Data-driven Digital Lighting Design for Residential Indoor Spaces

HAOCHENG REN, HANGMING FAN, and RUI WANG, State Key Lab of CAD&CG, Zhejiang University
YUCHI HUO, Zhejiang Lab and State Key Lab of CAD&CG, Zhejiang University
RUI TANG, KooLab, Manycore Tech Inc.
LEI WANG, RaysEngine Tech Inc.
HUJUN BAO, State Key Lab of CAD&CG, Zhejiang University

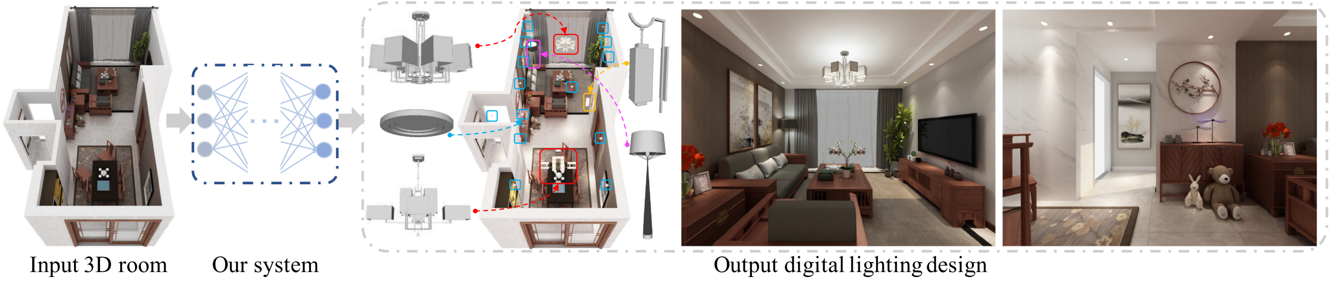


Fig. 1. Given an interior scene with furniture installed, our system automatically creates a digital lighting design that places different types of lights in the scenes as well as generates pleasing lighting effects.

Conventionally, interior lighting design is technically complex yet challenging and requires professional knowledge and aesthetic disciplines of designers. This article presents a new digital lighting design framework for virtual interior scenes, which allows novice users to automatically obtain lighting layouts and interior rendering images with visually pleasing lighting effects. The proposed framework utilizes neural networks to retrieve and learn underlying design guidelines and the principles beneath the existing lighting designs, e.g., a newly constructed dataset of 6k 3D interior scenes from professional designers with dense annotations of lights. With a 3D furniture-populated indoor scene as the input, the framework takes two stages to perform lighting design: (1) lights are iteratively placed in the room; (2) the colors and intensities of the lights are optimized by an adversarial scheme, resulting in lighting designs with aesthetic lighting effects. Quantitative and qualitative experiments show that the proposed framework effectively learns the guidelines and principles and generates lighting designs that are preferred over the rule-based baseline and comparable to those of professional human designers.

CCS Concepts: • **Computer Graphics → Interior Lighting Design**; **Synthetic Dataset;**

Additional Key Words and Phrases: Lighting design, interior design, data-driven approach, neural network, deep learning

ACM Reference format:
Haocheng Ren, Hangming Fan, Rui Wang, Yuchi Huo, Rui Tang, Lei Wang, and Hujun Bao. 2023. Data-driven Digital Lighting Design for Residential Indoor Spaces. *ACM Trans. Graph.* 42, 3, Article 28 (March 2023), 18 pages.
https://doi.org/10.1145/3582001

1 INTRODUCTION

With the wide usage of digital modeling and design software in interior design, digital lighting [Birn 2014] is one of the most important elements in designing an interior space. However, lighting design is technically complex and difficult. It conventionally requires the professional knowledge and aesthetic discipline of designer, because the design of lighting is not only used to lighten a space but also used to express certain artistry and atmosphere in interior design. Particularly in the case of interior digital lighting design, which aims to generate visually pleasing lighting configurations for virtual rooms, aesthetic consideration is often needed. In the interior design industry, interior designers need to show their design ideas using photorealistic and visually pleasing renderings or VR experiences. Nowadays, general users can also redesign their homes through online interior design tools [Coohom 2022; Planner5d 2022]. In a standard digital lighting design procedure, a designer must model and adjust the lighting in the 3D virtual indoor scene and perform various lighting experiments [Birn 2014] to achieve the desired lighting effect. Even with the latest commercial digital design tools [3ds Max 2021; Maya 2021; VRay 2021], the entire process is still iterative with trials and errors, making the

The work was partially supported by NSFC (No. 61872319), Key R&D Program of Zhejiang Province (No. 2023C01039), the Fundamental Research Funds for the Central Universities, Zhejiang Lab (121005-PI2101), and Information Technology Center and State Key Lab of CAD&CG, Zhejiang University.

Authors' addresses: H. Ren, H. Fan, R. Wang (corresponding author), and H. Bao, State Key Lab of CAD&CG, Zhejiang University, No. 866, Yuhangtang Road, Xihu District, Hangzhou, Zhejiang Province, 310058, China; emails: {swordigo, 12021033, ruiwang}@zju.edu.cn, bao@cad.zju.edu.cn; Y. Huo (corresponding author), Zhejiang Lab, No.1, Kechuang Avenue, Yuhang District, Hangzhou, Zhejiang Province, 311121, China; email: huo.yuchi.sc@gmail.com; R. Tang, KooLab, Manycore Tech Inc. No. 515, Yuhangtang Road, Gongshu District, Hangzhou, Zhejiang Province, 310000, China; email: ati@qunhemail.com; L. Wang, RaysEngine Tech Inc., No. 2301, Yuhangtang Road, Yuhang District, Hangzhou, Zhejiang Province, 311100, China; email: leowang@raysengine.com.

0730-0301/2023/03-ART28 $15.00
https://doi.org/10.1145/3582001

process time-consuming. An easy-to-use or even automatic digital lighting design approach is needed, especially for novice designers.

Many interactive or automatic digital lighting design tools and methods have been proposed to facilitate design by allowing designers to manipulate additional lighting features [Pellacini et al. 2002; Schoeneman et al. 1993], paint shadows and highlights [Lin et al. 2013; Pellacini et al. 2007], or explore different lighting plans with image galleries [Marks et al. 1997; Shapira et al. 2009]. Jin et al. [2019] proposed an automatic indoor lighting method with heuristic lighting design guidelines. Explicit principles and guidelines of lighting design are important in the aforementioned methods to guarantee accuracy and validity. However, with the aesthetic consideration in the design process, directly and explicitly obtaining these guidelines and principles from digital lighting designers and artists is difficult.

With the rapid development of deep learning-based techniques, data-driven approaches allow researchers to implicitly explore design principles and guidelines. Many data-driven interior design systems have been proposed. These systems seek to automate of interior furniture arrangement [Ritchie et al. 2019; Wang et al. 2019, 2018b], interior floorplan design [Hu et al. 2020; Wu et al. 2019], and architecture and planning design [Chaillou 2019]. Additionally, several datasets of 3D indoor scenes [Fu et al. 2021; Handa et al. 2016a; Li et al. 2021; Roberts et al. 2021; Song et al. 2017] have been presented along with these methods. The results show that the underlying design guidelines and principles in the data could be automatically retrieved and learned using neural networks.

Inspired by these methods, we propose a deep learning-based automatic digital lighting design framework in this article. However, several challenges still exist in the development of such a lighting design framework. First, none of the existing public datasets on 3D interior scenes can be directly used for the lighting design task. Most of these datasets [Avetisyan et al. 2019; Handa et al. 2016a, b; Li et al. 2021; Roberts et al. 2021; Song et al. 2017] focus on scene understanding, and the visual quality of their lighting effect is not guaranteed. Some datasets [Avetisyan et al. 2019; Handa et al. 2016b] lack important lighting information, such as labels of different lights, a list of light fixture models, and correct emission surface of light fixture models. Second, lighting design has unique design guidelines that are different from those for furniture layout or room planning. For example, some lights illuminate the entire space, providing a global atmosphere, while others lighten small regions, enabling local artistic effects. The lighting design should consider many factors of the scene, such as the room structure, furniture layout, and room style. Combining all these factors and learning in 3D space is still challenging.

To tackle the aforementioned challenges, we first construct a dataset with 6k 3D scenes, in which the lighting layouts are designed by professional lighting designers. Annotations of the lights, such as different light type, intensity, and color and emission surfaces, are all labeled and stored in these scenes. The dataset is available for online access[1] in the MINERVAS platform [Ren et al. 2022] to inspire more research. Then, we design an automatic lighting procedure with two steps: (1) selection and arrangement of lights in the scene; (2) optimization of the colors and intensities of lights. A light fixture arrangement pipeline extended from the iterative prediction scheme [Ritchie et al. 2019] is built in the first stage, where we extend the image-based scene representation [Wang et al. 2018b] to specifically represent the spatial information of the room for lighting design, such as encoding the ceiling and walls. Once the light placement is completed, another deep learning-based scheme is employed in the second stage to compute the intensity and color of lights. Technically, an adversarial network is trained and utilized to guide this optimization, where synthetic images and real indoor photographs are considered in the training process to facilitate the capture of additional lighting styles in real indoor scenes through optimization.

The experimental results demonstrate that the proposed framework effectively learns the lighting design principles in the dataset and generates quantitatively and qualitatively good results. User studies show that the lighting design results are comparable to those of professional human designers.

The main contributions of this article are as follows:

- The first deep learning-based automatic interior digital lighting design framework, which generates lighting design results comparable with those of human designers.
- An interior scene dataset including good lighting layouts with extra information and annotations of lights.
- An image-based representation of scenes for lighting design, which not only contains room layout information but also includes ceiling and walls.
- An adversarial light intensity optimization, including synthetic and real interior lighting design images, resulting in aesthetic lighting effects with diverse lighting styles.

2 RELATED WORK

Computer-aided lighting design. There are two main goals in the computer-aided lighting design. One goal is pleasing and aesthetic lighting of 3D scenes for rendering [Birn 2014], which is the aim of our work. Another goal is helping the real-world lighting design with accurate computer-based lighting simulation considering physical realization [Gordon 2015]. Both the lighting design in real-world and digital 3D scenes is a complex task requiring professional interior lighting design principles and guidelines [Birn 2014; Gordon 2015]. In both of these tasks, designers need to adjust the position, color, and intensity of each luminaire, which is considerably time-consuming, especially in a trial-and-error loop. Therefore, providing a good computer-aided design tool for lighting designers is a long-standing topic. The goal-based rendering method [Kawai et al. 1993] was proposed to provide designers with an intuitive method of lighting design to address this problem. Some studies [Pellacini et al. 2002; Schoeneman et al. 1993] enabled designers to directly manipulate lighting features, such as shadows and highlights, instead of specific light parameters. Pellacini et al. [2007] provided a painting interface to allow designers to paint the lighting effect. Kerr et al. [2009] gave a formal evaluation of the lighting design interface. Lin et al. [2013] used a coarse-to-fine strategy with a hierarchical light representation to determine the optimal lighting parameters given painting strokes of shadows and highlights. With this strategy, users can control the number of lights by providing different thresholds.

[1] https://coohom.github.io/MINERVAS/lighting-design.

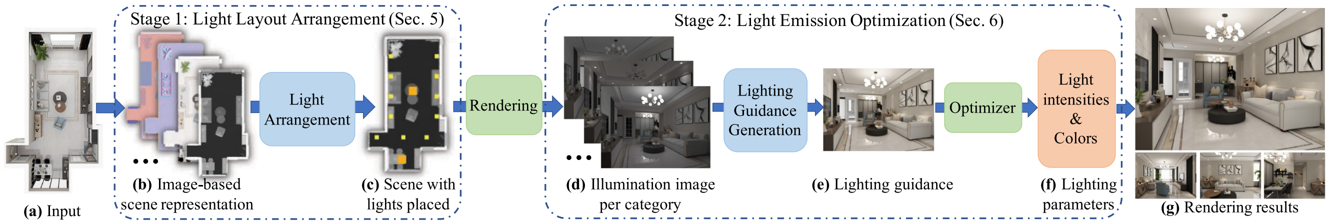

Stage 1: Light Layout Arrangement (Sec. 5)
Stage 2: Light Emission Optimization (Sec. 6)
Light Arrangement
Rendering
Lighting Guidance Generation
Optimizer
Light intensities & Colors
(a) Input
(b) Image-based scene representation
(c) Scene with lights placed
(d) Illumination image per category
(e) Lighting guidance
(f) Lighting parameters
(g) Rendering results

Fig. 2. Overview of our automatic lighting design framework. Given **(a)** an indoor scene with the furniture as input, our pipeline first extracts **(b)** an image-based scene representation and then predicts **(c)** the light arrangement in the room. Once all lights have been placed, we use a path tracer to render **(d)** images with each light on and the others off. Using these illumination images as input, we generate **(e)** a lighting guidance image and optimize **(f)** the intensity and color of each light. Finally, we obtain **(g)** the resultant lighting layout and rendering results with pleasing lighting effects.

In addition to lighting feature manipulation and user-painted goals, Schwarz et al. [2014] defined constraints for exterior lighting design procedurally using extensions to a grammar language, which is originally used to model buildings. These methods facilitate the lighting design process by allowing users to control specific and local lighting effects (e.g., shadows and highlights). A class of methods based on the image gallery has been proposed to help users control the overall tone and mood of the lighting [Marks et al. 1997; Shapira et al. 2009]. Technically, these systems use visual exploration instead of tweaking parameters, that is, providing a batch of recommendations and allowing users to iteratively refine the gallery to converge to their ideal design. Shimizu et al. [2019] proposed a system that aids exploratory theatrical lighting design. The color and intensity are determined using the statistics of reference images given by designers. The most recent advancement in lighting design tools is that of Walch et al. [2019], who presented an interactive interface with guidance to improve lighting design, provenance visualizations, and quality information for real-world lighting design workflow.

However, the aforementioned methods all require some human interaction and cannot generate lighting designs automatically. Shacked et al. [2001] provided an automatic lighting design approach for a single object by manually defining and optimizing a perception-based image quality objective function. Jin et al. [2019] had a similar goal to this paper. They designed a set of objective functions based on lighting design guidelines and used procedural and optimization-based approaches to generate the lighting layout. Such explicit guidelines may restrict the diversity and authenticity of the results.

Unlike these works on lighting for single object [Shacked and Lischinski 2001], exterior lighting [Schwarz and Wonka 2014], and real-world interior lighting [Shimizu et al. 2019; Walch et al. 2019], our system focuses on interior digital lighting like [Jin and Lee 2019; Lin et al. 2013], which aims to generate visually good lighting layouts for virtual 3D rooms. It is the first attempt to utilize neural networks to learn lighting design guidelines and principles from a dataset implicitly without defining them manually.

Data-driven interior design. Numerous studies are available in the field of automatic interior design. Most of these studies focus on indoor scene synthesis. Early investigations [Merrell et al. 2011; Yu et al. 2011] require pre-specified sets of objects as input and optimize position and rotation of furniture with interior design principles and statistical pair-wise relationships. Deep learning-based indoor scene synthesis methods have been proposed with the availability of a large-scale 3D scene dataset. Some methods represent the 3D scene and learn object arrangement using recursive neural networks [Li et al. 2019] or generative adversarial networks [Zhang et al. 2020b]. Wang et al. [2018b] proposed a top-down image-based representation and utilized convolutional neural networks to synthesize indoor scenes. Following this work, Ritchie et al. [2019] proposed a method to accelerate the synthesis process and boost the visual quality of synthesized scenes. PlanIT [Wang et al. 2019] is a framework that combines high-level relationship graphs for planning and spatial prior networks for instantiation. Similarly, Wu et al. [2019] generated floorplans given boundaries based on learned spatial prior in the field of floorplan design, and Hu et al. [2020] used graphs to represent floorplans for user-in-the-loop design. Most recently, transformer architecture is introduced in the interior design task to obtain faster [Wang et al. 2021] and order-independent scene synthesis [Paschalidou et al. 2021].

Unlike these data-driven interior design concepts, our work targets a different problem, automatic lighting design, which bears some similarity to automatic indoor scene synthesis but requires a different solution. To our knowledge, the proposed framework is the first deep learning-based interior lighting design work.

Image visual enhancement. The visual perception of a rendering image directly reflects the quality of lighting design. Bychkovsky et al. [2011] presented the MIT-Adobe FiveK dataset, which contains 5k images modified and retouched by five experts. They used this dataset to learn the global tonal adjustment. After this, many deep learning-based image enhancement methods emerged. Yan et al. [2016] proposed a deep learning-based photograph enhancement method. Ignotov et al. [2017] showed that the mapping between photos captured by mobile and DSLR cameras could be effectively learned from neural networks. In recent years, generative adversarial networks [Goodfellow et al. 2014] have achieved significant progress in image synthesis and have successfully generated photorealistic images [Karras et al. 2020]. GANs can also boost the visual perception of images, which can be regarded as a subproblem of image-to-image translation. Isola et al. [Isola et al. 2017] proposed a general image-to-image translation framework based on conditional adversarial networks [Mirza and Osindero 2014]. Two-way GANs, such as CycleGAN [Zhu et al. 2017a], DISCOGAN [Kim et al. 2017], and DualGAN [Yi et al. 2017], have been proposed to address the unpaired dataset problem. Numerous studies further improved the image quality with coarse-to-fine architecture [Wang et al. 2018a], global features in the generator [Chen et al. 2018], conditioned feature modulation [Park et al. 2019; Wang et al. 2018; Xu et al. 2019], and shading-albedo decomposition for rendering [Bi et al. 2019]. In addition to these end-to-end approaches, Hu et al. [2018] utilized deep reinforcement learning and GANs to predict a sequence of filters to enhance photos using the unpaired dataset. A good visual perception metric [Talebi and Milanfar 2018] is also an important factor in enhancing the style and quality of a photograph. Instead of generating a visually enhanced image with networks directly, our work utilizes enhanced images as guidance to optimize light parameters in the 3D scene. In this way, the generated interior lighting images are physically correct and have aesthetic lighting effects.

3 OVERVIEW

Given a 3D furniture-populated indoor scene as input, our digital lighting design framework automatically selects and places lights, computes their colors and intensities, and finally obtains an illuminated 3D scene and rendering results with pleasing lighting effects (Figure 1). The entire process is driven by principles and guidelines learned from existing lighting design data.

When we prepared the data, we found that none of the existing indoor 3D datasets include high-quality annotations of lights. Therefore, we constructed a dataset ourselves, specifically for the lighting design task (Section 4). This contains 6k 3D scenes with lighting layouts designed by professional designers, 8k synthetic images, 3k real interior photographs from the Internet, and a library of different types of lights.

Based on the collected data, we design a two-stage lighting design pipeline with a light arrangement stage (Section 5) and a light emission optimization stage (Section 6). The rationale behind such a two-stage pipeline is that the arrangement of lights is usually functional and sometimes plays a role in decoration. The underlying principles and guidelines of light placement are very relevant to the furniture, furniture layout, room structure, style of room design, and so on. However, the emission of light is different from the light arrangement. It is difficult to learn the emission directly from 3D scenes, whereas it can be learned from interior images. Synthetic images in virtual scenes and photographs captured in the real world provide good guidance on what good artistry lighting should be. Therefore, we design different and specific solutions to learn principles and guidelines of light placement and emission from different types of data.

The two-stage pipeline is illustrated in Figure 2. In the first stage, an iterative placement scheme is presented to place different types and numbers of lights, where an image-based scene representation (Figure 2(b)) is utilized to represent the 3D scene (Figure 2(a)), thereby simplifying the design of the neural networks. After the lights are placed (Figure 2(c)), in the second stage, we use an adversarial scheme to optimize the color and intensity of each light. A lighting guidance prediction network is used to predict an image with pleasing lighting effects from a camera view (Figure 2(e)), where synthetic images and real indoor photographs are both utilized to train the prediction network. Then, we use this lighting guidance as the target to determine the intensity and color of each light (Figure 2(f)). After the entire optimization process, we obtain the 3D room with the lighting layout and optimized lighting parameters (Figure 2(g)).

4 DATASET

For a deep learning-based approach, a well-labeled dataset is important and essential. After surveying most 3D interior scene datasets, we found that no dataset was specifically constructed for interior lighting design. Therefore, we built a dataset ourselves. Figure 3 shows the overview of the proposed dataset. First, it includes about 6k interior scenes (Figure 3(a)), which were all designed by professional interior designers. In the creation process, the professional designers considered both the realism and aesthetics of interior scenes to build realistic digital indoor scenes and generate pleasing renderings for interior design. Note that the interior designs in our dataset were mostly styled in Asia. Each piece of furniture in the scene was given a semantic label, and we used the NYUv2 40 label set [Silberman et al. 2012], which covers most common objects. Each scene includes designer set cameras, each of which corresponds to at least one high-quality synthetic image with elaborately designed digital lighting effects. In addition to these synthetic images, we collected real interior photographs from commercial websites. We also constructed a library of lights that stores all lights used in these scenes. We categorize these lights into groups, and for each light, we store the light type, color and intensity or luminous intensity described by IES profiles and labeled emission surfaces.

3D Scenes. In total, we collected 6,648 3D scenes with 8,177 camera views in these scenes, where each scene has at least one camera view. Each scene contains the complete information of the room structure, furniture layout, and lighting layout with light types and positions. All the scenes and furniture in our dataset are scaled to the same sizes in the real world.

2D images. The visual quality of lighting design is usually conveyed by interior images. Therefore, a set of high-quality, aesthetic interior images is important for lighting design. Since each 3D scene contains at least one manually set camera view, we render these views using our in-house Monte Carlo path tracer to obtain synthetic high-quality interior images, as shown in Figure 3(d). To represent the lighting effect of each light, we also render images with only one light on (see Figure 3(e)). In addition to these synthetic images, we collected a set of real photographs with high-quality interior lighting from *airbnb.com*. These photographs are usually captured by professional photographers and show realistic lighting effects with more details. This helps us to generate more visually pleasing digital lighting in the light emission optimization process (Section 6).

Library of Lights. When designers design interior lighting, they need a library of lights to select and place. While accomplishing this task, we also build a library of lights. We collect all light fixtures in our dataset and group them by type. In total, we have seven types of light fixtures: chandeliers, ceiling lamps, downlights, table lamps, floor lamps, wall lamps, and bedside pendant lights. Example 3D models of the light fixtures are shown in Figure 3(c). Each light model should have appropriate emission surfaces rather than

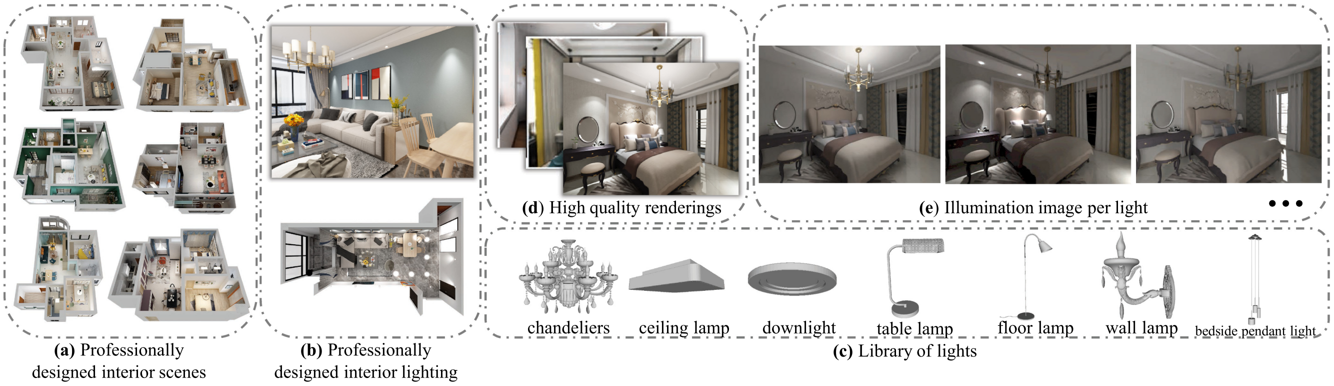
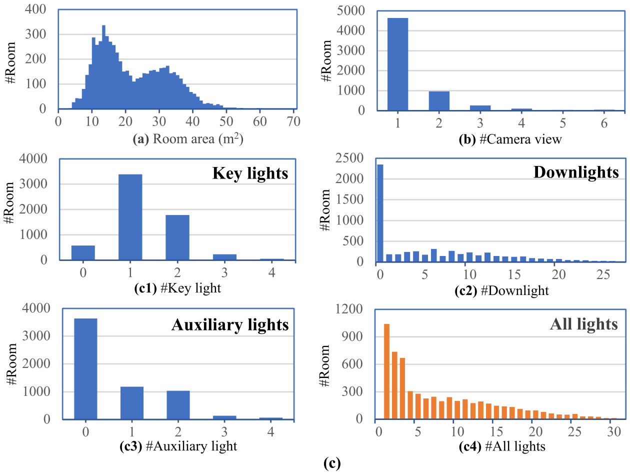

(d) High quality renderings
(e) Illumination image per light
• • •
chandeliers
ceiling lamp
downlight
table lamp
floor lamp
wall lamp
bedside pendant light
(c) Library of lights
(a) Professionally designed interior scenes
(b) Professionally designed interior lighting

Fig. 3. Overview of the dataset. Our dataset contains **(a)** professionally designed scenes and **(b)** lighting. We build **(c)** a library of lights, which covers all lights used in the dataset. We use a path tracer to render the scenes to obtain **(d)** high-quality synthetic images and **(e)** lighting images for each light.



Light type	#Room	#Average lamp per room
Ceiling lamp	1670	0.3220
Chandelier	4229	1.0007
Downlight	3778	6.2305
Table lamp	1215	0.2890
Floor lamp	560	0.0961
Wall lamp	653	0.1982
Bedside pendant lamp	328	0.0714

(d) #Room in different type of lights

Fig. 4. Statistics of our dataset: **(a)** the room areas, **(b)** the numbers of camera views per room, **(c)** the numbers of light categories per room, and **(d)** the numbers of rooms with different types of lights.

shining light from all surfaces. To ensure this, we create a labeling tool to fix light models with incorrect emission surfaces.

We split our dataset into training, validation, and test sets at a ratio of 70%-15%-15%. Some statistics of the dataset are presented in Figure 4. The distribution of room areas in the dataset is given in Figure 4(a). There are two peaks at approximately 30 m^2 and 13 m^2, corresponding to the standard sizes of a living room and the bedroom, respectively. Additionally, the wide range of room areas shows that our dataset includes a large diversity of rooms, such as study rooms, bathrooms, and balconies. The distribution of the number of cameras in each room is shown in Figure 4(b). Most scenes have only one camera, which is sufficient to cover most of the area, while others have multiple camera views. In Figure 4(c), we show the distribution of the number of rooms with specific categories of lights. We group these types to obtain categories according to their function. Chandeliers and ceiling lamps are named as key lights. Table lamps, floor lamps, wall lamps, and bedside pendant lights are named as auxiliary lights. The total numbers of rooms containing certain types of lights and the average numbers of lights per room are shown in Figure 4(d). The statistics show that downlights and chandeliers are the most frequently used lights in our dataset.

5 LIGHT LAYOUT ARRANGEMENT

In this section, we describe the algorithm to automatically select and place lights. Inspired by previous work in indoor scene furniture arrangement and floorplan design [Ritchie et al. 2019; Wang et al. 2018b; Wu et al. 2019], we first convert the 3D indoor scene into an image-based scene representation and then iteratively place lights in it, where the locations and types of lights are determined by neural networks specifically trained for arranging the light layout.

5.1 Image-based Indoor Scene Representation

Image-based indoor scene representations have been widely used in many interior design applications, such as furniture arrangement [Ritchie et al. 2019; Wang et al. 2018b] and floorplan synthesis [Hu et al. 2020; Wu et al. 2019], where a top-down view is utilized to represent a room, furniture, and floorplan. Unfortunately, such a representation is not sufficient for our lighting design task. For example, most key lights are placed on the ceiling, and a wall lamp is always placed on the wall. However, a 3D representation, such as voxels, is complex and data-intensive for networks. Therefore, we extend the top-down view representation to an image-based scene representation, which is still defined in 2D space but covers more of the structural information of the 3D interior scene.

An example of our image-based scene representation is shown in Figure 5. It is a set of images encoding several scene features:

Room structure images. We use a room mask image to indicate the occupancy of one room from a top-down view, where the pixels

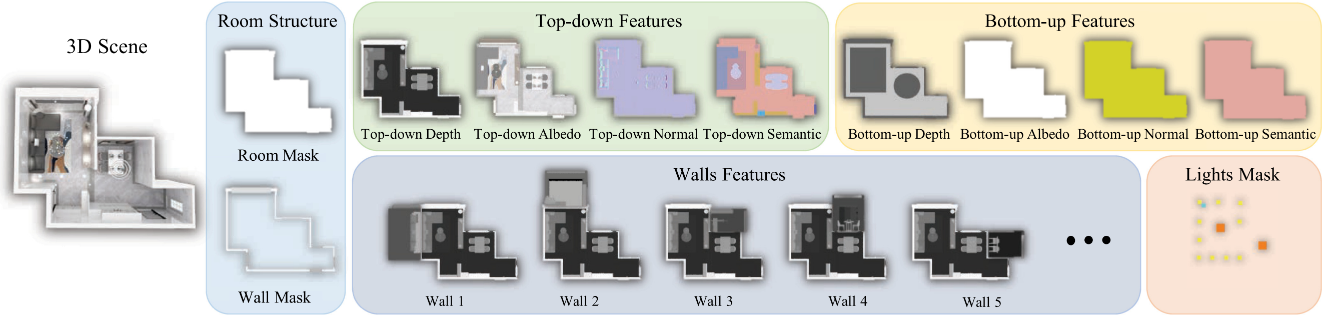


Fig. 5. Image-based scene representation. Our representation contains multiple features in the indoor scene, including two room structure images, eight top-down feature images, eight bottom-up feature images, eight light mask images, $C \times 9$ wall feature images, and C images of light masks for lights on the wall.

within the room are set to 1, otherwise 0. We employ a wall mask image to label walls, doors, and windows, where the pixels are set to 1.0, 0.5, and 0.75 for walls, doors, and windows, respectively.

Top-down feature images. Several top-down images are utilized to show the furniture layout and encode different pieces of furniture in the room, where depth, albedo, normal, and semantic labels of the room with furniture allow the network to better distinguish different objects.

Bottom-up feature images. Bottom-up feature images are similar to top-down images but contain information on the ceiling. This kind of feature image is important for obtaining an appropriate arrangement for chandeliers, ceiling lamps, and downlights.

Wall feature images. The wall is an essential feature in lighting design. Wall lamps should be placed on walls. Downlights also illuminate the wall and objects on it (e.g., artworks). Visual lighting effects on walls have been widely used for artistic and atmospheric purposes. To represent a wall in an image-based representation, we project each wall onto the floor along the outside room direction (similar to pushing the wall down) and record it with the entire room as a top-down image, where all objects and furniture on the wall are projected with the wall. We also encode the depth, albedo, normal, and semantic labels of walls into the wall feature images. Specifically, an extra mask image is used to indicate the region of the projected wall in the top-down image.

Light mask image. We use a light mask image to identify the locations and types of lights in the scenes. For each light, we use a square centered at the location of the light to act as the light mask. Such a mask simplifies the representation and training process. For key lights, we use a larger square (10 × 10), and we use a smaller square (5 × 5) for other lights. To separate lights with different categories, we store light masks in separate images, each of which encodes one light category. Different colors are used to represent different types of lights: red, orange, yellow, green, cyan, blue, and purple are used to represent ceiling lamps, chandeliers, downlights, table lamps, floor lamps, wall lamps, and bedside pendant lamps, respectively. This bears some similarity to the methods of handling different types of furniture in previous works [Ritchie et al. 2019; Wang et al. 2018b]. For each wall, we also project corresponding wall lamps to the wall feature images to obtain a light mask for wall lamps.

We use a path tracer with orthogonal cameras to obtain the aforementioned feature images of a 3D scene. The camera is placed in the center of the floor and set at 80% the height of the room. The camera renders top-down and bottom-up images by looking down and up, respectively. Although the camera position is set heuristically, it successfully splits the furniture and ceiling in all cases. To obtain wall feature images, we place the camera 1 *m* away from the wall, which captures almost all furniture or decoration adjacent to the wall. Each image has a resolution of 256 × 256 and represents a physical $15m \times 15m$ space, which covers most of the rooms in residential buildings. The image-based scene representation of a room consists of 26 images, including 2 room structure images, 8 top-down feature images (one channel for depth and semantic and three channels for albedo and normal), 8 bottom-up feature images, 8 light mask images (7 channels for lights of each type and 1 channel for all lights). For each wall, the image-based representation consists of 10 images, including 9 wall feature images (8 channels for top-down wall features and a single channel mask indicating the region of the projected wall) and an image of the light mask for the lights on the wall.

5.2 Light Arrangement

Inspired by learning-based scene synthesis work [Ritchie et al. 2019; Wang et al. 2018b; Wu et al. 2019], our light arrangement pipeline uses an iterative prediction scheme (Figure 6). Each iteration contains two basic steps: (1) the *Select Light Category* step determines the category of the light to be placed and selects one light; and (2) the *Predict Light Location* step places the selected light, where specific placement strategies for downlights and wall lamps are developed.

5.2.1 *Select Light Category.*

We utilize a neural network, named *NextCategoryNet*, to determine whether the existing light layout is satisfactory and the process can be stopped or whether more lights should be placed. The input of *NextCategoryNet* is the image-based scene representation along with a vector indicating the number of lights of each type in the existing light layout. The output is a distribution of placement possibilities for light categories, where we add "stop" as one category. Given the existing light layout, the network performs a classification task to make the decision.

The network architecture is a Resnet-18-based classification network, similar to that used in previous work [Ritchie et al. 2019]. As this is a standard classification task, we use cross-entropy loss. When training the network, we randomly remove lights from the scenes in our dataset to create a training set. We find that if we give a canonical order to place light fixtures rather than an entirely random order, then it will stabilize the training process. Therefore, we sort the lights in all scenes by a predefined type order (the order shown in Figure 4(d)). We find that such an order simulates the lighting design procedure of human designers in practice to some degree. Lighting designers usually place the key light first, increase the atmosphere using downlights, and then enhance local lighting using auxiliary lights. Note that this order is only used in the training process.

5.2.2 Place Light. When placing lights, we first develop a network, named *NextLocationNet*, to place the majority of types of lights, and then we design specific placement strategies for downlights and wall lamps (Figure 6).

This *NextLocationNet* is a Resnet-34-based network augmented with **atrous spatial pyramid pooling (ASPP)**, similar to that used in previous work [Wu et al. 2019]. The input of *NextLocationNet* is the image-based scene representation. The output is a $(2+6) \times 256 \times 256$ prediction map, which is a "heatmap" indicating the probability of light occurring per pixel, where 6 is the number of categories other than the downlight. Specifically, the prediction map is over (2+6) categories per pixel, which are the 6 light types (this pixel belongs to a certain light category), INSIDE (this pixel is inside the room but does not belong to any lights), and OUTSIDE (outside the room). The INSIDE and OUTSIDE categories are beneficial for this network, because they provide clues for placing lights only in the indoor region. Previous work [Ritchie et al. 2019] shows that even with one selected category, the generation of a prediction map of all categories rather than a prediction map of the selected category will help to obtain a stabilized distribution. Therefore, we also generate the prediction map for all light categories. Once we obtain the prediction map, we place the light at the location with maximum probability. Since this network performs a per-pixel classification task, we use averaged pixel-wise cross-entropy loss to train it.

Downlight Arrangement. The arrangement of downlights is more difficult than that of other lights. We observed in the dataset that the number of downlights is usually larger than that of other lights (as seen in Figure 4), and these downlights illuminate a relatively large region, which involves many conditions to be considered, such as the room structure, the shape of the ceiling, and the arrangement of furniture. In addition, designers usually design downlights in some symmetric patterns and utilize them to play an important role in determining the visual effect and lighting atmosphere upon basic lighting. We found that the iterative prediction scheme used for other lights is not good at capturing global arrangement patterns of downlights, such as alignment and symmetry. In Figure 16, we provide some comparisons to illustrate this. This problem was also mentioned in previous work [Wang et al. 2019, 2018b] where the iterative prediction module struggles to generate a symmetric arrangement of objects.

We propose a conditional generative adversarial network, named *DownlightGAN* (Figure 6 III), to predict the arrangement of all downlights at once rather than an iterative placement. In *DownlightGAN*, the discriminator learns to discriminate the generated result in a global perspective, such as by checking the overall alignment and symmetry, which is exactly what is preferable in the placement of downlights. After comparing many real and synthetic samples, the discriminator is able to discriminate the arrangement of downlights using important features such as symmetry. Thus, the symmetry feature is more likely to be learned by the generator network. Our network architecture is based on Pix2pixHD [Wang et al. 2018a] but is different in terms of the output, where the generator network of *DownlightGAN* outputs a per-pixel possibility. The network input is the image-based scene representation. The target is a labeled image that contains the square mask of downlights and the labels of pixels INSIDE and OUTSIDE. We use GAN loss and feature matching loss as in Wang et al. [2018a]. More details can be found in the supplementary document.

At runtime, the network generates a coarse layout of downlights in the prediction map (Figure 6(b)), and we vectorize it to obtain specific positions for individual downlights. Specifically, we fit the predicted noisy data to a set of squares using a scan line-based algorithm similar to that in Wu et al. [2019]. After vectorization, the centers of the squares represent the positions of downlights (Figure 6(c)). However, even with the generative adversarial network, some artifacts may exist; e.g., some downlights may not be exactly in a straight line. To handle these problematic cases, we align the downlights with heuristic lighting design guidelines. Specifically, we find the nearest wall of the downlights using a given threshold of 0.7 *m* as the default. Then, we align the downlights with the wall, where the downlights are set with the same distance to the wall and the same interval between two lights. For downlights that are not aligned with any wall (e.g., a sequence of downlights along the center of a corridor), we align them based on grid-based downlight arrangement design guidelines [Jin and Lee 2019]. Specifically, we align downlights vertically and horizontally. For vertical cases, we measure the horizontal axis values between all pairs of downlights and group downlights with similar values (given a threshold of 0.15 m) to form groups. We then align downlights within each group vertically. The same process is used in the horizontal direction. After the alignment, we obtain an enhanced version of the downlight arrangement (Figure 6(d)). Note that *DownlightGAN* predicts all downlights at once, which means there is no need to place downlights again for *NextCategoryNet*. We do not exclude the category selection phase directly, because the training data of *NextCategoryNet* imply that a downlight should no longer be chosen if it already exists, which can be learned by *NextCategoryNet*.

Wall Lamp Arrangement. A wall lamp is a good choice to create lighting in a room without wasting any valuable floor space. It plays both lighting and decoration roles. Wall lamps need to be placed on the wall, but it is challenging to find a single image that contains all wall features without overlap and preserves the relationships with the room structure. Therefore, we first

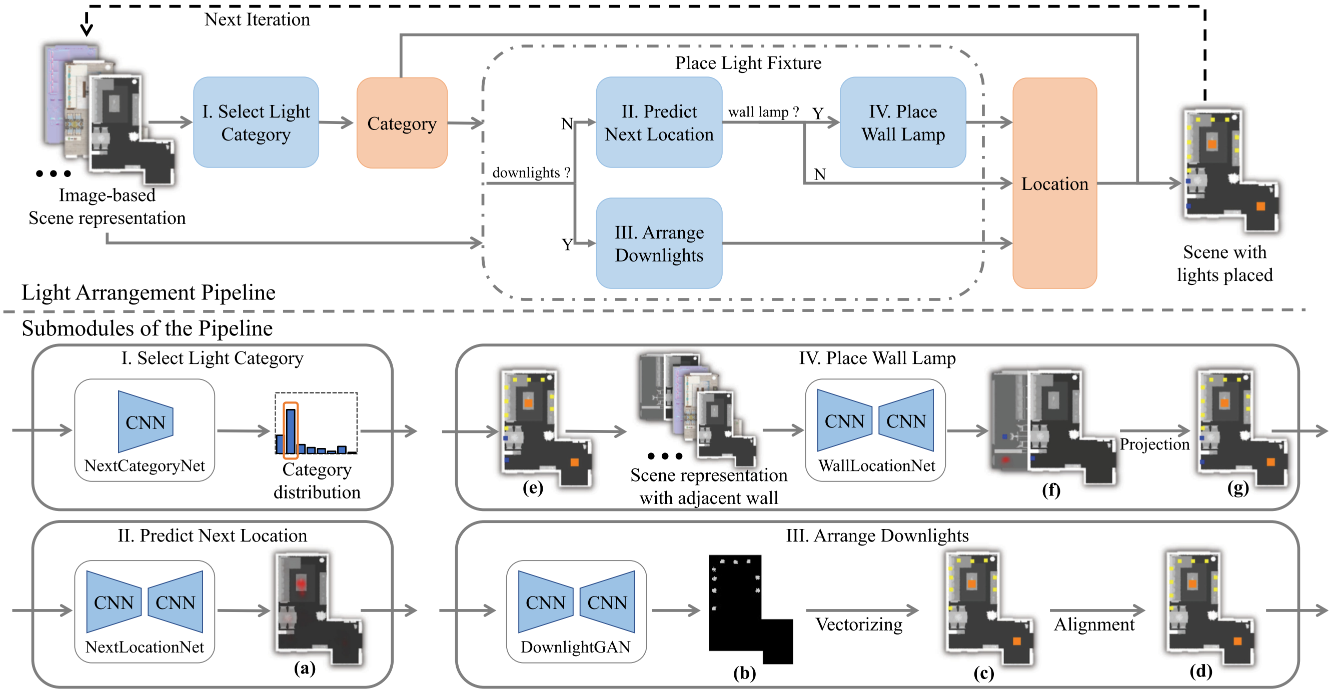


Fig. 6. Light layout arrangement pipeline. The pipeline inputs an image-based scene representation of the room and iteratively places lights in the scene in two steps: the *Select Light Category* step and the *Place Light* step. There are three submodules in the *Place Light* step: *Predict Next Location*, *Place Wall Lamp*, and *Arrange Downlights*. An orange square represents a chandelier, a yellow square represents a downlight, and a blue square represents a wall lamp.

use *NextLocationNet* to find the coarse position (Figure 6(e)), i.e., one wall, and then predict a precise location on the wall (Figure 6(f)). This procedure is shown in Figure 6 IV. To obtain the precise location, we design an end-to-end network, named *WallLocationNet*, to predict a distribution of locations on the wall similar to that of *NextLocationNet*.

The architecture of *WallLocationNet* is similar to that of *NextLocationNet*. The input is the image-based scene representation with wall feature images and wall lamp mask of the selected wall. The coarse wall lamp position in a top-down view and existing wall lamps on the selected wall are all included as light masks. Compared with *NextLocationNet*, we add more tags to label output pixels: OUTSIDE, INSIDE top-down view, INSIDE wall but not wall lamp, and WALL lamp. In some cases, the placement of wall lamps also exhibits asymmetric patterns. Therefore, we perform a similar alignment to that in the downlights arrangement by aligning wall lamps at the same height if there is more than one wall lamp on the same wall.

Neural networks in the light arrangement stage (Figure 6) are trained individually similar to previous works [Ritchie et al. 2019; Wang et al. 2018b; Wu et al. 2019].

5.3 Selection of the 3D Light Model

After arranging the locations of different lights, we place them in the scene. Specifically, we select the 3D model of each light from the light library, and then we place the light model to make sure the emission surface points in the correct direction, e.g., a chandelier points down, and a wall lamp points outward. Selecting a good 3D model of a light is a complex task that is worthy of further research [Chen et al. 2015; Liu et al. 2015]. In this article, we use a simple method to sample a model from the light library, similar to that proposed in Ritchie et al. [2019]. Technically, we first collect the light-object pairwise prior, i.e., the occurrence frequency of each light-object pair existing in the same room, and the light model occurrence frequency prior in the training set. Here, we use object to represent both furniture model and light model. Given a specific type of light to select, we calculate the probability of selecting each light model of this type from the pairwise prior and multiply them with the occurrence frequency of each light model to calculate a probability distribution for the models with the given category. Then, we randomly sample one light model from the computed probability distribution.

5.4 Exterior Lighting

In interior digital lighting design, exterior lighting also plays an important role in some cases [Birn 2014]. Therefore, we include a separate module after all interior lights have been arranged to handle exterior lighting. In particular, we model exterior lighting as an environment light (skylight) with a directional light (sunlight). We assume that the environment light is uniform with constant emission. For sunlight, we learn the direction from the dataset. In detail, similar to other lights, we first employ a network to determine whether to place sunlight, and then we use a network to predict the direction. The input of these networks is image-based scene representation without any light. Once the sunlight direction is determined, its emission is predicted along with all other lights in the light emission optimization stage. Please refer to the supplementary document for more details.

6 LIGHT EMISSION OPTIMIZATION

In this section, we describe the optimization used to compute the emission of each light. The proposed light emission optimization pipeline consists of two steps, as shown in Figure 7. First, we use a network to predict a lighting guidance image, which provides neural network-generated lighting effects. Since this is a synthetic result lacking the physical guarantee, we use it as the target image to solve the light emission optimization for the intensity and color of each light.

6.1 Generation of Lighting Guidance Image

Inspired by the human design process [Birn 2014; Shimizu et al. 2019], we first utilize neural networks to imagine a pleasing lighting image under a view as the guidance. Lighting guidance images can have diverse styles and be enhanced using real photographs.

6.1.1 Lighting Guidance Prediction Network. Even with all the lights that are placed, we find that it is still challenging to directly generate a good quality lighting guidance image. Therefore, we employ a progressive scheme that first generates a coarse image and then refines it to the final lighting guidance image. The entire process is shown in Figure 8. An evaluation of this progressive scheme is given in the supplementary document.

We use a network named *IntensityNet* to generate the coarse image. *IntensityNet* takes seven illumination per light category images and two illumination images for the environment light and sunlight as input (Figure 3(e)), where each light illuminates the scene using a unit emitted radiance (i.e., cd/m^2), and all illuminations of lights in the same category are summed to be one illumination per category image. Note that each image here is stored with high dynamic range without tone-mapping to preserve its emission properties. By obtaining these per-category images, *IntensityNet* predicts the scale and color of each category and then uses them to combine different category images and generate the coarse lighting guidance image. Here, we use the color temperature to represent the color instead of RGB, since we found that it could improve the robustness of the training. The network uses Resnet-34 to downsample the input lighting images to a latent 512 vector. Afterward, there are two network branches for predicting the intensity and color temperature of each light category separately, which consist of a sequence of fully connected layers. Please refer to the supplementary document for more details.

In generating the coarse image, the light intensities and colors are assumed to be the same for lights in the same category. Since this constraint is strong, we propose a second network named as *ShadingRefineNet* to refine the lighting to a more accurate prediction. We use conditioned feature modulation [Xu et al. 2019] to better integrate the features into the network. To better predict lighting in images rather than modifying the albedo of scenes [Bi et al. 2019], we split the shading and albedo before giving it to the network and multiply the albedo back after the output is obtained to determine the final predicted lighting guidance image.

Several types of loss are proven to be helpful in this image generation task. First, we use L1 loss on both coarse and refined lighting guidance to ensure that they are as close as possible to the target image. Then, we add VGG loss [Johnson et al. 2016] to better align with the feature and style in the target image. Additionally, we use GAN loss to enhance our image quality. Inspired by Bi et al. [2019], we use two discriminators to guide the predicted shading and refined lighting guidance.

The training of these two networks is performed progressively. We first train *IntensityNet* using loss on the coarse lighting guidance. Afterward, we add the *ShadingRefineNet* and loss on the refined lighting guidance to train them together.

6.1.2 Lighting Diversity. The lighting guidance prediction networks can be extended to a multimodal version and generate lighting styles with greater diversity. Specifically, we apply the loss of BicycleGAN [Zhu et al. 2017b] to our networks to generate diverse lighting guidance. A latent code is added as input of both *IntensityNet* and *ShadingRefineNet* to control the lighting style. With the ability of BicycleGAN loss, the mapping of the latent code and lighting style is ensured. Please refer to the supplementary document for the network architecture and loss function. Similar to the original training procedure, we train the multimodal version progressively. With this network, our system can generate diverse lighting styles, as shown in Figure 13.

6.1.3 Lighting Enhancement. As the lighting guidance image is used as a visual objective, we can further enhance it by considering more data, such as real interior photographs. We proposed an optional lighting enhancement stage (dotted box in Figure 7) to achieve this. We collect 3k photos from the Internet and utilize them to improve the guidance image. Unlike for the learning on synthetic data generated by 3D scenes, these photos are unpaired data and cannot be directly used to train the networks used in generating lighting guidance. Instead, we utilize CycleGAN [Zhu et al. 2017a], a classic image-to-image translation method for unpaired data. CycleGAN incorporates more realistic and visually pleasing lighting styles in the lighting guidance image and finally improves our lighting design results. Please refer to the Results section for more comparisons. Note that, though we use real photos to enhance lighting, any image collection with user-preferred style can also be utilized here to bake the style into the generated lighting.

6.2 Optimization of Intensity and Color

After obtaining a lighting guidance image, the optimization of the intensity and color of each light becomes relatively simple. The final image is a combination of the intensities of all lights with different RGB scale coefficients. As such, it can be treated as an optimization problem that takes the lighting guidance as the target and determines the scale coefficients of each light. Similar to previous work [Lin et al. 2013; Schoeneman et al. 1993], we solve this problem with a non-negative least squares solver [Nelder and Mead 1965]. We use gradient descent in the lighting enhancement stage, because the LDR target image and tone mapping break the problem’s linearity. Once all intensities and colors are obtained, we can use them to render the scene and obtain physically correct lighting images.

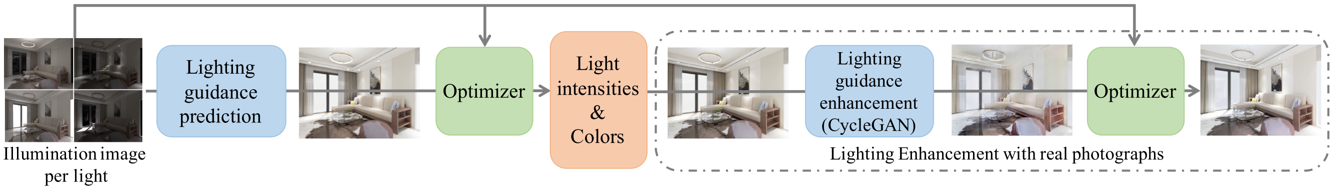

Illumination image per light
Lighting guidance prediction
Optimizer
Light intensities & Colors
Lighting guidance enhancement (CycleGAN)
Optimizer
Lighting Enhancement with real photographs

Fig. 7. Light emission optimization pipeline. Our light emission optimization pipeline consists of two steps, guidance image prediction and optimization, as shown in the blue and green blocks, respectively. Our pipeline can also utilize real-world photographs to enhance our lighting effects with CycleGAN.

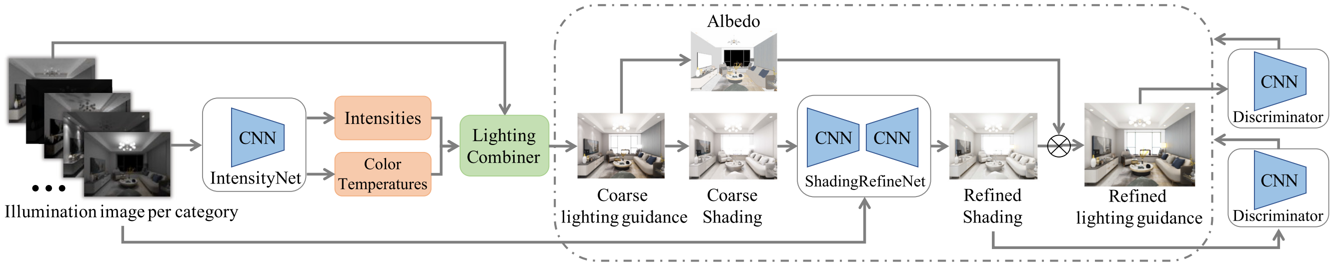


Fig. 8. Lighting guidance prediction networks. Our networks first produce coarse lighting guidance with *IntensityNet* and then use *ShadingRefineNet* to generate the final lighting guidance image. Adversarial discriminators are used to refine the lighting guidance.

6.3 Optimization for Whole-room Lighting

In our framework, we simulate the lighting procedure of human designers by performing lighting optimization at certain camera views. However, in some cases that the view covers a small space of the room, the lighting design according to only one view may bring unnatural lighting for other views. Additionally, some digital lighting applications may require a walk-through video rather than single rendering image. To address these cases, the aforementioned optimization at one camera view in our system could be extended to multiple camera views covering the whole room to ensure a good whole room lighting.

To enable whole-room lighting guidance prediction, we first partition the 2D room structure into a set of regions using an algorithm that can generate a minimized number of convex polygonal partitions [Greene 1983]. Then, we place six cameras to construct a panorama in the centroid of each partitioned polygon. Lighting guidance images are predicted in these views, which can cover the whole room. In the light emission optimization stage, we combine all views and optimize them simultaneously to obtain the whole-room lighting. However, according to the camera's position, the object near the camera may occupy the most area of the lighting guidance image in its view, which makes the optimization process view-dependent. Ideally, a pixel in the guidance image that covers more area of the room should be more important in the optimization. To address this problem, we weighted the loss pixel-wisely by the relative area of pixel footprints; thus, the light emission optimization becomes a weighted non-negative least squares problem. With automatic cameras generation with full room coverage and the view-independent optimization process, our system can automatically generate pleasing lighting for the whole room. Please refer to the supplementary document for more details. In rare cases where the centroids of partitions may collide with an object, we temporarily remove the object for proper rendering.

Note that our multimodal lighting guidance prediction network is also suitable for multi-view optimization. This network can generate consistent lighting styles with the same latent code as input, avoiding mixing multiple diverse styles in different views. Please refer to the supplementary video for the results of whole-room lighting with different styles.

7 RESULTS AND EVALUATION

We use PyTorch [Paszke et al. 2019] to implement all the proposed networks in this article, and we train and test our framework on a PC with an Intel Xeon E5-2630 v3 CPU and an NVIDIA GeForce RTX 2080Ti GPU. Please refer to the supplementary document for more details of the networks and the training process.

In the supplementary video, we show an automatic process using the proposed system to obtain the digital interior lighting of one 3D indoor scene. Once the scene is obtained, the system takes 5 seconds on average to construct the image-based representation, executes for around 3 to 10 seconds to arrange the lights according to different numbers of lights to be placed, and computes for about 2 seconds to optimize the emission of lights. The most time-consuming step of the entire lighting design process is path tracing-based rendering, which may take seconds to minutes due to different numbers of samples per pixel and the 3D scene model complexity.

7.1 Qualitative Evaluation

Comparison with designers. In Figure 9, we compare our results with those of professional lighting designers in the dataset. We show two results generated by our method. Figure 9(a) shows the initial scenes without light installed rendered using ambient occlusion. Figure 9(b) shows the results learned directly from the dataset, where the light emission is guided only by the prediction network (named "ours"). Figure 9(c) shows the results enhanced by

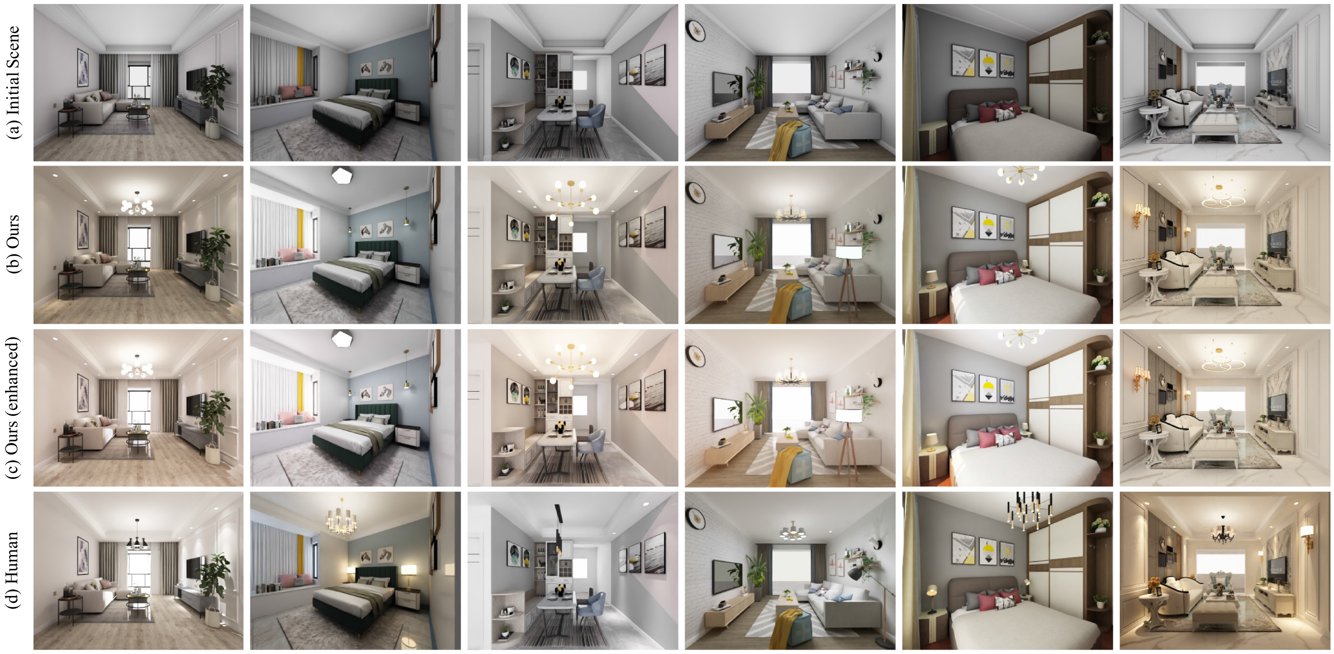


Fig. 9. Visual comparison between our results and lighting designs created by human designers. The first row shows the initial 3D scenes rendered with ambient occlusion. The second row shows images with lighting designs created with only synthetic data. The third row shows our results enhanced using interior photographs. The results created by human designers are shown in the last row.

real interior photographs (named “ours (enhanced)”). Comparing ours and ours (enhanced), in most cases the enhanced result has a slightly brighter and warmer lighting style. We also observed such a difference between the synthetic lighting images and captured real photographs. By comparing our results with those of human designers, we can see that our lighting layout arrangement is reasonable and naturally integrated into the room and furniture. In terms of lighting effects, both of our results show comparable lighting to that of the images from humans. In some cases, our enhanced results show even better aesthetic and pleasant lighting effects than those of professional designers.

Comparison with the rule-based method. To evaluate the effectiveness of our learning-based lighting design framework, we compare our system with a state-of-the-art rule-based lighting layout optimization method [Jin and Lee 2019]. This method optimizes the light placements and intensities simultaneously using simulated annealing optimization. Multiple rules derived from interior lighting guidelines are used to consider pairwise relations, hierarchy, circulation, illuminance, and collision. In this work, the target positions of key lights (ceiling lamps or chandeliers) are aligned with the centers of the furniture groups, which are constructed based on the connection strength [Xu et al. 2014] of the furniture. Downlights are placed on a uniform grid considering the room size and existing key lights. Other auxiliary lamps are placed in accordance with specified pairwise relations (e.g., the target position of the floor lamp is to the side of the sofa). Light emission is optimized to reach the target illuminance on the task plane of the furniture. To adapt this method to our scenes, we calculate the average illuminance for each type of furniture from our dataset to use as the target illuminance. As an optimization-based method, this method requires high-quality prespecified light objects for a scene as input. To satisfy this prerequisite for its best performance, we directly provide the set of light objects in each human-designed scene. Furthermore, because the original method lacks consideration of the room structure (e.g., no key light in a region with few pieces of furniture) and optimizes only white light and two-bounce lighting, we introduce additional considerations to enhance its results. Please refer to the supplementary document for more details about our adaptations and improvements to the original method [Jin and Lee 2019].

Figure 10 shows the visual comparisons of the results of the rule-based baseline method and our method. The first row shows the light layout visualizations in orthogonal views (top-down and bottom-up views). The following two rows show a set of rendered images from different views in a whole room. As seen from the bottom-up views, which show the light arrangement on the ceiling, our method can generate light arrangements that better fit the room structure and furniture arrangement compared to the rule-based baseline. Specifically, the output of the rule-based baseline lacks the downlights in the corridor (the first and fifth columns), and there may be multiple key lights (the first and third columns) or no key light (the fifth column) in some regions split by the ceiling pattern. The placement of the pendant lamps may also be not in the center part of the ceiling pattern (the first column). The main reason for these shortcomings of the rule-based baseline is that the complex ceiling pattern and other information, such as the room structure and furniture arrangement, are difficult to fully encode in explicit rules. As a result, incomplete rules will lead to problems in generalization. For example, the rules for downlight

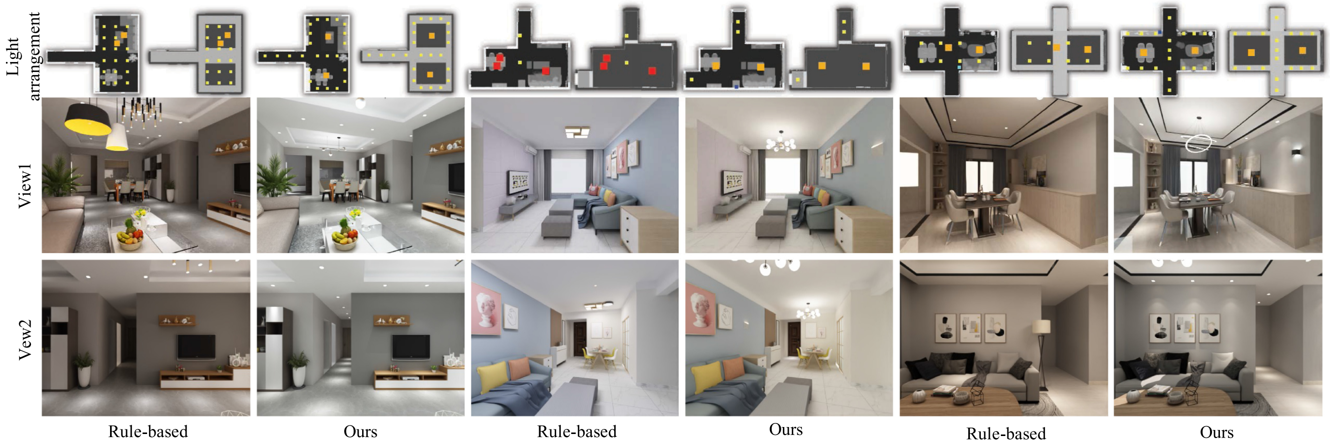


Fig. 10. Visual comparisons of the rule-based method and our method. The top row shows the visualizations of the light arrangements from top-down and bottom-up views. It is seen that our method can generate light layouts that better fit the room structure and furniture arrangement. Sufficient lighting for the whole room is achieved in our method, while the rule-based baseline often fails to place downlights in the corridor. Inappropriate arrangements of key lights (e.g., not suitable for the ceiling pattern, no or multiple key lights in one space) also appeared in the results of the rule-based baseline.

arrangement have difficulty considering all possible complex room structures. For the arrangement of key lights, we find that it is difficult to find a threshold for constructing the furniture groups that is suitable for all scenes, and thus, in some scenes, the rule-based approach may fail to produce the correct number of focal points. Furthermore, even if the target focal points are appropriately generated, key lights may be either excessively aggregated or lacking in some regions. A similar phenomenon has also been mentioned in the comparison of the arrangement baseline with a deep learning scene synthesis method [Wang et al. 2018b]; for example, two chairs may be aggregated to the same desk even though there are two desks in the room. This is a known failure mode caused by conflict between multiple pairings. More rules need to be introduced to help mitigate this phenomenon. To address these problems, we added a cost term to consider the missing factor of the room structure and another cost term to resolve pairing conflicts. However, we still observed similar failure cases in some results. Thus, we found that this behavior is also caused by instability of the optimization process. The optimized results may become trapped in local minima and thus may be sensitive to the initial randomized layouts. In contrast, our method encodes all corresponding information implicitly and does not suffer from the instability of the optimization process. Thus, we can produce better light arrangements.

As seen from the rendered images, our method can illuminate the whole room adequately, whereas in some results of the baseline method (the first and fifth columns), the corridor is dark due to missing downlights. Moreover, some lights do not achieve adequate emission (e.g., the floor lamp in the fifth column). Our method generates more accurate lighting guidance on more surfaces and thus can mitigate this phenomenon.

Perceptual Studies. Since it is very subjective to judge the lighting effect of renderings, we conducted two-alternative forced-choice perceptual studies to evaluate the quality of our lighting design. Unlike the perceptual studies for furniture arrangement and floorplans in previous works [Wang et al. 2018b; Wu et al. 2019], the perceptual evaluation of the whole-room lighting design is more complicated, because it is difficult to represent the design using a single image, and the evaluation needs to be done from multiple perspectives. To evaluate the lighting design in the whole room, we selected two or three views for each scene and ensured that the views could cover almost all the objects in the room. We used three or four images to represent a scene with its lighting design. One was the top-down image with annotated light masks and cameras, and the others were rendered images under different views. Then, we recruited participants to make a side-by-side comparison between two scenes. For each pair of comparisons, we designed three questions motivated by previous work [Jin and Lee 2019], which evaluate the lighting design as a whole.

- Q1: Which scene has a more appropriate light arrangement (position and number)?
- Q2: Which scene has a more visually comfortable lighting effect?
- Q3: Which scene has a more appropriate light placement and brightness to interact with the furniture?

Each participant performed 24 comparison tasks that are sorted randomly. In one out of 12 comparisons, we perform a “vigilance test,” where the participant is given an obviously unpleasant lighting design result (the lights were placed randomly with random intensity and color) to check whether the participant was paying attention. The results from participants who failed the “vigilance test” are filtered out.

We conducted two perceptual studies to compare the lighting designs of ours and ours (enhanced) with the designs from human designers separately. Since the light arrangement remains unchanged between ours and ours (enhanced), we only asked Q1 once in these two studies. We also conducted a perceptual study to compare our lighting designs with the designs from rule-based baseline. We control the light objects to be the same in this comparison by using the light objects from our lighting designs for the rule-based

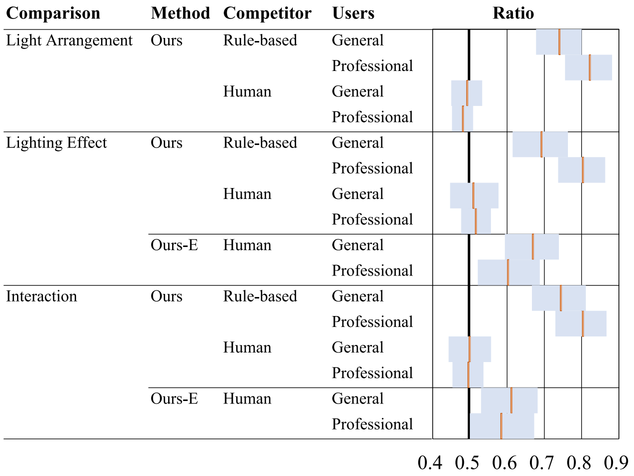


Fig. 11. Statistics of perceptual studies. Ours-E represents ours (enhanced). The orange lines show the ratio of users preferring the designs generated by our method to those of human designers. A 0.5 ratio shows a comparable preference, and a higher ratio indicates a greater preference for our method. The blue bar is the 95% confidence interval computed by Bootstrap [Efron and Tibshirani 1986].

baseline. The number of participants enrolled in these three perceptual studies was 32, 34, and 37, respectively. We obtained a total of 30 participants who passed the vigilance test in each perceptual study. Half of the participants in each perceptual study were professional interior designers recruited from interior design companies, and the remaining half were general users who were students in computer science. The age of the participants ranged from 19 to 42, and the professional designers enrolled had an average of 3.2 working years. Each participant who passed the vigilance test took on average 24.04 s to compare two scenes. Figure 11 shows the statistics of our perceptual studies for different evaluation aspects using ratio bars similar to the studies of Wang et al. [2018b].

As shown in the figure, neither general users nor professional designers show a preference between ours and human-created designs in the three aspects, which indicates that our system generates lighting designs comparable to those of human designers. In contrast, our method outperforms the rule-based baseline in all three aspects. Moreover, professional designers give higher scores than the general users in this comparison, which shows that our system learned professional lighting design principles better. In the study of ours (enhanced), the ratio indicates that users have a greater preference for our enhanced results than those of human-created designs in terms of lighting effect. This shows that the lighting learned from high-quality interior photographs generally brings more pleasing lighting effects than that from rendered synthetic images. This is mainly because real photographs include more realistic lighting effects and more visually pleasing aesthetic lighting styles, which is preferable. Although the pleasing aesthetic lighting style may also be caused by elaborate post-processing by the photographer, our system also successfully optimizes the lighting parameters to achieve enhanced lighting effects without the need to post-process the image manually. Note that the comparison results of enhanced lighting and human-created design only shows our system has a potential to generate such user-preferred lighting; the enhancement results under other collection of user-specified images are not guaranteed.

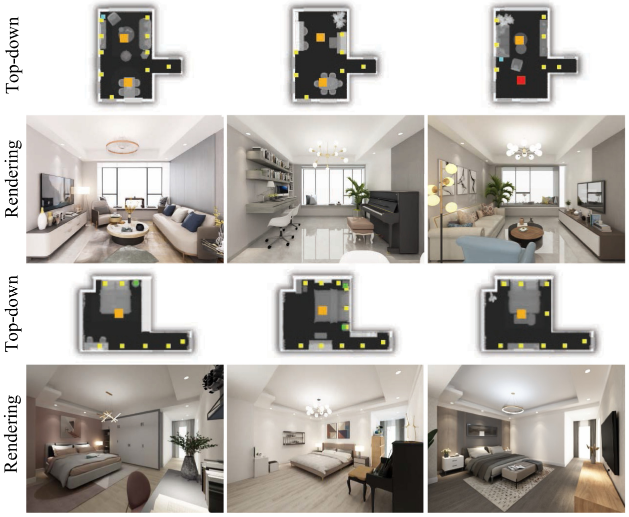


Fig. 12. Comparison of lighting design results in the same room structure with different furniture layouts. The red, orange, yellow, green, and cyan represent the ceiling lamps, chandeliers, downlights, table lamps, and floor lamps, respectively.

Evaluation of the structural relationship learned by the networks. Our approach learns lighting layouts from furniture and scenes. In Figure 12, we show the results under different furniture arrangements in the same room. The lighting design changes with the change in furniture layout, which implies that our network has learned the relationship between lights and furniture. It is also interesting that the location of the chandelier is almost fixed in the example scenes. This shows that the key light is learned mostly in relation to the room rather than the furniture, since the key light is usually designed to illuminate a large space. In contrast, local lights show a tighter relationship with furniture. As can be seen, the floor lamp is placed on the side of the cabinet or sofa in the first and third examples of the living room scene, and the desk lamps are placed on the nightstands in the bedroom scene. Some downlights also vary according to the furniture arrangement, such as the downlights above the piano in the second example of the living room and the downlights above the bedside in the bedroom.

Results of diverse lighting designs for the same scene. As shown in the first row of Figure 13, our light arrangement can generate different arrangements for the same scene by sampling from the probability distribution produced by *NextCategoryNet* and *NextLocationNet*, which is similar to previous works [Ritchie et al. 2019; Wang et al. 2018b; Wu et al. 2019]. In addition, our approach supports generating different lighting styles using the multimodal generative networks mentioned in Section 6.1.2. The second and third rows of Figure 13 show different lighting styles generated by our system.

Result visualization using t-SNE. We visualize the distribution of light arrangements in the test set using t-SNE [Van der Maaten and Hinton 2008], as shown in Figure 14. Specifically, we represent

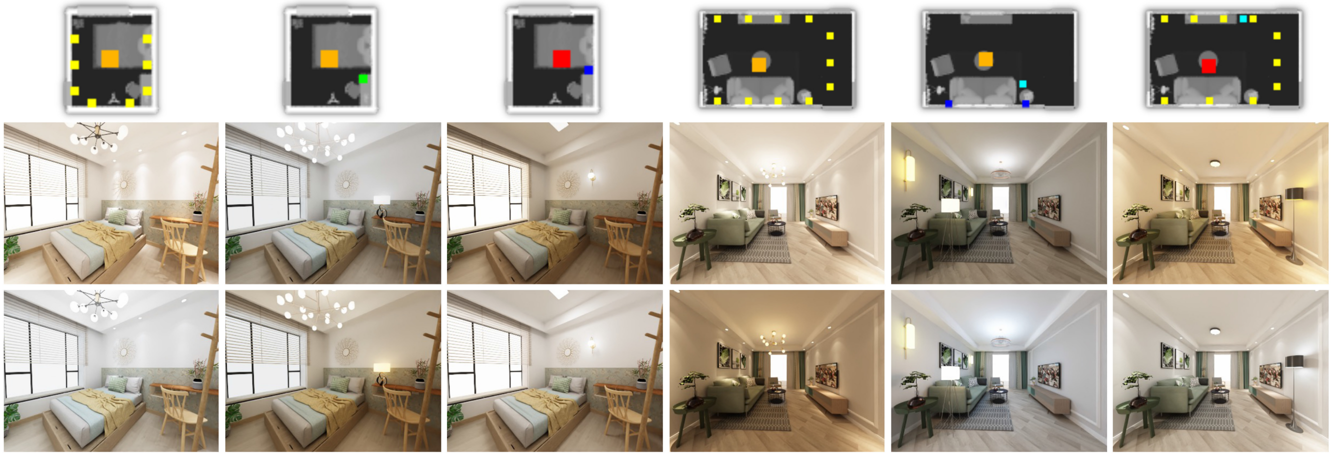


Fig. 13. Our system can generate different lighting designs for the same scene. The first row shows different light arrangements, and the second and third rows show different lighting effects produced under the same light arrangements.

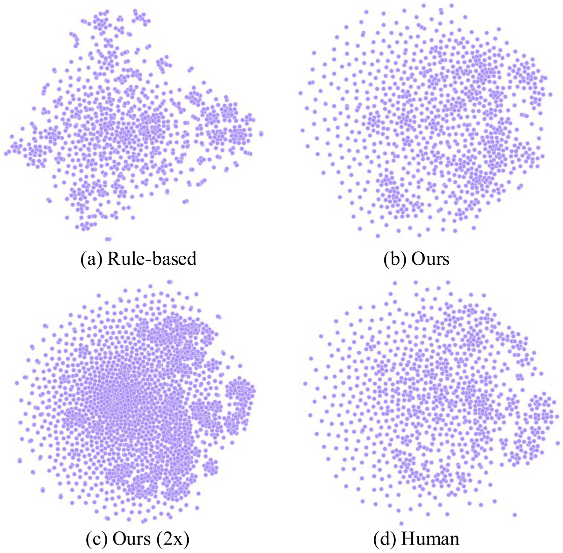


Fig. 14. Visualization of the distribution of light arrangements in the test set using t-SNE. Our method generates light arrangements (b) similar to the reference (d) and can generate more diverse light arrangements (c), which makes the distribution denser.

light arrangements using light mask images and then perform t-SNE dimension reduction with PCA initialization to visualize them. Our method is closer to the human results than the rule-based baseline, even though the rule-based baseline has the same number of lights as the human-designed scenes. Results also show that our method can generate a denser t-SNE visualization than the human one by generating more light arrangements for the same scene. Multiple patterns can also be found in the results, which shows the diversity of the room's structural space and the corresponding light arrangement.

Results of whole-room walk-through. Figure 15 shows a walk-through of a room with our predicted lighting. Pleasing and consistent lighting is obtained for the whole room. Room partitions and positions of panoramas for lighting prediction are visualized in Figure 15(a). Please refer to the supplementary video for more examples of whole-room walk-through.

7.2 Quantitative Evaluation

Besides qualitative evaluations, we conduct several quantitative evaluations.

Neural image assessment. To evaluate the quality of our lighting design, we use a quantitative metric known as neural image assessment [Talebi and Milanfar 2018]. We use a neural network trained on the **Aesthetic Visual Analysis (AVA)** dataset [Murray et al. 2012], which has been used as a reliable quantitative metric for aesthetic evaluation. Table 1 shows the statistics of the results generated by our method and the lighting designs in the dataset. This shows that our score is close to those of the professional designs both in the preference percentage and average score. Our enhanced results have higher scores in both metrics. The preference percentages are similar to the results in the perceptual study shown in Figure 11, which also shows the validity of both evaluations.

Evaluation of light arrangements. To evaluate the quality of the light arrangement, we compare our approach with two baseline methods. The first is an iterative prediction method similar to that of the previous work [Ritchie et al. 2019]. It uses iterative placement for downlights instead of our *DownlightGAN*. When adopting this method to our situation, the network is the same as ours except for the extra arrangement step for downlights. Another baseline method is called *LampGAN*. It uses a generative adversarial network to generate all lamps, where each lamp is extracted from the prediction map by the vectorization method in our *DownlightGAN*. The network architecture of *LampGAN* is the same as that of our *DownlightGAN*, which uses Pix2pixHD [Wang et al. 2018a].

Our approach can be regarded as a hybrid model of these two baselines. We use the iterative framework for most of the lights but use *LampGAN* to arrange downlights. The comparison results

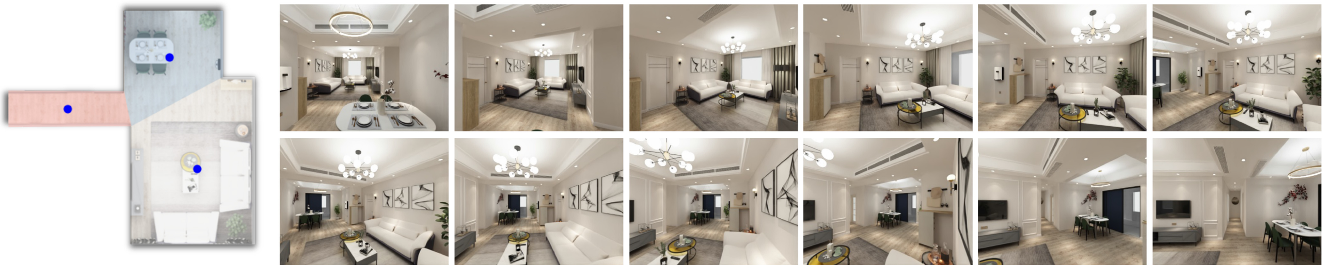

(a) Room partitions (b) Walk-through for the whole room

Fig. 15. The digital lighting design generated by our system can be experienced in a free-view walk-through (b); please refer to our supplementary video for the complete walk-through video. In this example, the room partitions and positions of panoramas are shown in (a).

Table 1. Aesthetic Evaluation Using Neural Image Assessment [Talebi and Milanfar 2018]

Metrics	Ours	Ours (enhanced)	GT
Preference percentage	49.23%	62.69%	-
Average score	5.329	5.384	5.337
Standard deviation	0.217	0.215	0.221

The metrics of our results are close to the human-designed results, and our enhanced results gain better scores.

Table 2. Comparison between Different Light Arrangement Methods

Metrics	Iterative model	LampGAN	Ours
classification	78.25	76.00	**62.75**
KL-divergence	0.0570	0.1062	**0.0072**

Our method achieves the best results in both classification accuracy and KL-divergence with the reference scenes in the dataset.

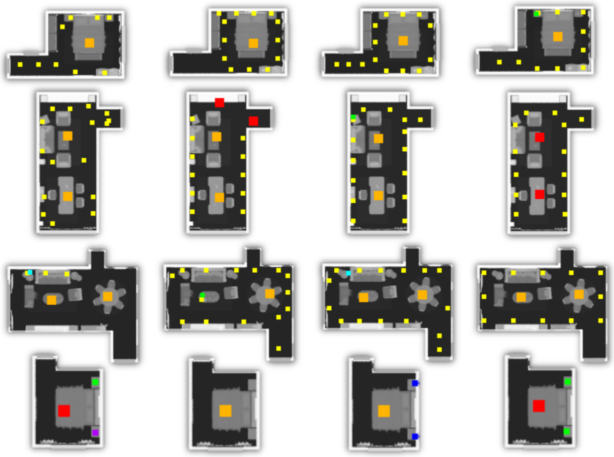

(a) Iterative (b) LampGAN (c) Ours (d) Human

Fig. 16. Comparison of light arrangements using (a) a pure iterative prediction model, (b) a generative adversarial network, (c) our lighting design pipeline, and (d) humans. Different colors indicate different categories of lights (Section 5.1). The results show that LampGAN usually has a reasonable downlight arrangement (Rows 1, 2, 3) but always ignores some categories of lights. The iterative model does not have such a symmetric arrangement, and downlights may be aggregated in some cases (Rows 1, 2). However, different categories of lights can be sampled reasonably (Rows 3, 4). In contrast, our method has all these advantages and produces light arrangements comparable to those of human designers.

of these three models are shown in Figure 16. Each row shows the results of different methods in the same room. The iterative model has difficulty obtaining a good result for the arrangement of downlights. It may aggregate multiple downlights. The quality of *LampGAN*'s result is much better, but it suffers from unbalanced light categories. For example, auxiliary lights rarely appear in the results. We guess this is due to the mode collapse of the GAN, where the overall distribution of the data is not grasped well by the network and some features in the data are lost. Our light arrangement combines the advantages of these two: the diversity of the layouts generated by the iteration-based model and the reasonable layout of downlights from *LampGAN*.

To further validate the results of our light arrangement, we use a classification network to quantitatively evaluate the results of our method and the baselines and take the lighting designs in the dataset as reference data. The closer our results are to the reference, the more difficult it is for the classification network to classify them, resulting in a lower classification accuracy. Similar to Ritchie et al. [2019], we use a Resnet-34-based classifier. The training dataset contains 1,500 results, half of which are the results we want to compare, while the other half are the results in the dataset from human designers. Four hundred images are used for testing. The classification accuracy of the different methods is shown in Table 2. The iterative model has a relatively high accuracy, indicating that it has a larger difference in the distribution compared to the reference. *LampGAN* has a lower accuracy but is still higher than ours. Our results are the most indistinguishable, showing the best consistency with the lighting designs in the dataset.

In addition, we evaluate the distribution of light categories generated in the results. Specifically, we calculate the KL divergence between results generated by different methods and scenes in the dataset, as shown in Table 2. Similar to what we discussed earlier, the category distribution of LampGAN is relatively poor. We found that for lights that do not appear frequently in the dataset, such as floor lamps and bedside pendant lamps, *LampGAN* generates hardly any of them. The category distribution of the iterative model is better than that of *LampGAN* but still worse than ours. We think this is because it is difficult for the *NextCategoryNet* in the iterative model to predict each downlight.

7.3 Discussion and Limitations

As the first deep learning-based method attempting to design interior lighting automatically, our method also has several limitations. Typical examples are shown in Figure 17. First, in some cases (Figure 17(a)), our approach does not take the orientation of light into account. This is because most lights are symmetrical to illuminate the environment evenly in all directions. This is not a problem for most chandeliers, ceiling lamps, downlights, and bedside pendant lamps. For wall lamps, we ensure that the up direction is from the floor to the ceiling. However, symmetrical placement may not be ideal for floor lamps or table lamps with an asymmetric model. In future work, our framework can be extended to consider the orientation using an orientation prediction module such as that in Ritchie et al.'s work [Ritchie et al. 2019]. Second, as a learning-based approach, our method may fail in some cases that are rare in the dataset. Figure 17(b) shows an example where the ceiling has a complex style pattern that has not been learned before (the downlights above the television overlap with the ceiling pattern, and the downlights in the corridor deviate from the center slightly). More training data may help in this case. Third, as we discussed in Section 5.3, we use a simple scheme to select light models, which may cause a style compatibility problem in some cases (such as Figure 17(c), where the two table lamps have different models). In most other cases, such a scheme selects the same table lamps because only one model has the highest possibility. However, more models with similar possibilities may result in incompatible models in some cases. We believe that, in future work, a better model selection scheme could be exploited.

8 CONCLUSION AND FUTURE WORK

In this article, we propose the first deep learning-based digital lighting design framework. It automatically generates visually pleasing lighting designs utilizing the guidelines and principles learned from an interior lighting dataset with 6k 3D scenes, 8k views, and 3k real photographs. Under the evaluation of a series of quantitative and qualitative experiments, our framework successfully adapts to various room structures and furniture arrangements and generates lighting designs that are comparable to those of professional human designers.

Future work. In addition to some future work mentioned in the limitations section, there are several directions available for further exploration. First, our framework is capable of generating lighting designs fully automatically. However, in some cases, the designer needs an interactive human-in-the-loop design. For example, more constraints may be added, such as the number of lights of each type and the relationship between the furniture and room structure. Our framework could also integrate more constraints by using a graph representation [Hu et al. 2020; Wang et al. 2019]. Designers could also control the lighting style using customized visual objectives [Shimizu et al. 2019]. Although our system aims for automatic lighting design, it can also be naturally integrated into the human design loop to facilitate the lighting design process. Specifically, human designers can adjust the light configurations manually on top of our result and then run the light emission optimization stage again to generate lighting design that more fits to their design goal. For predicting the whole-room lighting, our system places the cameras to cover the whole room by considering the occlusion from the room layout. Though most parts in the residential rooms can be covered in this way, some parts occluded by the furniture may be neglected. Some recent works on camera placement problems like Sun et al. [2021] can be utilized to obtain max-coverage placement considering complex occlusion in the scene. We would regard these as future works.

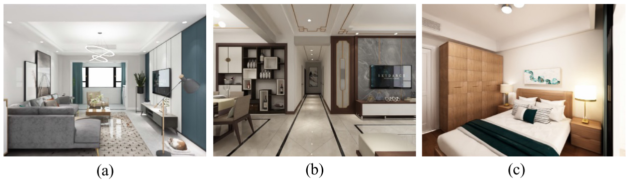


Fig. 17. Some failure cases of our method. (a) shows that some light fixtures have an inappropriate orientation (e.g., the floor lamp in the figure). (b) shows that our downlight arrangement struggles to handle complex ceiling patterns. (c) shows an incompatible model selection case.

Additionally, the framework is designed to generate lighting design in a room instead of a whole floorplan. Users need to design the lighting in rooms one- by-one to obtain a whole lighting design for an entire floorplan. A new lighting design approach for a whole floorplan is another future research direction. A larger space may bring more difficulty for neural networks. Our framework generates results across multiple types of residential rooms. Except for living room, dining rooms, and bedrooms shown above, more results for kitchen, kids' room and study room can be found in the supplementary document. In addition to residential rooms, our framework could be extended to other types of indoor scenes, such as restaurants and office spaces. However, the corresponding datasets should be available first. We would like to extend our dataset to include more interior scenes in the future. Although digital lighting designs with diverse styles can be obtained in our system, our lighting result is always a pleasing lighting effect learned from the dataset. In some cases, rare and special lighting effects of expressing a particular mood are also needed, especially in the movie and games industry. How to facilitate this design task deserves to be explored. Recent advances in differentiable rendering techniques [Jakob et al. 2022; Li et al. 2018; Zhang et al. 2020a] make it possible to optimize lighting parameters with physically based path tracing in gradient-based optimization. This is also deserved to be explored.

As our system aims for interior digital lighting design, the most important but challenging task for the future is to push the current approach to real-world lighting design by considering more factors in physical realization [Gordon 2015]. Several aspects need to be considered for achieving this goal: First, the light-fixture library in our dataset does not have an available set of discrete real-world power and color temperature parameters for each light. Additionally, not all light fixtures have an IES profile. A more realistic and better physically annotated light-fixture library should be built. Second, as lighting design is a complex task affecting different human activities, ergonomic aspects should be considered in real-world lighting designs (such as glare). Additionally, as real-world lighting corresponds to human activity and the time of day, lighting design supporting lighting modes for different activities deserves to be explored. Some more complex constraints, such as energy savings and cost savings, can also be considered in real-world lighting design.

ACKNOWLEDGMENTS

We would like to thank all reviewers for their insightful comments. We also thank Jialin Huang, Bing Xu for preparing the dataset and helpful discussions, Jacob Si for preparing the video, Yunjin Zhang for proofreading the article, and all participants in our perceptual studies.

REFERENCES

3ds Max. 2021. 3ds Max. Retrieved from https://www.autodesk.com/products/3ds-max/overview.

Armen Avetisyan, Manuel Dahnert, Angela Dai, Manolis Savva, Angel X. Chang, and Matthias Nießner. 2019. Scan2CAD: Learning CAD model alignment in RGB-D scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2614–2623.

Sai Bi, Kalyan Sunkavalli, Federico Perazzi, Eli Shechtman, Vladimir G. Kim, and Ravi Ramamoorthi. 2019. Deep CG2Real: Synthetic-to-real translation via image disentanglement. In *Proceedings of the IEEE International Conference on Computer Vision*. 2730–2739.

Jeremy Birn. 2014. *Digital Lighting & Rendering*. Pearson Education.

Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. 2011. Learning photographic global tonal adjustment with a database of input/output image pairs. In *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 97–104.

Stanislas Chaillou. 2019. *AI + Architecture: Towards a New Approach*. Master's thesis. Harvard School of Design.

Kang Chen, Kun Xu, Yizhou Yu, Tian-Yi Wang, and Shi-Min Hu. 2015. Magic decorator: Automatic material suggestion for indoor digital scenes. *ACM Trans. Graph.* 34, 6 (2015), 1–11.

Yu-Sheng Chen, Yu-Ching Wang, Man-Hsin Kao, and Yung-Yu Chuang. 2018. Deep photo enhancer: Unpaired learning for image enhancement from photographs with GANs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6306–6314.

Coohom. 2022. Coohom. Retrieved from https://www.coohom.com/.

Bradley Efron and Robert Tibshirani. 1986. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statist. Sci.* Feb. (1986), 54–75.

Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, and Hao Zhang. 2021. 3D-FRONT: 3D furnished rooms with layOuts and semaNTics. In *Proceedings of the IEEE International Conference on Computer Vision*. 10933–10942.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* 27 (2014), 2672–2680.

Gary Gordon. 2015. *Interior Lighting for Designers*. John Wiley & Sons.

Daniel H. Greene. 1983. The decomposition of polygons into convex parts. *Computat. Geom.* 1 (1983), 235–259.

Ankur Handa, Viorica Patraucean, Vijay Badrinarayanan, Simon Stent, and Roberto Cipolla. 2016a. Understanding real world indoor scenes with synthetic data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4077–4085.

Ankur Handa, Viorica Pătrăucean, Simon Stent, and Roberto Cipolla. 2016b. SceneNet: An annotated model generator for indoor scene understanding. In *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 5737–5743.

Ruizhen Hu, Zeyu Huang, Yuhan Tang, Oliver Van Kaick, Hao Zhang, and Hui Huang. 2020. Graph2Plan: Learning floorplan generation from layout graphs. *ACM Trans. Graph.* 39, 4 (2020), 118–1.

Yuanming Hu, Hao He, Chenxi Xu, Baoyuan Wang, and Stephen Lin. 2018. Exposure: A white-box photo post-processing framework. *ACM Trans. Graph.* 37, 2 (2018), 1–17.

Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. 2017. DSLR-quality photos on mobile devices with deep convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 3277–3285.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1125–1134.

Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, and Delio Vicini. 2022. DR. JIT: A just-in-time compiler for differentiable rendering. *ACM Trans. Graph.* 41, 4 (2022), 1–19.

Sam Jin and Sung-Hee Lee. 2019. Lighting layout optimization for 3D indoor scenes. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 733–743.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 694–711.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2020. Analyzing and improving the image quality of StyleGAN. In *Proceedings of the IEEE International Conference on Computer Vision*.

John K. Kawai, James S. Painter, and Michael F. Cohen. 1993. Radioptimization: Goal based rendering. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*. 147–154.

William B. Kerr and Fabio Pellacini. 2009. Toward evaluating lighting design interface paradigms for novice users. *ACM Trans. Graph.* 28, 3 (2009), 1–9.

Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. 2017. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*. 1857–1865.

Manyi Li, Akshay Gadi Patil, Kai Xu, Siddhartha Chaudhuri, Owais Khan, Ariel Shamir, Changhe Tu, Baoquan Chen, Daniel Cohen-Or, and Hao Zhang. 2019. GRAINS: Generative Recursive Autoencoders for Indoor Scenes. *ACM Trans. Graph.* 38, 2 (2019), 1–16.

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018. Differentiable Monte Carlo ray tracing through edge sampling. *ACM Trans. Graph.* 37, 6 (2018), 1–11.

Zhengqin Li, Ting-Wei Yu, Shen Sang, Sarah Wang, Meng Song, Yuhan Liu, Yu-Ying Yeh, Rui Zhu, Nitesh Gundavarapu, Jia Shi, Sai Bi, Hong-Xing Yu, Zexiang Xu, Kalyan Sunkavalli, Milos Hasan, Ravi Ramamoorthi, and Manmohan Chandraker. 2021. OpenRooms: An open framework for photorealistic indoor scene datasets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7190–7199.

Wen-Chieh Lin, Tsung-Shian Huang, Tan-Chi Ho, Yueh-Tse Chen, and Jung-Hong Chuang. 2013. Interactive lighting design with hierarchical light representation. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, 133–142.

Tianqiang Liu, Aaron Hertzmann, Wilmot Li, and Thomas Funkhouser. 2015. Style compatibility for 3D furniture models. *ACM Trans. Graph.* 34, 4 (2015), 1–9.

Joe Marks, Brad Andalman, Paul A. Beardsley, William Freeman, Sarah Gibson, Jessica Hodgins, Thomas Kang, Brian Mirtich, Hanspeter Pfister, Wheeler Ruml, et al. 1997. Design galleries: A general approach to setting parameters for computer graphics and animation. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*. 389–400.

Maya. 2021. Maya. Retrieved from https://www.autodesk.com/products/maya/overview.

Paul Merrell, Eric Schkufza, Zeyang Li, Maneesh Agrawala, and Vladlen Koltun. 2011. Interactive furniture layout using interior design guidelines. *ACM Trans. Graph.* 30, 4 (2011), 1–10.

Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).

Naila Murray, Luca Marchesotti, and Florent Perronnin. 2012. AVA: A large-scale database for aesthetic visual analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2408–2415.

John A. Nelder and Roger Mead. 1965. A simplex method for function minimization. *Comput. J.* 7, 4 (1965), 308–313.

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. 2019. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2337–2346.

Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. 2021. ATISS: Autoregressive Transformers for Indoor Scene Synthesis. *Adv. Neural Inf. Process. Syst.* 34 (2021).

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga and others. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*. 32, (2019).

Fabio Pellacini, Frank Battaglia, R. Keith Morley, and Adam Finkelstein. 2007. Lighting with paint. *ACM Trans. Graph.* 26, 2 (2007), 9–es.

Fabio Pellacini, Parag Tole, and Donald P. Greenberg. 2002. A user interface for interactive cinematic shadow design. *ACM Trans. Graph.* 21, 3 (2002), 563–566.

Planner5D. 2022. Planner5D. Retrieved from https://www.planner5d.com/.

Haocheng Ren, Hao Zhang, Jia Zheng, Jiaxiang Zheng, Rui Tang, Yuchi Huo, Hujun Bao, and Rui Wang. 2022. MINERVAS: Massive interior environments virtual synthesis. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 63–74.

Daniel Ritchie, Kai Wang, and Yu-an Lin. 2019. Fast and flexible indoor scene synthesis via deep convolutional generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6182–6190.

Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. 2021. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE International Conference on Computer Vision*. 10912–10922.

Chris Schoeneman, Julie Dorsey, Brian Smits, James Arvo, and Donald Greenberg. 1993. Painting with light. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques.* 143–146.

Michael Schwarz and Peter Wonka. 2014. Procedural design of exterior lighting for buildings with complex constraints. *ACM Trans. Graph.* 33, 5 (2014), 1–16.

Ram Shacked and Dani Lischinski. 2001. Automatic lighting design using a perceptual quality metric. In *Computer Graphics Forum*, Vol. 20. Wiley Online Library, 215–227.

Lior Shapira, Ariel Shamir, and Daniel Cohen-Or. 2009. Image appearance exploration by model-based navigation. In *Computer Graphics Forum*, Vol. 28. Wiley Online Library, 629–638.

Evan Shimizu, Sylvain Paris, Matt Fisher, Ersin Yumer, and Kayvon Fatahalian. 2019. Exploratory stage lighting design using visual objectives. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 417–429.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. 2012. Indoor segmentation and support inference from RGBD images. In *Proceedings of the European Conference on Computer Vision (ECCV).* Springer, 746–760.

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.

Yifan Sun, Qixing Huang, Dun-Yu Hsiao, Li Guan, and Gang Hua. 2021. Learning view selection for 3D scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 14464–14473.

Hossein Talebi and Peyman Milanfar. 2018. NIMA: Neural image assessment. *IEEE Trans. Image Process.* 27, 8 (2018), 3998–4011.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *J. Mach. Learn. Res.* 9, 11 (2008).

VRay. 2021. VRay.Retrieved from https://www.chaosgroup.com/.

Andreas Walch, Michael Schwärzler, Christian Luksch, Elmar Eisemann, and Theresia Gschwandtner. 2019. LightGuider: Guiding interactive lighting design using suggestions, provenance, and quality visualization. *IEEE Trans. Visualiz. Comput. Graph.* 26, 1 (2019), 569–578.

Kai Wang, Yu-An Lin, Ben Weissmann, Manolis Savva, Angel X. Chang, and Daniel Ritchie. 2019. PlanIT: Planning and instantiating indoor scenes with relation graph and spatial prior networks. *ACM Trans. Graph.* 38, 4 (2019), 1–15.

Kai Wang, Manolis Savva, Angel X. Chang, and Daniel Ritchie. 2018b. Deep convolutional priors for indoor scene synthesis. *ACM Trans. Graph.* 37, 4 (2018), 1–14.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018a. High-resolution image synthesis and semantic manipulation with conditional GANs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.*

Xinpeng Wang, Chandan Yeshwanth, and Matthias Nießner. 2021. SceneFormer: Indoor scene generation with transformers. In *Proceedings of the International Conference on 3D Vision (3DV).* IEEE, 106–115.

Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. 2018. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 606–615.

Wenming Wu, Xiao-Ming Fu, Rui Tang, Yuhan Wang, Yu-Hao Qi, and Ligang Liu. 2019. Data-driven interior plan generation for residential buildings. *ACM Trans. Graph.* 38, 6 (2019), 1–12.

Bing Xu, Junfei Zhang, Rui Wang, Kun Xu, Yong-Liang Yang, Chuan Li, and Rui Tang. 2019. Adversarial Monte Carlo denoising with conditioned auxiliary feature modulation. *ACM Trans. Graph.* 38, 6 (2019), 224–1.

Kai Xu, Rui Ma, Hao Zhang, Chenyang Zhu, Ariel Shamir, Daniel Cohen-Or, and Hui Huang. 2014. Organizing heterogeneous scene collections through contextual focal points. *ACM Trans. Graph.* 33, 4 (2014), 1–12.

Zhicheng Yan, Hao Zhang, Baoyuan Wang, Sylvain Paris, and Yizhou Yu. 2016. Automatic photo adjustment using deep neural networks. *ACM Trans. Graph.* 35, 2 (2016), 1–15.

Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. 2017. DualGAN: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision.* 2849–2857.

Lap Fai Yu, Sai Kit Yeung, Chi Keung Tang, Demetri Terzopoulos, Tony F. Chan, and Stanley J. Osher. 2011. Make it home: Automatic optimization of furniture arrangement. *ACM Trans. Graph.* 30, 4 (2011).

Cheng Zhang, Bailey Miller, Kan Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020a. Path-space differentiable rendering. *ACM Trans. Graph.* 39, 4 (2020).

Zaiwei Zhang, Zhenpei Yang, Chongyang Ma, Linjie Luo, Alexander Huth, Etienne Vouga, and Qixing Huang. 2020b. Deep generative modeling for scene synthesis via hybrid representations. *ACM Trans. Graph.* 39, 2 (2020), 1–21.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. 2017a. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision.* 2223–2232.

Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A. Efros, Oliver Wang, and Eli Shechtman. 2017b. Toward multimodal image-to-image translation. In *Proceedings of the International Conference on Advances in Neural Information Processing Systems.*

Received 15 May 2022; revised 26 October 2022; accepted 22 December 2022