Neural Rendering in a Room: Amodal 3D Understanding and Free-Viewpoint Rendering for the Closed Scene Composed of Pre-Captured Objects

BANGBANG YANG, State Key Lab of CAD&CG, Zhejiang University, China
YINDA ZHANG, Google, USA
YIJIN LI, State Key Lab of CAD&CG, Zhejiang University, China
ZHAOPENG CUI*, State Key Lab of CAD&CG, Zhejiang University, China
SEAN FANELLO, Google, USA
HUJUN BAO, State Key Lab of CAD&CG, Zhejiang University, China
GUOFENG ZHANG*, State Key Lab of CAD&CG, Zhejiang University, China

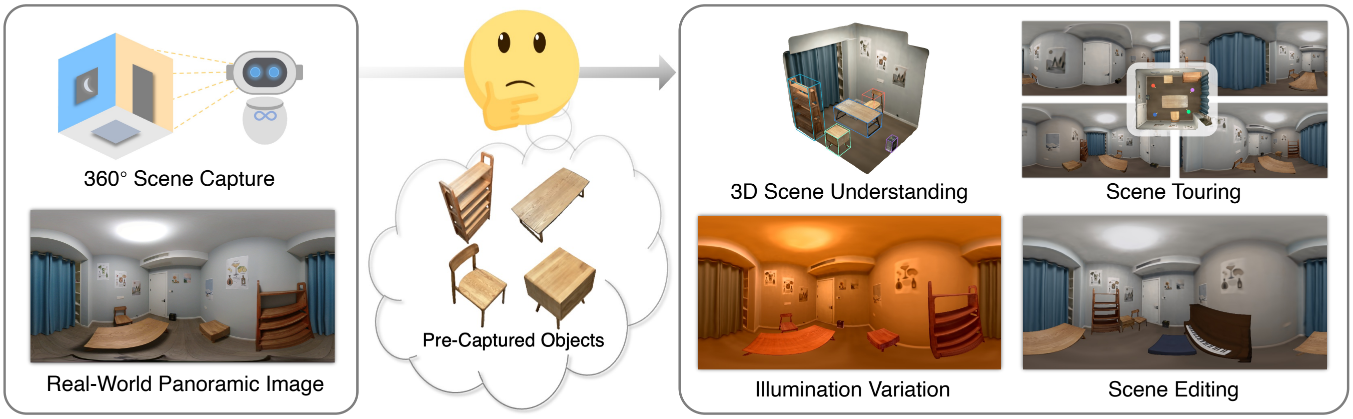


Fig. 1. **Motivation.** We focus on a scenario where a service robot operates in a specific indoor environment (*e.g.*, household, office, or museum). Therefore, it can collect information of the closed scene in an offline stage, then provide effective amodal scene understanding with a single panoramic capture of the room, which facilitates high-level tasks and delivers immersive synchronized free-viewpoint touring with illumination variation and scene editing.

We, as human beings, can understand and picture a familiar scene from arbitrary viewpoints given a single image, whereas this is still a grand challenge for computers. We hereby present a novel solution to mimic such human perception capability based on a new paradigm of amodal 3D scene understanding with neural rendering for a closed scene. Specifically, we first learn the prior knowledge of the objects in a closed scene via an offline stage, which facilitates an online stage to understand the room with unseen furniture arrangement. During the online stage, given a panoramic image of the scene in different layouts, we utilize a holistic neural-rendering-based optimization framework to efficiently estimate the correct 3D scene layout and deliver realistic free-viewpoint rendering. In order to handle the domain gap between the offline and online stage, our method exploits compositional neural rendering techniques for data augmentation in the offline training. The experiments on both synthetic and real datasets demonstrate that our two-stage design achieves robust 3D scene understanding and outperforms competing methods by a large margin, and we also show that our realistic free-viewpoint rendering enables various applications, including scene touring and editing. Code and data are available on the project webpage: https://zju3dv.github.io/nr_in_a_room/.

CCS Concepts: • **Computing methodologies** → **Computer vision**; **Rendering**.

Additional Key Words and Phrases: Neural rendering, 3D scene understanding, Amodel perception

ACM Reference Format:
Bangbang Yang, Yinda Zhang, Yijin Li, Zhaopeng Cui, Sean Fanello, Hujun Bao, and Guofeng Zhang. 2022. Neural Rendering in a Room: Amodal 3D Understanding and Free-Viewpoint Rendering for the Closed Scene Composed of Pre-Captured Objects. *ACM Trans. Graph.* 41, 4, Article 101 (July 2022), 10 pages. https://doi.org/10.1145/3528223.3530163

*Guofeng Zhang and Zhaopeng Cui are corresponding authors.

Authors' addresses: Bangbang Yang, ybbbbt@gmail.com, State Key Lab of CAD&CG, Zhejiang University, China; Yinda Zhang, yindaz@gmail.com, Google, USA; Yijin Li, eugenelee@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, China; Zhaopeng Cui, zhpcui@gmail.com, State Key Lab of CAD&CG, Zhejiang University, China; Sean Fanello, seanfa@google.com, Google, USA; Hujun Bao, baohujun@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, China; Guofeng Zhang, zhangguofeng@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, China.

0730-0301/2022/7-ART101
https://doi.org/10.1145/3528223.3530163

1 INTRODUCTION

Given a photo of our living room, as human beings, we can vividly picture the whole layout in our mind, including how the furniture is placed in 3D space and how the environment looks from any viewpoint, even when objects are re-arranged differently in the room. Granting the computer similar skills would require reliable indoor scene 3D semantic understanding and free-viewpoint rendering capabilities, which ideally are all fulfilled from widely available input, *e.g.*, a single photo. Over decades, enormous efforts have been made in the field of computer vision and graphics [Dai et al. 2020; Gortler et al. 1996; Levoy and Hanrahan 1996; Mildenhall et al. 2020; Nie et al. 2020; Zhang et al. 2021a], yet the gap with the human perception is still huge. Despite this, we argue that likely humans are better at this task for places they are familiar with, and the learned prior knowledge on the objects and their arrangement in a closed room are the key to the success.

In this paper, we present a novel solution for reliable 3D indoor scene understanding and free-viewpoint rendering in a closed scene – a.k.a. a room with a fixed set of pre-captured objects but placed under unknown arrangements and diverse illuminations. Inspired by human amodal perception, our method takes advantage of an offline stage to collect prior knowledge of the target scene, where models for each object, *e.g.*, for localization or neural rendering, can be built with an affordable workload and then fine-tuned in the specific scenario for better performance. With the help of this strong prior knowledge, during the online stage, our method only needs light-weighted input, *i.e.*, a single panoramic image taken from the scene, and can reliably recognize and localize objects in 3D space and render the scene from arbitrary camera viewpoints via amodal 3D understanding. While general scene understanding [Nie et al. 2020; Zhang et al. 2021a] makes the best effort to make predictions under unseen environments but still suffers from generalization issues, our amodal scene understanding aims at an accurate and reliable scene understanding for familiar scenes.

A flexible yet effective scene representation is critical to the considered task. Traditional representations such as textured meshes [Izadinia et al. 2017; Liu et al. 2019; Waechter et al. 2014] or voxels [Kim et al. 2013; Song et al. 2017] generally have some drawbacks, *e.g.*, limited rendering quality [Liu et al. 2019] and resolution [Song et al. 2017], requiring pre-built CAD furniture model for scene reconstruction [Izadinia et al. 2017] and explicit lighting/material definitions for lighting variations [Li et al. 2020; Matusik et al. 2003], which prohibits fine-grained scene rendering and understanding. We thus choose the neural implicit representation [Mildenhall et al. 2020] as it enables geometric reconstruction with photo-realistic volumetric rendering, and it could be extended to support functionalities such as appearance variation [Martin-Brualla et al. 2021] and scene graph decomposition [Ost et al. 2021] with rendering-based optimization.

Specifically, we first build object detection and 3D pose estimation models for all the objects of interest as well as a neural rendering model for each object, including the empty room. At run-time, given a panoramic image taken from the room stuffed with pre-captured objects in a new arrangement, the scene understanding task can be achieved by 3D object detection and pose estimation, followed by an optimization via differentiable rendering using the neural rendering models. Additionally, the per-object neural rendering models can be plugged in to support full scene free-viewpoint rendering. While this framework is technically plausible, we find it suffers from several challenges as follows, which we will address in this work:

Intensive Computation. Neural volume rendering methods are typically computationally intensive since a tremendous number of network queries are required for points densely sampling along pixel rays, making it prohibitive for back-propagation-based optimization, like pose estimation, where the rendering needs to be done repetitively. iNeRF [Yen-Chen et al. 2021] mitigates this issue by restricting sample pixels inside the detected region of interest, which reduces the computation cost and enables the camera pose estimation with respect to a single object on a commodity-level GPU. However, this is still not practical for room-scale scenarios when multiple objects need to be jointly optimized in order to handle mutual occlusions or physical relations. To tackle this challenge, we learn an implicit surface model jointly with its radiance field, inspired by NeuS [Wang et al. 2021a], which allows us to perform efficient sphere tracing [Liu et al. 2020] at the early stage of the rendering, leveraging the estimated ray-to-surface distances. Points can then be sampled from regions close to the surface, and a small number of points is sufficient for the optimization. In this way, we significantly reduce the computational cost and make it feasible to finish the joint optimization with multiple objects on a single GPU in a reasonable amount of computation.

Incorrect Physical Relationship. Even though machine learning models are trained per-scene, they could still make obvious mistakes like breaking the physical rules and resulting in implausible novel view rendering, *e.g.*, objects flying in the air or intersecting with walls. To solve this problem, we propose several novel physical losses and integrate a physics-based optimization into the neural-rendering-based optimization, where the conformity to prior knowledge and even pre-defined rules (*e.g.*, a bed should attach to the wall) are jointly optimized with the photometric error between the rendered image and the observation. This significantly helps fix errors made on individual objects and improves the overall object pose accuracy, which further delivers context abides rendering.

Domain Gap. The lighting condition may inevitably vary in the scene, and object renderings from the models trained at offline stage may not be consistent with the environment, which will further influence the rendering-based optimization. To mitigate this, we propose to exploit compositional neural rendering to augment the training data. In particular, we augment the pre-captured data with environment maps sourced from polyhaven.com [Zaal et al. 2020], and learn the neural rendering models conditioned on lighting represented in a latent space. During the neural rendering based optimization, the neural rendering model is able to respond to novel illumination other than the one during the pre-capture stage, and both the environment lighting and object pose can be successfully optimized. We also synthesize objects with different scene layouts and render photo-realistic images for the training of object prediction, which empirically enhances model robustness.

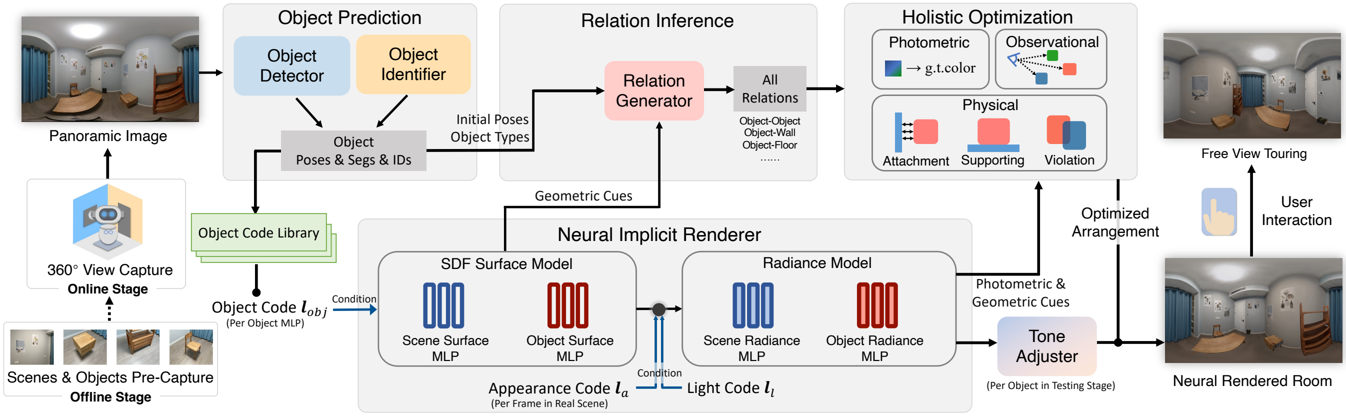


Fig. 2. During the offline stage, we learn a neural implicit renderer and object detectors with pre-captured scene and objects. During the online stage, given a panoramic capture of a room, we first recognize object identities and estimate object meta information. Then, we generate object relations based on the prediction and the geometric cues from the renderer. Finally, we conduct a holistic optimization to obtain 3D scene understanding by jointly optimizing all the photometric and geometric cues.

Our contributions can be summarized as follows. We present a practical solution for a novel task which aims at amodal 3D scene understanding and free-viewpoint rendering for indoor environments from a single panoramic image. We design a two-stage framework in which per-object pre-trained models are learned offline, and a neural-rendering-based optimization is exploited for online 3D understanding. We analyze the technical challenges for this novel task and propose mitigation techniques to improve run-time efficiency, add physical constraints in the model, handle illumination changes, and increase the data diversity that is hard to be ensured physically. Extensive experiments show that our method can achieve significantly better 3D scene understanding performance than state-of-the-art general 3D scene understanding methods and meanwhile, deliver free-viewpoint rendering capability that supports high-level applications like scene editing and virtual touring.

2 RELATED WORK

2.1 3D Scene Understanding

3D scene understanding is a popular topic in computer vision. Early works mainly focus on room layout estimation with Manhattan World [Coughlan and Yuille 1999; Ramalingam et al. 2013; Sun et al. 2019; Yan et al. 2020; Zou et al. 2018] or cuboid assumption [Dasgupta et al. 2016; Mallya and Lazebnik 2015]. Song *et al.* [Song et al. 2015] attempts to reconstruct and recognize scene objects from a domestic robot but requires laborious crowd-sourced annotations. With the advance of neural networks, many works propose to estimate both object poses and the room layout [Du et al. 2018; Huang et al. 2018a; Zhang et al. 2017]. To recover object shapes, some methods [Chen et al. 2019; Groueix et al. 2018; Wang et al. 2018] reconstruct meshes from a template, and others [Huang et al. 2018b; Izadinia et al. 2017] adopt shape retrieval approaches to search from a given CAD database. Recently, some approaches [Dahnert et al. 2021; Nie et al. 2020; Popov et al. 2020; Yang and Zhang 2016; Yang et al. 2019; Zhang et al. 2021b] enable 3D scene understanding by generating a room layout, camera pose, object bounding boxes, or even meshes from a single view, automatically completing and annotating scene meshes [Bokhovkin et al. 2021] or predicting object alignments and layouts [Avetisyan et al. 2020] from an RGB-D scan. Inspired by PanoContext [Zhang et al. 2014] that panoramic images contain richer context information than the perspective ones, Zhang *et al.* [Zhang et al. 2021a] propose a better 3D scene understanding method with panoramic captures as input. For amodal scene completion, Zhan *et al.* [Zhan et al. 2020] propose to decompose cluttered objects of an image into individual identities. However, these works still suffer from limited generalization in real-world environments and do not allow fine-grained scene presence from arbitrary views.

2.2 Neural Rendering

Neural rendering methods aim at synthesizing novel views of objects and scene by learning scene representation from 2D observations in various forms, such as voxels [Lombardi et al. 2019; Sitzmann et al. 2019a], point clouds [Dai et al. 2020], meshes [Riegler and Koltun 2020, 2021], multi-plane images [Mildenhall et al. 2019; Tucker and Snavely 2020; Wang et al. 2021b] and implicit functions [Mildenhall et al. 2020; Niemeyer et al. 2020; Sitzmann et al. 2019b]. NeRF [Mildenhall et al. 2020] uses volume rendering to achieve photo-realistic results; follow up works extend the model to multiple tasks, such as pose estimation [Yen-Chen et al. 2021], dense surface reconstruction [Oechsle et al. 2021; Wang et al. 2021a; Yariv et al. 2021] and scene editing [Granskog et al. 2021; Guo et al. 2020; Yang et al. 2021]. Meanwhile, other methods [Riegler and Koltun 2020, 2021] also show impressive free-viewpoint rendering capability in the wild, or scene rendering [DeVries et al. 2021; Luo et al. 2020] of indoor environments. However, existing neural rendering pipelines either need to be trained for a static scene thus do not generalize to dynamic environments, or require domain prior [Wang et al. 2021b; Yu et al. 2021], limiting the free-viewpoint rendering in unconstrained settings.

3 METHOD

Given a panoramic image of a closed environment with unknown furniture placement, our goal is to achieve reliable 3D scene understanding, including instance semantic detection, 3D geometry of each object, and their arrangements (*i.e.*, object positions) in the room, utilizing the data pre-captured beforehand. We split the whole pipeline into an offline stage and an online stage. During the **offline stage**, we scan each object and the scene background with an RGB-D camera, train a neural implicit renderer for every object of interest in the room, and then fine-tune object detectors via compositional neural rendering. In the **online stage**, as shown in Fig. 2, we first predict object meta information (*i.e.*, poses, segmentation and IDs) from the panoramic image, and then follow pre-defined rules to generate object-object and object-room relations based on the object prediction and geometric cues (*e.g.*, physical distances obtained from the encoded neural implicit model). Finally, to correctly estimate the scene arrangement and lighting condition that visually fits the input panorama, we perform holistic optimization with all the photometric and geometric cues, which further enables free-viewpoint scene touring and scene editing.

3.1 Offline Stage

3.1.1 Neural Implicit Renderer.

Neural Implicit Model for Scene and Objects. We use a neural implicit renderer that hierarchically encodes the room. Practically, we choose the SDF-based implicit field for geometry representation [Wang et al. 2021a; Yariv et al. 2021][1], since it provides an exact surface to facilitate geometric optimization, *e.g.*, for collision detection, while NeRF's density field is too noisy or uncertain to support a similar objective. As shown in Fig. 2, we separately express geometry in SDF values (with SDF surface model F_{SDF}) and colors (with radiance model F_{R}). We will show later that this formulation enables efficient neural-rendering-based optimization by providing geometric cues like ray intersection distances with sphere tracing. Motivated by Yang *et al.* [Yang et al. 2021], we encode scene background and objects in two branches, and use the object code $\boldsymbol{l}_{\text{obj}}$ to control the visibility of a certain object, rather than per-model per-object training. We render the object k with sampled points $\{\mathbf{x}_i | i = 1, ..., N\}$ along the ray $\boldsymbol{r}$, which is defined as:

$$\hat{C}(\boldsymbol{r})_{\text{obj}} = \sum_{i=1}^{N} T_i \alpha_i \mathbf{c}_{\text{obj}_i}, \quad T_i = \prod_{j=1}^{i-1} (1 - \alpha_{\text{obj}_j}),$$
$$\alpha_{\text{obj}_j} = \max\left(\frac{\Phi_s(\text{SDF}(\mathbf{x}_i)_j) - \Phi_s(\text{SDF}(\mathbf{x}_{i+1})_j)}{\Phi_s(\text{SDF}(\mathbf{x}_i)_j)}, 0 \right). \tag{1}$$

Note that we omit the object index k for brevity. T_i is the accumulated transmittance, Φ_s is the logistic density distribution, $\text{SDF}(\mathbf{x}) = F_{\text{SDF}}(\mathbf{x}, l_{\text{obj}})$, and α_{obj} is the opacity value derived from the SDF surface model. $\mathbf{c}_{\text{obj}}$ is the color defined as $\mathbf{c}_{\text{obj}} = F_{\text{R}}(\mathbf{x}, \text{v}, \boldsymbol{l}_{\text{obj}}, \boldsymbol{l}_a, \boldsymbol{l}_l)$, where v is the viewing direction, $\boldsymbol{l}_a$ is the appearance code [Martin-Brualla et al. 2021] that handles per-frame sensing variations (*e.g.*, white balance and auto-exposure on real-world data), $\boldsymbol{l}_l$ is the light code introduced later. We supervise the renderer with color, depth and object masks, and jointly render multi-objects and scene background by ordering the distance of samples along the ray directions and render pixels $\hat{C}(\boldsymbol{r})$ following the quadrature rules. More details can be found in the supplementary material.

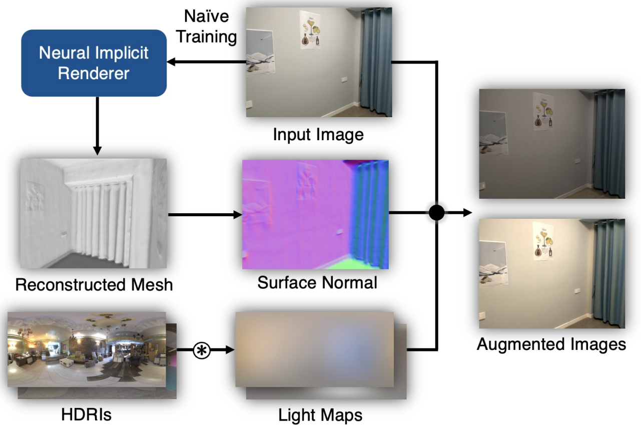


Fig. 3. **Lighting augmentation.** We show how to leverage the neural rendering model and pre-convolved HDR maps to synthesize novel lighting conditions. See text for details.

Lighting Augmentation & Light Code Learning. We learn a neural renderer conditioned on a latent lighting space $\boldsymbol{l}_l$, aiming at modeling scene-level illumination variation and adapting to the target scene depicted in the given panorama. Since it is non-trivial to capture real-world images with thorough lighting variation, we synthetically augment the captured image with diffuse shading rendered from realistic HDR environment maps. Practically, we gather 100 HDRI indoor environment maps from the Internet [Zaal et al. 2020] and convolve them to diffuse irradiance maps [Debevec 2006]. Then we compute the per-pixel surface normal in the world coordinate and retrieve the corresponding light intensity from the light map. Finally, we multiply the light intensity to the input images. However, for real-world data, reliable surface normals are not readily available. To tackle this problem, we leverage a two-stage pipeline by first training a naïve neural renderer without augmentation, and then extracting mesh from the model for normal computation. We show this procedure in Fig. 3, where the captured image has been naturally augmented with two different light maps. During the training stage, we randomly augment the input images with pre-convolved light maps and feed the radiance model F_{R} with a learnable light code $\boldsymbol{l}_l^m$, where m is the index of the light map. Although such geometry-aware augmentation does not cover every important aspect of the real-world physics and provides augmented data only up to an approximation, it brings convenience to the offline stage: the training data is collected only once under a mild lighting condition, and the network empirically adapts to unseen lighting decently (see Sec. 4.3).

3.1.2 3D Object Prediction Fine-tuning.

Module Design. As illustrated in Fig. 2 (on top left), we adopt the object detector (ODN) from Zhang *et al.* [Zhang et al. 2021a, 2014]

[1] In our paper, we use the formulation from NeuS [Wang et al. 2021a], but VolSDF [Yariv et al. 2021] is also applicable.

and Nie *et al.* [Nie et al. 2020] to detect scene objects and estimate object poses w.r.t. the camera, and use an object identifier based on NetVLAD feature similarity [Arandjelovic et al. 2016] to recognize previously seen objects.

Data Augmentation with Neural Rendering. At training time for each scene, instead of physically moving objects in the real-world, we exploit compositional neural rendering with implicit neural renderer (Sec. 3.1.1) to render labeled panorama for training, where objects are randomly placed following user-defined rules (*e.g.*, bed and table should attach to the floor). Then, we perform a fast fine-tuning for the pre-trained ODN network from Zhang *et al.* [Zhang et al. 2021a] and also store the NetVLAD features for each object of different views.

3.2 Online Stage

3.2.1 Bottom-up Initialization.

Object Prediction. We first feed panoramic images to the object detector, and obtain object meta information including initial pose estimation, instance / semantic segmentation and object identities.

Relation Generation. As demonstrated in [Nie et al. 2020; Zhang et al. 2021a, 2014], indoor scenes are commonly well-structured (*e.g.*, beds and nightstands are usually attached to the wall, desks and chairs are often supported by the floor), and such prior knowledge can be formulated as various relations to enhance arrangement optimization. Therefore, we also generate a series of relations for physical constraints optimization (Sec. 3.2.4), including object-object support, object-wall attachment and object-floor support. Practically, we directly infer relations based on object meta information and geometric cues (extracted bounding boxes, ray intersection distance and normal) from the SDF surface model with user-defined rules (see supplementary material). In theory, our method can also work with rules or scene context learned in a data-driven way [Zhang et al. 2021a], which we leave for future work.

Camera Pose Estimation. Since the optimization is based on a known neural implicit model, we need to locate camera poses to ensure background rendering is aligned with the input image. To do so, we transform the panorama to multiple perspective views (*i.e.*, similar to "equirectangular to cubemap" conversion by warping pixels according to ray directions) and employ the method from Sarlin *et al.* [Sarlin et al. 2019, 2020] for visual localization.

Object Pose Parameterization. We optimize poses $\hat{\mathbf{T}}^k \in \mathrm{SE}(3)$ for K objects, where the rotation $\hat{\mathbf{R}}^k$ is parameterized as Zhou *et al.* [Zhou et al. 2019], and the position (a.k.a. object center) $\hat{\mathbf{p}}^k$ is directly expressed in Euclidean space.

3.2.2 Photometric Constraint Optimization.

Tone Adjuster. To better adapt the lighting condition to the input panorama at the online stage, we introduce a per-object tone adjuster which explicitly models lighting variations and helps to reduce the burden of light code optimization. In practice, we additionally optimize a learnable shifting factor $\mathbf{t}^k_{\text{obj}}$ and scaling factor $\mathbf{s}^k_{\text{obj}}$ for each object j as: $\tilde{\mathbf{c}}^k_{\text{obj}} = (\hat{\mathbf{c}}^k_{\text{obj}} - \mathbf{t}_{\text{obj}})^k / \mathbf{s}^k_{\text{obj}}$, which can be regarded as color transformation [Reinhard et al. 2001] but in a per-object manner. We find this explicit representation benefits the lighting adaptation, as demonstrated in our experiments.

Photometric Loss with Joint Rendering. We use photometric constraint by leveraging joint rendering where the photometric loss is back-propagated to optimize per-object poses and light parameters. For each input image, we sample N rays on the object masks and $0.2N$ rays on the background so as to ensure the convergence of both objects and background. The photometric loss is defined as the squared distance between rendered colors $\hat{C}(\boldsymbol{r})$ and pixel colors $C(\boldsymbol{r})$ from the input panorama for all the sampled rays $\boldsymbol{r} \in N_r$:

$$L_{pho} = \frac{1}{|N_r|} \sum_{\boldsymbol{r} \in N_r} ||\hat{C}(\boldsymbol{r}) - C(\boldsymbol{r})||_2^2. \tag{2}$$

Safe-Region Volume Rendering. However, neural volume rendering requires hundreds of network queries for each ray, which restricts tasks like pose estimation [Yen-Chen et al. 2021] by only sampling a small bunch of rays due to the limitation of GPU memory. This is particularly true in our task as one ray might go through 2 or 3 objects at a time when object to object occlusions happen, which results in 2 or 3 times more queries than a single object case. Fortunately, as our renderer learns an SDF-based representation of the geometry, we can easily determine ray intersections using sphere tracing at the early stage of the rendering. Inspired by Liu *et al.* [Liu et al. 2020], we propose a safe-region volume rendering by first computing ray-to-surface distances with efficient sphere tracing and then sampling much fewer points near the surface for differentiable volume rendering. Our experiments demonstrate that this strategy significantly reduces network query times and allows us to jointly optimize more objects in cluttered scenes. Please refer to the supplementary material for more details.

3.2.3 Observation Constraint Optimization.

Observation Loss. The initial poses from object prediction may be inaccurate on the dimension of camera-to-object distance due to scale ambiguity, but the observing angles (a.k.a. object center re-projection) on the panoramic view are usually reliable. Thus, we also add an observation constraint by encouraging closer observing angles of objects between initial pose estimation and the optimized pose, as:

$$L_{obs} = \sum_{k=1}^{K} ||1 - \mathrm{sim}(\mathbf{p}^k_{\text{init}} - \mathbf{p}_{\text{cam}}, \hat{\mathbf{p}}^k - \mathbf{p}_{\text{cam}})||^2, \tag{3}$$

where $\mathrm{sim}(\cdot)$ denotes cosine similarity, and $\mathbf{p}_{\text{cam}}$ is the camera center estimated in Sec. 3.2.1.

3.2.4 Physical Constraint Optimization.

Prior scene understanding works [Nie et al. 2020; Zhang et al. 2021a,b] mainly build physical constraints upon object bounding boxes and room layout under Manhattan assumption. Thanks to the precise geometries encoded in the neural SDF model, we can define physical constraints to optimize physical conformity at a finer-grained level.

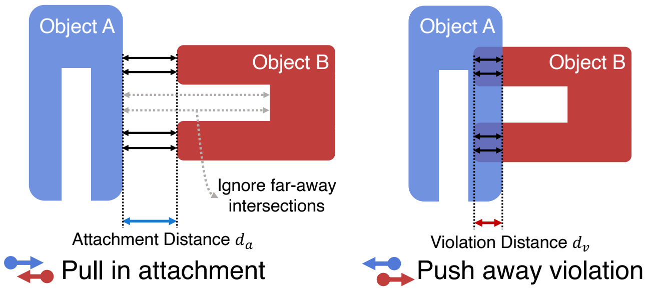


Fig. 4. **Magnetic Loss.** For the attachment relation, the loss pulls in when two instances are far from each other and pushes away when the violation happens.

Magnetic Loss. We introduce a novel magnetic loss that fully leverages neural renderer's SDF model to optimize generated relations (*e.g.*, attachment and support) from Sec.3.2.1. As the name suggests, the magnetic loss encourages two opposite surfaces of the attached objects to be close to each other without violation. Practically, we shoot a set of probing rays from one object surface plane to another with the shooting direction from the generated relations (Sec. 3.2.1) and compute ray-to-surface distances via sphere tracing. Then, as illustrated in Fig. 4, we define two distances to diagnose surface-surface relations: 1) attachment distance d_a which measures the surface distance between two objects by summing up the distances of partial nearest intersections while ignoring far-away intersections, 2) violation distance d_v which indicates potential violation of two objects by summing up all the violation part of the surfaces. To this end, we defined magnetic loss as:

$$L_{mag} = \frac{1}{K} \sum_{k=1}^{K} \max(d_a^k, 0) + \max(d_v^k, 0). \tag{4}$$

Please refer to the supplementary materials for more details.

Physical Violation Loss. To mitigate physical occlusions that the magnetic loss does not cover (*e.g.*, chair under the desk), inspired by Zhang *et al.* [Zhang et al. 2021b], we add a physical violation loss based on the neural SDF model. Different from Zhang *et al.* [Zhang et al. 2021b] which uniformly samples points inside the bounding boxes, we only sample points on the visible surface with outside-in sphere tracing, so as to make the optimization more efficient when two objects only collide partially at a shallow level. The physical violation loss is defined by punishing P surface points for each object k when an object's points lie inside its O neighbor objects by querying the corresponding SDF surface model F_{SDF} as:

$$L_{vio} = \sum_{k=1}^{K} \sum_{o=0}^{O} \sum_{p=1}^{P} \min(\text{SDF}(\mathbf{x}_{kp})_o + \epsilon, 0), \tag{5}$$

where $o = 0$ denotes the scene background, and we set $\epsilon = 0.025$, $O = 3$ and $P = 1000$ in our experiment.

Gravity Direction Loss. In real-world scenarios, many furniture like beds and tables only rotate around the gravity direction (*i.e.*, rotation uncertainty only on the yaw angle). So we also add the gravity-direction energy term to the physical constraints for those objects as:

$$L_g = \sum_{j=1}^{K} \text{sim}(\hat{\mathbf{R}}^k \mathbf{g}, \mathbf{g}), \tag{6}$$

where $\mathbf{g} = [0, 0, 1]^\top$ is the gravity direction .

The overall physical loss is defined as: $L_{phy} = L_{mag} + L_{vio} + L_g$.

3.2.5 *Holistic Optimization.*

In holistic optimization, we seek for per-object poses $\hat{\mathbf{T}}^k$, object and background appearance codes $\boldsymbol{l}_a^k$ and light codes $\boldsymbol{l}_l^k$ that satisfy the input panoramic image. To fulfill this goal, we jointly optimize photometric loss, observation loss and physical constraint losses at the online stage, as:

$$L = \lambda_{pho} L_{pho} + \lambda_{obs} L_{obs} + \lambda_{phy} L_{phy}. \tag{7}$$

We use $\lambda_{pho} = 1$, $\lambda_{lbs} = 100$, and $\lambda_{phy} = 1$ in our experiment. The total optimization takes about 10-15 minutes (depending on the frequency of object occlusions) for a panoramic image with 500 iterations on an Nvidia RTX3090-24G graphics card. More discussion of the time-consuming and the possible improvement at the online stage can be found in Sec. 5.

4 EXPERIMENTS

In this section, we first compare our scene arrangement prediction with DeepPanoContext [Zhang et al. 2021a] and evaluate the scene lighting adaptation ability both quantitatively and qualitatively. Then, we perform ablation studies to analyze the design of our framework. Finally, we demonstrate the applicability of our method on scene touring, scene illumination interpolation, and scene editing.

4.1 Dataset

iG-Synthetic. We use iGibson [Shen et al. 2020] simulator to synthesize labeled images with depth, segmentation and 3D bounding boxes for training and testing. For training object-centric models for identification, pose estimation or neural rendering, we generate 360° views around each object (similar to Realistic Synthetic 360° in NeRF [Mildenhall et al. 2020]). For the background scenes, we leverage the toolbox from Zhang *et al.* [Zhang et al. 2021a] to generate panoramic views of the iGibson scenes. Since many rooms in iGibson are either too empty (*e.g.*, bathroom and storage-room) or filled with fixed stuff (*e.g.*, basin and oven in kitchen), we thus select four representative scenes (*i.e.*, bedroom, lobby, child's room and home office) which already covers most of the movable object types in the dataset.

Fresh-Room. To demonstrate the efficacy in real-world scenes, we create a new dataset named Fresh-Room, which contains RGB-D posed training images for 5 objects and the room background captured by iPad Pro. We also capture multiple panoramic testing images under 4 different setups with varying arrangements and lighting conditions using a 360°camera (Insta360 ONE-R). We utilize the SfM system with mesh reconstruction [Kazhdan et al. 2006;

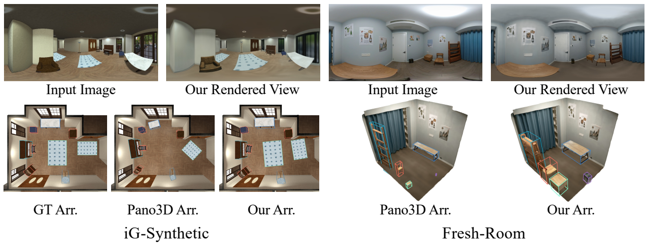


Fig. 5. Scene arrangement visualization with textured meshes, where objects are scaled by the estimated bounding boxes. Note that for the Fresh-Room, we use the object meshes extracted from our neural implicit renderer.

Table 1. Quantitative evaluation on scene arrangement prediction.

Scene	DeepPanoContext			Ours		
	IoU (%) ↑	ARE (°) ↓	APE (cm) ↓	IoU (%) ↑	ARE (°) ↓	APE (cm) ↓
Lobby	26.34	52.80	23.47	**44.48**	**33.99**	**10.08**
Bedroom	40.61	30.33	11.43	**55.42**	**25.88**	**9.14**
Child's Room	27.76	34.64	17.78	**48.63**	**29.57**	**9.08**
Home Office	38.04	27.47	11.45	**47.91**	**19.95**	**6.46**
Average	33.19	36.31	16.03	**49.11**	**27.35**	**8.69**

Schönberger and Frahm 2016] and ARKit metadata[2] to recover camera poses with real-world scale and obtain 2D segmentation by projecting annotated labels from 3D meshes for training data.

4.2 Scene Arrangement Prediction

We first evaluate scene arrangement prediction on iG-Synthetic dataset and our Fresh-Room dataset. For iG-Synthetic dataset, we reorganize the scene arrangement following room examples [Shen et al. 2020], producing unseen arrangements for four scenes, and synthesizing testing data with 5 unseen indoor illuminations based on the iGibson PBR engine. Since there is no amodal scene understanding approach for comparison, we take DeepPanoContext [Zhang et al. 2021a] (Pano3D) as a reference, which is a SOTA method for general holistic 3D scene understanding with a panoramic input. The Intersection over Union (IoU), Average Rotation Error (ARE) and Average Position Error (APE) are used as evaluation metrics. As demonstrated in Fig. 5 and Tab. 1, our method consistently achieves better scene arrangement prediction quality both quantitatively and qualitatively under a closed scene, where the furniture like shelf and piano are faithfully placed with accurate size, while the general scene understanding method (DeepPanoContext) struggles to produce satisfying results (*e.g.*, the desk and the piano are tilted in iG-Synthetic, and the size of shelf and nightstand are distorted in Fresh-Room). This experiment shows that our amodal 3D understanding approach makes a further step towards a perfect 3D understanding, which benefits from the offline preparation stage.

4.3 Scene Lighting Adaptation

Since we decouple lighting variation implicitly in a latent space ($\boldsymbol{l}_l$) and explicitly via tone adjuster, hence we can adjust the renderer to fit the lighting condition at test time. As shown in Fig. 6, the input images (first row) have dramatically different lighting variations such as local highlight and global warm light. When the lighting adaptation is disabled (second row), the rendered results are close to the pre-captured training views, where the rendered furniture comes up with inconsistent shininess (*e.g.*, the shelf in the third column are inevitably brighter than the real image). By introducing implicit light code optimization (third row), the rendered scenes are closer to the input ground-truth but struggles to adapt to the extremely warm light in the last column, where the global tone has turned yellow but the floor color and the curtain color are distorted. By enabling the tone adjuster (fourth row) only, we can also handle a certain degree of lighting variation (*e.g.*, carpet and desk lit by yellow light in the second row, warm light in fourth column), but fails to adapt the local lighting variation (*e.g.*, scene background partially lit by strong light in first column). When the tone adjuster and the light code optimization are both enabled, we successfully render images with local highlight and global consistent tone, and also achieve the best metric performance as demonstrated in Tab. 2. We believe that the tone adjuster effectively reduces the burden of latent space optimization, and the combination of explicit and implicit optimization enhances the lighting adaptation ability.

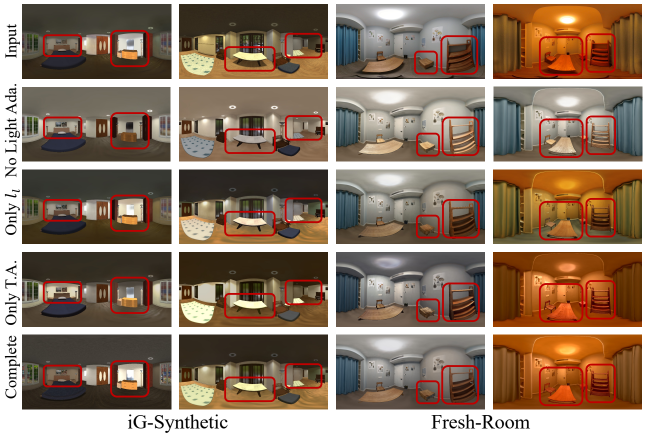


Fig. 6. **Lighting Adaptation.** We can adapt lighting condition to the input panorama with light code optimization ($\boldsymbol{l}_l$) and tone adjuster (T.A.).

Table 2. Ablation study of light code optimization (l_l) and tone adjuster (T.A.) of light adaptation.

Config.	iGibson-Synthetic			Real-Room		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
No light ada.	13.939	0.674	0.359	13.832	0.592	0.480
Only T.A.	16.836	0.732	0.334	17.302	0.615	0.318
Only l_l	23.015	0.809	0.302	19.471	0.669	0.345
Complete	**24.073**	**0.821**	**0.279**	**21.341**	**0.683**	**0.288**

4.4 Ablation Studies

Data Augmentation. We first analyze the effectiveness of our data augmentation with compositional neural rendering for object prediction. Specifically, we use the ODN network [Nie et al. 2020; Zhang

[2] https://developer.apple.com/documentation/arkit/arcamera

Table 3. Ablation studies for the data augmentation and various constraints.

Config.	iGibson / Lobby			iGibson / Bedroom		
	IoU (%) ↑	ARE (°) ↓	APE (cm) ↓	IoU (%) ↑	ARE (°) ↓	APE (cm) ↓
ODN	10.75	58.46	38.29	16.30	33.89	26.33
ODN w aug.	13.01	39.17	45.50	20.43	32.12	23.70
w/o aug.	37.01	50.93	13.53	56.99	38.20	8.27
w/o L_{pho}	42.34	33.92	10.17	51.54	26.10	10.31
w/o L_{phy}	16.93	43.81	35.61	23.65	37.69	23.31
w/o L_{obs}	33.45	34.88	16.76	44.91	24.48	15.40
only L_{pho}	21.05	42.24	34.00	20.57	34.22	28.99
Complete	44.48	33.99	10.08	55.42	25.88	9.14

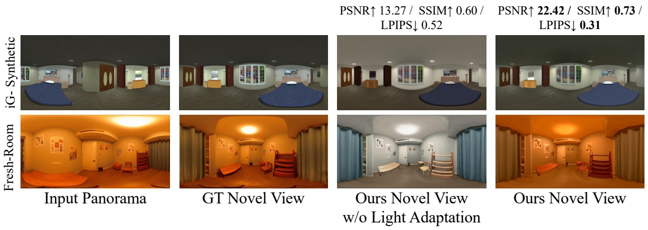

PSNR↑ 13.27 / SSIM↑ 0.60 / LPIPS↓ 0.52
PSNR↑ 22.42 / SSIM↑ 0.73 / LPIPS↓ 0.31
iG- Synthetic
Fresh-Room
Input Panorama
GT Novel View
Ours Novel View w/o Light Adaptation
Ours Novel View

Fig. 7. **Free-viewpoint scene touring** on the iG-Synthetic dataset and Fresh-Room dataset.

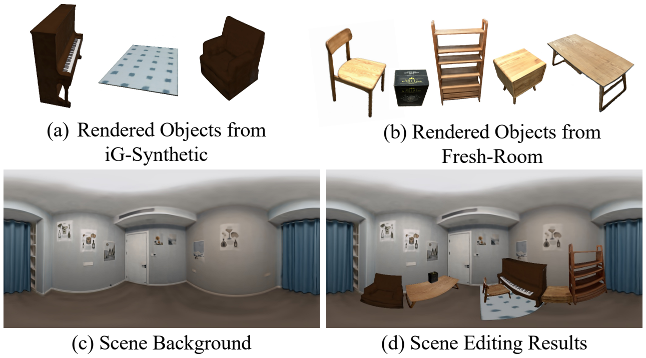
et al. 2021a] as a baseline and fine-tune on the labeled panoramic images rendered by our neural implicit renderer. The results in Tab. 3 show that our proposed data augmentation improves the object prediction quality on the rotation error and IoU (first two rows), and also boosts the performance of scene arrangement prediction in holistic optimization (third row and last row).

Various Constraints. We then inspect the efficacy of various constraints in our holistic optimization, including photometric constraint in Sec. 3.2.2, observation constraint in Sec. 3.2.3 and physical constraint in Sec. 3.2.4. Note that we exclude lighting adaptation as this process is mainly for better rendering quality and performed after the object pose optimization (see supplementary for more details). As shown in Tab. 3 (last five rows), all these loss terms improve the overall scene arrangement quality. Furthermore, we evaluate the performance when only a bare photometric loss is enabled. It is clear that without physical and observation constraints, a simple photometric loss is prone to be unstable in such cluttered scenes.

Safe-Region Volume Rendering. We also compare our proposed safe-region volume rendering with the classical volume rendering pipeline (*e.g.*, used by iNeRF [Yen-Chen et al. 2021]) in the pose optimization task with an Nvidia RTX-3090-24GB graphics card. During the experiment, we optimize object poses by jointly rendering background scenes and target objects with back-propagation, and report the GPU memory usage by varying the number of sample rays and target objects. As shown in Tab. 4, our proposed strategy significantly reduces the number of network queries and GPU memory consumption and can simultaneously optimize 10 objects, while the classical volume rendering fails due to out of memory. We verify the impact on the pose estimation quality in the supplementary material, which shows that this strategy maintains similar pose convergence performance as classical volume rendering.

Table 4. Ablation study for the safe-region volume rendering (S.R.) with different number of target objects and rays. Bg.+1/10 Obj. denotes joint rendering with background and 1 or 10 objects. × denotes out of memory.

Config.	S.R./Bg.+1 Obj.	no S.R./Bg.+1 Obj.	S.R./Bg.+10 Obj.	no S.R./Bg.+10 Obj.
# Rays	# Query / GPU Memory			
256	**5.2M / 2.5G**	19.7M / 6.4G	**18.1M / 3.3G**	51.4M / 7.8G
512	**23.0M / 3.4G**	86.4M / 11.1G	**70.8M / 4.3G**	220.8M / 13.6G
1024	**91.9M / 5.2G**	346M / 20.2G	**287.0M / 6.4G**	×
2048	**274.0M / 6.9G**	×	**1170.2M / 10.5G**	×


(a) Rendered Objects from iG-Synthetic

(b) Rendered Objects from Fresh-Room

(c) Scene Background

(d) Scene Editing Results

Fig. 8. **Scene Editing.** We insert virtual objects (piano, sofa chair and carpet) into the real-world.

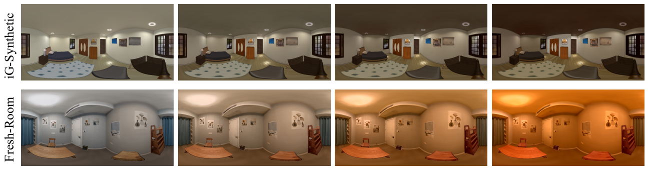


Fig. 9. **Illumination interpolation** on the iG-Synthetic dataset and Fresh-Room dataset.

4.5 Free-viewpoint Scene Touring

Once we resolve the scene arrangement and scene lighting condition, it is feasible to re-render the room in any arbitrary view, which enables virtual touring of the room. To inspect the rendering quality for this task, we conduct a scene re-rendering experiment by fitting input images (first column in Fig. 7) and render another view with the fitting results. Thanks to our neural scene representation, the rendered novel views (last column in Fig. 7) vividly reproduce scene appearance and lighting conditions (*e.g.*, local highlight and global warm light) of the corresponding ground-truth novel views (second column in Fig. 7). As a comparison, when ablating the lighting adaptation from the representation, we can still achieve realistic novel view rendering results, but the specific lighting conditions (*e.g.*, local highlight and warm tone) are no longer kept (third column in Fig. 7), which also results in lower metric performances.

4.6 Scene Editing & Illumination Interpolation

Since our neural implicit renderer has already learned to render the scene background and objects, we can easily edit or composite novel scenes upon this. As shown in Fig. 8, we perform scene editing by inserting virtual objects learned from iG-Synthetic into the real scene Fresh-Room, and the rendered image of novel view demonstrates correct space relationship with seamless object-object occlusion. We also conduct the illumination interpolation experiment in Fig. 9, where the scene lighting temperature can be naturally turned from day to night (first row), or from cold to warm (second row).

5 CONCLUSION

We propose a novel 3D scene understanding and rendering paradigm for closed environments. Given a single panoramic image as input, our method can reliably estimate 3D scene arrangement and the lighting condition via a holistic neural-rendering-based optimization framework. The proposed method also enables free-viewpoint scene touring and editing by changing illumination or objects' placement. Despite the novel capabilities provided by our method, it still has its limitations. First, since we assume the neural implicit models are pre-built, our method cannot handle the cases with unobserved objects. Second, the computational efficiency of the online stage is currently not ready for real-time performance, which is due to the intensive network queries of MLPs. There are some existing approaches accelerating neural volumetric rendering from 0.06FPS to 200FPS, e.g., by using local-bounded representation [Reiser et al. 2021], cached coefficients [Garbin et al. 2021; Sara Fridovich-Keil and Alex Yu et al. 2022], or multiresolution voxel-hashing [Müller et al. 2022], and they can be applied for real-time rendering and fast optimization, which is a promising future direction. Third, the proposed method still cannot handle deformed/recolored objects and extremely harsh lighting that severely violates photometric consistency, or render transparent surfaces and fine-grained light effects like shadows and indirect illumination. Finally, our lighting augmentation is not well-defined for glossy materials like mirrors and glasses, which can be improved by introducing material estimation in the future.

ACKNOWLEDGMENTS

We thank Hanqing Jiang and Liyang Zhou for their kind help in object reconstruction. This work was partially supported by NSF of China (No. 62102356, No. 61932003) and Zhejiang Lab (2021PE0AC01).

REFERENCES

Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. 2016. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 5297–5307.

Armen Avetisyan, Tatiana Khanova, Christopher Choy, Denver Dash, Angela Dai, and Matthias Nießner. 2020. SceneCAD: Predicting Object Alignments and Layouts in RGB-D Scans. In *European Conference on Computer Vision.* Springer, 596–612.

Alexey Bokhovkin, Vladislav Ishimtsev, Emil Bogomolov, Denis Zorin, Alexey Artemov, Evgeny Burnaev, and Angela Dai. 2021. Towards Part-Based Understanding of RGB-D Scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 7484–7494.

Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. 2019. Learning to Predict 3D Objects with an Interpolation-based Differentiable Renderer. *Advances in Neural Information Processing Systems* 32.

James M Coughlan and Alan L Yuille. 1999. Manhattan World: Compass Direction from a Single Image by Bayesian Inference. In *Proceedings of the seventh IEEE international conference on computer vision*, Vol. 2. IEEE, 941–947.

Manuel Dahnert, Ji Hou, Matthias Nießner, and Angela Dai. 2021. Panoptic 3D Scene Reconstruction From a Single RGB Image. *Advances in Neural Information Processing Systems* 34.

Peng Dai, Yinda Zhang, Zhuwen Li, Shuaicheng Liu, and Bing Zeng. 2020. Neural Point Cloud Rendering via Multi-Plane Projection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 7830–7839.

Saumitro Dasgupta, Kuan Fang, Kevin Chen, and Silvio Savarese. 2016. DeLay: Robust Spatial Layout Estimation for Cluttered Indoor Scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 616–624.

Paul E. Debevec. 2006. Image-based Lighting. In *International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2006, Boston, Massachusetts, USA, July 30 - August 3, 2006, Courses,* John W. Finnegan and Dave Shreiner (Eds.). ACM, 4.

Terrance DeVries, Miguel Angel Bautista, Nitish Srivastava, Graham W Taylor, and Joshua M Susskind. 2021. Unconstrained Scene Generation with Locally Conditioned Radiance Fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 14304–14313.

Yilun Du, Zhijian Liu, Hector Basevi, Ales Leonardis, Bill Freeman, Josh Tenenbaum, and Jiajun Wu. 2018. Learning to Exploit Stability for 3D Scene Parsing. In *NeurIPS.* 1733–1743.

Stephan J Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. 2021. FastNeRF: High-Fidelity Neural Rendering at 200FPS. In *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 14346–14355.

Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. 1996. The Lumigraph. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques.* 43–54.

Jonathan Granskog, Till N Schnabel, Fabrice Rousselle, and Jan Novák. 2021. Neural Scene Graph Rendering. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–11.

Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 2018. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 216–224.

Michelle Guo, Alireza Fathi, Jiajun Wu, and Thomas Funkhouser. 2020. Object-Centric Neural Scene Rendering. *arXiv preprint arXiv:2012.08503* (2020).

Siyuan Huang, Siyuan Qi, Yinxue Xiao, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. 2018a. Cooperative Holistic Scene Understanding: Unifying 3D Object, Layout, and Camera Pose Estimation. *Advances in Neural Information Processing Systems* 31.

Siyuan Huang, Siyuan Qi, Yixin Zhu, Yinxue Xiao, Yuanlu Xu, and Song-Chun Zhu. 2018b. Holistic 3D Scene Parsing and Reconstruction from a Single RGB Image. In *Proceedings of the European conference on computer vision (ECCV).* 187–203.

Hamid Izadinia, Qi Shan, and Steven M Seitz. 2017. IM2CAD. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 5134–5143.

Michael M. Kazhdan, Matthew Bolitho, and Hugues Hoppe. 2006. Poisson Surface Reconstruction. In *Proceedings of Eurographics Symposium on Geometry Processing.* 61–70.

Byung-soo Kim, Pushmeet Kohli, and Silvio Savarese. 2013. 3D Scene Understanding by Voxel-CRF. In *Proceedings of the IEEE International Conference on Computer Vision.* 1425–1432.

Marc Levoy and Pat Hanrahan. 1996. Light Field Rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques.* 31–42.

Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. 2020. Inverse Rendering for Complex Indoor Scenes: Shape, Spatially-Varying Lighting and SVBRDF From a Single Image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2475–2484.

Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. 2019. Soft Rasterizer: A Differentiable Renderer for Image-Based 3D Reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 7708–7717.

Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. 2020. DIST: Rendering Deep Implicit Signed Distance Function with Differentiable Sphere Tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2019–2028.

Stephen Lombardi, Tomas Simon, Jason M. Saragih, Gabriel Schwartz, Andreas M. Lehrmann, and Yaser Sheikh. 2019. Neural Volumes: Learning Dynamic Renderable Volumes from Images. *ACM Trans. Graph.* 38, 4 (2019), 65:1–65:14.

Andrew Luo, Zhoutong Zhang, Jiajun Wu, and Joshua B Tenenbaum. 2020. End-to-End Optimization of Scene Layout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 3754–3763.

Arun Mallya and Svetlana Lazebnik. 2015. Learning Informative Edge Maps for Indoor Scene Layout Prediction. In *Proceedings of the IEEE international conference on computer vision.* 936–944.

Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. 2021. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 7210–7219.

Wojciech Matusik, Hanspeter Pfister, Matthew Brand, and Leonard McMillan. 2003. A Data-Driven Reflectance Model. *ACM Trans. Graph.* 22, 3, 759–769.

Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. 2019. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. *ACM Trans. Graph.* 38, 4 (2019), 29:1–29:14.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *European conference on computer vision*. Springer, 405–421.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *arXiv preprint arXiv:2201.05989* (2022).

Yinyu Nie, Xiaoguang Han, Shihui Guo, Yujian Zheng, Jian Chang, and Jian Jun Zhang. 2020. Total3DUnderstanding: Joint Layout, Object Pose and Mesh Reconstruction for Indoor Scenes From a Single Image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 55–64.

Michael Niemeyer, Lars M. Mescheder, Michael Oechsle, and Andreas Geiger. 2020. Differentiable Volumetric Rendering: Learning Implicit 3D Representations Without 3D Supervision. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3501–3512.

Michael Oechsle, Songyou Peng, and Andreas Geiger. 2021. UNISURF: Unifying Neural Implicit Surfaces and Radiance Fields for Multi-View Reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5589–5599.

Julian Ost, Fahim Mannan, Nils Thuerey, Julian Knodt, and Felix Heide. 2021. Neural Scene Graphs for Dynamic Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2856–2865.

Stefan Popov, Pablo Bauszat, and Vittorio Ferrari. 2020. CoReNet: Coherent 3D Scene Reconstruction from a Single RGB Image. In *European Conference on Computer Vision*. Springer, 366–383.

Srikumar Ramalingam, Jaishanker K Pillai, Arpit Jain, and Yuichi Taguchi. 2013. Manhattan Junction Catalogue for Spatial Reasoning of Indoor Scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3065–3072.

Erik Reinhard, Michael Adhikhmin, Bruce Gooch, and Peter Shirley. 2001. Color Transfer Between Images. *IEEE Computer graphics and applications* 21, 5 (2001), 34–41.

Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. 2021. KiloNeRF: Speeding up Neural Radiance Fields with Thousands of Tiny MLPs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14335–14345.

Gernot Riegler and Vladlen Koltun. 2020. Free View Synthesis. In *European Conference on Computer Vision*. Springer, 623–640.

Gernot Riegler and Vladlen Koltun. 2021. Stable View Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12216–12225.

Sara Fridovich-Keil and Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. 2022. Plenoxels: Radiance Fields without Neural Networks. In *CVPR*.

Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. 2019. From Coarse to Fine: Robust Hierarchical Localization at Large Scale. In *CVPR*.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. SuperGlue: Learning Feature Matching with Graph Neural Networks. In *CVPR*.

Johannes L. Schönberger and Jan-Michael Frahm. 2016. Structure-from-Motion Revisited. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 4104–4113.

Bokui Shen, Fei Xia, Chengshu Li, Roberto Martín-Martín, Linxi Fan, Guanzhi Wang, Claudia Pérez-D'Arpino, Shyamal Buch, Sanjana Srivastava, Lyne Tchapmi, et al. 2020. iGibson 1.0: A Simulation Environment for Interactive Tasks in Large Realistic Scenes. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 7520–7527.

Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer. 2019a. DeepVoxels: Learning Persistent 3D Feature Embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2437–2446.

Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. 2019b. Scene Representation Networks: Continuous 3D-Structure-Aware Neural Scene Representations. In *Proceedings of Advances in Neural Information Processing Systems*. 1119–1130.

Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. 2017. Semantic Scene Completion from a Single Depth Image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1746–1754.

Shuran Song, Linguang Zhang, and Jianxiong Xiao. 2015. Robot In a Room: Toward Perfect Object Recognition in Closed Environments. *CoRR, abs/1507.02703* (2015).

Cheng Sun, Chi-Wei Hsiao, Min Sun, and Hwann-Tzong Chen. 2019. HorizonNet: Learning Room Layout With 1D Representation and Pano Stretch Data Augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1047–1056.

Richard Tucker and Noah Snavely. 2020. Single-View View Synthesis With Multiplane Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 551–560.

Michael Waechter, Nils Moehrle, and Michael Goesele. 2014. Let There Be Color! — Large-Scale Texturing of 3D Reconstructions. In *Proceedings of the European Conference on Computer Vision*. Springer.

Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. 2018. Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 52–67.

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021a. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. *NeurIPS*.

Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. 2021b. IBRNet: Learning Multi-View Image-Based Rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4690–4699.

Chenggang Yan, Biyao Shao, Hao Zhao, Ruixin Ning, Yongdong Zhang, and Feng Xu. 2020. 3D Room Layout Estimation From a Single RGB Image. *IEEE Transactions on Multimedia* 22, 11 (2020), 3014–3024.

Bangbang Yang, Yinda Zhang, Yinghao Xu, Yijin Li, Han Zhou, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. 2021. Learning Object-Compositional Neural Radiance Field for Editable Scene Rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13779–13788.

Hao Yang and Hui Zhang. 2016. Efficient 3D Room Shape Recovery from a Single Panorama. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5422–5430.

Shang-Ta Yang, Fu-En Wang, Chi-Han Peng, Peter Wonka, Min Sun, and Hung-Kuo Chu. 2019. DuLa-Net: A Dual-Projection Network for Estimating Room Layouts From a Single RGB Panorama. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3363–3372.

Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. 2021. Volume Rendering of Neural Implicit Surfaces. *Advances in Neural Information Processing Systems* 34.

Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. 2021. iNeRF: Inverting Neural Radiance Fields for Pose Estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 1323–1330.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 2021. pixelNeRF: Neural Radiance Fields From One or Few Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4578–4587.

Greg Zaal, Sergej Majboroda, and Andreas Mischok. 2020. Poly Haven. https://polyhaven.com/. Accessed: 2022-05-03.

Xiaohang Zhan, Xingang Pan, Bo Dai, Ziwei Liu, Dahua Lin, and Chen Change Loy. 2020. Self-Supervised Scene De-Occlusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3784–3792.

Cheng Zhang, Zhaopeng Cui, Cai Chen, Shuaicheng Liu, Bing Zeng, Hujun Bao, and Yinda Zhang. 2021a. DeepPanoContext: Panoramic 3D Scene Understanding With Holistic Scene Context Graph and Relation-Based Optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 12632–12641.

Cheng Zhang, Zhaopeng Cui, Yinda Zhang, Bing Zeng, Marc Pollefeys, and Shuaicheng Liu. 2021b. Holistic 3D Scene Understanding from a Single Image with Implicit Representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8833–8842.

Yinda Zhang, Mingru Bai, Pushmeet Kohli, Shahram Izadi, and Jianxiong Xiao. 2017. DeepContext: Context-Encoding Neural Pathways for 3D Holistic Scene Understanding. In *Proceedings of the IEEE international conference on computer vision*. 1192–1201.

Yinda Zhang, Shuran Song, Ping Tan, and Jianxiong Xiao. 2014. PanoContext: A Whole-Room 3D Context Model for Panoramic Scene Understanding. In *European conference on computer vision*. Springer, 668–686.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. 2019. On the Continuity of Rotation Representations in Neural Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5745–5753.

Chuhang Zou, Alex Colburn, Qi Shan, and Derek Hoiem. 2018. LayoutNet: Reconstructing the 3D Room Layout From a Single RGB Image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2051–2059.