataset providing an image, a visible surface pointmap, and a layout surface pointmap for each camera pose; and
2. a feed-forward scene reconstruction model demonstrating effective room layout estimation using this dataset.

Table 1. Comparison of dataset features across indoor scene datasets. We show which datasets provide camera poses, point clouds, depth maps (first depth and layout depth), mesh data, semantic labels, and surface normals. Room Envelope is the only dataset that provides both first layer and layout depth representations for comprehensive 3D reconstruction training.

Datasets	#Scenes	Type	Camera Poses	Point Cloud	First Depth	Layout Depth	Mesh	Semantic Labels	Normals
Matterport3D [2]	90	Real	✓	✗	✓	✗	✓	✓	✗
ScanNet++ [29]	1,006	Real	✓	✓	✓	✗	✓	✓	✗
ScanNet [3]	1,513	Real	✓	✗	✓	✗	✓	✓	✗
ARKitScenes [1]	1,661	Real	✓	✓	✓	✗	✓	✗	✗
RealEstate10K [34]	74K	Real	✓	✗	✗	✗	✗	✗	✗
MegaSynth [6]	700K	Synthetic	✓	✗	✓	✗	✗	✗	✗
Structured3D [33]	22K	Synthetic	✓	✗	✓	✗	✗	✓	✓
3D-Front [4]	19K	Synthetic	✗	✗	✗	✗	✓	✗	✗
HyperSim [18]	461	Synthetic	✓	✗	✓	✗	✓	✓	✓
Room Envelopes (Ours)	461	Synthetic	✓	✓	✓	✓	✗	✓	✓

2 Related Work

2.1 Indoor Scene Datasets

Indoor scene understanding relies on both real-world and synthetic datasets, each with distinct advantages and limitations. We provide a detailed comparison of Room Envelopes with existing indoor datasets in Table 1. Real-world datasets [34,3,29,2] face significant collection challenges that limit their scale and quality. Ground truth acquisition is difficult due to depth sensor limitations, including restricted resolution, errors with reflective surfaces, artifacts at object boundaries, and these datasets require extensive manual effort for data capture and post-processing.

Synthetic datasets provide perfect ground truth at scale with reduced manual effort. Recent procedural generation advances enable automatic scene layout generation [17], but often lack natural spatial arrangements that characterise human-designed spaces. Artist-designed synthetic environments offer a compelling middle ground [33,6,18,4], however visual realism remains a key consideration. HyperSim [18] uses V-Ray rendering to achieve photorealistic imagery per view. Room Envelopes builds on this foundation while introducing layout surface maps for each view.

2.2 Scene and Layout Reconstruction

A modern approach to scene reconstruction estimates pixel-aligned pointmaps from images, where each pixel in the output pointmap represents a 3D point in the scene [30]. The pioneering feed-forward pointmap reconstruction method DUSt3R [27] takes pairs of unposed views and estimates pointmaps in the same coordinate system, encoding their geometry and pose. This pairwise representation can be extended to estimate geometry for large scenes with many views, including as part of a SLAM system [16]. The follow-up work MASt3R [10]

improved upon this approach by introducing local feature matching and modifying the training methodology to output metric-scale representations. MoGe [25] extended this paradigm to single-view pointmap estimation, with MoGe v2 [26] providing metric-scale output. LaRI [11] adapted the training concept from MoGe to estimate layered pointmap representations, attempting to predict the depth of occluded surfaces with a feed-forward model. VGGT [23] predicts comprehensive 3D scene attributes (such as pointmaps, camera poses, and depth maps) from single or multiple images of a scene without post-processing, achieving state-of-the-art performance in feed-forward 3D reconstruction.

The goal of complete 3D scene reconstruction is to generate plausible representations of entire environments, including both visible and occluded regions, from limited input views. Regression-based methods include 3D Gaussian Splatting [9] feed-forward approaches [21,20,31], as well as encoder–decoder approaches that create intermediate scene representations for novel view synthesis or complete geometric reconstruction [8,5,24]. These methods learn by comparing the rendered predicted representation with other views of the same scene in their ground truth dataset, indirectly supervising the occluded parts of the scene. These methods often produce impressive visual results near the input views, however, they struggle with view extrapolation, leading to poor predictions for areas that were occluded in the source views. In contrast, generative models are able to produce sharper and often more accurate representations of the unseen areas, but compromise speed and require significant training resources [28,22,12,19]. We suspect that generative approaches are less necessary for layout estimation than object reconstruction, since the layout is typically simple, repetitive and often planar. Both approaches, however, require data with layout supervision, which is addressed by the Room Envelopes dataset.

Layout estimation is the task of reconstructing the structure of an indoor scene from an RGB image [14,15]. Traditional approaches often rely on Manhattan world assumptions, where every wall is at right angles to every other wall, modeling scenes as the best fit 3D box defined by simple 3D planes [32]. In contrast, our Room Envelopes approach provides complete geometric details of the first layout surface for each pixel, rather than just 3D scene bounds. This enables a more comprehensive scene reconstruction learning signal that supports scene completion from a single input image, treating layout estimation as a holistic scene completion task rather than only identifying the 3D extents of a room.

3 The Room Envelopes Dataset

In this section, we outline the properties of the Room Envelopes dataset, the construction procedure, and the data format and statistics.

3.1 Hypersim Dataset

The dataset is built upon the Hypersim dataset [18], which provides images of high-quality synthetic indoor environments, containing 77,400 images across

461 indoor scenes with detailed per-pixel annotations. Hypersim includes images generated from fly-through trajectories of each scene, with multiple camera viewpoints per scene providing comprehensive coverage of each environment. Each frame includes precise depth maps, surface normals, ground truth pointmaps providing world coordinates for each pixel location, instance-level semantic segmentations, and detailed lighting information that captures view-dependent effects such as glossy surfaces and specular highlights. The scenes are rendered using V-Ray, a professional physically-based renderer that produces photorealistic imagery. One limitation of this dataset is that the underlying 3D mesh assets are not freely available, and so they cannot be used as part of our pipeline.

3.2 From Multi-view Images to Room Envelopes

Our processing pipeline transforms the multi-view Hypersim data into Room Envelope representations through point cloud aggregation, semantic filtering, and view-specific re-rendering. We extract pointmaps, surface normals, camera poses, and semantic segmentations from each frame in the Hypersim dataset, then aggregate these into point clouds.

Given pointmaps $\mathbf{P}_i \in \mathbb{R}^{H\times W\times 3}$, RGB images $\mathbf{C}_i \in \mathbb{R}^{H\times W\times 3}$, surface normals $\mathbf{N}_i \in \mathbb{R}^{H\times W\times 3}$, and semantic labels $\mathbf{S}_i \in \mathbb{Z}^{H\times W}$ for each frame i, we directly aggregate all world coordinate points with their associated attributes from all frames in the scene

$$\mathcal{P} = \bigcup_{i=1}^{N} \{(\mathbf{p}_{u,v}^{(i)}, \mathbf{c}_{u,v}^{(i)}, \mathbf{n}_{u,v}^{(i)}, s_{u,v}^{(i)}) : \mathbf{p}_{u,v}^{(i)} \in \mathbf{P}_i\}, \tag{1}$$

where $\mathbf{p}_{u,v}^{(i)} \in \mathbb{R}^3$ represents the world coordinate point, $\mathbf{c}_{u,v}^{(i)} \in \mathbb{R}^3$ the RGB colour, $\mathbf{n}_{u,v}^{(i)} \in \mathbb{R}^3$ the surface normal, and $s_{u,v}^{(i)} \in \mathbb{Z}$ the semantic label at pixel (u, v) in frame i, with only valid points included. We then downsample the pointcloud to remove redundant points using a voxel downsampling method with voxel size ρ, while preserving the associated attributes for each point.

Next, we filter the aggregated point cloud $\mathcal{P}$ to retain only layout-relevant points: walls, floors, doors, windows, and ceilings, such that

$$\mathcal{P}_{\text{layout}} = \{\mathbf{p} \in \mathcal{P} : s(\mathbf{p}) \in \mathcal{S}_{\text{layout}}\}, \tag{2}$$

where $s(\mathbf{p})$ denotes the semantic class of point $\mathbf{p}$ and $\mathcal{S}_{\text{layout}}$ is the set of layout-relevant semantic classes. We align the filtered points to each original camera viewpoint by transforming them through a series of coordinate conversions to account for Hypersim's non-standard camera model

$$\tilde{\mathbf{p}}_{\text{cam}} = \mathbf{T}_{\text{rend}} \mathbf{T}_{\text{w2c}} \tilde{\mathbf{p}}_{\text{w}}, \tag{3}$$

where $\tilde{\mathbf{p}}$ represents points in homogeneous coordinates, $\mathbf{T}_{\text{w2c}}$ is the world-to-camera transformation matrix, and $\mathbf{T}_{\text{rend}}$ handles renderer-specific coordinate conversions and camera model corrections. We then rasterize the scene by projecting the 3D points into pixel image space. Since layout elements may project

to the same pixel we isolate the closest layout surface using depth thresholding. For each pixel (u, v), we first identify all layout points that project to this pixel,

$$\mathcal{P}_{u,v} = \{\mathbf{p} \in \mathcal{P}_{\text{layout}} : \text{proj}(\mathbf{p}) = (u, v)\}, \tag{4}$$

where $\text{proj}(\mathbf{p})$ is the projection of 3D point $\mathbf{p}$ to image coordinates. We filter these candidate points by depth proximity to select points near the closest surface

$$\mathcal{P}^{\tau}_{u,v} = \{\mathbf{p} \in \mathcal{P}_{u,v} : |z_{\mathbf{p}} - z_{\min}| \leq \tau\}, \tag{5}$$

where $z_{\mathbf{p}}$ is the camera-space depth (z-coordinate) of point $\mathbf{p}$, $z_{\min} = \min_{\mathbf{p} \in \mathcal{P}_{u,v}} z_{\mathbf{p}}$, and τ is the depth threshold. From this candidate set of points we select the point most aligned with the camera ray (the 3D ray from the camera centre through pixel (u,v)), minimising the distance between the point and its projection onto the ray. This process is repeated for each pixel, resulting in a complete layout surface pointmap.

A visualisation of the resulting room envelope is shown in Fig. 1. It inherently contain gaps where layout elements are occluded across all viewpoints, for example, floor partially covered by a rug. A visualisation highlighting these holes is shown in Fig. 2. We observe that missing data of this type would also occur for real-world data collection, for which our processing pipeline could be used with minimal alterations.

3.3 Dataset Format and Statistics

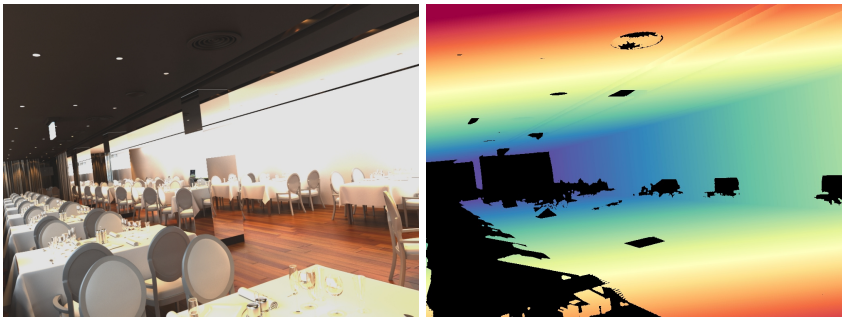
The dataset comprises 77,400 images across 461 indoor scenes. Each data sample consists of an RGB image paired with two pointmaps: a visible surface pointmap capturing all directly observable geometry, and a layout surface pointmap representing structural elements as they would appear without occlusion. Additionally we retain metric scale point clouds of each scene, camera information, and surface normals per pixel. Our approach maintains flexibility for future applications by preserving the completed point clouds, enabling extraction of alternative layers or conversion to other 3D representations as needed.

4 Experiments

4.1 Experimental Setup

We evaluate baseline methods on our Room Envelopes dataset using a comprehensive experimental setup designed to assess layout reconstruction performance. Our dataset is split into 80/10/10 train/validation/test partitions by scene, resulting in 365, 46, and 46 scenes respectively. This scene-level split ensures that test performance reflects the model's ability to generalise to unseen indoor environments rather than memorising specific room configurations.

We fine-tune a pre-trained MoGe v1 model as our base architecture, leveraging its established capabilities in single-view pointmap estimation. Training is



(a) RGB Image with occluded layout (b) Surface layout with missing data

Fig. 2. Missing data and occlusion patterns. (a) Original RGB image showing furniture occluding wall surfaces. (b) Corresponding layout depth image with holes (missing data) where no camera view captured the structural elements.

conducted on 4 NVIDIA L40S 48GB GPUs with a batch size of 32, fine-tuning for 75,000 training steps over approximately 72 hours. We use the AdamW optimiser [13] with a learning rate of 1×10^{-6} for the backbone and 1×10^{-5} for the prediction head, and train with the global pointcloud loss, mask loss, and normal loss from MoGe [25].

4.2 Quantitative Evaluation

Our quantitative evaluation focuses on layout surface reconstruction accuracy across different visibility conditions. For comparison, we evaluate against two other methods. We compare against a pre-trained MoGe v1 model which estimates a pointmap from a single image [25]. Note that MoGe was trained on Hypersim among other datasets. We also compare against a pre-trained LaRI scene estimation model [11], which was trained on layered ray intersection images processed from the 3D-Front dataset [4]. We concatenate the 5 layers of depth predictions from LaRI's output to create a pointcloud, and evaluate the quality of this pointcloud against our Room Envelopes dataset.

For fair comparison, predicted pointclouds are aligned to ground truth pointclouds using scale-shift alignment with least squares optimisation, where a single scale factor is applied to all coordinates and a 3D z translation vector is added. All pointclouds are scaled to metres using the scaling parameters provided in the Hypersim dataset [18]. We evaluate reconstruction quality using Chamfer Distance (CD) and F-Scores (F) similar to previous works [11]. For LaRI evaluation, we take the best one-directional chamfer distance as the evaluation metric, since their method predicts many more points than in the ground truth.

Our evaluation distinguishes between "seen", "unseen", and "overall" layout regions. Seen regions correspond to layout elements that are fully visible in the source image, while unseen regions represent areas occluded by furniture and other objects but reconstructed using information from other viewpoints. Overall regions include all layout elements, regardless of visibility. In our test set,

Table 2. Comparison of layout estimation performance on layout pixels seen in the input image, unseen in the input image, and seen or unseen in the input image (overall). CD denotes the chamfer distance and F denotes the F-score.

	Seen			Unseen			Overall		
Method	CD ↓	F@0.1↑	F@0.05↑	CD↓	F@0.1↑	F@0.05↑	CD↓	F@0.1↑	F@0.05↑
MoGe [25]	**0.229**	**0.657**	0.507	1.037	0.114	0.059	**0.330**	0.565	0.431
LaRI [11]	0.948	0.412	0.286	1.522	0.400	0.283	0.925	0.410	0.285
Ours	0.362	0.647	**0.512**	**0.753**	**0.512**	**0.390**	0.401	**0.633**	**0.499**

Table 3. Surface Normal Likelihood Analysis. We construct a kernel density estimator using von Mises–Fisher kernels from normals in seen layout regions and evaluate the likelihood of normals in unseen regions under this distribution. We compare our method against uniformly distributed random normals (Baseline-uniform) and normals fixed to the most frequent direction in seen regions (Baseline-dominant). Higher average likelihood values indicate more geometrically consistent normal predictions in unseen areas.

Method	Average Likelihood ↑
Baseline-uniform	0.0795
Baseline-dominant	0.8788
Ours	0.4093

unseen pixels comprise 23.57% of all layout pixels, representing a substantial reconstruction challenge. The quantitative results are summarised in Table 2, demonstrating our method's effectiveness particularly in reconstructing occluded layout geometry.

We evaluate the geometric consistency of our predicted layout surfaces by analysing how well normals in occluded regions conform to the distribution of normals from visible regions. Normal vectors are computed from the estimated point cloud using local surface fitting. We randomly sample 5,000 normal vectors from the seen layout regions to construct a kernel density estimator using von Mises–Fisher kernels, where each sampled normal serves as a kernel centre with equal weight, and concentration parameter $\kappa = 15$. We then evaluate the likelihood of 5,000 randomly sampled normals in unseen layout regions under this kernel density estimator. This process is repeated for each test image and the resulting likelihood values are averaged across the entire test split. We compare against two baselines: uniformly distributed random normals on the unit sphere, and normals fixed to the most frequent direction in the seen regions of each test sample. Table 3 shows that our estimated normals in occluded regions are substantially more likely under the seen region distribution than uniformly random normals, though not as consistent as simply using the dominant normal direction from visible regions. This indicates that our method produces both geometrically plausible point distributions in occluded areas and maintains geometric variation rather than defaulting to a single dominant orientation.

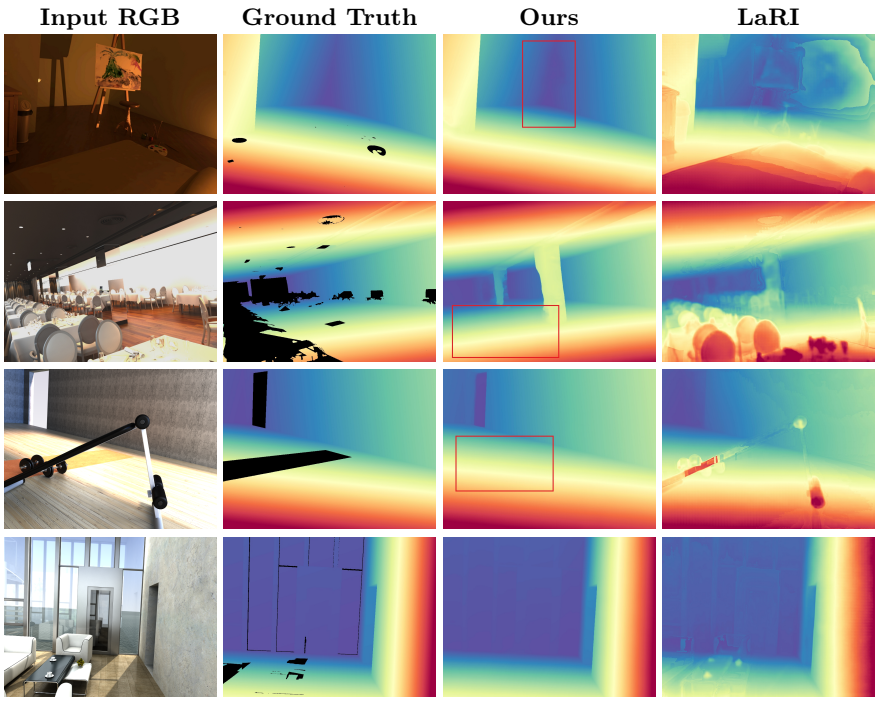


Fig. 3. Qualitative comparison of 3D first surface layout results as depth images. From left to right: input RGB image, ground truth layout geometry, our fine-tuned model trained on Room Envelopes, LaRI [11]. Our method shows superior reconstruction of occluded layout elements. Red boxes highlight regions where our method successfully reconstructs layout geometry that is completely occluded in the input image.

4.3 Qualitative Results

Figure 3 shows qualitative comparisons of our layout reconstruction against LaRI [11] on several test set scenes. Our Room Envelopes approach demonstrates superior reconstruction of layout elements, particularly in occluded regions where furniture blocks the direct view of walls and structural elements. For LaRI, we visualise the best possible permutation of their 5 predicted depth images that best aligns to the ground truth. Additional in-the-wild qualitative results on real indoor images are provided in Section A.

5 Conclusion

We have introduced Room Envelopes, a synthetic dataset that addresses fundamental limitations in indoor scene reconstruction by providing dual pointmap representations: visible surfaces and layout surfaces. This dual representation enables direct supervision for layout reconstruction tasks, eliminating the ambiguity inherent in layered depth approaches and providing clear geometric targets

for feed-forward models. This work opens several directions for future research, including extension to complete scene geometry reconstruction within camera frustums, integration with real-world data collection pipelines, and application to downstream tasks such as robotic navigation and augmented reality.

References

1. Baruch, G., Chen, Z., Dehghan, A., Dimry, T., Feigin, Y., Fu, P., Gebauer, T., Joffe, B., Kurz, D., Schwartz, A., et al.: ARKitScenes: A diverse real-world dataset for 3D indoor scene understanding using mobile RGB-D data. arXiv preprint arXiv:2111.08897 (2021)
2. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3D: Learning from RGB-D data in indoor environments. arXiv preprint arXiv:1709.06158 (2017)
3. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In: CVPR (2017)
4. Fu, H., Cai, B., Gao, L., Zhang, L.X., Wang, J., Li, C., Zeng, Q., Sun, C., Jia, R., Zhao, B., et al.: 3d-front: 3d furnished rooms with layouts and semantics. In: CVPR. pp. 10933–10942 (2021)
5. Jiang, H., Tan, H., Wang, P., Jin, H., Zhao, Y., Bi, S., Zhang, K., Luan, F., Sunkavalli, K., Huang, Q., et al.: RayZer: A self-supervised large view synthesis model. arXiv preprint arXiv:2505.00702 (2025)
6. Jiang, H., Xu, Z., Xie, D., Chen, Z., Jin, H., Luan, F., Shu, Z., Zhang, K., Bi, S., Sun, X., et al.: MegaSynth: Scaling up 3D scene reconstruction with synthesized data. In: CVPR. pp. 16441–16452 (2025)
7. Jiang, Z., Liu, B., Schulter, S., Wang, Z., Chandraker, M.: Peek-a-boo: Occlusion reasoning in indoor scenes with plane representations. In: CVPR. pp. 113–121 (2020)
8. Jin, H., Jiang, H., Tan, H., Zhang, K., Bi, S., Zhang, T., Luan, F., Snavely, N., Xu, Z.: LVSM: A large view synthesis model with minimal 3D inductive bias. In: ICLR (2025), https://openreview.net/forum?id=QQBPWtvtcn
9. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D Gaussian splatting for real-time radiance field rendering. ACM TOG **42**(4), 139–1 (2023)
10. Leroy, V., Cabon, Y., Revaud, J.: Grounding image matching in 3D with MASt3R. In: ECCV (2024)
11. Li, R., Zhang, B., Li, Z., Tombari, F., Wonka, P.: LaRI: Layered ray intersections for single-view 3D geometric reasoning. In: arXiv preprint arXiv:2504.18424 (2025)
12. Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3D object. In: ICCV. pp. 9298–9309 (2023)
13. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR (2019)
14. Mathew, B.: Review on room layout estimation from a single image. International Journal of Engineering Research and Technology **V9** (06 2020). https://doi.org/10.17577/IJERTV9IS060820
15. Meng, M., Zhu, Y., Zhao, Y., Li, Z., Zhu, Z.: 3D indoor scene geometry estimation from a single omnidirectional image: A comprehensive survey. Computational Visual Media **11**(3), 431–464 (2025). https://doi.org/10.26599/CVM.2025.9450438
16. Murai, R., Dexheimer, E., Davison, A.J.: MASt3R-SLAM: real-time dense SLAM with 3d reconstruction priors. In: CVPR. pp. 16695–16705 (2025)

17. Raistrick, A., Mei, L., Kayan, K., Yan, D., Zuo, Y., Han, B., Wen, H., Parakh, M., Alexandropoulos, S., Lipson, L., Ma, Z., Deng, J.: Infinigen indoors: Photorealistic indoor scenes using procedural generation. In: CVPR. pp. 21783–21794 (June 2024)
18. Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R., Susskind, J.M.: Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In: ICCV. pp. 10912–10922 (2021)
19. Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: MVDream: Multi-view diffusion for 3D generation. arXiv preprint arXiv:2308.16512 (2023)
20. Smart, B., Zheng, C., Laina, I., Prisacariu, V.A.: Splatt3R: Zero-shot Gaussian splatting from uncalibrated image pairs (2024), https://arxiv.org/abs/2408.13912
21. Szymanowicz, S., Insafutdinov, E., Zheng, C., Campbell, D., Henriques, J., Rupprecht, C., Vedaldi, A.: Flash3D: Feed-forward generalisable 3D scene reconstruction from a single image. In: 3DV (2025)
22. Szymanowicz, S., Zhang, J.Y., Srinivasan, P., Gao, R., Brussee, A., Holynski, A., Martin-Brualla, R., Barron, J.T., Henzler, P.: Bolt3D: Generating 3D scenes in seconds. arXiv:2503.14445 (2025)
23. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: VGGT: Visual geometry grounded transformer. In: CVPR (2025)
24. Wang, Q., Zhang, Y., Holynski, A., Efros, A.A., Kanazawa, A.: Continuous 3D perception model with persistent state. In: CVPR (2025)
25. Wang, R., Xu, S., Dai, C., Xiang, J., Deng, Y., Tong, X., Yang, J.: MoGe: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In: CVPR (2025)
26. Wang, R., Xu, S., Dong, Y., Deng, Y., Xiang, J., Lv, Z., Sun, G., Tong, X., Yang, J.: MoGe-2: Accurate monocular geometry with metric scale and sharp details (2025), https://arxiv.org/abs/2507.02546
27. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: DUSt3R: geometric 3D vision made easy. In: CVPR (2024)
28. Xiang, J., Lv, Z., Xu, S., Deng, Y., Wang, R., Zhang, B., Chen, D., Tong, X., Yang, J.: Structured 3D latents for scalable and versatile 3D generation. In: CVPR (2025)
29. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: ScanNet++: A high-fidelity dataset of 3D indoor scenes. In: ICCV. pp. 12–22 (2023)
30. Zhang, J., Li, Y., Chen, A., Xu, M., Liu, K., Wang, J., Long, X.X., Liang, H., Xu, Z., Su, H., Theobalt, C., Rupprecht, C., Vedaldi, A., Pfister, H., Lu, S., Zhan, F.: Advances in feed-forward 3D reconstruction and view synthesis: A survey (2025), https://arxiv.org/abs/2507.14501
31. Zhang, K., Bi, S., Tan, H., Xiangli, Y., Zhao, N., Sunkavalli, K., Xu, Z.: GS-LRM: Large reconstruction model for 3D Gaussian splatting. In: ECCV. pp. 1–19. Springer (2024)
32. Zhang, W., Qiao, Y., Liu, Y., Song, R., Zhang, W.: Fast 3D room layout estimation based on compact high-level representation. IEEE TIP (2025)
33. Zheng, J., Zhang, J., Li, J., Tang, R., Gao, S., Zhou, Z.: Structured3d: A large photo-realistic dataset for structured 3d modeling. In: ECCV (2020)
34. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification. ACM TOG **37**(4), 1–12 (2018)

A In-the-Wild Qualitative Results

We present additional qualitative results on real-world indoor images captured from indoor environments in Fig. 4. These examples show how our layout estimation model performs on real indoor environments with different lighting conditions and realistic furniture arrangements. We convert the estimated pointmap into a depth image, and estimate surface normals through local surface fitting on the pointmap. The results illustrate the model's ability to reconstruct plausible layout geometry even when applied to real images that differ from the synthetic training data, highlighting the value of our layout supervision approach for practical applications.

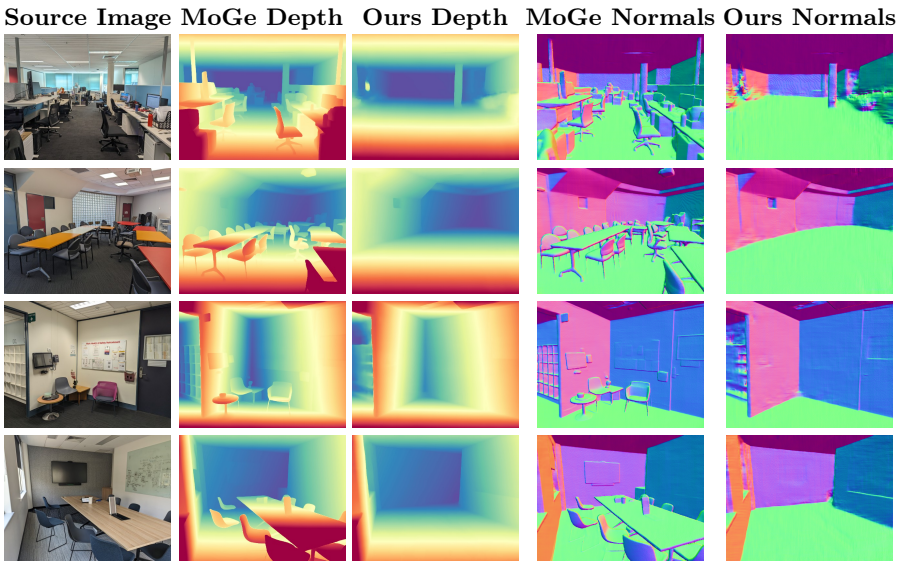


Fig. 4. Qualitative comparison on in-the-wild images captured by a phone camera comparing MoGe and our layout trained model on real indoor images. Normals are estimated using local surface fitting and colourised by mapping the x, y, z components to red, green, blue channels respectively. Please zoom in to see finer details.