The Visual Computer (2022) 38:4251–4265
https://doi.org/10.1007/s00371-021-02292-2

ORIGINAL ARTICLE

Object-based illumination transferring and rendering for applications of mixed reality

Di Xu[1] · Zhen Li[2] · Qi Cao[3]

Accepted: 19 August 2021 / Published online: 7 October 2021

Abstract
In applications of augmented reality or mixed reality, rendering virtual objects in real scenes with consistent illumination is crucial for realistic visualization experiences. Prior learning-based methods reported in the literature usually attempt to reconstruct complicated high dynamic range environment maps from limited input, and rely on a separate rendering pipeline to light up the virtual object. In this paper, an object-based illumination transferring and rendering algorithm is proposed to tackle this problem within a unified framework. Given a single low dynamic range image, instead of recovering lighting environment of the entire scene, the proposed algorithm directly infers the relit virtual object. It is achieved by transferring implicit illumination features which are extracted from its nearby planar surfaces. A generative adversarial network is adopted in the proposed algorithm for implicit illumination features extraction and transferring. Compared to previous works in the literature, the proposed algorithm is more robust, as it is able to efficiently recover spatially varying illumination in both indoor and outdoor scene environments. Experiments have been conducted. It is observed that notable experiment results and comparison outcomes have been obtained quantitatively and qualitatively by the proposed algorithm in different environments. It shows the effectiveness and robustness for realistic virtual object insertion and improved realism.

Keywords Lighting estimation · Illumination transferring · Virtual object rendering · Mixed reality applications

1 Introduction

Compositing realistic virtual objects rendered into real scenes and illumination estimation is fundamental, but challenging problems in computer vision and computer graphics. Emerging applications, such as augmented reality (AR), mixed reality (MR), live streaming, or film production, demand realistic graphical visualization and rendering [3]. Conventionally, the problem consists of two steps, i.e., lighting estimation and virtual objectrendering. The high dynamic range (HDR) environment maps are usually adopted to record the illumination of the entire scene. It reproduces a great dynamic range of illumination which is even higher than that of the human visual system. However, direct capture of HDR images is not feasible for most cases, as it requires tedious setups and expensive devices [29]. On the contrary, commercial AR tools, e.g., Google's ARCore or Apple's ARkit, provide lightweight mobile applications to estimate scene illuminations. But these techniques only consider the camera exposure information and are regarded to be rudimentary.

In order to achieve realistic rendering, prior works in the literature try to obtain the HDR environment maps in various ways. Some works are reported to insert certain objects into scenes [19], such as light probes [8,9], 3D objects [14,35] with known properties, or human faces [4,42]. Some works assume that additional information is available, e.g., panoramas [43], depth [25,36], or user input [21]. Although these reported methods work well in certain scenarios, such requirements would not always be feasible for most of practical applications. Therefore, recent works are reported to infer HDR environment maps from limited input information by learning. Works are reported to

✉ Qi Cao
qi.cao@glasgow.ac.uk

Di Xu
di.xu@ivglass.com

Zhen Li
yodlee@mail.nwpu.edu.cn

1 AI Lab of Shadow Creator Inc, Beijing, China

2 Northwestern Polytechnical University, Xi'an, China

3 School of Computing Science, University of Glasgow, Singapore Campus, Singapore, Singapore

recover HDR environment maps from a single limited field-of-view (FOV) low dynamic range (LDR) image for indoor scenes [12,13,31]. Prior works make use of the sky model and try to infer the outdoor lighting in [17,18,44]. Although the learning-based works achieve plausible results, recovering the illumination of the entire scene is still a highly ill-posed problem. It is mainly due to the complexity of HDR environment maps and the missing information from the input LDR RGB image. The illumination of a scene is resulted in by many factors, including various lighting sources, surface reflectance, scene geometry, and object inter-reflections. The limited FOV would only capture about 6% of the panoramic scene according to [23]. It makes the problem even more challenging, since light sources are very likely not captured in the input LDR image. More importantly, HDR environment maps only account for illumination incidents from every direction at a particular point, which is often violated for spatially varying lighting in the scene [11]. Consequently, describing illumination of the entire scene with a single HDR environment map may fall short for realistic rendering of virtual objects.

In this research, an object illumination estimation algorithm is proposed by transferring the lighting conditions from 3D planes detected in real scenes to virtual objects. Instead of learning an entire HDR environment map, the proposed algorithm directly infers the relit virtual object itself. It is inspired and conceived by two important observations as follows.

- Planar regions are quite common in both indoor and outdoor scenes. They offer important geometric and photometric cues in tasks, such as scene understanding [34], scene reconstruction [5], and navigation [24]. Such cues could also help the illumination estimation.
- The per-vertex object lighting model, overall illumination (OI) introduced by [40], can be utilized to represent the overall effect of all incident lights at the particular 3D point.

Taking advantage of the easy-to-obtain OI from planes in the scenes, we propose to adopt a novel generative adversarial network (GAN) to transfer the implicit illumination features. In the proposed algorithm, a 3D primitive layer between planes and virtual objects is introduced to make lighting feature transferring more robust to complicated geometry. It can be regarded as a bridge between planes and objects during the transfer learning. Planar surfaces and the desired virtual objects are used to train an autoencoder network to learn implicit illumination features from OI. Then, the GAN in the proposed algorithm is guided to transfer OI among different objects with the photo-consistency constraint. Finally, relit objects are rendered from the predicted OI.

Given a single LDR RGB image, without recovering an entire environment map, the proposed algorithm directly infers rendered virtual objects by transferring the implicit illumination features of planar surfaces in real scenes. It generates realistic rendering with spatially varying illumination. An example of rendered scene effects using our proposed algorithm is shown in Fig. 1, where three virtual bunnies are rendered into different locations of a real-world kitchen scene. It is observed that different and realistic illumination effects are rendered for each virtual bunny according to the corresponding real world illuminations. The proposed algorithm will be very useful in the applications of AR or MR which overlay virtual objects with real world scenes.

Fig. 1 An example of effects using our proposed algorithm for three virtual bunnies being rendered into different locations of a real scene

The main contributions of this paper are as follows:

1. Rather than recovering the complicated HDR environment map from a single RGB image, a novel algorithm is proposed that directly infers the rendered virtual object by transferring the illumination features of planar surfaces in real scenes.
2. The nature of our proposed transfer learning and planar surface illumination estimation approach makes it more robust to different scenarios in both indoor and outdoor scenes with spatially varying illumination. It is more versatile than prior works reported [11–13,17,18,31,44] which only focus on particular cases.
3. Although the proposed algorithm is trained with planar surfaces in this research, it is also feasible for other common geometries for illumination transferring.

To benefit readers of this paper for research and comparison purposes, we have published our source codes of the proposed algorithm at Github with the web link: https://github.com/xudi1227/illumiNet.

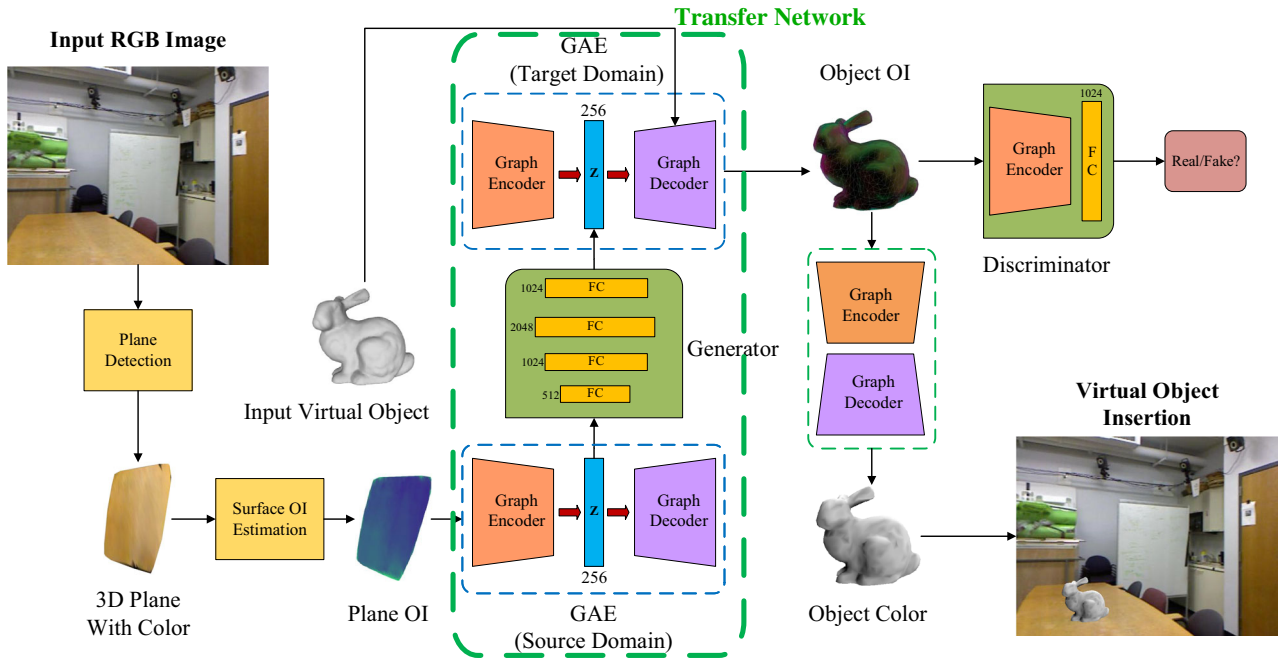

Fig. 2 The proposed algorithm of object-based illumination transferring and rendering

2 Related works

Illumination estimation from a scene is a long-standing topic and has been extensively studied. The problem is complicated, even ill-posed sometimes, since it depends on multiple factors, including lighting, scene geometry, surface material, reflectance, etc. Direct capture methods are reported by Debevec [8,9] by taking photographs of a polished metal ball in the scene. An omnidirectional HDR radiance map with great dynamic range of luminosity is then reconstructed. It can be applied to render virtual objects into scenes. However, inserting such an additional object into the scene is infeasible for most practical scenarios and difficult to scale. Other than HDR environment map, spherical harmonics (SH) are often exploited to parameterize incident illumination [2,20,37,38,41]. Geometry and reflectance properties are jointly estimated with lighting [45]. Due to heavy computational costs of higher order components of SH, usually only the low-frequency parts are considered during the optimization, e.g., 2nd-order SH in [38] or 5th-order in [13,41]. Even though the use of SH can simplify the formulation of incident illumination, it still requires estimating the visibility and albedo maps. It actually takes a considerable amount of time, especially for a dense mesh [37]. Xu et al. introduce a novel concept named vertex overall illumination vector, to represent overall effect of all incident lights at each individual 3D point of the object [40]. In that work, there is no need to recover lighting of the entire scene. However, its improvement over SH is only demonstrated in terms of shape-from-shading. Meanwhile, lighting estimation is also studied as an intermediate result for specific purposes. For example, the lighting is estimated for the purpose of image enhancement [10] and [28]. Han et al. present their works to normalize the illumination on human faces, in order to improve the performance of face recognition [16]. Illumination is estimated for the light consistency to render virtual pedestrians into real environment scenes [30].

Thanks to the rapid development of deep learning, recent works have been tried to directly estimate illumination from a single LDR image with limited FOV. An end-to-end convolutional neural network (CNN) is reported to recover environment maps from a single view-limited LDR image in an indoor scene [12]. In that work, a large number of LDR panoramas with source light position labels are employed to train and predict the position of the light source first. It then uses a small number of HDR panoramas to fine-tune the network and estimate light intensity. Learning from a single image is utilized to predict parameters of the Hosek-Wilkie sky model to get the outdoor scene illumination [18]. A more sophisticated Lalonde–Matthews (L–M) outdoor light model is introduced to predict model parameters from LDR images for an outdoor HDR panorama [44]. Two pictures taken by the front and rear mobile phone cameras are utilized to estimate low-frequency lighting [6]. A special camera device is employed to take scene photographs and three balls with the same dimension but different materials, in order to collect pairs of images and HDR environment map data [23]. Calian *et al.* make use of a Sun+Sky model and a LDR face photograph as an outdoor light probe to estimate illumination conditions [4]. But this approach is prone

to local minima. When the light source is behind the person, the model estimates the wrong result, as it is unable to get enough illumination information from the backlit face photograph. Three sub-neural networks are designed to progressively estimate geometry, LDR panoramas, and final HDR environment map based on input image and locale [31]. A method is reported for estimating the spatially varying indoor illumination in real time, which combines global features and local features to predict spherical harmonics coefficients [13]. However, due to complexity and unknownness of real scenes, especially when light sources are not captured in the input image, inconsistent illumination of predicted panoramic HDR is inevitable. Essentially, most of these works usually get the mapping of input images to environment maps or SH coefficients. While our proposed algorithm directly predicts illumination effects of the inserted virtual object itself, making the problem less error prone.

Gardner et al. presented their work to replace the HDR environment maps with parametric representations [11]. The idea sounds similar to ours, but there are two major differences. Firstly, their lighting model is a set of discrete 3D lights describing the entire scene, while our proposed approach directly transfers the vertex overall illumination from detected planes to virtual objects. Secondly, their work only applies to the indoor illumination. It is less versatile compared to our proposed approach that works for both indoor and outdoor scenes.

Some recent works have also been reported to estimates HDR lighting environment maps from more complicated inputs. A data-driven model is introduced that estimates lighting from a spherical panorama [15]. Narrow-baseline stereo pairs of images are utilized to estimate a 3D volumetric RGB model, followed by the estimation of incident illumination [32]. An inverse rendering pipeline is presented to take RGB-Depth (RGBD) video frames as input [36]. Tarko et al. [33] employ inverse tone mapping to recover HDR environment maps and reproduce real-world lighting conditions, when inserting virtual objects into a moving-camera 360° video. However, it is tricky to set reference objects, as it is required to be homogeneous material and mostly convex shape. While these reported works achieve realistic results, the inputs are usually more difficult to obtain, compared to ordinary RGB images captured by general commercial cameras, such as mobile phone cameras.

3 Proposed object-based illumination transferring and rendering

In the proposed algorithm, taking a single LDR image as the input, planar surfaces are detected in the scene and compute its overall illumination (OI) [40]. The implicit illumination features are extracted by a graph autoencoder (GAE), and then transferred to the virtual object using a GAN. Finally, the virtual object is rendered and inserted to the image according to the predicted OI.

In our research, the goal is transferring illumination and lighting effects of common structures in the scene, e.g., planar surfaces, to rendered virtual objects. The proposed algorithm is shown in Fig. 2. Given a single input LDR RGB image, planar surfaces of any specific region in the scene will be first detected for virtual object compositing. The OI of a particular plane is then computed according to its shading cues [40]. In the next step, two graph autoencoders (GAE) [22] are applied to extract the corresponding implicit illumination features. The proposed transfer network is shown in the big green rectangle in Fig. 2. Such illumination features are then transferred from planes to virtual objects. Finally, the virtual object is rendered into the image scene by the last GAE with its corresponding OI.

In the remaining parts of this section, each sub-module, network architecture, as well as the implementation details will be illustrated one by one.

3.1 Overall illumination

The concept of vertex OI is introduced to describing the overall effect of all incident lights at each point of the object [39]. In the work reported in [40], the reflectance model is approximately described as shown in Eq. (1).

$$I_o(v) = \int_{\Omega(v)} \rho(v) I_i(v, \omega) \max(\omega \cdot \mathbf{n}(v), 0) V(v, \omega) \mathrm{d}\omega \quad (1)$$

where $I_o(v)$ is the reflected radiance of the object at vertex v; $\rho(v)$ is the albedo or an approximation of the bidirectional reflectance distribution function (BRDF); ω is the incident direction; $I_i(v, \omega)$ is the incident radiance along ω; $\mathbf{n}(v)$ is the unit surface normal at v; $\Omega(v)$ represents a hemisphere of incident directions at v; and $V(v, \omega)$ stands for a binary visibility function of vertex v to direction ω.

Let $\Omega'(v)$ denote the subset of Ω for which $\omega \cdot \mathbf{n}(v) > 0$ and $V(v, \omega) = 1$. Then, the model representing in Eq. (1) can be rewritten as Eq. (2).

$$I_o(v) = \left(\int_{\Omega'(v)} \rho(v) I_i(v, \omega) \omega \mathrm{d}\omega \right) \cdot \mathbf{n}(v). \quad (2)$$

Let

$$L(v) = \int_{\Omega'(v)} \rho(v) I_i(v, \omega) \omega \mathrm{d}\omega. \quad (3)$$

$L(v)$ is called the vertex overall illumination vector at v.

As shown in Fig. 3a, at each point v_i on the surface, the vertex overall illumination vector $L(v_i)$ represents the overall

Fig. 3 **a** Vertex OI vector $L(v_i)$ at each point v_i on the surface. **b** Visualization of $L(v)$ on 3D bunny and buddha objects, with 3D vector being directly mapped to RGB color space

effect of all incident lights such as $l_1, l_2, \ldots, l_m$ from different directions. For a given vertex v on a 3D model, its OI is denoted as a 3D vector $L(v)$; while its unit surface normal is denoted as $\mathbf{n}(v)$, and the reflected radiance of v is denoted as $I(v)$. Then, the reflected radiance $I(v)$ can be computed as: $I(v) = L(v) \cdot \mathbf{n}(v)$.

For the method reported in [40], the OI is applied on Debevec's light probe images [8]. A light probe image is an omnidirectional HDR image that records the incident illumination conditions at a particular point in space. Such images are usually captured under general and natural illumination conditions. The OI of each vertex for object compositing purpose is inferred in the proposed algorithm. There is no need to estimate individual light sources in the scene and compute the visibility function of each vertex. Some examples for visualizing object-based OI are shown in Fig. 3b, where the 3D vectors of OI are directly mapped to RGB color space under different lighting for a better understanding. It shows that the bunny and buddha models with color mapped $L(v)$ under the corresponding lighting.

3.2 Plane detection

Different from prior works reported in the literature that are required to insert certain geometries [4,8,9,14,35,42], our proposed algorithm makes use of the *planes* that already exist in the scene. Planar surfaces with different sizes and shapes, e.g., floors, walls, tables, or the ground, are some most commonly seen geometries in indoor and outdoor scenes.

How to detect if there is any plane in the scenes? Our approach does not reinvent the wheel, but utilizes an existing plane detection method in the literature, named PlaneRCNN reported in [24]. The PlaneRCNN is a detection-based neural network for piecewise planar reconstruction from a single RGB image. In order to boost the performance, the PlaneRCNN learns for detecting planar regions, regressing plane parameters and instance masks, refining segmentation masks globally with a nearby view during training. PlaneRCNN outperforms existing state-of-the-art methods with significant

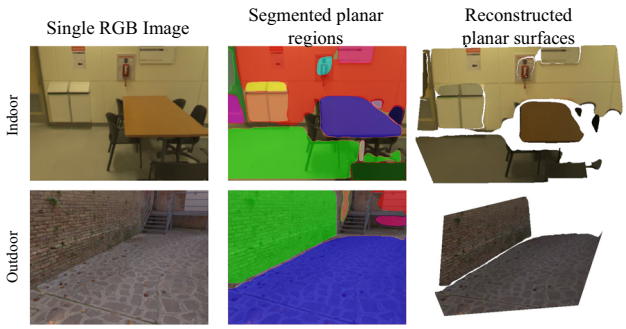

Fig. 4 Few examples of plane detection results

margins in the plane detection [24]. This is the reason for our approach to choose the PlaneRCNN method for plane detection.

Given a single RGB image input to the system, the PlaneRCNN is used to detect the planes first. The 3D piecewise planar surfaces are then reconstructed and the corresponding 3D coordinates are estimated, as shown in Fig. 4. Then, the OI of the planes can be computed according to its shading cues. For the shape-from-shading problem in [40], the object geometry is refined from its estimated OI by solving a total variation minimization formulation.

In our proposed algorithm, the similar procedure is adopted. But it is for a reversed goal, as the OI is computed from known geometry, i.e., 3D planes. It is worth highlighting that our proposed algorithm is able to achieve illumination transferring from various geometries, not only the planar surfaces, but also non-planar geometries like spheres, hemisphere, cylinders, cube, ring, etc. Illumination deep features can be transferred from various geometries to virtual object targets.

As a quick verification to show the robustness of our proposed algorithm, the planes are replaced by other geometries such as sphere, hemisphere, cube, ring, and cylinder. The quantitative evaluation results under 1000 different lighting environments are shown in Table 1. It is observed for the proposed algorithm that good performances with low mean

Table 1 Quantitative evaluation on illumination transferring from various geometries

Geometry	MAE of recovered OI
Sphere	0.019
Hemisphere	0.0665
Ring	0.0664
Cube	0.066
Cylinder	0.0223
Cone	0.0221
Plane	0.045

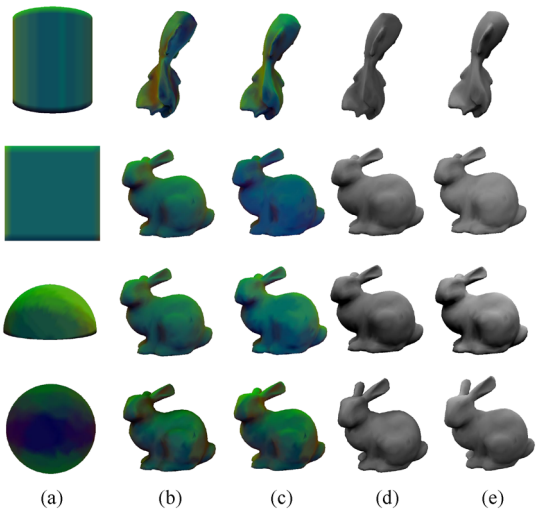

Fig. 5 Qualitative evaluation on illumination transferring from various geometries with a particular shape at each row

absolute error (MAE) are achieved for various geometries which is similar to the planar surface with the MAE value being about 0.045.

Besides, the qualitative evaluation results for four types of geometries are also side by side compared with the ground truths, as shown in Fig. 5. The OI of each input shape, i.e., source domain of our transfer network is shown in Column (*a*) of Fig. 5. The predicted OI of the virtual models, i.e., target domain of our transfer network, is listed in Column (*b*), according to each input geometry. The ground truth OI of the virtual model is shown in Column (*c*). Next, the predicted relit virtual models are listed in Column (*d*). The ground truth relit virtual models are shown in Column (*e*) in Fig. 5. It is observed that our predicted results listed in Column (*b*) and Column (*d*) are similar with ground truths in Column (*c*) and Column (*e*).

The quantitative evaluation results in Table 1 and qualitative evaluation results in Fig. 5 illustrate that our proposed algorithm is robust to compute well for planar surface as well as other geometries. Note that although results of some geometries are better than plane surfaces, they are less commonly seen in most real-world images.

3.3 Implicit illumination features extraction

The graph convolution network (GCN) model reported in [22] is based on a variant of convolutional neural networks which operate directly on graph. It is capable of encoding both graph structure and node features for semi-supervised classification. According to this work, the single layer of the graph convolutional neural network is defined in Eq. (4).

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{4}$$

where $H^{(l)} \in \mathbb{R}^{(N\times D)}$ represents the matrix of activations in the input of lth layer network, with the initial input as $H^{(0)}$. $\tilde{A} = A + I_N$ is added self-joining adjacency matrix. I_N is the identity matrix, while N is the number of nodes in the graph. Each node is represented by the feature vector with D dimension. $\tilde{D}$ is a degree matrix, derived by $\tilde{D}_{(ii)} = \Sigma_j \tilde{A}_{(ij)}$. $W^{(l)} \in \mathbb{R}^{(D\times D)}$ is the parameter to be trained. σ is the corresponding activation function, and tanh is used.

Inspired by the GCN reported in [22], we design our own GAE structure in this research work. Before transferring the object illumination in our work, we need to extract its implicit illumination features from the object OI. Our designed GAE structure is independently trained to learn the OI feature representation of the source object and the target object.

As shown in Fig. 6, our designed encoder–decoder consists of a two-layer graph convolution and a fully connected (FC) layer. The latent feature vector is 256-dimensional. Each GAE contains a unique representation of this domain object, including shapes, normals, poses, etc. Since the input data to our proposed network are 3D models, our designed graph is defined as an undirected graph $G = (V, E)$, where E is the adjacency matrix of the graph; V is the feature matrix with a dimension of six times of the vertex number, including normal, OI, and RGB information.

3.4 Object illumination transferring

A method to stabilize GAN is reported in [27], i.e., Unrolled GAN, which is capable of solving the common problem of GAN mode collapse. This technique defines the generator objective with respect to an unrolled optimization of the discriminator. This technique will be adopted in our proposed algorithm, presented in this subsection as follows.

Our proposed transfer network for object illumination is based on a GAN. As shown in Fig. 7a, the generator of our designed GAN is a multilayer perceptron (MLP) consisting of 5 layers of FC. Except for the last layer, each layer is followed by a batch normalization (BN) layer and a LeakyReLU layer. The parameter of the LeakyReLU layer is set as 0.2.

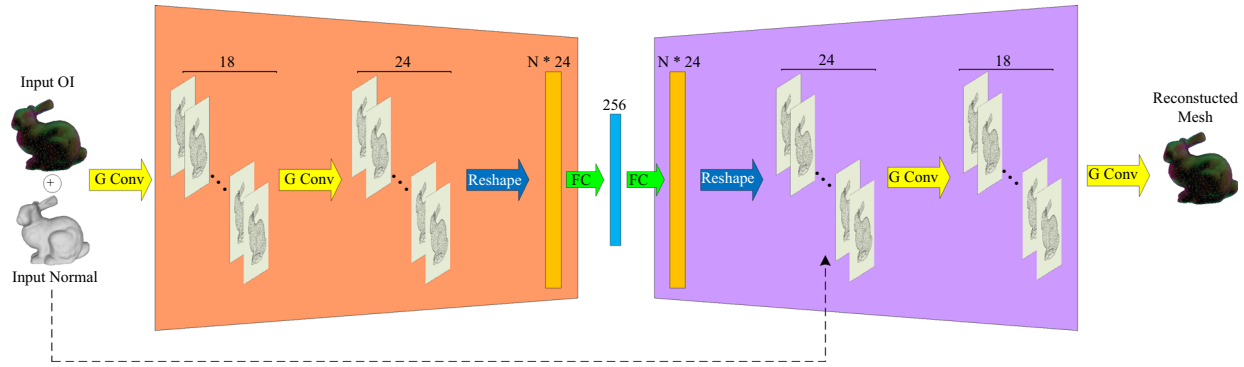

Fig. 6 Structure of our designed graph autoencoder, which is used to extract the implicit illumination features for illumination transferring

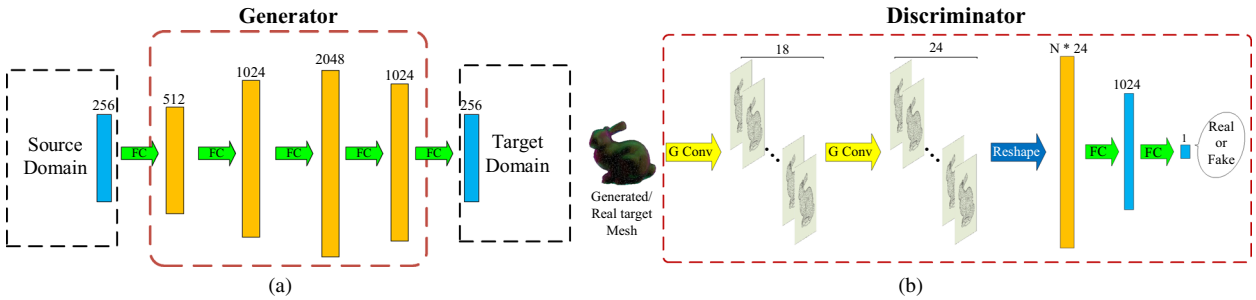

Fig. 7 **a** Generator of our GAN: translates the latent space vector of the source domain to the latent space of the target domain. **b** Discriminator of our GAN: discriminates whether the input data matches the distribution of the target domain

Our designed discriminator structure shown in Fig. 7b consists of two layers of convolution and two layers of FC layers, which is similar to the graph autoencoder. All layers except for the last layer are connected to the BN layer. All the convolution layers are connected to the tanh layer.

In our algorithm, the generator transfers the latent feature vector of target domain from that of input source domain. The decoder obtains the OI of the virtual object. The discriminator determines whether the generated OI conforms to the distribution of the target domain. Through this minimax game, the final generator produces properties of real target objects.

In order to alleviate the mode collapse, we utilize the technique of Unrolled GAN [27]. Following that technique, our generator function G updates itself by predicting the future responses of the discriminator D in advance. It makes the discriminator D more difficult to respond to the generator function G's update, avoiding the problem of mode skipping.

3.5 Rendering

The goal of the work reported in [40] is 3D reconstruction. It can only infer OI from the color of 3D models, but not vice versa. Meanwhile, conventional rendering pipelines which take HDR environment maps or SH as the global lighting are infeasible for implicit illumination rendering.

In order to address such issues and improve the performance of the proposed algorithm in this work, we design another GAE as the renderer. It learns the color of 3D models from the corresponding OI. Its structure is almost the same as our designed GAE described in Sect. 3.3. The only difference is that this GAE generates the feature of $N\times1$, which is then expanded to $N\times3$, in order to get the intensity value of the final object.

To cast the shadow correctly, the dominant OI region on the virtual object is first computed. Next the corresponding lighting direction is synthesized based on the dominant OI region. The shadows are finally cast according to the given geometry of the virtual object and the plane underneath. An example scene of a rendered virtual bunny with shadow is cast on a real plane in this work shown in Fig. 8.

3.6 Loss functions of the proposed algorithm

GAE: In the proposed algorithm, the designed GAE eventually outputs the OI features with dimension of N×3. According to the input 3D object P, the designed GAE network reconstructs object $\hat{P}$. The reconstruction loss function of the designed GAE is defined in Eq. (5).

$$L_{\text{recons}} = \frac{1}{ND}\sum_{i=0}^{N}\sum_{k=0}^{D}\left|p_i^k - \hat{p}_i^k\right| \tag{5}$$

Fig. 8 Example scene of a rendered virtual object with shadow cast on a real plane

where N is the number of points in the model, and D is the number of OI features. The value of D is set as 3 in the proposed approach. The notation k represents the kth OI feature.

GAN: In the proposed algorithm, the initial objective function of the designed GAN is the squared error. It means that the essence of our proposed transfer network is Least Squares Generative Adversarial Networks (LSGAN) [26]. Data(y) is defined to represent the data of the target domain T, i.e., $y \in T$. While data(x) is represented the data of the source domain S, i.e., $x \in S$. For real data, y is defined as 1. For fake data, G(x) is defined as 0. The loss of the designed GAN is defined in Eq. (6).

$$L_{\text{LSGAN}} = E_{y\sim\text{data}(y)}[(D(y)-1)^2] + E_{x\sim\text{data}(x)}[(1-D(G(x)))^2] \tag{6}$$

Besides, several additional factors are also included in our proposed algorithm. These factors are described by the additional functions as follows.

In order to generate the features of the corresponding target object from the source object, an additional term, pairing loss L_{pair} is added. It is represented in Eq. (7).

$$L_{\text{pair}} = E_{x\sim\text{data}(x),y\sim\text{data}(y)} = [|y - G(x)|] \tag{7}$$

Moreover, we make use of the photo-consistency by adding a shading term as described in Eq. (8).

$$E_{sh} = \sum_{i=1}^{N} \| L(v_i)\cdot \mathbf{n}(v_i) - c_i \|^2 \tag{8}$$

where $L(v)\cdot\mathbf{n}(v)$ describes OI as illustrated in Sect. 3.1. The notation c_i is the average intensity values in all the multi-view images corresponding to the vertex v_i. It is the intensity error measuring the difference between the computed reflected radiance and the average captured intensities.

Since the OI is supposed to be piecewise smooth, the smooth loss of the 3D model can be calculated, which is defined in Eq. (9).

$$\nabla_M = \sum_{i=0}^{N}\sum_{k=0}^{D} \left| \left(\frac{1}{d_i}\sum_{j\in N_i} p_j^k\right) - p_i^k \right| \tag{9}$$

where d_i represents the degree of the ith node and N_i represents all neighbor nodes of the ith node.

The matrix form of the smooth loss can be represented in Eq. (10).

$$\nabla_M = \text{average}(D^{-1}AM - M) \tag{10}$$

where D is the degree matrix; A is the feature matrix; and M is the adjacency matrix.

In taking consideration of all these terms mentioned above, the total loss function can be revised and computed in Eq. (11).

$$L_{\text{total}} = L_{\text{LSGAN}} + \beta_{\text{pair}} L_{\text{pair}} + \beta_{\text{shading}} E_{sh} + \beta_{\text{smooth}} \nabla_M \tag{11}$$

where β_{pair}, β_{shading}, and β_{smooth} are the term coefficients of the total loss function.

3.7 Implementation details of the proposed algorithm

Dataset: In the literature, Debevec [7] reports a technique for approximating a light probe image as a constellation of light sources based on a median cut algorithm, which can realistically represent a complex lighting environment. A HDR environment dataset, SHlight, with various illumination conditions is reported in [6] for training and evaluation of predicting illuminations. A dataset of HDR environment maps, Laval Indoor Dataset is reported in [12] to learn a direct mapping from image to lighting and predict HDR scene illumination. A shading algorithm is reported in [40], to generate high fidelity 3D models from images under general lighting conditions. It is able to estimate both geometry and illumination using the same objective function. These prior works are very useful and applied into the dataset generation of our proposed approach. It will be discussed in this subsection as follows.

In the implementation stage of the proposed algorithm, synthetic data are generated for training purpose. A total of 10,000 sets of synthetic lighting environment are generated randomly to model the indoor and outdoor illuminations. For

each lighting condition, 32 synthetic point light sources are randomly placed in the 3D space. The corresponding OI and intensity of the virtual object are then computed. Additionally, a rotation perturbation is applied to the virtual object, so that our designed GAE can learn feature representation with various poses.

For real-world dataset used for the proposed algorithm, we use Debevec's median cut algorithm [7] to generate 3292 real environment illuminations from the real-world HDR environment maps of SHlight [6], and Laval Indoor Dataset [12], which are represented as 32-point sources.

In the proposed algorithm, these point sources are then randomly rotated three times to generate 9135 augmented data, among which 1000 (i.e., about 9%) are used as validation data. We apply the shading algorithm [40] to generate the corresponding OI and intensity for different objects as our fine-tuning data. The LDR images are cropped with random pitch, yaw and exposure. The HDR lighting is obtained under this setting. The ground truth lighting and predicted lighting from prior works are utilized to render the target objects, respectively. There are 247 sets of test data.

Training: The training is conducted with four GTX 1080TI GPUs. The entire procedure takes around four hours. Each sub-network is trained separately and finally fine-tuned as a whole. In the proposed algorithm, the GAE of the planar surfaces and that of the virtual object are trained separately. After that the renderer of the virtual object is trained. Finally, the transfer network is trained. GAE and the renderer are trained for 400 epochs with a batch size of 256, using the ADAM optimizer with the values of betas as (0.9, 0.999). The learning rate is set as 0.001. For the transfer network, the ADAM optimizer is set with the values of betas as (0.5, 0.99), the learning rate of G and D as 0.0001 and 0.0004, respectively, and 100 epochs training. The term coefficients of the loss function are set as β_{pair} being 1.0, β_{smooth} being 2.5, and β_{shading} being 0.3. For all dropout layers in the GCN layer, the parameter is set as 0.2.

4 Experiments

In this research, experiments have been conducted to evaluate the proposed algorithm quantitatively and qualitatively on several datasets. To illustrate the effectiveness and robustness of the proposed algorithm, the experiment results are compared with prior works reported in different scenarios, namely indoor [12,23,41], outdoor [17,18,23,41], and spatially varying environments [13].

It is worth mentioning that prior works in the literature generally account for the relighting error from a single view. It means that a 2D image is cropped from a particular view with the virtual object in the scene. Its relighting error on a pixel-wise basis is then calculated. We argue that measuring the rendered objects in 3D space is more reasonable, as nowadays multi-user AR or MR applications have gained broader popularity. For example, Microsoft's *Azure Spatial Anchors* [1] allows multiple users to place virtual contents in the same physical location, where rendered virtual objects can be seen on different devices in the same position or orientation relative to users' environment. Such scenarios require realistic rendering not only the object front views, but also the back views. In such a case, pixel-wise measurement from a single view is insufficient. The proposed object relighting algorithm is able to measure 3D relighting errors directly.

In the experiments, there are totally 1000 sets of testing lighting conditions in the synthetic validation data. For the real-world data, there are 141 indoor lighting scenes from SHlight [6] and Laval Indoor Dataset [12]. There are 106 outdoor lighting scenes from SHlight [6]. There are 76 lighting scenes from spatially varying data in [13]. For some rare cases if the plane detection module is failed, a synthetic plane is placed in the image as the source for our feature transferring framework.

4.1 Quantitative results

One of the evaluations in quantitative analysis is conducted on computing relighting errors of virtual objects. The relighting errors are computed in 3D space, where all the vertices on the virtual object are considered. It is observed in Table 2, our proposed algorithm achieves better results on indoor, outdoor, and spatially varying data. The proposed algorithm exhibits substantial performance improvements with much lower MAE and root mean squared error (RMSE), compared to the prior works for indoor [12,23,41] and those for outdoor [17,18,23,41]. As for the spatially varying data, the MAE and RMSE results of the proposed algorithm are also lower than [13]. The overall performance on these validation datasets illustrates the robustness and effectiveness of the proposed algorithm.

The effectiveness to include loss terms on shading loss and smooth loss in the proposed algorithm has also been studied. Table 3 shows that the relighting results are improved with inclusion of shading loss and smooth loss, as the shading loss enforces the photo-consistency of the rendered virtual object, based on its geometry and lighting. While the smooth loss measures that the lighting is expected to be piece-wise smooth.

4.2 Qualitative results

Besides the quantitative analysis, the qualitative analysis and comparisons have also been conducted. As mentioned in the previous section, the proposed algorithm relights the object from all directions in 3D space, instead of a single view. Therefore, not only the front views but also the back views

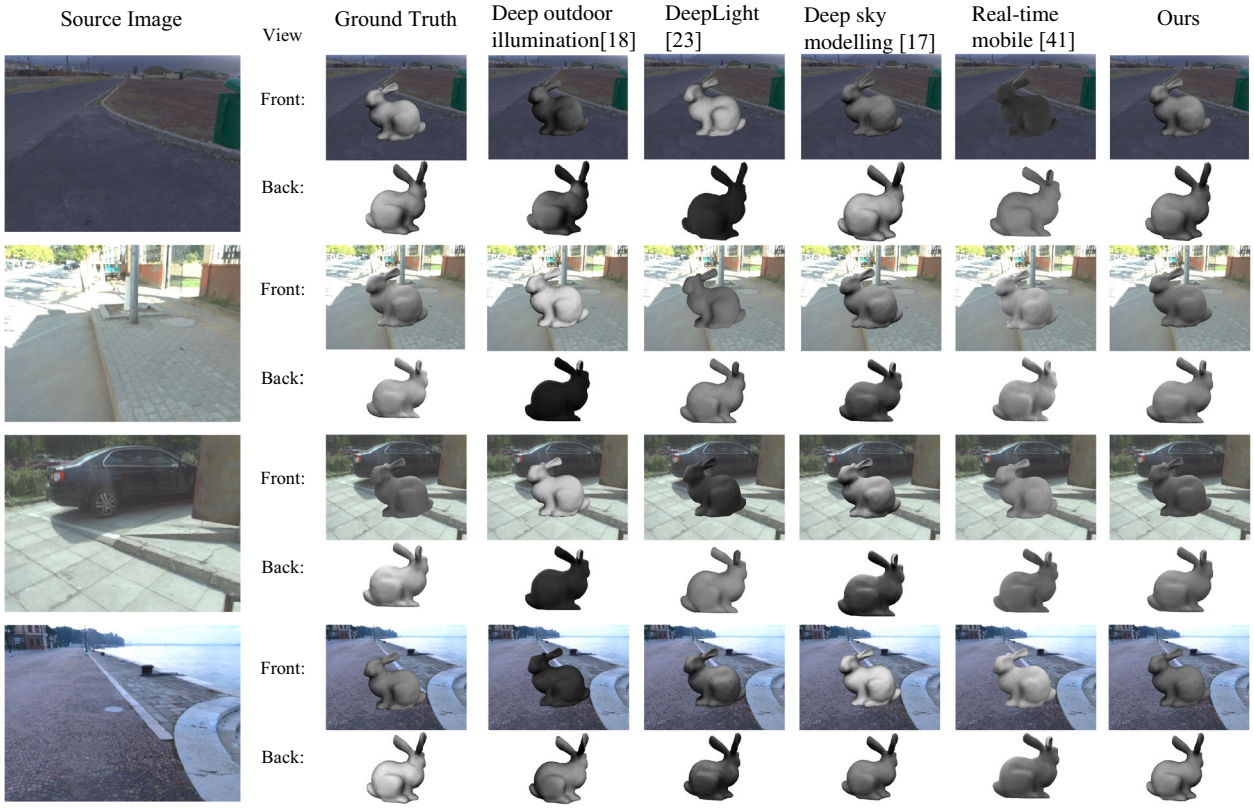

Fig. 9 Outdoor results. From left to right: ground truth, results from [17,18,23,41] and our results. Note that some models may look realistic from the frontal-view, but their back-views are quite different from the ground truth

Table 2 Quantitative comparison between the proposed algorithm with prior works in three lighting environments

	Indoor		Outdoor		Spatially varying	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
Deep outdoor illumination [18]	–	–	0.159	0.202	–	–
Indoor illumination [12]	0.142	0.179	–	–	–	–
DeepLight [23]	0.122	0.151	0.116	0.143	–	–
Deep sky modeling [17]	–	–	0.109	0.132	–	–
Real-time mobile [41]	0.103	0.122	0.102	0.121	–	–
Fast spatially varying [13]	–	–	–	–	0.072	0.089
Ours	0.066	0.081	0.061	0.076	0.056	0.070

Table 3 Effects of different losses. The proposed shading loss and smooth loss improve the relighting results

	Indoor		Outdoor		Spatially varying	
Loss terms	MAE	RMSE	MAE	RMSE	MAE	RMSE
Original	0.066	0.082	0.065	0.081	0.059	0.073
$L_{\text{shading}} + L_{\text{smooth}}$	0.066	0.081	0.061	0.076	0.056	0.070

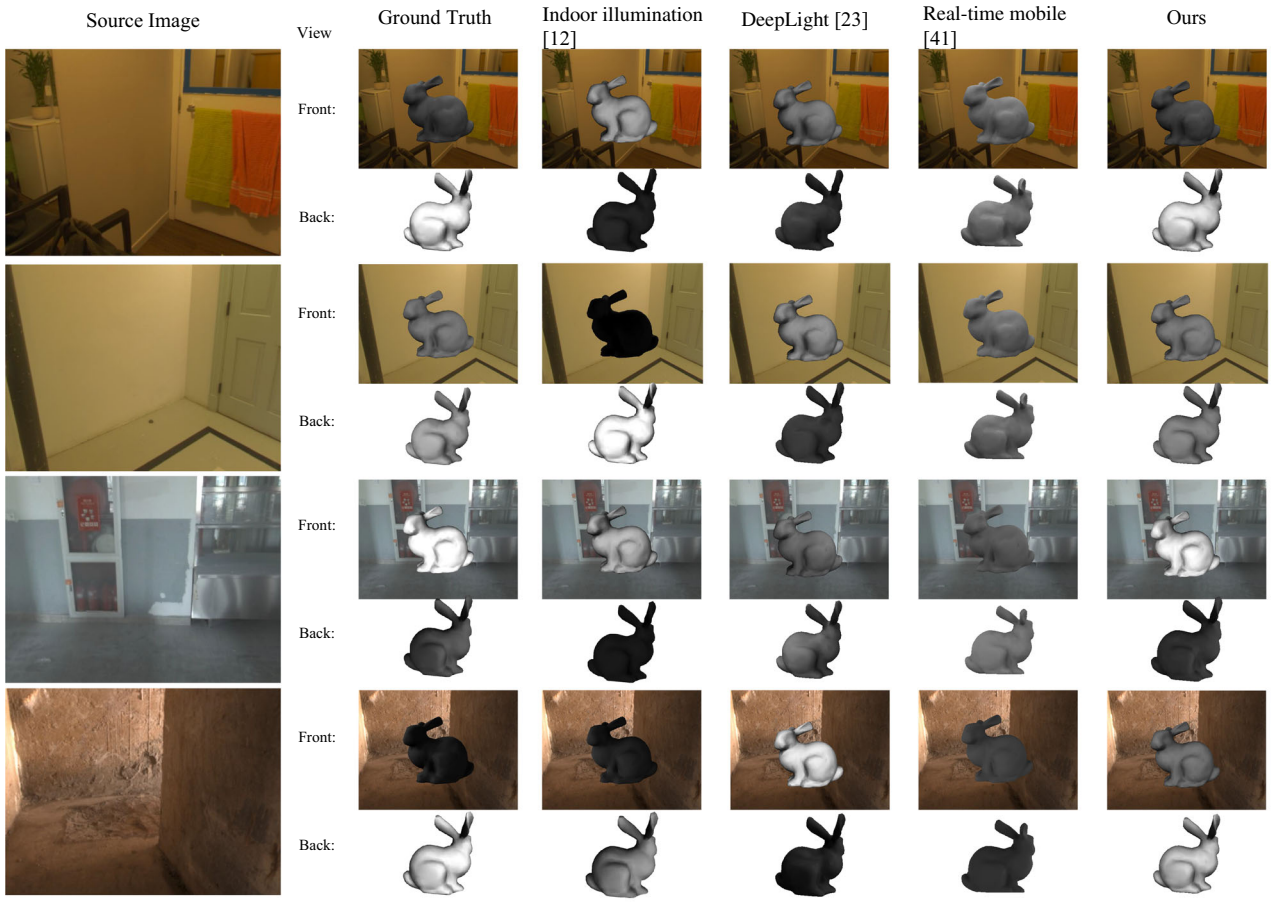

Fig. 10 Indoor results. From left to right: ground truth, results from [12,23,41] and our results

of rendered models are compared in all scenes. It is worth highlighting that this step is proven to be quite important. As some prior works may perform well on the front view, their back view may not be realistic compared to the ground truth.

According to four scenes taken at various outdoor environments, the qualitative comparisons between the proposed algorithm and prior works reported in [17,18,23,41] are shown in Fig. 9. It is observed that some prior works produce good qualitative results in either front view or back view. But very few obtains good results in both views. In general, the proposed algorithm is observed producing comparable qualitative results in both views, except for few cases. As observed from Fig. 9, outdoor methods based on sun and sky-model [17,18] are difficult to generate satisfactory results if the sun is not captured in the input images, due to the nature of their methods. In fact, there are many outdoor images captured without the sun or sky inside. The performance of methods using sun and sky-model may be affected in such situations.

There are another four scenes taken in indoor environments, which are used for qualitative comparisons between the proposed algorithm and prior works reported in [12,23,41] are shown in Fig. 10. It is observed that very few prior works produce good results in both views, while the proposed algorithm is able to obtain relatively better results in both views for most cases.

For the qualitative analysis in the spatially varying lighting conditions, three scenes are taken with three locations being selected in each scene for the experiments. The result comparisons are performed between the proposed algorithm and the prior work reported in [13], as shown in Fig. 11. It is observed that the proposed algorithm achieves comparable results. Similar with [13], the proposed algorithm also demonstrates spatially varying capability. At different locations in the same scene, the rendered model appears differently according to its relative position to the light sources. Although our improvement may not look so significant compared to the results of [13] in Fig. 11, the method of [13] is meant more for indoor scenes. While the proposed algorithm exhibits better robustness, as it works well not only indoor scenes, but also in outdoor and spatially varying scenes.

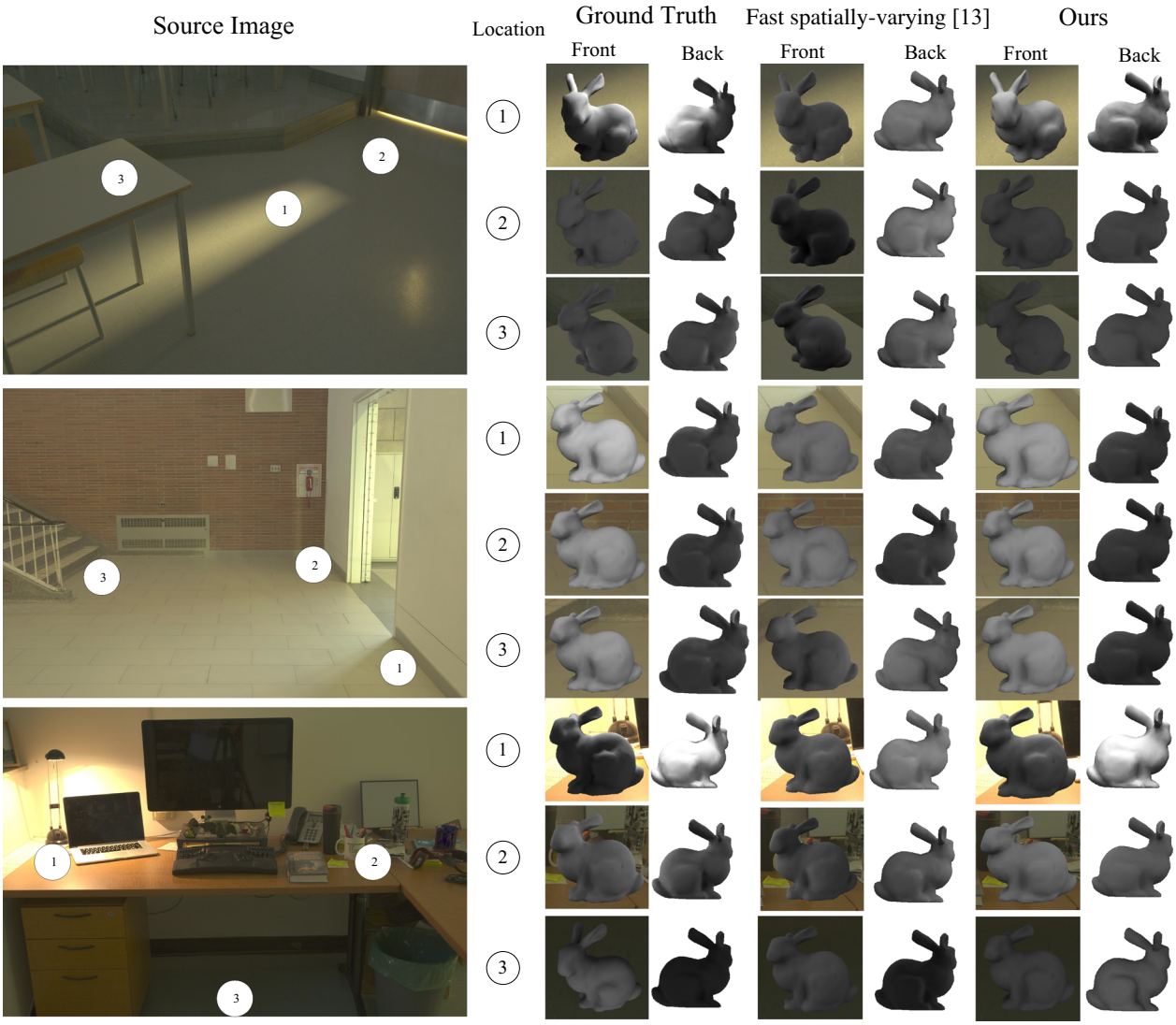

Fig. 11 Spatially varying results. The numbers indicate different rendering positions of the virtual object. From left to right: source image with position marks, ground truth, results from [13] and our results

It is observed in the qualitative comparisons, producing good results at both front view and back view is challenging. It is particularly obvious for Deeplight [23], which can be seen in both Figs. 9 and 10, as their back-view results look relatively dark compared to the ground truth or other methods. One possible reason would be caused by their training data, which are likely captured with mobile phone cameras from a very short distance to the light probe. In such a setup, if there is a strong light source, e.g., the sun, located just behind the light probe seeing from the camera view, the generated environment map would fail to record this light source. It may explain why the back views of their results are relatively dark. It shows that capturing HDR environment maps with light probes may cause potential issues under certain circumstances. It may be overlooked by previous works, and worth further exploring to avoid such potential issues.

With the proposed algorithm, the illumination transferring and rendering of various objects besides the bunny can be achieved, as shown in Fig. 12.

4.3 User study

We further conduct a user study to evaluate the realism of results. A total number of 170 unique computer science undergraduate students take part in the user study in this research. Seventeen scenes with rendered virtual objects are presented to all participants. These scenes mix with various environments including indoor, outdoor, and spatially varying ones. In each scene, there are four pairs of front view and back view images besides the ground truth. These four pairs of images are the predictions from the four approaches, i.e., our proposed algorithm and prior works [13,17,23]. These

Fig. 12 Examples of different virtual objects rendered into real scenes by our proposed algorithm

approaches are independent to each other and scored separately. In the user study procedure, participants are displayed with pairs of images (i.e., front view and back view) compared to the ground truth in every scene, and they are asked to choose the more realistic pair.

According to the choices made by participants, results are presented as percentages, denoting the fraction that each approach is preferred to the ground truth results (the higher the better). For spatially varying scenes, our approach scores 32.5%, compared to 28% of approach reported in [13]. For the indoor and outdoor scenes, our approach scores 48.8%, compared to 39.6% for [23] and 41.4% for [17], respectively. Overall, it is observed from the user study scores, participants have higher preferences to the predictions of our proposed algorithm.

4.4 Limitations

Since our framework is object-based using implicit illumination, each individual virtual object needs to be trained accordingly, which takes around four hours on our PC. However, it is arguable that virtual objects for most AR or MR applications are already installed or pre-defined. It means that the geometries can be known in advance, which means it is practical to make offline training beforehand. Moreover, unlike conventional methods, the proposed algorithm is able to reduce system complexity without needs of running a separate rendering pipeline.

The nature of original OI is designed for Lambertian surfaces only. As such, the training data we generated is only Lambertian. As a result, the implicit illumination features extracted in the proposed network only exhibits the diffuse lighting, that is a limitation. However, we would like to argue that it is mainly due to the training data, not the proposed framework itself. We will leave non-Lambertian training data generation for the future work.

The proposed network is implemented in Pytorch. The inference time is per image without optimization. The inference time of our illumination transferring and rendering network is below 30 ms, for bunny and all other object showed in this paper. The plane detection module, i.e., PlaneRCNN, takes about 400 ms in our case. We leave the network conversion (TensorRT) and optimization for the future work.

The proposed algorithm is not capable of simulating light inter-reflections between multiple virtual objects, if there are multiple virtual objects rendered into the same real scene. It can only simulate the estimated illumination features detected and transferred from spatially varying planar surfaces in real scenes.

In the qualitative comparison with prior works in Sect. 4.2, the illumination features detected from spatially varying planar surfaces in real scenes are simulated and compared. But, the shadow effects in the scenes are not simulated, since it is a separate pipeline as described in Sect. 3.5. This is another limitation of the work.

5 Conclusion

In this paper, a novel algorithm for virtual object illumination transferring and rendering is proposed. It does not need reconstructing the lighting of the entire real scene. The implicit illumination is directly transferred from existing planar surfaces to virtual objects within a unified framework, with plane detection, OI estimation, and the GAN based feature transferring algorithm. Extensive experiments have been conducted in various environments including indoor, outdoor, and spatially varying scenes. It is observed from the quantitative and qualitative results of the experiments that the proposed algorithm can accurately estimate and render the illumination of virtual objects in real scenes. The experiment results show realistic rendering at both front view and back view in these environments. It illustrates the effectiveness and robustness of the proposed algorithm compared to prior works reported in the literature which obtain good results only in either front view or back view. The proposed algorithm is able to render realistic virtual objects into scenes which will be very useful to applications of AR and MR.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China under Grant 61801391, in part by Open Project Program of the National Laboratory of Pattern Recog-

nition under Grant 202000025, in part by China Postdoctoral Science Foundation under Grant 2018M631193.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Azure spatial anchors. https://azure.microsoft.com/en-us/services/spatial-anchors/
2. Barron, J.T., Malik, J.: Intrinsic scene properties from a single RGB-D image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 17–24 (2013)
3. Bui, G., Le, T., Morago, B., Duan, Y.: Point-based rendering enhancement via deep learning. Vis. Comput. **34**, 829–841 (2018). https://doi.org/10.1007/s00371-018-1550-6
4. Calian, D.A., Lalonde, J.-F., Gotardo, P., Simon, T., Matthews, I., Mitchell, K.: From faces to outdoor light probes. Comput. Graph. Forum (2018). https://doi.org/10.1111/cgf.13341
5. Chauve, A.-L., Labatut, P., Pons, J.-P.: Robust piecewise-planar 3D reconstruction and completion from large-scale unstructured point data. In: 2010 IEEE computer society conference on computer vision and pattern recognition, pp. 1261–1268. IEEE (2010)
6. Cheng, D., Shi, J., Chen, Y., Deng, X., Zhang, X.: Learning scene illumination by pairwise photos from rear and front mobile cameras. Comput. Graph. Forum (2018). https://doi.org/10.1111/cgf.13561
7. Debevec, P.: A median cut algorithm for light probe sampling. In: ACM SIGGRAPH 2008 Classes, pp. 1–3 (2008)
8. Debevec, P.: Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In: ACM SIGGRAPH 2008 Classes, p. 32. ACM (2008)
9. Debevec, P., Graham, P., Busch, J., Bolas, M.: A single-shot light probe, pp. 10:1–10:1 (2012). https://doi.org/10.1145/2343045.2343058
10. Gao, Y., Hu, H.-M., Li, B., Guo, Q.: Naturalness preserved nonuniform illumination estimation for image enhancement based on retinex. IEEE Trans. Multimed. **20**(2), 335–344 (2017)
11. Gardner, M.-A., Hold-Geoffroy, Y., Sunkavalli, K., Gagné, C., Lalonde, J.-F.: Deep parametric indoor lighting estimation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 7175–7183 (2019)
12. Gardner, M.-A., Sunkavalli, K., Yumer, E., Shen, X., Gambaretto, E., Gagné, C., Lalonde, J.-F.: Learning to predict indoor illumination from a single image. ACM Trans Graph (SIGGRAPH Asia)
13. Garon, M., Sunkavalli, K., Hadap, S., Carr, N., Lalonde, J.-F.: Fast spatially-varying indoor lighting estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
14. Georgoulis, S., Rematas, K., Ritschel, T., Fritz, M., Tuytelaars, T., Gool, L. Van.: What is around the camera? In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5170–5178 (2017)
15. Gkitsas, V., Zioulis, N., Alvarez, F., Zarpalas, D., Daras, P.: Deep lighting environment map estimation from spherical panoramas. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 640–641 (2020)
16. Han, X., Yang, H., Xing, G., Liu, Y.: Asymmetric joint GANs for normalizing face illumination from a single image. IEEE Trans. Multimed. **22**(6), 1619–1633 (2019)
17. Hold-Geoffroy, Y., Athawale, A., Lalonde, J.-F.: Deep sky modeling for single image outdoor lighting estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6927–6935 (2019)
18. Hold-Geoffroy, Y., Sunkavalli, K., Hadap, S., Gambaretto, E., Lalonde, J.-F.: Deep outdoor illumination estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7312–7321 (2017)
19. Jacobs, K., Nielsen, A.H., Vesterbaek, J., Loscos, C.: Coherent radiance capture of scenes under changing illumination conditions for relighting applications. Vis. Comput. **26**, 171–185 (2010). https://doi.org/10.1007/s00371-009-0360-2
20. Johnson, M.K., Adelson, E.H.: Shape estimation in natural illumination. In: CVPR 2011, pp. 2553–2560. IEEE (2011)
21. Karsch, K., Hedau, V., Forsyth, D., Forsyth, D.:Hoiem. Rendering synthetic objects into legacy photographs. In: ACM Transactions on Graphics (TOG), vol. 30, p. 157. ACM (2011)
22. Kipf, T., Welling, M.: Semi-supervised classification with graph convolutional networks (2017)
23. LeGendre, C., Ma, W.-C., Fyffe, G., Flynn, J., Charbonnel, L., Busch, J., Debevec, P.: Deeplight: learning illumination for unconstrained mobile mixed reality. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5918–5928 (2019)
24. Liu, C., Kim, K., Gu, J., Furukawa, Y., Kautz, J.: Planercnn: 3D plane detection and reconstruction from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4450–4459 (2019)
25. Maier, R., Kim, K., Cremers, D., Kautz, J., Nießner, M.: Intrinsic3d: high-quality 3D reconstruction by joint appearance and geometry optimization with spatially-varying lighting. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3114–3122 (2017)
26. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z., Smolley,S.P.: Least squares generative adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2813–2821 (2017)
27. Metz, L., Poole, B., Pfau, D., Sohldickstein, J.: Unrolled generative adversarial networks. arXiv: Learning (2016)
28. Pei, S.-C., Shen, C.-T.: Color enhancement with adaptive illumination estimation for low-backlighted displays. IEEE Trans. Multimed. **19**(8), 1956–1961 (2017)
29. Reinhard, E., Heidrich, W., Debevec, P., Pattanaik, S., Ward, G., Myszkowski, K.: High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting. Morgan Kaufmann, Burlington (2010)
30. Ren, Z., Gai, W., Zhong, F., Pettré, J., Peng, Q.: Inserting virtual pedestrians into pedestrian groups video with behavior consistency. Vis. Comput. **29**, 927–936 (2013). https://doi.org/10.1007/s00371-013-0853-x

31. Song, S., Funkhouser, T.: Neural illumination: lighting prediction for indoor environments. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
32. Srinivasan, P.P., Mildenhall, B., Tancik, M., Barron, J.T., Tucker, R., Snavely, N.: Lighthouse: Predicting Lighting Volumes for Spatially-Coherent Illumination. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8080–8089 (2020)
33. Tarko, J., Tompkin, J., Richardt, C.: Omnimr: omnidirectional mixed reality with spatially-varying environment reflections from moving 360 video cameras. In: 2019 IEEE conference on virtual reality and 3D user interfaces (VR), pp. 1177–1178. IEEE (2019)
34. Tsai, G., Xu, C., Liu, J., Kuipers, B.: Real-time indoor scene understanding using bayesian filtering with motion cues. In: ICCV, pp. 121–128 (2011)
35. Weber, H., Prévost, D., Lalonde, J.-F.: Learning to estimate indoor lighting from 3D objects. In: 2018 International Conference on 3D Vision (3DV), pp. 199–207. IEEE (2018)
36. Wei, X., Chen, G., Dong, Y., Lin, S., Tong, X.: Object-based illumination estimation with rendering-aware neural networks. arXiv preprint arXiv:2008.02514 (2020)
37. Wu, C., Wilburn, B., Matsushita, Y., Theobalt, C.: High-quality shape from multi-view stereo and shading under general illumination. In: CVPR 2011, pp. 969–976. IEEE (2011)
38. Wu, C., Zollhöfer, M., Nießner, M., Stamminger, M., Izadi, S., Theobalt, C.: Real-time shading-based refinement for consumer depth cameras. ACM Trans. Graph. (ToG) **33**(6), 200 (2014)
39. Xu, D., Duan, Q., Zheng, J., Zhang, J., Cai, J., Cham, T.-J.: Recovering surface details under general unknown illumination using shading and coarse multi-view stereo. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1526–1533 (2014)
40. Xu, D., Duan, Q., Zheng, J., Zhang, J., Cai, J., Cham, T.-J.: Shading-based surface detail recovery under general unknown illumination. IEEE Trans. Pattern Anal. Mach. Intell. **40**(2), 423–436 (2018)
41. Xu, D., Li, Z., Zhang, Y.: Real-time illumination estimation for mixed reality on mobile devices. In: 2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), pp. 703–704. IEEE (2020)
42. Yi, R., Zhu, C., Tan, P., Lin, S.: Faces as lighting probes via unsupervised deep highlight extraction. In: The European Conference on Computer Vision (ECCV) (2018)
43. Zhang, J., Lalonde, J.-F.: Learning high dynamic range from outdoor panoramas. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4519–4528 (2017)
44. Zhang, J., Sunkavalli, K., Hold-Geoffroy, Y., Hadap, S., Eisenman, J., Lalonde, J.-F.: All-weather deep outdoor lighting estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
45. Zhu, M., Morin, G., Charvillat, V., Ooi, W.T.: Sprite tree: an efficient image-based representation for networked virtual environments. Vis. Comput. **33**, 1385–1402 (2017). https://doi.org/10.1007/s00371-016-1286-0

Di Xu is the R&D Director of Shadow Creator Inc., Beijing, China. He received his B.S. degree in electrical and electronic engineering in 2011 and the Ph.D. degree in computer engineering in 2016, both from Nanyang Technological University, Singapore. From 2017 to 2020, he has been an Assistant Professor with the School of Computer Science, Northwestern Polytechnical University, China. His research interests include 3D vision, neural rendering, and mixed reality.

Zhen Li received his B.S. degree from Northwestern Polytechnical University, China, in 2018. He is currently a masters student in Northwestern Polytechnical University. His research interests include computer vision, computer graphics, and augmented reality.

Qi Cao is an Assistant Professor with the School of Computing Science, University of Glasgow, Singapore campus. He obtained his Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2007. He obtained his Bachelor of Engineering degree from HuaZhong University of Science & Technology (HUST), Wuhan, China, in 2000. His research interests include virtual reality (VR) and augmented reality (AR), signal processing, data analytics, and computational intelligence.