IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 2, 2020, accepted October 22, 2020, date of publication November 2, 2020, date of current version November 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3035213

SIR-Net: Self-Supervised Transfer for Inverse Rendering via Deep Feature Fusion and Transformation From a Single Image

TIANTENG BI[1], JUNJIE MA[2], (Member, IEEE), YUE LIU[1,3], (Member, IEEE), DONGDONG WENG[1,3], AND YONGTIAN WANG[1,3], (Member, IEEE)

[1]Beijing Engineering Research Center of Mixed Reality and Advanced Display, School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China
[2]Department of Computer Science and Technology, Tsinghua University, Beijing National Research Center for Information Science and Technology (BNRist), Beijing 100084, China
[3]AICFVE of Beijing Film Academy, Beijing 100088, China

Corresponding author: Yue Liu (liuyue@bit.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61731003 and Grant 61960206007, in part by the Key-Area Research and Development Program of Guangdong Province under Grant 2019B010149001, and in part by the 111 Project under Grant B18005.

ABSTRACT Measuring the material, geometry, and ambient lighting of surfaces is a key technology in the object's appearance reconstruction. In this article, we propose a novel deep learning-based method to extract such information to reconstruct the object's appearance from an RGB image. Firstly, we design new deep convolutional neural network architectures to improve the performance by fusing complementary features from hierarchical layers and different tasks. Then we generate a synthetic dataset to train the proposed model to tackle the problem of the absence of the ground-truth. To transfer the domain from the synthetic data to the specific real image, we introduce a self-supervised test-time training strategy to finetune the trained model. The proposed architecture only requires one image as input when inferring the material, geometry, and ambient lighting. The experiments are conducted to evaluate the proposed method on both the synthetic data and real data. The results show that our trained model outperforms the existing baselines in each task and presents obvious improvement in final appearance reconstruction, which verifies the effectiveness of the proposed methods.

INDEX TERMS Inverse rendering, attention, feature fusion, lighting recovery, material estimation, deep learning, image-based rendering.

I. INTRODUCTION

An object's appearance is determined by the ambient light, geometry, and material of its surfaces. Acquiring them is very important for the appearance reconstruction that has wide applications in such fields as mixed reality, robotics, and artistic creation, which generally requires complex optical devices to conduct dense measurements of the target object [1]. However, usually, such devices can only be used for a certain class of objects and the measurements need an extremely strict experiment environment, which is a costly effort and merely used under limited scenarios.

A more general and efficient method is to infer these properties from the objects' images [2], namely the inverse rendering. However, it is a highly ill-posed problem as quite a few combinations among the inferred factors may lead to the same observed image. How to address this issue is the key to acquire these properties from images.

In this article, we extract these properties from an object's image using Deep Convolutional Neural Networks (DCNNs). The object has variable materials and complex shapes. The image is captured under natural illumination. We present an end-to-end learning architecture with feature fusion and transformation to regress material, geometry, and ambient light from a single image of an object. Our design is composed of the geometry material regression module and illumination inference module.

Training such modules requires large amounts of labeled data that are extremely difficult to acquire in practice. Therefore, we generate a synthetic training corpus consisting

The associate editor coordinating the review of this manuscript and approving it for publication was Xian Sun.

of rendered object images, their corresponding ground truth.

However, the model trained with synthetic data cannot be directly applied to real image data and requires a domain adaption procedure. Hence, a self-supervised test-time training strategy is introduced to transfer the domain.

In summary, we make the following three main contributions:

- We propose an attention-based feature fusion unit to improve the accuracy of the geometry and material prediction. This unit is composed of existing DCNN components and ensures the end-to-end training.
- We design a feature transformation module to introduce skip connections into the basic encoder-decoder architecture when inferring the ambient light from the object's image. The feature transformation block converts the learning from the object's low-level space to the semantic illumination space and enables the skip connection between these levels.
- We also introduce a self-supervised test-time training strategy to finish the domain adaptation and the appearance reconstruction is dramatically improved by this method. This strategy finetunes the model by comparing the output with the corresponding input without the ground-truth.

The remaining parts of this article are organized as follows: in section 2, we introduce the related works. In section 3, the overview of our proposed system is presented. In section 4, we describe the designed architectures in detail. In section 5, the self-supervised test-training strategy is explained. In section 6, the evaluation and analysis of our methods are presented. In section 7, we make a summary and conclusion.

II. RELATED WORK

Inverse rendering is a fundamental problem in computer vision and graphics. Certain existing methods solve such a challenging problem by constructing a set of priors over the constituted properties or assuming one of such the components as the object's geometry and the lighting condition being known, and then iteratively optimize a hand-crafted mathematical model to find the solutions [3]–[5]. Geometry is relatively simple to capture as a result of the availability of depth sensors [6]. The illumination also can be approximated by inserting a light probe into the scene [7], [8] or low-dimensional representation based lighting model [9]. The lighting condition also could become a controllable factor such as image sequences under static illumination or varying illumination [10], [11]. Besides, the user input can also be used as an additional prior to guide the intrinsic decomposition [12]. However, these approaches need either the depth sensor or image sequences from different illuminations and viewpoints as well as inputs from the user, and cannot solve the problem from a single image automatically.

Compared with the above-mentioned methods, recently DCNNs make great progress in this task. Tang *et al.* propose a deep lambertian model to predict diffuse material, point light direction, and orientation map from a single image with Gaussian Restricted Boltzmann Machines [13]. Georgoulis *et al.* predict the normal maps and reflectance maps from a single image with the designed DCNNs [14]. They propose a direct architecture and an indirect architecture respectively to regress the reflectance maps. Both architectures are based on encoder-decoder strategies developed to regress dense label maps [15], [16]. Particularly, in the indirect way, they stack two networks to infer the reflectance map by using the output of the first network as the input of the followed one. Liu *et al.* design three separate DCNNs to predict the material parameters, normal maps as well as environment maps from a single image, which is followed by a differentiable rendering layer [17]. In these works, they either use the stacked way or consider the task separately. All of them neglect that fusing intermediate related features would benefit the performance of all tasks, which have been verified by various tasks [18]–[20]. Besides, these methods tackle this problem under the assumption that the objects are covered by only one kind of material. However, it should be noticed that many objects in the real world are covered by several different materials. Moreover, these methods use synthetic data to train their model, which makes the model trained suffers from a domain gap.

Besides, the differentiable renderer is often embedded in the DCNNs to construct the perceptual reconstruction loss between the input and the output as an additional constraint to generate a result of higher quality in the view of perception for inverse rendering [21], [22]. Such a reconstruction loss can not only enable the learning from a mixture of labeled synthetic data and unlabeled real image data and approaching the physical image generation model [23] but also, more importantly, make the unsupervised learning more effective with Siamese architecture from a single images or images sequences [24], [25].

III. OVERVIEW OF THE PROPOSED SYSTEM

In our study, the DCNNs infer the material, geometry, and ambient light simultaneously from a 2D image of an object. Such inverse rendering is the inversion of the real world image formation which can be formulated as:

$$(\mathbf{M}, \mathbf{N}, \mathbf{L}) = \mathcal{F}^{-1}(\mathcal{I}) \tag{1}$$

where the function $\mathcal{F}^{-1}$ represents the inverse rendering that obtains the object's material $\mathbf{M}$, geometry $\mathbf{N}$ and the ambient light $\mathbf{L}$ from its image $\mathcal{I}$ without any prior.

In order to train a DCNN to learn these properties from an image, the material, geometry, and illumination are represented as the learnable forms.

A. MATERIAL REPRESENTATION

In this work, we focus on opaque objects without considering transmitted and scattered light, so the material can be represented by the reflectance properties that can be fully formulated as the bidirectional reflection distribution

function (BRDF). BRDF directly describes how the incident light is reflected off the surface rather than limits the appearance of a fixed material under a fixed illumination and it can be defined as:

$$f(x, \omega_{\mathbf{i}}, \omega_{\mathbf{o}}) = \frac{d\mathcal{L}_o(x, \omega_{\mathbf{o}})}{dE(x, \omega_{\mathbf{i}})} \tag{2}$$

which is the ratio of the radiance $\mathcal{L}_o$ leaving the surface at point x in direction $\omega_{\mathbf{o}}$ and irradiance $\mathcal{E}_i$ arriving at x from direction $\omega_{\mathbf{i}}$.

Non-parametric models adopt a lookup table to store the reflectance information with high accuracy, but they are generally more computationally intensitive [1]. Besides, they are not differentiable as to support the back-propagation in deep learning. Consequently, in this article, we use the directional statistics BRDF (DSBRDF) to physically describe the reflectance in the real world [26]. It has a small set of parameters and an analytic expression to model a wide range of real-world isotropic BRDF accurately. Compared with the existing micro-facets based models, DSBRDF achieves higher accuracy without the linear combination of different parametric models [27], [28]. Specifically, the DSBRDF model is composed of a set of hemispherical exponential power distributions known as lobes that enable encoding a variety of the BRDFs. In a typical setting, the number of the lobes is 3 and there are 108 coefficients in total in such a model. Because we extend the scenario to an extreme condition in which each pixel can own its material, the corresponding material tensor $\mathbf{M}$ has a dimension of $height \times width \times 108$.

B. GEOMETRY REPRESENTATION

We adopt the normal map to represent the geometry information $\mathbf{N}$ of the object. Specifically, $\mathbf{N}$ is a $height \times width \times 3$ map of which each channel stores the x, y, and z coordinate respectively of the point on the object. During the training and testing stage, the normal map is normalized to $[-1, 1]$.

C. ILLUMINATION REPRESENTATION

In this article, the widely used high dynamic range (HDR) environment map is considered as the illumination representation [17], [29]. We assume distant lighting and the absence of self-shadowing, and there is no emitted or reflected light from the object to the environment. Each pixel in the environment map can be transferred to spherical coordinates and thus represents an incoming light direction. The value of each pixel represents the intensity of the light from this direction. Note that we only regress the light intensity instead of the direction of light sources. Since the object is not a light source, i.e., it is not emitting light, we only model global changes in scene brightness. The interreflections accounting for the energy exchange within the object itself are typically ignored.

Finally, a differentiable renderer is inserted after the DCNNs to accept the material tensor, normal map, and environment map for the self-supervised test-time training strategy. The system diagram is shown in Fig. 1.

IV. DESIGNED NETWORK ARCHITECTURE

Under the above assumption, we design an end-to-end learning architecture to regress material tensor, normal map, and environment map from a single image of an object. Neither empirically considering separate network architectures [17] nor heuristically sharing the layers from different networks [23], we propose a fusion unit and a transformation module that can automatically learn how to fuse and how to transfer the feature. Our whole system is composed of the geometry material regression part and the illumination inference part.

A. INTERMEDIATE FEATURE FUSION BETWEEN MATERIAL AND GEOMETRY

Our motivation is to fuse the features from the geometry network and material network to improve respective accuracy [30], which can be implemented by stages of feature fusion and recalibration. The feature fusion takes the learned feature maps from layers of two networks and then fuses them to learn a global feature over these two tasks. To improve the performance, the new feature should keep the original characteristics of the task while absorbing the complementary information from the other task. Consequently, after learning the global feature, the global feature needs to be further recalibrated according to the specified task.

We denote the normal feature in the j^{th} block of the normal network as $n^{(j)}$, and the material feature in the j^{th} block of the material network as $m^{(j)}$. Then the fused global feature $g^{(j)}$ of this block is:

$$g^{(j)} = u^{(j)}([n^{(j)}; m^{(j)}]) \tag{3}$$

where $u^{(j)}$ represents the fusion operation and $[n^{(j)}; m^{(j)}]$ is a concatenation of the normal feature and material feature. Here, $u^{(j)}$ is convolutional layers with batch normalization, followed by a non-linear ReLU activation, which is composed of $[1 \times 1]$ kernels representing the feature fusion in the j^{th} block.

The purpose of recalibrating the global feature is to screen the task-related information. To keep the spatial correspondence between the geometry feature and material feature, we introduce squeeze and excitation (SE) mechanism [31] to implement channel-wise attention to refine the global feature to generate the new normal features $\hat{n}^{(j)}$ and the material features $\hat{m}^{(j)}$:

$$\hat{n}^{(j)} = g^{(j)} \odot a_n^{(j)} \tag{4}$$

$$\hat{m}^{(j)} = g^{(j)} \odot a_m^{(j)} \tag{5}$$

where $a^{(j)}$ is the learned attention weights of different tasks and $\odot$ means element-wise multiplication.

Channel attention can automatically learn to recalibrate channel-wise feature responses, which models the interdependencies between tasks. Here the SE is implemented by a global pooling layer and two convolutional units. The convolution is implemented by $[1 \times 1]$ kernels and followed by a sigmoid activation function that limits the learned attention maps $a^{(j)} \in [0, 1]$, which ensures that the performance will

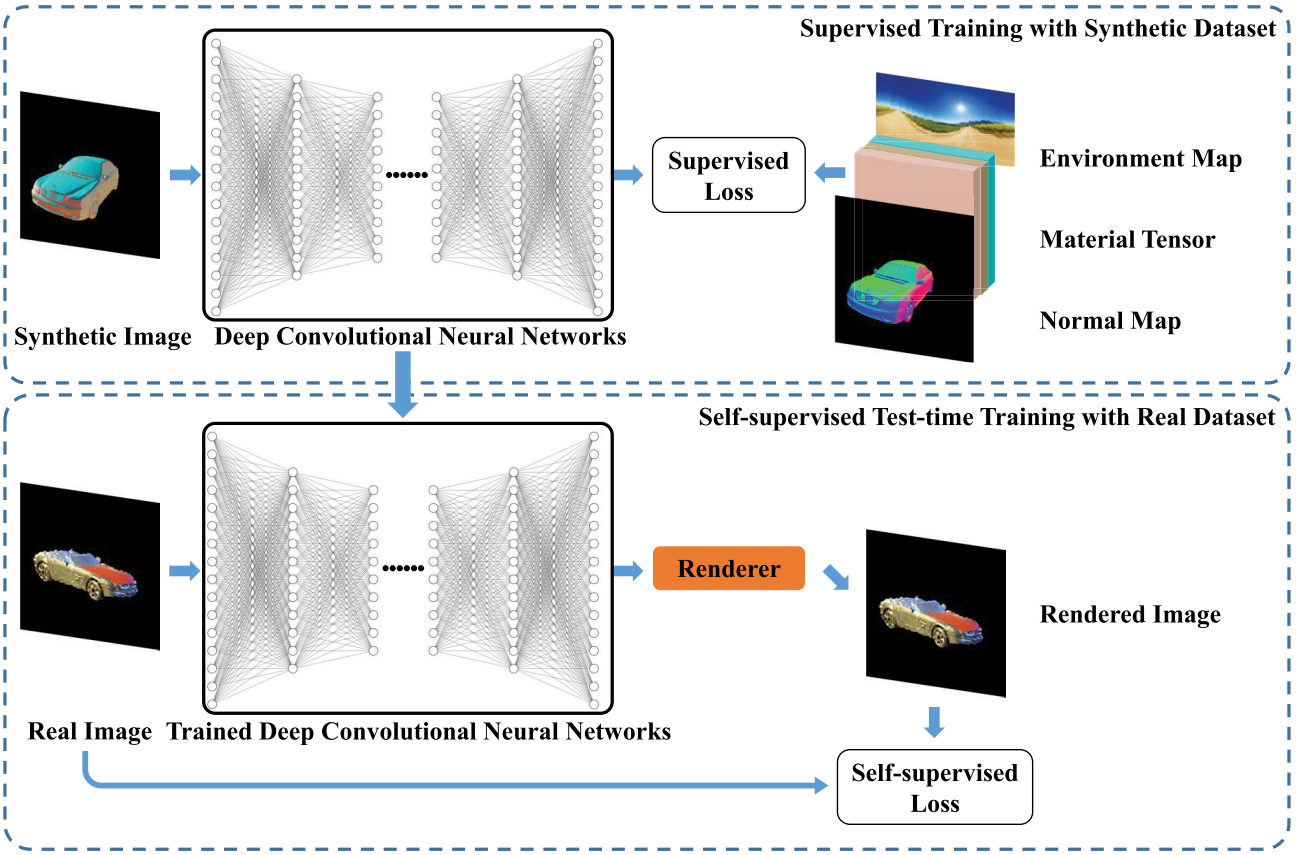

FIGURE 1. Our system takes an RGB image as input and predicts its normal, material, and environment map. Combined with the differentiable renderer, self-supervised learning can be used to finetune the model to transfer the domain and improve the performance over a specific target.

not be worse. The details of this feature fusion operation is shown in Figure. 2

Finally, we adopt two U-nets as the backbone [32] to construct the overall framework for geometry and material regression and insert this unit after the down-sampling group as shown in Fig. 3.

B. FEATURE TRANSFORMATION FOR ILLUMINATION INFERENCE

When inferring the illumination, Gardner *et al.* uses an end-to-end network to regress the panoramic illumination images [33]. Weber *et al.* propose a "T-network" to estimate the panoramic map from a 2D image of 3D objects given the geometry information [29]. These works simply try to learn the latent feature vectors from the images and convert them to the desired lighting information without considering the fusion of the features from hierarchical layers, which ignores the fact that pixel-wise accuracy can be improved by sufficient fusing low-level spatial information and high-level semantic information [34], [35].

Therefore, our method takes the feature fusion and transformation into consideration to improve the regression performance, while avoiding the mismatch from the different spatial scales between the 2D image input and the panoramic output. Specifically, in our system, the illumination module predicts

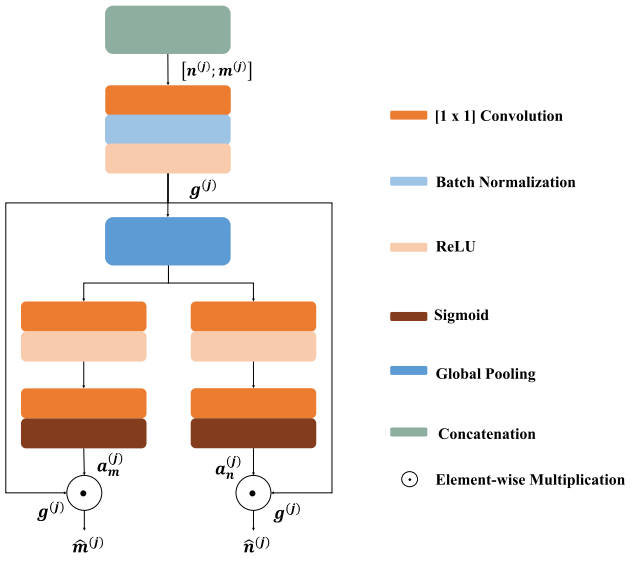

FIGURE 2. Details of our feature fusion unit.

a panoramic image from an object's image, which infers the light intensity from the image pixel while recovering the panoramic scene structure.

To introduce multi-scale feature fusion into this problem, a feature transformation block is proposed to transfer the

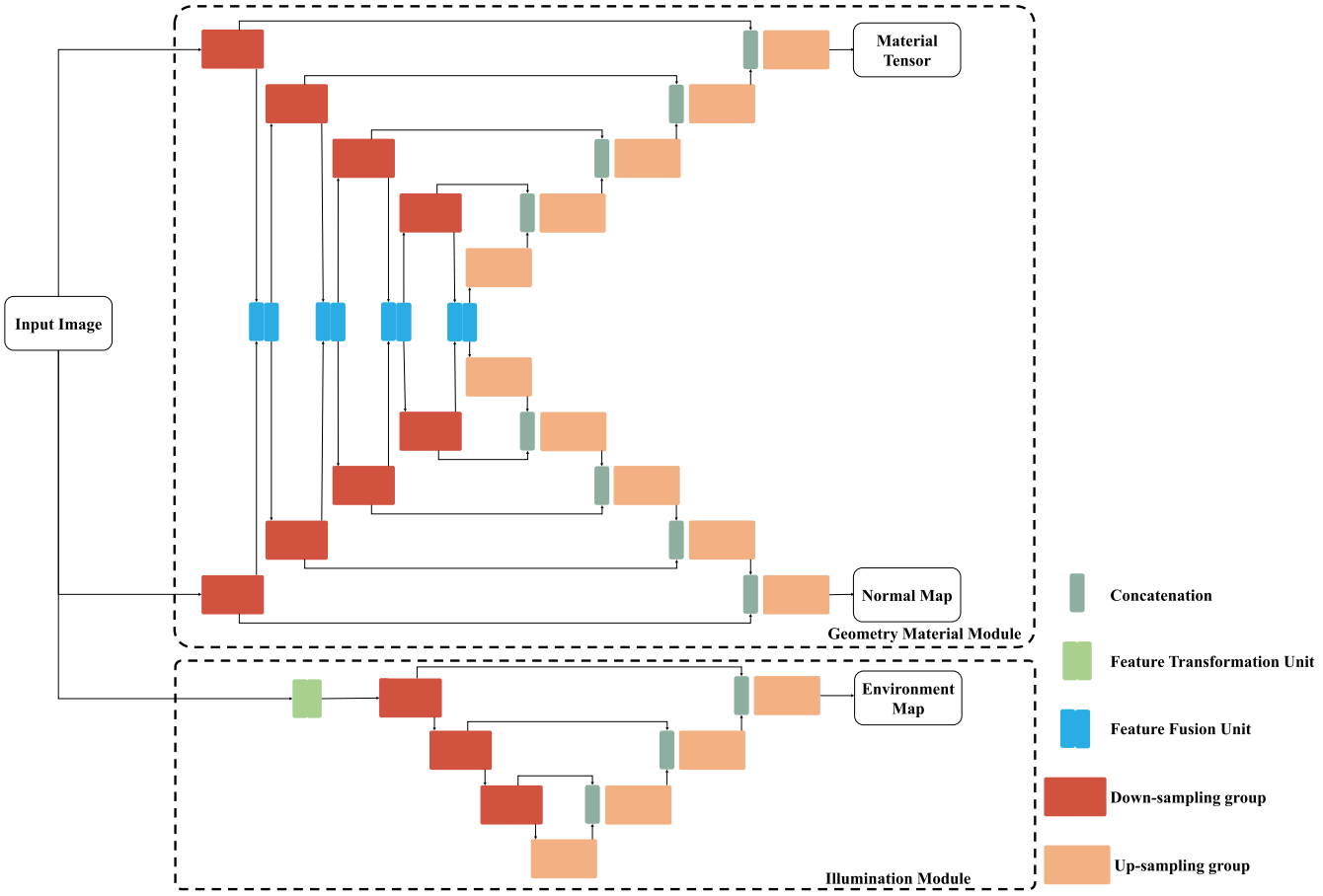

FIGURE 3. Our feature fusion unit fuses the features from two networks to learn a global feature and then recalibrate it according to the task-specific property. The illumination module is composed of a feature transformation unit and a backbone network. The transformation unit learns the lighting information from low-level features and then warps the feature scales.

learned low-level information from the object's image to its ambient lighting space. Such a transformation not only learns to transfer the feature space but also warps the feature map to the available scale for the subsequent fusion, which converts the learning from the object's low-level feature space to the illumination space and bridges the skip connection between the different levels.

The illumination module is composed of the transformation block and the backbone which takes as input the output of the transformation block. Furthermore, the transformation block can be simply implemented with two convolution units in our architecture. Fig. 3 shows the details of the architecture.

C. LOSS FUNCTIONS

The geometry material module can be evaluated by L1 or L2 loss. To balance the material loss and geometry loss, the L1 loss is used for material:

$$l_{material} = \sum_{x} |\mathbf{m}_x - \mathbf{m}'_x| \tag{6}$$

where $\mathbf{m}'_x$ is the predicted material and $\mathbf{m}_x$ is the material ground-truth at point x on the object.

L2 loss is used for normal:

$$l_{normal} = \sum_{x} (\mathbf{n}_x - \mathbf{n}'_x)^2 \tag{7}$$

where $\mathbf{n}'_x$ represents the predicted normal and $\mathbf{N}_x$ is the normal ground-truth at point x on the object.

The total loss function for the material geometry module is:

$$l = \lambda_1 l_{material} + \lambda_2 l_{normal} \tag{8}$$

where λ_1 and λ_2 are weights of the two losses. Empirically, we set $\lambda_1 = 1, \lambda_2 = 10000$.

Because the environment maps are in the HDR domain, we use the *log* to compress the potentially very high dynamic range of lighting to alleviate fluctuations from the errors. The Euclidean distance is defined in the *log* domain to constrain the illumination module:

$$l_{illumination} = \sum_{i} (log(1 + L_i) - log(1 + L'_i))^2 \tag{9}$$

where L_i and L'_i are the ground-truth and inferred environment maps of the ith pixel in the environment map respectively.

V. SELF-SUPERVISED TEST-TIME TRAINING STRATEGY

Learning the material, normal and environment maps from images requires amounts of labeled data in a supervised way. However, it is extremely difficult or even impossible in practice to acquire these data. For example, to obtain the object's reflection properties, it generally requires complex optical devices to conduct a great dense measurement represented by a set of BRDF samples at a specified incident and outgoing directions of the target object [1]. Consequently, synthetic data is widely adopted to address this problem, which can provide available labels to supervise the training procedure. However, the synthetic data is not the same as the real one, which is not obtained by the real image formation process. So the model trained by synthetic data cannot be directly performed on real data and the model needs further finetuning.

It can be seen from certain some previous self-supervised learning works [36]–[39] that a self-supervised task could be solved by test-time training. In our system, a differentiable renderer can be inserted after the geometry material and the illumination modules to implement the self-supervised learning while combining with the test-time training to transfer the domain. Specifically, the geometry material and the illumination modules first infer the material tensor, normal map, and environment map from an input image, and the renderer accepts such predicted elements to generate an input-like image. We then define a pixel-wise Euclidean distance over the input image and the generated one as an original constraint. Afterward, we finetune the model on this input image until the loss is small enough. This test-time training procedure can transfer the model trained on synthetic data to the real test image and avoid an additional post-optimization.

A. PHYSICALLY BASED RENDERER

Our renderer obeys the physical image formation theory [40], which can be formulated as:

$$\begin{aligned}\mathcal{L}_o(x,\omega_{\mathbf{o}}) &= \mathcal{L}_e(x,\omega_{\mathbf{o}}) + \mathcal{L}_r(x,\omega_{\mathbf{o}}) \\ &= \mathcal{L}_e(x,\omega_{\mathbf{o}}) + \int_{\Omega} f(x,\omega_{\mathbf{i}},\omega_{\mathbf{o}})\mathcal{L}_i(x,\omega_{\mathbf{i}})(\omega_{\mathbf{i}},\mathbf{n}_x)d\omega_{\mathbf{i}}\end{aligned} \tag{10}$$

where $\mathcal{L}_o$ expresses the radiance leaving the point at x with normal $\mathbf{n}$ in direction $\omega_{\mathbf{o}}$ as the emitted $\mathcal{L}_e$ and reflected radiance $\mathcal{L}_r$ which is a function of incoming light $\mathcal{L}_i$ over the hemisphere from direction $\omega_{\mathbf{i}}$. f represents the reflectance describing the material.

The rendering equation in the discrete domain can be formulated as:

$$\mathcal{I}(x,\omega_{\mathbf{o}}) = \sum_{i=1}^{m} f(x,\omega_{\mathbf{i}},\omega_{\mathbf{o}})L_i \max(0,\omega_i\cdot\mathbf{n}_x)w_i \tag{11}$$

where $\mathcal{I}$ is the pixel value in the image and m is the number of pixels and L_i represents the intensity of the i*th* pixel in the environment map. The weights w_i describe the contribution of the pixels at the different longitude of the spherical coordinates, which can be computed by:

$$w_i = \sin\frac{\lfloor i/W_{L_i}\rfloor}{H_{L_i}} \tag{12}$$

where W_{L_i} and H_{L_i} represent the width and height of the environment map respectively.

B. DYNAMIC RANGE-DEPENDED SELF-SUPERVISED LOSS FUNCTIONS

The differentiable renderer as a layer of the neural network accepts the material tensor, the normal map, and the environment map to generate an image, then the errors can be back-propagated through it. In our framework, the physically-based rendering layer computes how much light is reflected by the object's surface. And the pixel on the surface contains the physical lighting intensity. When defining the loss function between the output image and the input one for self-supervised training, both input cases should be considered: i.e. low dynamic range (LDR) input and HDR input.

When the input is an HDR image, both the input and output include rich physical information. So we can directly define the image reconstruction loss using the output of the rendering layer as the self-supervised loss function in *log* domain:

$$l_{rec}^{HDR} = \sum_x (\ln(\beta+\alpha*\mathcal{F}(\mathbf{m}'_x,\mathbf{n}'_x,\mathbf{L}')) - \ln(\beta+\alpha*\mathcal{I}_x^{input}))^2 \tag{13}$$

where, on a specific point x, $\mathbf{m}'_x, \mathbf{n}'_x, \mathbf{L}'$ are the predicted material tensor, normal map and environment map, $\mathcal{I}^{input}$ is the input image. We set $\beta = 1$ to ensure the pixel value not to be negative in *log* domain and $\alpha = 1$ to benefit the learning by not over-scaling the data.

For the more general LDR input, it is transformed from the HDR version by many operations, for example, gamma correction, which loses its original physical property. Using such input to constrain the self-supervised learning could force the network system to learn an additional inverse mapping from LDR to HDR. Consequently, to keep the learning stable and converging, the output HDR image should be normalized to the corresponding range. Here, the linear normalization is used to compress the range while keeping the physical property:

$$l_{rec}^{LDR} = \sum_x (\lambda\mathcal{F}(\mathbf{m}'_x,\mathbf{n}'_x,\mathbf{L}') - \mathcal{I}_x^{input})^2 \tag{14}$$

where λ represents a channel-wise normalization factor to scale the rendered image. It can be written as:

$$\lambda^k = \frac{255*\mathcal{F}^k(\mathbf{M}^k,\mathbf{N},\mathbf{L}^k)}{\max\mathcal{F}^k(\mathbf{M}^k,\mathbf{N},\mathbf{L}^k)} \tag{15}$$

where $k = 1, 2, 3$ means the color channels.

VI. EXPERIMENTS AND DISCUSSIONS

We implement the whole system on Caffe with a GTX 2080Ti graphics card from scratch [41]. The back-propagated gradients are obtained by AdaDelta with delta parameter δ = 1e-6

and momentum parameter $\beta = 0.9$ [42]. We set the original learning rate according to the specified tasks.

A. DATASET

To evaluate our model, we synthesize a dataset containing complicated shapes covered by multi-materials under different illuminations. Specifically, we use the normal maps in [14] and split the data into training and testing sets according to the original definition. The size of the normal map is $256 \times 256 \times 3$ which leads to the synthetic image with the same dimension. 60 DSBRDFs fitted from the MERL dataset [4] are used for training and 20 are used for testing. To cover different materials over the objects, we use k-means to cluster the normal maps for 3 categories. For each category, we randomly choose one material in the DSBRDF dataset. The illumination is composed of 100 free HDR outdoor environment maps of $512 \times 1024 \times 3$ downloaded from the Internet. The high resolution of the environment maps can generate better rendering results, but it also reduces the training efficiency due to the traversing the environment map during the rendering. Empirically, we scale the downloaded environment map to the size of $64 \times 128 \times 3$ to make a trade-off between efficiency and effectiveness. We define a split to classify the environment maps into 80 training samples and 20 testing samples according to their scenes. Moreover, random rotation is adopted to augment the lighting data. Finally, a total of 50,440 training images and 9,930 testing images is generated and no materials, normal maps, and environment maps are shared between them. The visualized examples from our synthetic dataset are shown in Fig. 4.

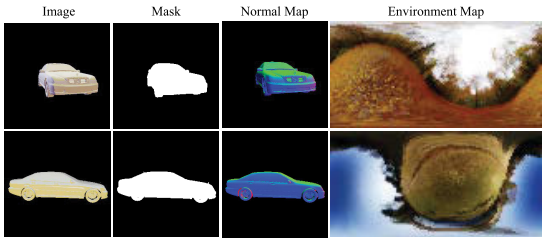

FIGURE 4. Two examples from our synthetic dataset (from left to right): image $\mathcal{I}$, segmentation mask, normal map **N**, and environment map **L**.

B. EVALUATION FOR THE INTERMEDIATE FEATURE FUSION

We evaluate our proposed feature fusion method on the synthetic data and compare it with the baseline approach as well as several previous works, which are briefly discussed below.

U-net. Our first baseline architecture is the U-net architecture, which is widely used in the image-to-image transformation task. Its skip connections allow sharing the different level's features in the encoder with corresponding layers in the decoder. Specifically, we design 4 down-sampling and up-sampling groups. Each down-sampling group is composed of 2 convolution units including a convolution layer and a ReLU activation layer, followed by a MAX pooling layer.

TABLE 1. Comparison between our approach and the two related baseline methods for geometry material model.

	Normal		Material	Rendering	
	Mean	Median	MSE	SSIM	MSE
U-net	8.1723	5.2167	1.0750	0.9604	129.1860
Liu's net	16.3332	11.4996	1.9041	0.8931	2542.9
SETD-net	8.2804	4.8407	0.9242	0.9588	123.9954
Ours	**7.2476**	**4.2020**	**0.8864**	**0.9623**	**119.1847**

The up-sampling group contains 2 convolution units and an up-sampling unit including a deconvolution layer and a ReLU layer, which also concatenates the features from the previous layers by the skip connection. It is trained with the synthetic ground truth from scratch.

Liu's net. The second baseline is inspired by the work of Liu *et al.* [17] which predicts normal, material, and illumination with three individual networks from a single image. However, baseline 2 can only address the object covered by one material, which means that their material network maps the image to a feature vector representation rather than generates our material tensors. Consequently, we modify their material network to adapt to our scenario by adding some up-sampling groups. Furthermore, to make it comparable with ours, we train their networks with our synthetic dataset from scratch.

SETD-net. The heuristic shared model is often used in inverse rendering paper [25]. This model generally adopts a single encoder to learn shared information and then two decoders convert the shared feature to different semantic targets. In our evaluation, a single encoder and two decoders (SETD) is constructed to infer material and normal. We use U-net architecture as the backbone. The encoder extracts shared information of material and normal as well as the two decoder output material tensor and normal map respectively. The skip connections between the encoder and decoders fuse the high-level and low-level features.

For comparison, we train our network with the same synthetic dataset. The quantitative and visual results are provided in Table 1 and Fig. 5. For the quantitative results of normals, we compute the mean and median error over the angular error between the predicted normals and ground-truth. The mean square error (MSE) is used to evaluate the accuracy of the materials. We rendered the images with the inferred material to present the visual difference from the material error, given the ground-truth normal and environment map. Then we compute the MSE and the structural similarity (SSIM) [43] metrics between the rendered image and the ground-truth to show the visual error quantitatively. SSIM is a widely used metric to evaluate the similarity between two images, whose value is between 0 and 1. The higher the value of SSIM, the more similar the two images.

It can be seen from the experimental results that the material of Liu's net fails to predict the correct material, which verifies that a low dimensional feature vector cannot contain

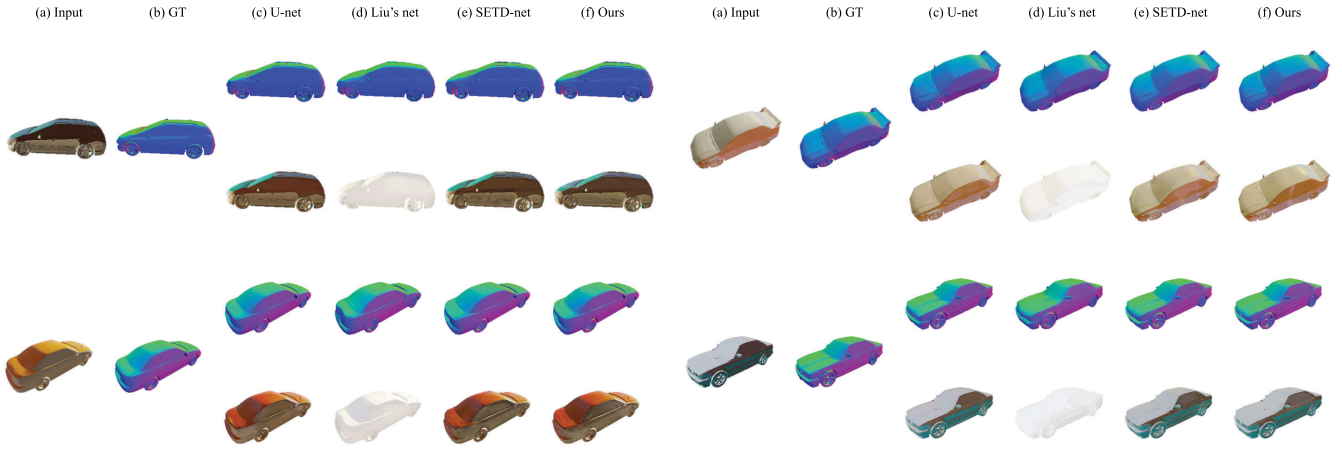

FIGURE 5. Qualitative comparison of our approach on synthetic data. Each result shows (a)the input image; (b) the ground-truth normal; the estimated normal map and rendered images with the output of (c) U-net, (d) Liu's net, (e) SETD-net, and (f) our method.

the complete information of the pixel-wise materials and therefore results in poor performance. Compared with our backbone network that fuses the low-level and high-level features with skip connections, the proposed geometry and material regression module further improves the performance by learning how to fuse and share features from the layers of two tasks automatically. This result also shows that these tasks indeed have relationships that can be used to improve accuracy.

C. EVALUATION FOR THE ILLUMINATION MODULE

We compare our illumination module with the other three methods: 1) T-net 2) encoder-decoder net 3) U-net. All of them learn a latent feature vector encoding the environment maps from the object's image.

SESD-net. The first compared baseline is the lighting prediction branch from Liu's work [17], namely the single encoder and single decoder (SESD) net without skip connections. Specifically, the encoder maps the image of an object to a feature vector and the decoder uses this vector to generate an environment map of size $64 \times 128 \times 3$. The encoder includes 7 convolutions followed by the ReLU activation functions and 2 fully connected layers. The decoder is composed of a sequence of deconvolutional layers.

T-net. T-net is inspired by the work of Weber *et al.* [29], which solves this problem through two stages. They first train an auto-encoder to learn to compress the image to a latent vector compactly modeling the indoor lighting then use a DCNN to generate the environment maps based on the latent feature space. The DCNN takes as input a single image of an object and its normal map. Besides, the DCNN and the auto-encoder share the same decoding part. The original system is performed over an indoor lighting dataset and we retrain it with our outdoor environment maps. The auto-encoder is firstly trained on our data to converge and then the DCNN composed of an encoder network and the trained decoder of the auto-encoder are trained again. Note that the same encoder and decoder network architecture with SESD-net is adopted to eliminate the effects from the depth of the network and ensure that the results are comparable.

TABLE 2. Comparison between our proposed architecture and the related networks for illumination module.

	Environment maps		Rendering	
	MAE	SSIM	SSIM	MSE
SESD-net	0.6252	0.2603	0.9696	168.6778
T-net	0.6254	0.2605	0.9696	168.9825
U-net	0.6158	0.2552	0.9704	161.8387
Ours	**0.5940**	**0.3037**	**0.9743**	**148.6923**

U-net. U-net is the backbone of our proposed architecture. It is considered as a baseline to show the advantages of the feature transformation block. In this experiment, the same U-net structure with the material geometry module is adopted and more details can be found in the section of evaluation for the intermediate feature fusion. Besides, we adjust the last 3 layers of the U-net to match the resolution of the environment map.

Our proposed network is composed of the feature transformation block and the U-net. All the networks are trained with the same data for comparisons. The rendering results with the predicted environment maps are generated to show the effectiveness. The quantitative and qualitative results are provided in Fig. 6 and Table 2. Both the mean absolute error (MAE) and the SSIM metric between the inferred illumination and the ground truth are computed. Given the ground-truth normal and material, we also present the SSIM and MSE error between the rendered image with the predicted environment maps and the original one.

It can be seen from the experimental results that our proposed method outperforms the others. With the proposed feature transformation block, the learned feature maps of the objects in the image are successfully transferred to the

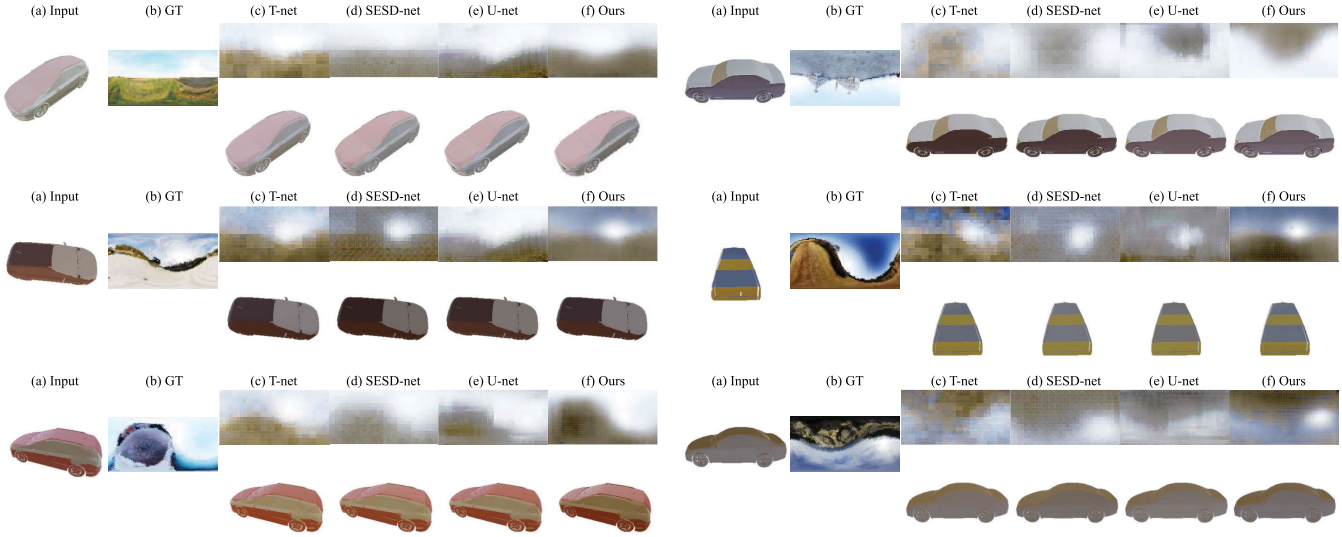

FIGURE 6. Qualitative comparison of our approach on synthetic data. Each result shows (a)the input image; (b) the ground-truth lighting; the rendered images with predicted environment maps of (c) T-net, (d) SESD-net, (e) U-net, and (f) our method.

lighting space. Moreover, the following layers learn the related feature from the transferred information and improve the performance by fusing the low-level and high-level features. The experiments show that the fusion of different hierarchical features is indeed helpful to recover the environment maps from a single image of an object.

Compared with the other baselines' methods, the T-net has a better decoder due to the previous training with an auto-encoder, but its recovered environment maps are not smooth enough as shown in the qualitative results. Our approach eliminates the trellis effect in the transformation and therefore obtains better visualization. SESD-net intuitively maps the image to a latent vector to generate an environment map, but because of the ambiguities of such a problem, the learned vector usually includes more information about the object, which results in poor performance of the predicted environment map. U-net fuses the hierarchical features by skip connections, but these features are more related to the low-level information rather than the semantic one. On contrary, our feature transformation block can automatically choose the illumination information, and then the U-net can use such features to improve the performance.

D. EFFECTIVENESS OF SELF-SUPERVISED TEST-TIME TRAINING

We insert the differentiable rendering equation (11) with the DSBRDF as the renderer of the proposed architecture. Note that this is different from the work in [17], we further extend it to support the per-pixel material rendering. In such a setting, the errors can be back-propagated to each channel of one pixel for updating the weights, which allows our model to learn to predict multi-materials.

It is well known that acquiring large amounts of labeled data is extremely infeasible in practice and, on the other side, training the network with the differentiable renderer from scratch results in the collapse among the elements. Consequently, we first train the system with our synthetic data to initialize the learning to the desired direction. Because the synthetic data is generated by a simplified physically based rendering procedure, the rendered images are different from the real images captured by the camera. The different distributions of such two datasets lead to the domain gap which the model has to overcome if we want to perform the trained model on the real images.

As a result, we exploit the test-time training strategy to finish the domain adaption and improve the performance in a self-supervised way. We use the single material images [14] and multi-material images [44] to finetune our model trained on the synthetic data. A combined subset from these two datasets is extracted as our self-supervised test-time training dataset. The images in this dataset are captured by a camera in a natural scene.

Because the dataset is purely LDR images, we use the deep learning-based method [45] to preprocess the LDR data to transfer them to HDR images. Such a method proposes a reverse tone mapping based on deep learning to recover the lost information caused by camera sensors and generate visually convincing HDR results. After the preprocessing, the self-supervised learning uses (13) as the loss function.

We show the performance of our proposed method by comparing it with the other baseline. Because of lacking the normal and light ground-truth of the real images, the SSIM and MSE metrics on the final rendered images are used as quantitative results. The inferred normal map, environment map, and the rendered input image are provided as the qualitative results. The rendered image with the predicted normal map, environment map, and material tensor shows the qualitative material and final appearance reconstruction.

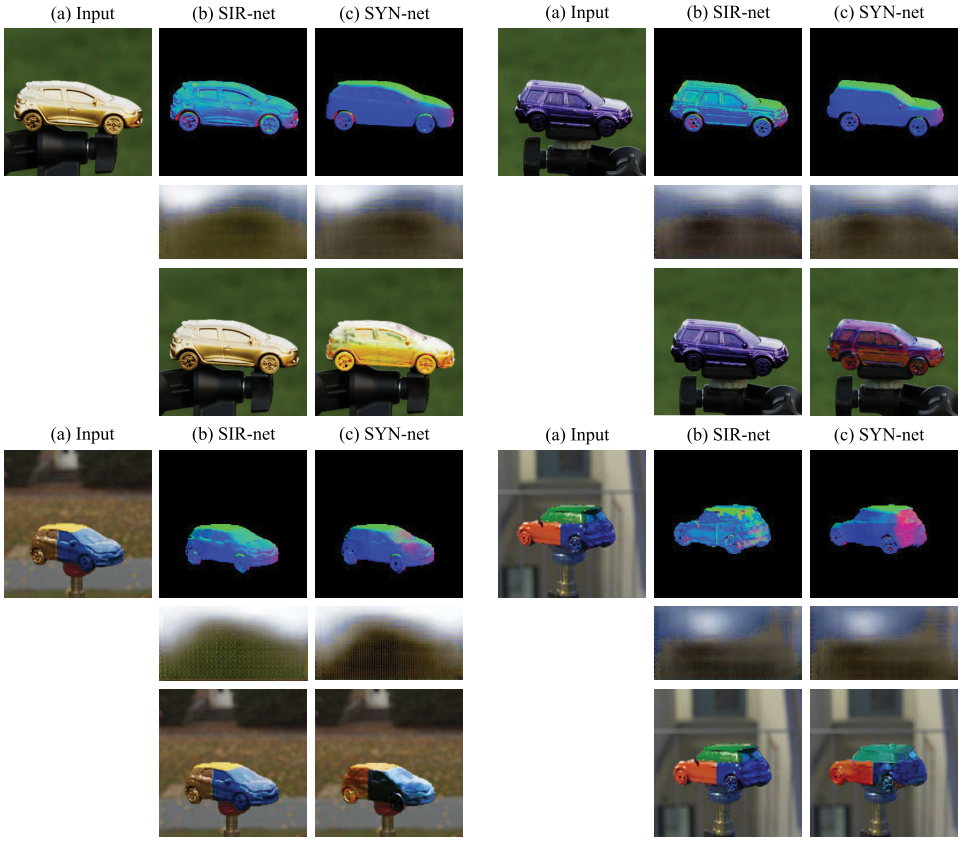

FIGURE 7. Qualitative comparison of our method on real data. Each result shows (a)the input image; (b) the output of SIR-net and the rendered image with the output; (c) the estimated normal map, environment map of SYN-net, and the rendered image with the output elements.

TABLE 3. Difference between the baseline and our method.

	Rendering	
	SSIM	MSE
SYN-net	0.8851	443.8661
SIR-net	**0.9914**	**6.3516**

SYN-net. SYN-net is the model completely trained with synthetic data in a supervised way. We directly use real test images to evaluate its performance.

SIR-net. SIR-net is the proposed network finetuned by the test-time training with the real images. During the test-time training, we firstly keep the weights of the illumination module fixed and train the geometry material module independently. After converging, the modules are combined to be trained for a few iterations.

The qualitative and quantitative results are presented in Fig. 7 and Table 3. respectively. It can be seen from the experimental results that the proposed SIR-net achieves a dramatic improvement, which also implicitly shows the huge gap between the synthetic data and the real one. In contrast, SYN-net generates many wrong inferences and loses the specular property of the material. For all the samples our SIR-net can recover both the multi-materials and the single material, which shows the advantages of the extended multi-material form. Besides, because the image reconstruction loss constrains the network in terms of the perception consistency, it may result in some intermediate errors to mislead the inferred material, normal, and illumination.

E. DYNAMIC RANGE FOR INVERSE RENDERING

As what is explained in section V-B, the dynamic range of input images could affect the performance, we provide a comparison to show the importance of the dynamic range. Here the geometry material model and the illumination model are trained with the corresponding LDR version of the synthetic data to reveal how the accuracy changes with the dynamic range in supervised learning. The learning procedure uses (14) as the loss function. Both the quantitative and qualitative results are presented in Table 4 and Fig. 8. We also provide the qualitative result from the self-supervised test-time training on real images in Fig. 9. Besides, we present the corresponding HDR results to show the difference explicitly.

We conclude that the HDR data contains most brightness information which contributes to recovering the environment map, so the DCNN is feasible to learn the correct illuminance intensity from such data. The LDR input has lost the most original properties of the scene, thus it is more difficult to infer the physical results.

TABLE 4. The comparison between different dynamic ranges for the geometry, material, and environment map on the synthetic dataset.

	Normal		Material	Environment maps		Rendering	
	Mean	Median	MSE	MAE	SSIM	SSIM	MSE
LDR	**7.0583**	**4.0583**	0.8960	0.6267	0.2562	0.9468	153.8095
HDR	7.2476	4.2020	**0.8864**	**0.5940**	**0.3037**	**0.9615**	**139.5654**

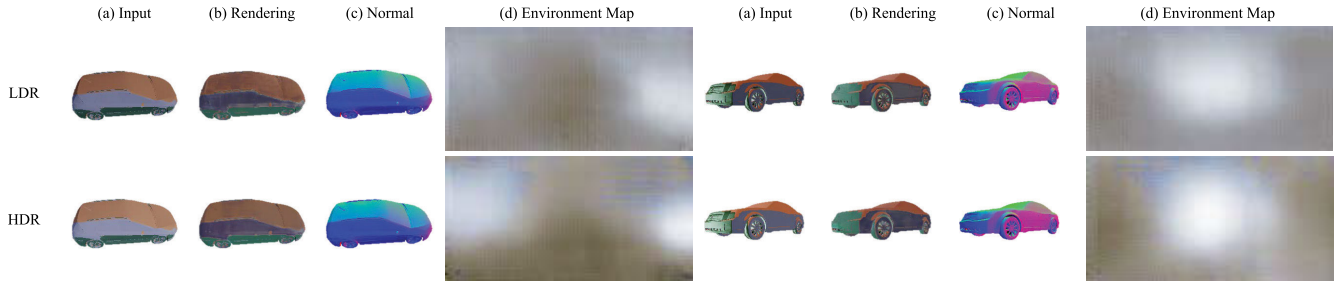

FIGURE 8. Qualitative comparison of input with different dynamic ranges on synthetic data. Each result shows (a)the input image; (b) the rendered image; (c) the estimated normal map and (d) the inferred environment map under the different dynamic range.

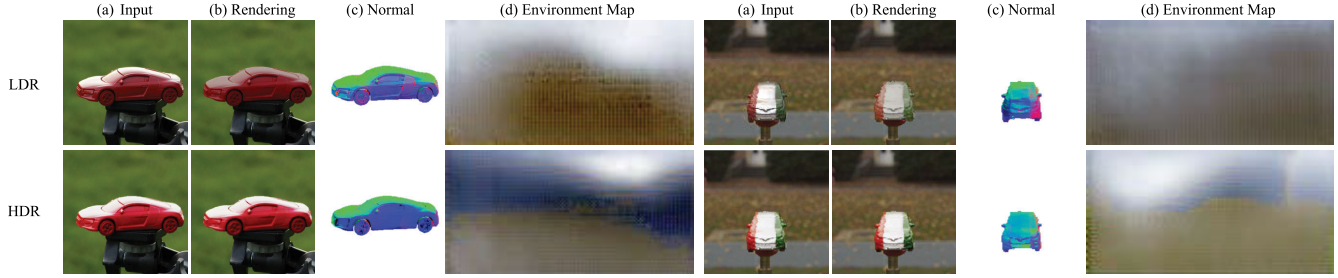

FIGURE 9. Qualitative comparison of input with different dynamic ranges on real data. Each result shows (a)the input image; (b) the rendered image; (c) the estimated normal map and (d) the inferred environment map under the different dynamic range.

For the geometry material module, the normal map is mainly related to the scene's structure feature, but the high light intensity may cover this feature in images. Consequently, the result shows that the accuracy of normal maps changes with the dynamic range. For the material, it can be found that LDR input loses the specular property, which reveals that the lower dynamic range sets an obstacle for learning the high reflectance. Overall, for a stable scene, both the HDR input and LDR input can generate similar material and normal, which shows the dynamic range-invariant characteristics of the geometry material module.

It can also be found from the above results that the HDR input provides more information and thus decreases the error from the brightness. However, capturing an HDR image is infeasible in most scenarios. Furthermore, it is impossible to transfer the existing LDR images to their real HDR version. Although some algorithms can finish this task in terms of better visualization, the original physical lighting intensity cannot be recovered. In comparison, the LDR data lacks physical information and therefore the self-supervised training could impose an additional mapping from LDR to HDR on the learning procedure. Consequently, it decreases the performance of the whole system and leads to more visual incredibility. In summary, HDR input can provide better results than LDR, but it is more difficult to capture.

VII. CONCLUSION

In this article, a designed DCNN is utilized to obtain the material, the geometry, and the ambient light of an object from a single image. Based on these acquired elements, the proposed architecture can reconstruct an object's appearance in a self-supervised test-time training way. The BRDF and the normal map are introduced to represent the material and geometry. A feature fusion unit with a channel-wise attention mechanism is proposed to improve their accuracies. The HDR environment maps describe the ambient light and the feature transformation block is adopted to fuse the low-level and high-level features of the illumination module. To tackle the absence of labeled data, the synthetic data is generated to train the DCNN, and then a self-supervised test-time training strategy is adopted to transfer the domain by inserting a differentiable renderer after the proposed modules. The designed architectures are evaluated and the results show that our methods outperform the other baselines.

In our system, the geometry is related to the depth information. Consequently, this model could be further combined with the depth estimation or even 3D reconstruction. Apart from the absence of labeled data, direct self-supervised training from scratch could result in the collapse of each element, so how to solve this problem and constructing a large dataset are promising directions in the future.

ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers for their valuable suggestions.

REFERENCES

[1] W. Matusik, H. Pfister, M. Brand, and L. Mcmillan, "A data-driven reflectance model," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 759–769, Jul. 2003. [Online]. Available: http://doi.acm.org/10.1145/882262.882343

[2] E. A. Khan, E. Reinhard, R. W. Fleming, and H. H. Bülthoff, "Image-based material editing," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 654–663, Jul. 2006. [Online]. Available: http://doi.acm.org/10.1145/1141911.1141937

[3] J. T. Barron and J. Malik, "Shape, illumination, and reflectance from shading," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1670–1687, Aug. 2015.

[4] S. Lombardi and K. Nishino, "Reflectance and natural illumination from a single image," in *Computer Vision*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Germany: Springer, 2012, pp. 582–595.

[5] S. Lombardi and K. Nishino, "Single image multimaterial estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 238–245.

[6] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu, "Real-time geometry, albedo, and motion reconstruction using a single RGB-D camera," *ACM Trans. Graph.*, vol. 36, no. 3, pp. 1–13, Jul. 2017. [Online]. Available: http://doi.acm.org/10.1145/3083722

[7] P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," in *Proc. 25th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, New York, NY, USA, 1998, pp. 189–198. [Online]. Available: http://doi.acm.org/10.1145/280814.280864

[8] D. A. Calian, K. Mitchell, D. Nowrouzezahrai, and J. Kautz, "The shading probe: Fast appearance acquisition for mobile AR," in *Proc. SIGGRAPH Asia Tech. Briefs (SA)*, New York, NY, USA, 2013, pp. 20:1–20:4. [Online]. Available: http://doi.acm.org/10.1145/2542355.2542380

[9] R. Ramamoorthi and P. Hanrahan, "An efficient representation for irradiance environment maps," in *Proc. 28th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, New York, NY, USA, 2001, pp. 497–500. [Online]. Available: http://doi.acm.org/10.1145/383259.383317

[10] R. Xia, Y. Dong, P. Peers, and X. Tong, "Recovering shape and spatially-varying surface reflectance under unknown illumination," *ACM Trans. Graph.*, vol. 35, no. 6, pp. 187:1–187:12, Nov. 2016. [Online]. Available: http://doi.acm.org/10.1145/2980179.2980248

[11] T. Y. Wang, T. Ritschel, and N. J. Mitra, "Joint material and illumination estimation from photo sets in the wild," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2018, pp. 22–31.

[12] A. Meka, G. Fox, M. Zollhofer, C. Richardt, and C. Theobalt, "Live user-guided intrinsic video for static scenes," *IEEE Trans. Vis. Comput. Graphics*, vol. 23, no. 11, pp. 2447–2454, Nov. 2017.

[13] Y. Tang, R. Salakhutdinov, and G. E. Hinton, "Deep lambertian networks," in *Proc. ICML*, 2012, pp. 1–8.

[14] S. Georgoulis, K. Rematas, T. Ritschel, E. Gavves, M. Fritz, L. Van Gool, and T. Tuytelaars, "Reflectance and natural illumination from single-material specular objects using deep learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1932–1947, Aug. 2018.

[15] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[16] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 239–248.

[17] G. Liu, D. Ceylan, E. Yumer, J. Yang, and J.-M. Lien, "Material editing using a physically based rendering network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2261–2269.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[19] F. Xia, P. Wang, X. Chen, and A. L. Yuille, "Joint multi-person pose estimation and semantic part segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6080–6089.

[20] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2650–2658.

[21] A. Meka, M. Maximov, M. Zollhofer, A. Chatterjee, H.-P. Seidel, C. Richardt, and C. Theobalt, "LIME: Live intrinsic material estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6315–6324.

[22] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau, "Single-image SVBRDF capture with a rendering-aware deep network," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 1–15, Aug. 2018. [Online]. Available: http://www-sop.inria.fr/reves/Basilic/2018/DADDB18

[23] S. Sengupta, A. Kanazawa, C. D. Castillo, and D. W. Jacobs, "SfSNet: Learning shape, refectance and illuminance of faces in the wild," in *Proc. Comput. Vis. Pattern Regognition (CVPR)*, 2018, pp. 6296–6305.

[24] W.-C. Ma, H. Chu, B. Zhou, R. Urtasun, and A. Torralba, "Single image intrinsic decomposition without a single intrinsic image," in *Computer Vision*. Cham, Switzerland: Springer, 2018, pp. 211–229.

[25] Z. Li and N. Snavely, "Learning intrinsic image decomposition from watching the world," 2018, *arXiv:1804.00582*. [Online]. Available: http://arxiv.org/abs/1804.00582

[26] K. Nishino, "Directional statistics BRDF model," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 476–483.

[27] R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," *SIGGRAPH Comput. Graph.*, vol. 15, no. 3, pp. 307–316, Aug. 1981, doi: 10.1145/965161.806819.

[28] K. Nishino and S. Lombardi, "Directional statistics-based reflectance model for isotropic bidirectional reflectance distribution functions," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 28, no. 1, pp. 8–18, Jan. 2011. [Online]. Available: http://josaa.osa.org/abstract.cfm?URI=josaa-28-1-8

[29] H. Weber, D. Prévost, and J.-F. Lalonde, "Learning to estimate indoor lighting from 3D objects," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2018, pp. 199–207.

[30] S. Ruder, "An overview of multi-task learning in deep neural networks," *CoRR*, vol. abs/1706.05098, p. 14, Jun. 2017. [Online]. Available: http://arxiv.org/abs/1706.05098

[31] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[32] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham, Switzerland: Springer, 2015, pp. 234–241.

[33] M.-A. Gardner, K. Sunkavalli, E. Yumer, X. Shen, E. Gambaretto, C. Gagné, and J.-F. Lalonde, "Learning to predict indoor illumination from a single image," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 176:1–176:14, Nov. 2017. [Online]. Available: http://doi.acm.org/10.1145/3130800.3130891

[34] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.

[35] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Computer Vision*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 483–499.

[36] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[37] V. Jain and E. Learned-Miller, "Online domain adaptation of a pre-trained cascade of classifiers," in *Proc. CVPR*, Jun. 2011, pp. 577–584.

[38] Y. Chen, C. Schmid, and C. Sminchisescu, "Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7062–7071.

[39] R. G. Cinbis, J. Verbeek, and C. Schmid, "Unsupervised metric learning for face identification in TV video," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1559–1566.

[40] J. T. Kajiya, "The rendering equation," *ACM SIGGRAPH Comput. Graph.*, vol. 20, no. 4, pp. 143–150, Aug. 1986. [Online]. Available: http://doi.acm.org/10.1145/15886.15902

[41] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," 2014, *arXiv:1408.5093*. [Online]. Available: http://arxiv.org/abs/1408.5093

[42] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*. [Online]. Available: http://arxiv.org/abs/1212.5701

[43] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[44] S. Georgoulis, K. Rematas, T. Ritschel, M. Fritz, T. Tuytelaars, and L. Van Gool, "What is around the camera?" in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5170–5178.
[45] G. Eilertsen, J. Kronander, G. Denes, R. K. Mantiuk, and J. Unger, "HDR image reconstruction from a single exposure using deep CNNs," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 1–15, Nov. 2017, doi: 10.1145/3130800.3130816.

TIANTENG BI received the B.S. degree in optical information science and technology from the Taiyuan University of Technology, Shanxi, China, in 2012. He is currently pursuing the Ph.D. degree with the School of Optics and Photonics, Beijing Institute of Technology, Beijing, China. His research interests include computer vision, computer graphics, virtual reality, and augmented reality.

JUNJIE MA (Member, IEEE) received the B.Eng. degree from the Taiyuan University of Technology, in 2012, and the Ph.D. degree from the Beijing Institute of Technology, in 2020. He was funded by the China Scholarship Council and an Internship Student at Nanyang Technological University, from 2017 to 2019. He is currently a Postdoctoral Researcher with Tsinghua University. His research interests include computer vision and deep learning.

YUE LIU (Member, IEEE) received the Ph.D. degree in telecommunication and information system from Jilin University, Jilin, China, in 2000. He is currently a Professor of Optical Engineering with the School of Optics and Photonics, Beijing Institute of Technology, Beijing. His research interests include human–computer interaction, virtual and augmented reality, accurate tracking of the pose of camera, 3D display systems, camera calibration, and so on.

DONGDONG WENG received the Ph.D. degree in optical engineering from the Beijing Institute of Technology, Beijing, China, in 2006. He is currently a Professor of Optical Engineering with the School of Optics and Photonics, Beijing Institute of Technology. His research interests include virtual reality and augmented reality, human–computer interaction, new media entertainment theme park, precise location tracking algorithm and the corresponding devices, and so on.

YONGTIAN WANG (Member, IEEE) received the B.Sc. degree in precision instrumentation from Tianjin University, China, in 1982, and the Ph.D. degree in optics from the University of Reading, U.K., in 1986. He is currently a Professor with the School of Optics and Photonics, Beijing Institute of Technology. His research interests include optical design and CAD, optical instrumentation, image processing, virtual reality, and augmented reality.