
OpenRooms: An End-to-End Open Framework for Photorealistic Indoor Scene Datasets

Zhengqin Li¹ Ting-Wei Yu¹ Shen Sang¹ Sarah Wang¹ Sai Bi¹
Zexiang Xu¹ Hong-Xing Yu³ Kalyan Sunkavalli² Miloš Hašan²
Ravi Ramamoorthi¹ Manmohan Chandraker¹
¹UC San Diego ²Adobe Research ³Sun Yat-sen University

Abstract

Large-scale photorealistic datasets of indoor scenes, with ground truth geometry, materials and lighting, are important for deep learning applications in scene reconstruction and augmented reality. The associated shape, material and lighting assets can be scanned or artist-created, both of which are expensive; the resulting data is usually proprietary. We aim to make the dataset creation process for indoor scenes widely accessible, allowing researchers to transform casually acquired scans to large-scale datasets with high-quality ground truth. We achieve this by estimating consistent furniture and scene layout, ascribing high quality materials to all surfaces and rendering images with spatially-varying lighting consisting of area lights and environment maps. We demonstrate an instantiation of our approach on the publicly available ScanNet dataset. Deep networks trained on our proposed dataset achieve competitive performance for shape, material and lighting estimation on real images and can be used for photorealistic augmented reality applications, such as object insertion and material editing. Importantly, the dataset and all the tools to create such datasets from scans will be released, enabling others in the community to easily build large-scale datasets of their own. All code, models, data, dataset creation tool will be publicly released on our [project page](#).

1 Introduction

Indoor scenes constitute an important environment for visual perception tasks ranging from low-level scene reconstruction to high-level scene understanding. However, indoor scene appearance is a complex function of multiple factors such as shape, material and lighting, and demonstrates complex phenomena like significant occlusions, large spatial variations in lighting and long range interactions between light sources and room geometries. Thus, reasoning about these underlying, entangled factors requires large-scale high-quality ground truth, which remains hard to acquire. While it is possible to capture ground truth geometry using a commercial 3D scanner, this process is expensive. Further, it is extremely challenging (if not nearly impossible) to accurately capture complex spatially-varying material and lighting of indoor scenes.

An alternative is to render large-scale synthetic datasets. Recent works show that deep networks trained on synthetic data can generalize to real scenarios, especially if the synthetic data is physically meaningful. However, large-scale synthetic datasets of indoor scenes with plausible configurations of scene geometry, materials and lighting are also non-trivial to create. Previous works achieve this by using artist-designed scene configurations and assets. This is expensive and brings up copyright issues, which makes it hard to share such data with the research community.

The goal of our work is to democratize indoor scene reconstruction research by making it easier to create high-quality, photorealistic datasets of complex indoor scenes. We present tools to do this from (potentially noisy) scans acquired using commodity sensors and by leveraging publicly available

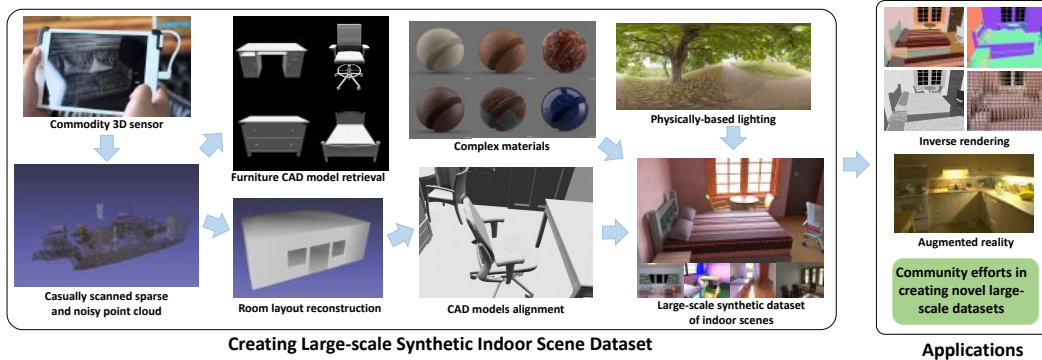


Figure 1: Our framework for creating a synthetic dataset of complex indoor scenes with ground truth shape, SVBRDF and SV-lighting, along with the resulting applications. Given possibly noisy scans acquired with a commodity 3D sensor, we generate consistent layouts for room and furniture. We ascribe per-pixel ground truth for material in the form of high-quality SVBRDF and for lighting as a spatially-varying physically-based representation. We render a large-scale dataset of images associated with this ground truth, which can be used to train deep networks for inverse rendering. We further motivate applications for augmented reality and suggest that the open source tools we make available can be used by the community to create large-scale datasets of their own, using just casually-acquired scans of indoor scenes.

shape, material, and lighting assets, thus allowing anyone to create their own datasets. Unlike most other indoor scene datasets, we provide high-quality ground truth for spatially-varying materials and complex spatially-varying lighting (modeling both area and environment lights and full global illumination). Note that our goal is not to mimic the input images, rather to create ground truth geometry, material and lighting that leads to photorealistic synthetic images of indoor scenes. Figure 1 shows our overall framework for dataset creation and example images from the dataset.

We illustrate an instance of such dataset creation using existing repositories of 3D scans from ScanNet [15], reflectance [1] and illumination [19, 20]. The resulting dataset contains 118343 HDR images with ground-truth per-pixel depth, normal, spatially-varying reflectance and spatially-varying lighting. Since all the above inputs are publicly available, our dataset will be publicly released. Our custom physically-based GPU renderer to synthesize photorealistic images will also be released.

To demonstrate the efficacy of the dataset, we train a state-of-the-art network that decomposes an input image into shape, material and lighting factors. We achieve results close to state-of-the-art on each task, enabling novel object insertion and editing applications (noting that previous works have relied on the artist-created SUNCG dataset[42] which currently faces licensing issues). More importantly, our dataset is open source and can be significantly extended through future community efforts, for example, by adding more and better scans, shapes and materials. In supplementary material, we demonstrate generality by conducting a similar process on the SUN-RGBD dataset [41].

We believe that our effort will significantly accelerate research in multiple areas. “Inverse rendering” tasks are directly related, including single-view [16] and multi-view [49] depth prediction, intrinsic decomposition [28, 8], material classification [6] and lighting estimation [17, 18, 29]. But other indoor reasoning tasks such as scene understanding and navigation will also benefit, where our dataset creation tools can naturally and easily augment existing ones such as House3D [47] and Minos [37] with greater photorealism and scene variations. Finally, learning disentangled representations is a problem of wider interest, where prior works largely consider images of simpler objects such as faces [26, 13]. Indoor scenes constitute a harder challenge due to the complexity of underlying factors and their interactions, where we believe our dataset and tools will allow learning better disentangled representations.

2 Related Work

Indoor scene datasets. The importance of indoor scene reconstruction and understanding tasks has lead to the creation of a number of indoor scene datasets. These includes both real captured datasets [35, 15, 10, 48, 44] as well as synthetic scenes [33, 42, 27]. While real captured datasets are by nature photorealistic, capturing and annotating such datasets is expensive; as a result, these datasets

are smaller (ranging from tens to a few hundred scenes) and only capture some scene information (usually RGB images, geometry in the form of depth and semantic labels). However, we are interested in the problem of estimating scene geometry, reflectance and illumination; the latter two in particular, are extremely challenging to capture/annotate and absent in all real datasets.

This void can be addressed by synthetic datasets where ground truth material and lighting annotations can be automatically generated. However, most of these datasets do not offer the ability to render arbitrary ground truth data [27], or do not have real-world scene layout [33] or material appearance [42]. To the best of our knowledge, the only existing dataset with complex materials and spatially-varying lighting annotations is from Li et al. [29], but this dataset is built on proprietary data [42] that is not publicly accessible. Our goal is to address this problem by creating photorealistic indoor scene datasets with meaningful scene layouts and real-world geometry, reflectance and illumination. Moreover, we propose a pipeline to build synthetic scenes from real world RGBD scans, thus allowing anyone to create such a dataset using off-the-shelf commodity hardware.

Building CAD models from real data. Several recent methods can build CAD models for indoor scenes from a single image [22] or a scanned point cloud [2, 9, 12]. Our dataset creation framework relies on [2] to locate furniture. However, our framework goes beyond replacing geometries and also automatically assigns real-world materials and lighting to create photorealistic synthetic scenes.

Inverse rendering for indoor scenes. Indoor scene inverse rendering seeks to reconstruct geometry, reflectance and/or lighting from (in our case, monocular) RGB images. In particular, estimating geometry, in the form of scene depth or surface normals, has been widely studied [16, 3, 49, 32].

Most scene material estimation methods either recognize material classes [7] or only reconstruct diffuse albedo [28, 4, 25]. Scaling these methods to real-world images requires scene datasets with complex physically-based materials. Li et al. [29] demonstrate this by training a physically-motivated network on a large-scale indoor scene dataset with ground-truth complex reflectance annotations. However, their dataset is built on top of proprietary data [43]. In this work, we demonstrate that we achieve comparable inverse rendering performance from our using their method but trained on our dataset developed using publicly available assets.

Previous indoor scene lighting estimation methods only predict shading (which entangles geometry and lighting) [28], require RGBD inputs [4], or rely on hand-crafted heuristics [24, 25]. More recently, deep network-based lighting estimation methods have shown great progress for estimating both global [17] and spatially-varying lighting [18, 40, 29] from single RGB images. The latter set of methods largely rely on proprietary synthetic data to generate spatially-varying lighting annotations. We demonstrate that we can achieve comparable performance by training on our dataset.

3 Building a Photorealistic Indoor Scene Dataset

In this section, we describe our framework for building a synthetic complex indoor scene dataset. We start from a large collection of 3D scans of indoor scenes, reconstruct the room layout, replace detected objects in the scans with CAD models, and finally assign complex materials and lighting to the scenes. We mainly demonstrate our pipeline on ScanNet, a large-scale repository of real indoor scans [15]. However, our pipeline can be extended, using existing tools, to other datasets [41][21] of 3D scans of indoor scenes, as demonstrated in the supplementary.

3.1 Creating CAD Models from 3D Scans

We demonstrate using ScanNet [15] to create a synthetic indoor scene dataset, which achieves a balance between the degree of automation and the generalization ability of deep networks trained on this dataset. ScanNet contains RGBD images for 1506 diverse scenes covering 707 distinct spaces. We can fuse the depth maps from different views of a scene to obtain a single point cloud.

Initial furniture placement. Our first step is to locate and replace furniture in the scans with a CAD model. Previously, Scan2cad [2] has provided the ground-truth poses that aligned CAD models from ShapeNet [11] to the ScanNet point clouds. Therefore, we directly use their annotations. We do not require the appearance of the CAD models to be a close match to the ScanNet images, but focus on generating layouts and shapes with plausible appearances with as much automation as possible.

Reconstructing the room layout. The next step is to reconstruct the room layout, i.e. the locations of the walls, floor and ceiling. There are several previous works [12, 50, 9] that aim at this problem. However, probably due to the significantly lower quality of point clouds in casually



Figure 2: (Top) Rendered examples of assigning complex materials to ShapeNet CAD models. (Bottom-left) Synthetic scenes rendered with different materials and different lighting. (Bottom-right) A synthetic scene rendered with from different views selected by our algorithm. A video is included in the supplementary.

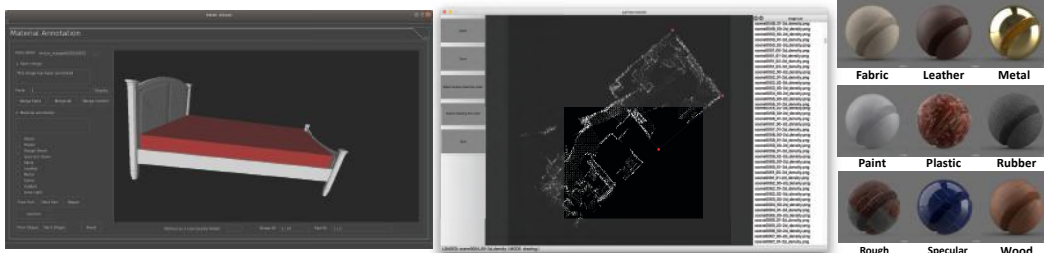


Figure 3: (Left) UI for room layout annotation. (Middle) UI for material category annotation. (Right) Material examples from each category. Please zoom for better visualization.

acquired sequences, directly applying these methods to ScanNet point clouds leads to artifacts such as hallucinating non-existent walls or splitting a single room into multiple ones. To mitigate this, we design a UI for quick layout annotation, which allows us to labeling the whole dataset efficiently in 1-2 days. The user interface is shown in Figure 3 (Middle). In this tool, we first project the 3D point clouds to the floor plane and obtain the contour for the walls by manually selecting a polygon. While this step is manual, it is fast and does not require artistic effort. The tool will be open sourced.

Moreover, we train the network from prior work [12] on our annotations for automatic layout estimation. This leads to significantly better layout estimation for noisy point clouds. Our trained network can be extended to other real 3D scans [41][21]. More results are included in the supplementary. After locating the walls, we find the floor by using RANSAC to fit the lowest horizontal plane and find the ceiling by simply setting the height of the rooms to be 3 meters.

Windows and doors. Windows and doors are important sources of illumination in indoor scenes. Thus, special consideration is needed to localize them in our dataset creation process. We utilize the segmentation provided by ScanNet to locate them. We first project the 3D points labeled as doors and windows to the closest wall and then divide the wall into a set of segments $\{S_i\}$ with equal length. Let $S_i(W)$ and $S_i(D)$ be the number of window points and door points in S_i . A wall segment will be labeled as a door or window if

$$S_i(W) > \tau_W, \quad S_i(D) > \tau_D \quad (1)$$

where τ_W and τ_S are thresholds. To determine the number and range of doors and windows, we find the connected components of wall segments and each connected component becomes one door (window) instance. After locating the windows and doors, we randomly pick one CAD model from each of these categories in ShapeNet [11] and assign it to the scene.

Consistency of room layout and furniture poses. In the initial room layouts and furniture poses, the furniture can intersect with walls and floors, or float above the floor. We solve these conflicts by computing the distance between the bounding boxes of furniture and walls or floors and move the furniture accordingly. Let D_W , D_F be the directional distance between the bounding box of the

furniture item to the wall and floor, where $D < 0$ suggests that the object is inside the room. Let \mathbf{n}_W and \mathbf{n}_F be the normal of the wall and floor planes, pointing into the room. The translations \mathbf{t}_W and \mathbf{t}_F to solve the conflicts with walls and floors are defined as

$$\mathbf{t}_W = \max(D_W, 0) \mathbf{n}_W \quad \mathbf{t}_F = D_F \mathbf{n}_F \quad (2)$$

If the furniture being moved supports another furniture, we move the supported furniture recursively until every furniture finds its support.

3.2 Assigning Complex Materials to Indoor Scenes

One of the major contributions of our dataset is to provide ground-truth annotation of complex material parameters for indoor scenes. Previous work typically provides material annotations in the form of simple diffuse or Phong reflectance [42, 39]. While recent works like [29] use physically accurate microfacet SVBRDFs, they rely on proprietary artist-created assets from [42]. We instead use 1331 spatially varying complex materials from Adobe Stock assets [1], defined by a parametric microfacet BRDF model [46].

Assigning materials to ShapeNet. Many ShapeNet CAD models do not have texture coordinates; we use cube mapping [14] to compute texture coordinates for these models automatically.

To assign materials to models, we propose to split CAD models into semantically meaningful parts and assign different materials to different parts. This is inspired by [36], who use a similar strategy to map complex materials to chair models. However, they use the initial part segmentation provided by ShapeNet, which we find to be usually noisy and not semantically meaningful. Thus, we use the segmentation from PartNet [34], which provides high-quality semantically meaningful part segmentation of 24 categories of ShapeNet models.

Material annotation UI. After obtaining the part segmentation, we use a custom UI tool to manually assign material category annotations to each part, as shown in Figure 3 (Right). It has two functions. The first function is part combination, in which the annotators will be asked to choose the parts that they believe should share the same material and group them into a single part. This is necessary because many CAD models from PartNet have overly fine-grained segmentations.

The second function is to annotate each part with possible material categories. Similar to [31] and [29], we divide 1078 kinds of spatially-varying materials from the Adobe stock dataset into 9 categories based on their appearances.¹ The examples of material categories are shown in Figure 3 (Right). We ask annotators to label each part of the CAD model with one or multiple material categories. Figure 2 (Top) shows rendered examples of our material assignment results. Instead of attempting to match the appearance of input images, our random assignment simply respects broad material categories, which maintains a plausible appearance. The resulting images are not optimized for aesthetic quality; however, to enable training of inverse rendering networks that generalize to unseen real images, the resulting appearance diversity may in fact be beneficial. Our tools and the annotations will be released for future research.

Assigning materials to room layouts. We find that not all the materials look plausible on large-scale room structures (walls, floors, ceilings); we choose a subset of materials for them.

3.3 Assigning Appropriate Lighting to Indoor Scenes

In order to achieve high-quality lighting estimation, we create our dataset using diverse and physically-based lighting models. We use 2 kinds of light sources: environment lighting coming from the windows, and area lighting. We use 414 high-resolution HDR panoramas of natural outdoor scenes, collected from [20] and [19]. This set is 2 times larger than in previous work [29].

We also appropriately handle indoor lights. Scan2CAD does not include ceiling lights. Thus, we place a randomly chosen ceiling light model from PartNet in each room. Unlike previous synthetic datasets which randomly sample the spectrum (color) of the area lights [29, 51, 28], we follow the more physically plausible black-body model, where the spectrum of the light source is decided by its temperature. In our dataset, we randomly sample between 4000K and 8000K.

¹Around 300 materials are not used because they are only relevant for outdoor scenes.

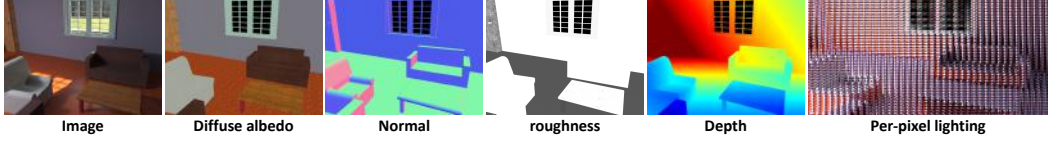


Figure 4: One of our rendered images with ground-truth geometry, spatially-varying material and lighting.

3.4 View Selection

ScanNet provides the camera pose of each RGBD image. However, the view distribution is biased towards views that are very close to the scene geometry and therefore can only cover a small region of the room, which achieves optimal coverage for scanning. On the contrary, we prefer views covering larger regions with more non-trivial geometry, matching typical human viewing conditions. To achieve this, we first randomly sample different views along the wall approximately facing the center of the room. For each view, we render its depth map and normal map. Let d_p and n_p be the depth and normal of pixel p , $\text{Grad}(n_p)$ be the gradient of the normal. We choose the view based on computing a score defined as

$$\sum_{p \in \mathcal{P}} \text{Grad}(n_p) + 0.3 \sum_{p \in \mathcal{P}} \log(d_p + 1). \quad (3)$$

Views with higher scores are used to create the dataset. An example of our view selection results is shown in Figure 2 (bottom right). Details are included in the supplementary.

3.5 Rendering the Dataset with a Physically-based Renderer

In order to minimize the domain gap between synthetic and real data, it is necessary to use a physically-based renderer that models complex light transport, such as inter-reflections and soft shadows. Therefore, we render our dataset using a custom-built physically-based GPU renderer. We render HDR images, the ground-truth SVBRDFs and the geometry. To create ground-truth spatially varying lighting, we follow the same lighting representation as in [29]. For every pixel, we find its corresponding 3D point on the object surface and render a 16×32 environment map at that location. The typical time of rendering one image and one spatially varying lighting is between 1-2 minutes, which is one order faster than a typical CPU-based renderer. Our renderer will be open sourced. More details are included in the supplementary.

3.6 Dataset Statistics

We pick 1287 scenes from the 1506 ScanNet scenes to create our dataset. 219 scenes are discarded because their boundaries are not clear. We randomly sample 1178 scenes for training and 109 scenes for validation. For each scene, we choose 2 sets of views using our view selection method. For each set of views, we render the image with different material and lighting configurations, leading to three set of images as shown in Figure 2 (Bottom-left). We render 118343 HDR images at 480×640 resolution, 108269 of them in the training set and 10074 of them in the validation set. We also render ground-truth spatially-varying lighting for all of them, at a grid of 120×160 spatial locations, where the resolution of each lighting probe is 16×32 . Both the number of HDR images and spatially varying lighting in our dataset are larger than [29], especially spatially-varying lighting.

4 Experimental Validation on Inverse Rendering Tasks

In this section, we verify the effectiveness of our dataset for inverse rendering tasks by testing networks trained on our dataset on various benchmarks.

Network architecture. To demonstrate the effectiveness of our dataset, we use a state-of-the-art network architecture for inverse rendering of indoor scenes, as proposed by [29], which achieves superior performances in capturing complex material reflectance and spatially varying lighting. Please refer to the supplementary material for more details.



Figure 5: Comparisons with previous state-of-the-art on intrinsic decomposition (albedo prediction shown).

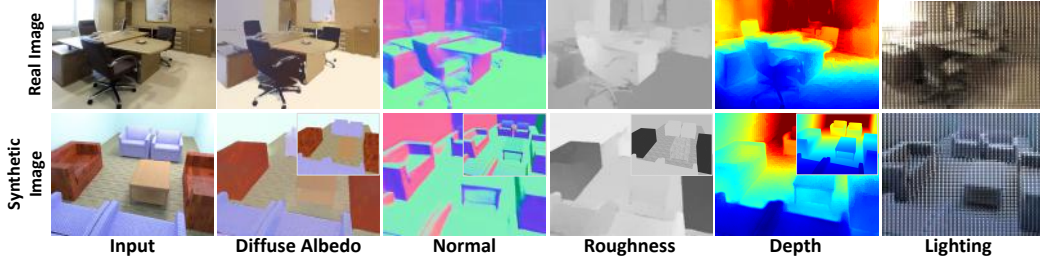


Figure 6: Inverse rendering results on a real example and a synthetic example. The insets in the bottom row are the ground truth.

		WHDR
Ours	Ours + IIW	16.4
Li18[28]	CGI + IIW	17.5
Sen.19[38]	CGP + IIW	16.7
Li20[29]	CGM + IIW	15.9

Table 1: Intrinsic decomposition on IIW [5].

Method	Mean($^{\circ}$)	Median($^{\circ}$)	Depth(Inv.)
Ours	25.3	18.0	0.171
Li20[29]	24.1	17.3	0.184
Sen.19[38]	21.1	16.9	
Zhang17[51]	21.74	14.8	

Table 2: Normal and depth predictions on NYU dataset [35].

4.1 Inverse Rendering on Real Datasets

In this section, we test the networks on various real data benchmarks. Both qualitative and quantitative comparisons show that networks trained on our synthetic dataset can generalize well to real data and can achieve performance comparable to the state-of-the-art non-public dataset.

Intrinsic decomposition. We compare our intrinsic decomposition results with 3 previous approaches. The qualitative comparison is shown in Figure 5 while the quantitative comparison is shown in Table 1. Our method is comparable to prior state-of-the-arts. All of these previous methods are based on SUNCG and therefore can no longer be used due to severe licensing restrictions.

Depth and normal estimation. We evaluate the normal and depth estimation on the NYU dataset. The quantitative evaluation is in Table 2. We perform slightly worse than Li et al.’s dataset, possibly because their SUNCG-based dataset has more diverse and complex geometry compared to our ShapeNet-based furnitures.

4.2 Insertion and Editing in Real Images

Object insertion. Photorealistic synthetic object insertion is a key application in augmented reality. It requires high-quality estimation of geometry, material and lighting. We tested our trained network on the datasets from [18], which contains the around 80 ground-truth spatially-varying light probes. Some results are shown in Figure 7. Our network outperforms those methods that cannot handle spatially-varying or high-frequency lighting well. It can even generate more consistent lighting color compared to [30], which is trained on a SUNCG-based dataset. That is probably because our dataset has more diverse outdoor lighting and handle the indoor lighting in a physically meaningful way. A quantitative user study summarized in Table 3 also suggests that network trained on our dataset performs better on object insertion.



Figure 7: Object insertion results on real data from dataset of [18]. We observe that our dataset leads to photorealistic insertion results comparable to state-of-the-arts [30][18]. Please zoom in for more details.



Figure 8: Material replacement results on real data. We observe that specular effects and spatially-varying lighting are handled well by the network trained on our dataset.

Material editing. We further illustrate material editing examples by replacing the material of a planar surface in Figure 8. We show that spatially-varying lighting effects and some amount of specularity in the synthetic material are handled quite well. The material editing results are comparable to ones obtained by [29], even though our dataset is created based only on noisy scans acquired with a commodity sensor.

5 Conclusion and Future Work

We have proposed methods that enable user-generated photorealistic datasets for complex indoor scenes. Previous synthetic datasets required large amounts of artistic effort to create, and have licensing issues preventing their open distribution. In contrast, our dataset is created without artist involvement, from existing public repositories of 3D scans, 3D shapes and materials.

We illustrate the process on over 1000 interior scenes from ScanNet, and proceed by matching CAD models to detected furniture, detecting room layout, assigning doors and windows, assigning complex spatially-varying materials from a database, and creating lighting through a combination of indoor area lights and outdoor environment lighting. The resulting dataset has 118343 HDR images with ground truth per-pixel depth, normal and material coefficients. We also provide spatially varying lighting for each image (a set of ground-truth HDR environment maps sampled at a dense set of locations in the scene). We demonstrate the efficacy of our dataset by training a state-of-the-art inverse rendering network on it, achieving performance comparable to the previous state-of-the-art training dataset (which is not publicly available). We further demonstrate augmented reality applications such as object insertion and material editing in complex indoor scenes.

	Barron13 [4]	Gardner17 [17]	Garon19 [18]	Li20 [30]	Ours
Barron13 [4]	-	23.37%	13.25%	13.60%	11.81%
Gardner17 [17]	76.63%	-	36.25%	39.54%	33.84%
Garon19 [18]	86.75%	63.75%	-	42.28%	43.47%
Li20 [30]	86.40%	60.46%	57.72%	-	45.23%
Ours	88.19%	66.16%	56.53%	54.77%	-

Table 3: User study on object insertion. Here we perform pairwise comparisons between different methods. The number $X\%$ in row I column J means that in $X\%$ of total cases, human annotators think method I performs better than method J . The comparisons with our method are labeled with (blue) color in the table. Our method outperforms all previous state-of-the-arts. More details and comparisons are in the supplementary.

Our dataset will be publicly released, along with all the tools for its creation, which will allow the rest of the community to extend the dataset or create new ones based on their own scans.

A Video

The video included with the supplementary material illustrates various steps of the proposed dataset creation framework and the side-by-side comparisons between our rendered results and the original ScanNet images. The comparisons further demonstrate the high-quality of our rendered dataset.

B Synthetic Dataset Creation using SUNRGBD Data



Figure 9: Synthetic scene reconstruction results using scanned indoor scenes from SUNRGBD dataset. We visualize the reconstructed scenes rendered from different views with different material assignments.

To demonstrate that our framework can generalize to other datasets, we present our scene reconstruction results based on scanned indoor scenes from the SUNRGBD dataset. Unlike ScanNet [15], SUNRGBD only contains partial scans of the rooms with extremely incomplete and sparse point clouds. Moreover, unlike Scan2CAD [2], SUNRGBD only has 3D bounding box annotations for furniture locations and lacks full poses. Thus, we design a method for furniture retrieval and pose estimation. We utilize the ground-truth bounding box annotation as an initialization. We first align the bounding box of the CAD models with the ground truth bounding boxes provided by the SUNRGBD dataset. Then we select the CAD model and adjust its pose by using grid search to minimize the Chamfer distance between the CAD model and the point cloud in the bounding box. In some cases, intersections may arise due to inaccurate bounding box annotations in the SUNRGBD dataset (unlike in the case of ScanNet), which we handle by simply manual adjustment of furniture positions to resolve conflicts. Then, we reconstruct the room layout and assign appropriate materials and lighting to the CAD models, as described in the main paper.

In Figure 9, we visualize the reconstruction results by rendering the created scenes from different viewpoints with different material assignments. While the rendered images may not have high aesthetic value, they present diverse appearances with plausible material and lighting assignments, with complex visual effects such as soft shadows and specularities being correctly handled. We posit that through future community efforts to produce more accurate scene layout and furniture pose annotation, our framework for high-quality synthetic dataset creation can be significantly enhanced.

C Further Comparisons and Results

This section includes: (a) more user study results for object insertion under spatially-varying lighting, (b) comparisons of normal estimation with prior works on real datasets, (c) comparisons for layout estimation, (d) ablation study for the network on our proposed dataset, (e) qualitative visualization of inverse rendering network outputs on synthetic and real data, when trained on a synthetic dataset created from ScanNet using the proposed dataset creation method.

User study on object insertion under SV-lighting. We conducted a user study to quantitatively evaluate our object insertion performance on the real dataset of [18], consisting of 20 real images.

Barron13 [4]	Gardner17 [17]	Garon19 [18]	Li20 [30]	Ours
11.5%	28.07%	29.15%	34.84%	38.89%

Table 4: Summary of quantitative numbers of user study on object insertion. Here we compare the lighting prediction results of different methods against ground truth lighting and report the % of times that users picked a particular method as being more realistic than ground truth; ideal performance is 50%. Our result is marked in (blue). Similar to the results in the main paper, our trained network outperforms previous state-of-the-arts.



Figure 10: Qualitative comparisons of object insertion in [18]’s dataset.

Some qualitative and quantitative results have been included in Table 3 and Figure 7 in the main paper. We provide more comparisons in Table 4 and Figure 10 in the supplementary material.

In Table 4, we summarize comparisons for different methods against ground-truth lighting. Ideal performance for this task is 50%, which indicates that the predicted lighting and the ground-truth lighting are indistinguishable. The best two previous methods [30] (34.84%) and [18] (29.15%) are both trained on the SUNCG-related datasets and therefore can no longer be used. Again, our method (38.89%) outperforms both of them. In Figure 10, we show more qualitative comparison. The network trained on our dataset achieves realistic high frequency shading the consistent lighting color.

In conclusion, the dataset created by our framework enables high-quality object insertion with performance better than the methods built on previous datasets created from proprietary repositories. More importantly, our dataset is free to use and can be easily extended and enhanced by the community, using the dataset creation process and tools proposed by the paper.

Normal prediction. Figure 11 shows qualitative comparisons with [38] and [51] on three real examples from [38]. Since both methods are trained on SUNCG-related datasets, quantitative comparison with those methods on datasets other than NYU [35] is hard without the trained models. From Figure 11, we observe that even though [51] achieves the best accuracy on NYU dataset, it might overfit to that specific dataset and might not generalize well to images from other sources. On the contrary, both [38] and our network achieve less noisy normal predictions. Our network may sometimes over-smooth the normal, probably since our scenes are built from Scan2CAD annotations that usually contain only a small number of large items of furniture in each room. Therefore, there may be less geometric details in our synthetic dataset. This can probably be solved in the future by procedurally adding small objects to the rooms to increase the complexity of the dataset.

Layout prediction. In order to reduce labeling effort, we experimented with automatic layout prediction using Floor-SP [12]. It accepts a 2D top-down projection of the point cloud and its mean surface normal as inputs. In the subsequent steps, the room segmentation is predicted and room loops are formed. We omit the loop merging step since ScanNet scans generally contain a single room. We refer the reader to [12] for more details. Since the point cloud generated by RGB-D scans contain higher levels of noise compared to the training data used by the authors, we trained a randomly initialized model on a subset of ScanNet consisting of 1069 scenes with human-annotated layout as ground-truth.

The final layout is evaluated on 103 held-out scenes in terms of corner precision and recall, edge precision and recall, as well as intersection-over-union of the room segmentation. A corner prediction is deemed correct if its distance to the closest ground truth corner is within 10 pixels. An edge

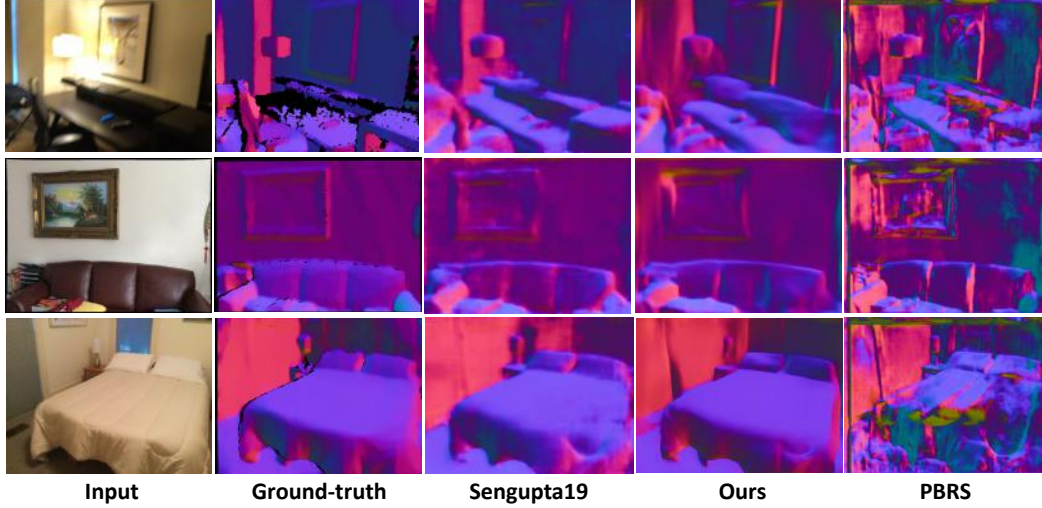


Figure 11: Qualitative comparisons of our normal estimation with previous methods Sengupta19 [38] and PBRS [51], on real images from [38].

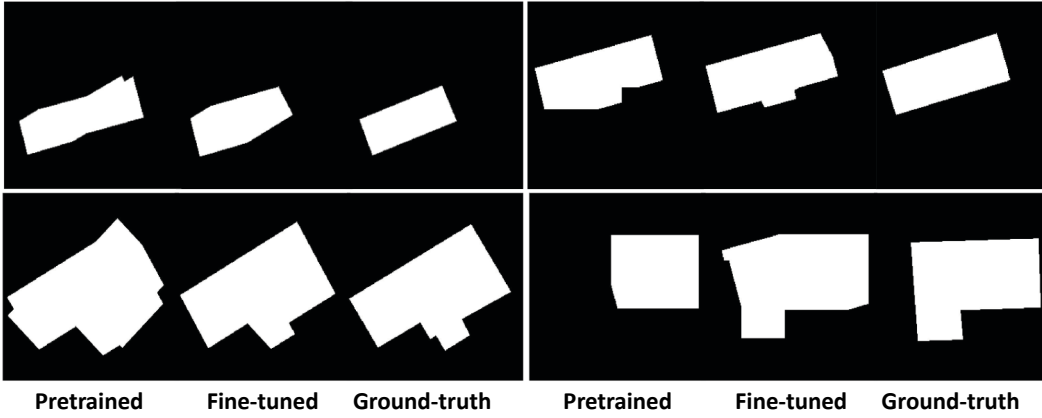


Figure 12: Comparison of layout reconstruction using the original network from [12] and the network trained on our ScanNet annotation.

prediction is deemed valid if its two endpoints pass the criterion for corners and the edge belongs to the set of ground-truth edges.

Table 5 shows the comparison between the model trained on ScanNet and the pre-trained weights provided by the authors. Figure 12 shows a qualitative comparison. Note that the room segmentation performs moderately well despite the low precision and recall of the corners and edges. We believe that this is caused by the ambiguities during layout annotation. Since we require the walls to be arranged such that they form a closed loop, for the scans that do not cover the entire room, the human annotator would have to add false corners and edges that pass through open areas where the scan is incomplete, thereby affecting the evaluation of the corner and edge predictions. On the other hand, false corners and edges do not affect IoU since it measures the area covered by the room, rather than the occurrence of predictions.

Inverse rendering on testing set of proposed dataset. Table 6 quantitatively evaluates the performance of the network trained and then tested on the proposed synthetic dataset created from ScanNet. We observe that both the cascade structure and bilateral solver can improve the accuracy of prediction of most intrinsic components. Figure 13 shows a few inverse rendering results on our synthetic testing set. From the figure, we observe that through iterative refinement, the cascade structure can effectively

	Corner		Edge		Room
	Precision	Recall	Precision	Recall	IOU
Chen19 [12]	0.358	0.524	0.151	0.191	0.734
Trained on ScanNet	0.531	0.716	0.254	0.316	0.858

Table 5: Comparison of Floor-SP [12] models with pre-trained weights provided by [12] and weights trained on 1069 ScanNet scenes.

	Albedo(10^{-3})	Normal(10^{-2})	Depth(10^{-2})	Roughness(10^{-2})	Lighting
Cascade0	9.99	4.51	5.18	6.59	0.150
Cascade1	9.43	4.42	4.89	6.64	0.146
Bilateral solver	9.29	-	4.86	6.57	-

Table 6: Ablation study for the network architecture on our proposed dataset. We report the scale invariant L2 loss for Albedo, L2 loss for normal, scale invariant log L2 loss for depth, L2 loss for roughness and scale invariant $\log(x + 1)$ L2 loss for per-pixel lighting.

remove noise and recover high-frequency signals, especially for lighting and normal prediction. The bilateral solver also helps remove noise by enhancing the smoothness prior.

Further examples of inverse rendering on real images. Figure 14 shows inverse rendering results on several real images. We observe that even though the network is trained on a synthetic dataset, it can generalize well to real data. For real data, the effectiveness of the cascade structure and bilateral solver is more apparent, probably due to noisier initial predictions on real data.

D Microfacet BRDF Model

We use the simplified microfacet BRDF model of [23]. Let A , N , R be the diffuse albedo, normal and roughness. Our BRDF model $f(A, N, R)$ is defined as

$$f(A, N, R, l, v) = \frac{A}{\pi} + \frac{\mathbf{D}(h, R)\mathbf{F}(v, h)\mathbf{G}(l, v, N, R)}{4(N \cdot l)(N \cdot v)}, \quad (4)$$

where v and l are the view and light directions, while h is the half angle vector. Further, $\mathbf{D}(h, R)$, $\mathbf{F}(v, h)$ and $\mathbf{G}(l, v, N, R)$ are the distribution, Fresnel and geometric terms, respectively, which are defined as

$$\mathbf{D}(h, R) = \frac{\alpha^2}{\pi((N \cdot h)^2(\alpha^2 - 1) + 1)^2} \quad (5)$$

$$\alpha = R^2 \quad (6)$$

$$\mathbf{F}(v, h) = (1 - F_0)2^{(-5.55473(v \cdot h) - 6.98316)v \cdot h} + F_0 \quad (7)$$

$$\mathbf{G}(l, v, N, R) = \mathbf{G}_1(v, N)\mathbf{G}_1(l, N) \quad (8)$$

$$k = \frac{(R + 1)^2}{8} \quad (9)$$

$$\mathbf{G}_1(v, N) = \frac{N \cdot v}{(N \cdot v)(1 - k) + k} \quad (10)$$

$$\mathbf{G}_1(l, N) = \frac{N \cdot l}{(N \cdot l)(1 - k) + k}, \quad (11)$$

We set $F_0 = 0.05$, following [23].

E Physically-Based GPU Renderer

We render our dataset efficiently using a physically-based GPU renderer. There is one design choice that improves the rendering speed while maintainss the rendering quality. When rendering the spatially-varying lighting, we not only uniformly sample the hemisphere, but also sample the light sources. The contributions of the two sampling methods can be combined together using the standard

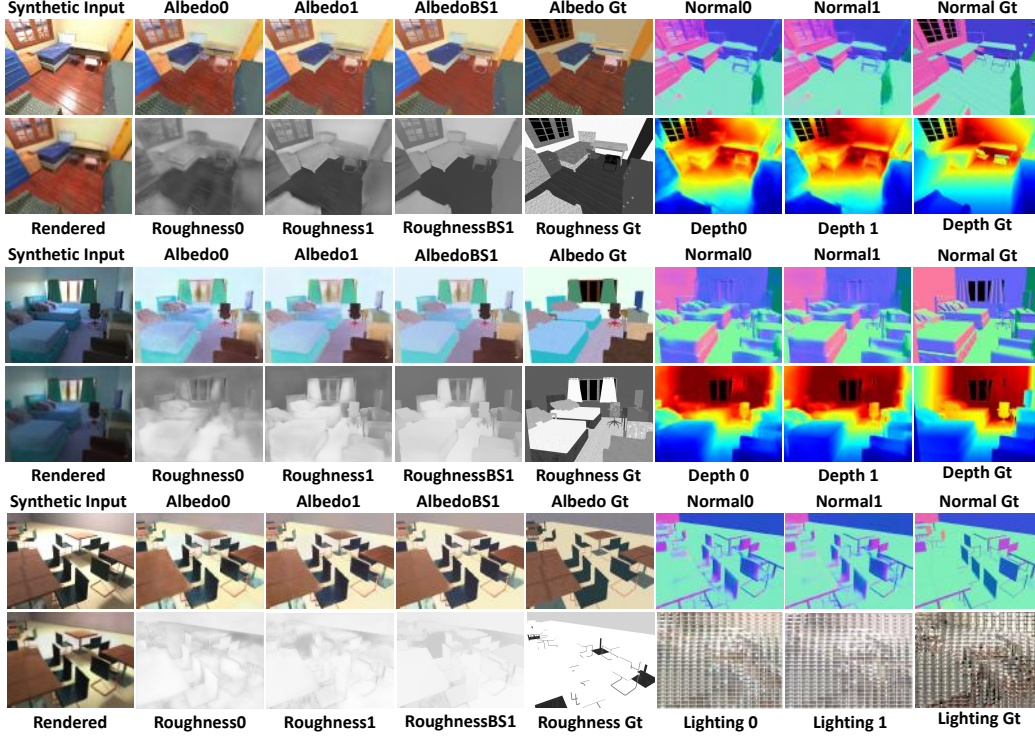


Figure 13: Qualitative visualization of inverse rendering results on synthetic images from the testing set of the proposed dataset.

power rule in multiple importance sampling [45]. This allows us to capture the radiance from small light sources in the scenes with far fewer samples. More formally, let η be the ray direction, $\mathbf{P}_L(\eta)$ be the probability of sampling η when sampling the light sources, $\mathbf{P}_U(\eta)$ be the probability of uniformly sampling the hemisphere and \mathcal{I} be an indicator function that is equal to 1 when a light source is sampled and 0 otherwise. Let $\mathbf{L}(\eta)$ be the radiance. Then, the contribution of sampling η towards the corresponding pixel on the hemisphere can be written as:

$$I \cdot \frac{\mathbf{P}_L^2(\eta)}{\mathbf{P}_L^2(\eta) + \mathbf{P}_U^2(\eta)} \frac{\mathbf{L}(\eta)}{\mathbf{P}_L(\eta)} + (1 - I) \cdot \frac{\mathbf{P}_U^2(\eta)}{\mathbf{P}_L^2(\eta) + \mathbf{P}_U^2(\eta)} \frac{\mathbf{L}(\eta)}{\mathbf{P}_U(\eta)}. \quad (12)$$

References

- [1] Adobe stock, 2017. 2, 5
- [2] Armen Avetisyan, Manuel Dahnert, Angela Dai, Manolis Savva, Angel X Chang, and Matthias Nießner. Scan2cad: Learning cad model alignment in rgb-d scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2614–2623, 2019. 3, 9
- [3] Aayush Bansal, Bryan Russell, and Abhinav Gupta. Marr revisited: 2D-3D model alignment via surface normal prediction. In *CVPR*, 2016. 3
- [4] Jonathan T Barron and Jitendra Malik. Intrinsic scene properties from a single rgb-d image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 17–24, 2013. 3, 8, 10
- [5] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *ACM Transactions on Graphics (TOG)*, 33(4):159, 2014. 7
- [6] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the materials in context database. *Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

- [15] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 2, 3, 9
- [16] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 2, 3
- [17] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Trans. Graphics*, 9(4), 2017. 2, 3, 8, 10
- [18] Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-François Lalonde. Fast spatially-varying indoor lighting estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6908–6917, 2019. 2, 3, 7, 8, 9, 10
- [19] HDRI HAVEN. 100% free hdris, for everyone., 2017. 2, 5
- [20] Yannick Hold-Geoffroy, Akshaya Athawale, and Jean-François Lalonde. Deep sky modeling for single image outdoor lighting estimation. In *CVPR*, 2019. 2, 5
- [21] Binh-Son Hua, Quang-Hieu Pham, Duc Thanh Nguyen, Minh-Khoi Tran, Lap-Fai Yu, and Sai-Kit Yeung. Scenenn: A scene meshes dataset with annotations. In *International Conference on 3D Vision (3DV)*, 2016. 3, 4
- [22] Hamid Izadinia, Qi Shan, and Steven M Seitz. Im2cad. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5134–5143, 2017. 3
- [23] Brian Karis and Epic Games. Real shading in unreal engine 4. *Proc. Physically Based Shading Theory Practice*, 2013. 12
- [24] Kevin Karsch, Varsha Hedau, David Forsyth, and Derek Hoiem. Rendering synthetic objects into legacy photographs. *ACM Transactions on Graphics*, 30(6):1, 2011. 3
- [25] Kevin Karsch, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, Hailin Jin, Rafael Fonte, Michael Sittig, and David Forsyth. Automatic scene inference for 3d object compositing. *ACM Transactions on Graphics*, pages 32:1–32:15, 2014. 3
- [26] Tejas D Kulkarni, William F Whitney, Pushmeet Kohli, and Josh Tenenbaum. Deep convolutional inverse graphics network. In *Advances in neural information processing systems*, pages 2539–2547, 2015. 2
- [27] 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. *arXiv preprint arXiv:1809.00716*, 2018. 2, 3
- [28] Zhengqi Li and Noah Snavely. Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In *ECCV*, pages 371–387, 2018. 2, 3, 5, 7
- [29] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. *CVPR*, 2020. <https://arxiv.org/abs/1905.02722>. 2, 3, 5, 6, 7, 8
- [30] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. 2020. 7, 8, 10
- [31] Zhengqin Li, Kalyan Sunkavalli, and Manmohan Chandraker. Materials for masses: Svbrdf acquisition with a single mobile phone image. In *ECCV*, 2018. 5
- [32] Chen Liu, Jimei Yang, Duygu Ceylan, Ersin Yumer, and Yasutaka Furukawa. Planenet: Piece-wise planar reconstruction from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2579–2588, 2018. 3

- [33] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J. Davison. Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? 2017. 2, 3
- [34] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In *CVPR*, June 2019. 5
- [35] Pushmeet Kohli, Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgb-d images. In *ECCV*, 2012. 2, 7, 10
- [36] Keunhong Park, Konstantinos Rematas, Ali Farhadi, and Steven M Seitz. Photoshape: photo-realistic materials for large-scale shape collections. *ACM Transactions on Graphics (TOG)*, 37(6):192, 2019. 5
- [37] Manolis Savva, Angel X. Chang, Alexey Dosovitskiy, Thomas Funkhouser, and Vladlen Koltun. MINOS: Multimodal indoor simulator for navigation in complex environments. *arXiv:1712.03931*, 2017. 2
- [38] Soumyadip Sengupta, Jinwei Gu, Kihwan Kim, Guilin Liu, David W Jacobs, and Jan Kautz. Neural inverse rendering of an indoor scene from a single image. *arXiv preprint arXiv:1901.02453*, 2019. 7, 10, 11
- [39] Jian Shi, Yue Dong, Hao Su, and X Yu Stella. Learning non-Lambertian object intrinsics across ShapeNet categories. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 5844–5853. IEEE, 2017. 5
- [40] Shuran Song and Thomas Funkhouser. Neural illumination: Lighting prediction for indoor environments. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6918–6926, June 2019. 3
- [41] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *CVPR*, pages 567–576. IEEE Computer Society, 2015. 2, 3, 4
- [42] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. *Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 3, 5
- [43] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1746–1754, 2017. 3
- [44] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 2
- [45] Eric Veach. *Robust Monte Carlo methods for light transport simulation*, volume 1610. Stanford University PhD thesis, 1997. 13
- [46] Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. Microfacet models for refraction through rough surfaces. In *EGSR*, 2007. 5
- [47] Yi Wu, Yuxin Wu, Georgia Gkioxari, and Yuandong Tian. Building generalizable agents with a realistic and rich 3d environment. *arXiv preprint arXiv:1801.02209*, 2018. 2
- [48] Fei Xia, Amir R. Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: real-world perception for embodied agents. In *Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on*. IEEE, 2018. 2

- [49] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018. 2, 3
- [50] Edward Zhang, Michael F Cohen, and Brian Curless. Emptying, refurnishing, and relighting indoor spaces. *ACM Transactions on Graphics (TOG)*, 35(6):1–14, 2016. 3
- [51] Yinda Zhang, Shuran Song, Ersin Yumer, Manolis Savva, Joon-Young Lee, Hailin Jin, and Thomas Funkhouser. Physically-based rendering for indoor scene understanding using convolutional neural networks. *CVPR*, 2017. 5, 7, 10, 11