Object-based Illumination Estimation with Rendering-aware Neural Networks

Xin Wei[1,2], Guojun Chen[1], Yue Dong[1], Stephen Lin[1], and Xin Tong[1]

[1] Microsoft Research Asia
[2] Zhejiang Univiersity

Abstract. We present a scheme for fast environment light estimation from the RGBD appearance of individual objects and their local image areas. Conventional inverse rendering is too computationally demanding for real-time applications, and the performance of purely learning-based techniques may be limited by the meager input data available from individual objects. To address these issues, we propose an approach that takes advantage of physical principles from inverse rendering to constrain the solution, while also utilizing neural networks to expedite the more computationally expensive portions of its processing, to increase robustness to noisy input data as well as to improve temporal and spatial stability. This results in a rendering-aware system that estimates the local illumination distribution at an object with high accuracy and in real time. With the estimated lighting, virtual objects can be rendered in AR scenarios with shading that is consistent to the real scene, leading to improved realism.

1 Introduction

Consistent shading between a virtual object and its real-world surroundings is an essential element of realistic AR. To achieve this consistency, the illumination environment of the real scene needs to be estimated and used in rendering the virtual object. For practical purposes, the lighting estimation needs to be performed in real time, so that AR applications can accommodate changing illumination conditions that result from scene dynamics.

Traditionally, lighting estimation has been treated as an inverse rendering problem, where the illumination is inferred with respect to geometric and reflectance properties of the scene [32, 31, 4, 29]. Solve the inverse rendering problem with a single image input is ill-conditioned, thus assumptions are made to simplify the lighting model or to limit the supported material type. Despite the ambiguity among shape, material, and lighting. Solving the optimization involved in inverse rendering entails a high computational cost that prevents real-time processing, since the forward rendering as a sub-step of such optimization already difficult to reach real-time performance without compromising the accuracy.

Recent methods based on end-to-end neural networks provide real-time performance [18, 11, 7, 26, 10, 35], specifically, they regard the input image as containing partial content of the environment map and estimate high resolution

environment light based on those contents, rich content in the input image is critical to infer the surrounding environment without ambiguity. However, many AR applications have interactions focused on a single object or a local region of a scene, where the partial content of the environment map in the input image is very limited. Thinking about guessing an environment map based on the image of a toy putting on the ground (like Figure 3.h), although the ground is part of the environment map, it provides limited clues to rule out ambiguities when inferring the surroundings, yielding unstable estimation results.

On the contrary, given such a scenario, the appearance of the object and the shadow cast by the object provide strong cues for determining the environment light. Utilizing such cues with neural networks is however challenging, due to the complex relationship between the lighting, material, and appearance. First, the neural network needs to aware of the physical rules of the rendering. Second, physically based rendering is computationally intensive, simple analysis by synthesis is not suitable for a real-time application. Third, the diffuse and specular reflections follow different rules and the resulting appearance are mixed together in the input based on the unknown material property of the object. Previous methods already prove optimizing an end-to-end model for such a complex relationship is challenging and inefficient [35].

In this paper, we present a technique that can estimate illumination from the RGBD appearance of an individual object and its local image area. To make the most of the input data from a single object, our approach is to integrate physically-based principles of inverse rendering together with deep learning. An object with a range of known surface normals provides sufficient information for lighting estimation by inverse rendering. To deal with the computational inefficiency of inverse rendering, we employ neural networks to rapidly solve for certain intermediate steps that are slow to determine by optimization. Among these intermediate steps are the decomposition of object appearance into reflection components – namely albedo, diffuse shading, and specular highlights – and converting diffuse shading into a corresponding angular light distribution. On the other hand, steps such as projecting specular reflections to lighting directions can be efficiently computed based on physical laws without needing neural networks for speedup. Estimation results are obtained through a fusion of deep learning and physical reasoning, via a network that also accounts for temporal coherence of the lighting environment through the use of recurrent convolution layers.

In this way, our method takes advantage of physical knowledge to facilitate inference from limited input data, while making use of neural networks to achieve real-time performance. This rendering-aware approach moreover benefits from the robustness of neural networks to noisy input data. Our system is validated through experiments showing improved estimation accuracy from this use of inverse rendering concepts over a purely learning-based variant. The illumination estimation results also compare favorably to those of related techniques, and are demonstrated to bring high-quality AR effects.

2 Related Work

A large amount of literature exists on illumination estimation in computer graphics and vision. Here, we focus on the most recent methods and refer readers to the survey by Kronander et al. [24] for a more comprehensive review.

Scene-based lighting estimation Several methods estimate lighting from a partial view of the scene. In earlier works, a portion of the environment map is obtained by projecting the viewable scene area onto it, and the rest of the map is approximated through copying of the scene area [22] or by searching a panorama database for an environment map that closely matches the projected scene area [21].

Deep learning techniques have also been applied to this problem. Outdoor lighting estimation methods usually take advantage of the known prior distribution of the sky and predict a parametric sky model [18, 44] or a sky model based on a learned latent space [17]. Another commonly used low-parametric light model is the spherical harmonic (SH) model, adopted by [7, 12]. However, an SH model is generally inadequate for representing high frequency light sources. Recently, Gardner et al. [10] proposed a method estimating a 3D lighting model composed of up to 3-5 Gaussian lights and one global ambient term, improving the representation power of the parametric light model. Instead of depending on a low-order parametric model, our method estimates high resolution environment maps without the limitations of such models.

Recent scene-based methods [11, 26, 35] regard the input image as containing partial content of the environment map and estimate high resolution environment light based on those contents. The input image is usually regarded as a warped partial environment map that follows the spherical warping [11] or warping based on depth [35]. The quality their estimated light depends on the amount of content in the known partial environment, since fewer content in the partial environment leads to stronger ambiguity of the missing part, increasing the estimation difficulties. By contrast, our work seeks to estimate an environment map from only the shading information of an object, without requiring any content of the environment map, which is orthogonal to scene-based methods and could be a good complement.

Object-based lighting estimation Illumination has alternatively been estimated from the appearance of individual objects in a scene. A common technique is to include a mirrored sphere in the image and reconstruct the lighting environment through inverse rendering of the mirrored reflections [9, 40, 39]. This approach has been employed with other highly reflective objects such as human eyes [30], and extended to recover multispectral lighting using a combination of light probes and color checker charts [27].

Avoiding the use of special light probes, several inverse rendering methods utilize general objects instead. Some estimate lighting from diffuse objects for which rough geometry is measured by a depth sensor [16, 3, 15], reconstructed

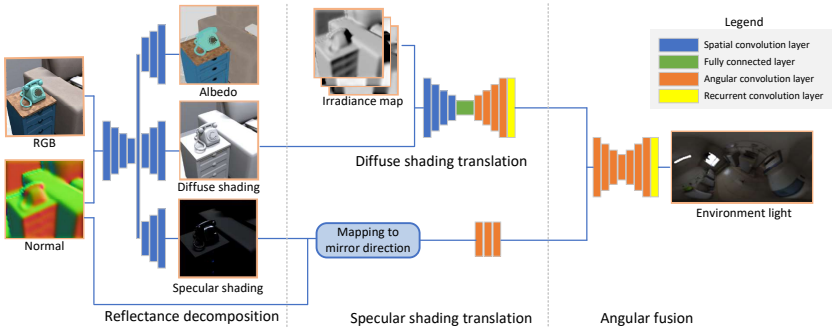

Fig. 1. Overview of our system. The input RGB image crop is first decomposed into albedo, diffuse shading and specular shading maps. This decomposition is assisted by shape information from a normal map, calculated from rough depth data. The diffuse shading is translated into the angular lighting domain with the help of auxiliary irradiance maps computed from the rough depth, and the specular shading is geometrically mapped to their mirror direction. The translated features are then processed in the angular fusion network to generate the final environment light estimate.

by multi-view stereo [42], or jointly estimated with the lighting [4]. For homogeneous objects with a known shape, the BRDF has been recovered together with all-frequency illumination [31, 29]. Cast shadows from objects with known geometry have also been analyzed to extract high-frequency lighting information [32, 20]. With a complete scan and segmentation of a scene, inverse path-tracing has been used to estimate both the light and object BRDFs [2]. Inverse rendering approaches such as these have proven to be effective, but are too computationally slow for real-time AR applications due to iterative optimization. Recently, Sengupta et al. [33] trained a inverse rendering neural network; however, their direct rendering component assumes diffuse shading only and their method produces a low-resolution environment map limited by the direct renderer.

Deep learning has also been applied for illumination estimation from objects. Many of these works specifically address the case of human faces, for which prior knowledge about face shape and appearance can be incorporated [38, 5, 43, 37, 45, 34, 41, 36]. For more general cases, neural networks have been designed to estimate an environment map from homogeneous specular objects of unknown shape [14] and for piecewise constant materials while considering background appearance [13]. By contrast, our approach does not make assumptions on the surface reflectance, and makes greater use of physical principles employed in inverse rendering.

3 Overview

Our method takes RGBD video frames from an AR sensor as input. For a cropped object area within a frame, an illumination environment map is estimated for the object's location. Illustrated in Fig. 2, the estimation process consists of

three main components: reflectance decomposition, spatial-angular translation, and angular fusion.

Reflectance decomposition aims to separate object appearance into albedo, diffuse shading, and specular reflections. Such a decomposition facilitates inverse rendering, as diffuse and specular reflection arise from different physical mechanisms that provide separate cues about the lighting environment.

Spatial-angular translation then converts the computed diffuse and specular shading maps from the spatial domain into corresponding lighting distributions in the angular domain. Since the relationship between lighting and shading is different for diffuse and specular reflections, we perform the spatial-angular translation of diffuse and specular shading separately. For diffuse shading, a neural network is trained to infer a low-resolution environment map from it. On the other hand, the translation from the specular map to lighting angles is computed geometrically based on mirror reflection and a normal map of the object, calculated from its depth values.

The angular lighting maps from diffuse and specular reflections are then merged through an *angular fusion* network to produce a coherent environment map based on the different reflectance cues, as detailed in Sec. 4. To ensure temporal consistency of the estimated environment maps over consecutive video frames, we incorporate recurrent convolution layers [19, 6] that account for feature maps from previous frames in determining the current frame's output.

4 Network structures

Reflectance decomposition network The input of the reflectance decomposition network is a cropped RGB input image and the corresponding normal map computed from the bilateral filtered input depth map as the cross-product of depths of neighboring pixels. After a series of convolutional, downsampling and upsampling layers, the network produces a set of decomposed maps including the diffuse albedo map, diffuse shading map, and specular shading map. Our reflectance decomposition is similar to traditional intrinsic image decomposition, but additionally separates shading into diffuse and specular components. Moreover, the normal map taken as input provides the network with shape information to use in decomposition, and the network also generates a refined normal map as output.

In practice, we follow the network structure and training loss of Li et al. [28] with the following modifications. We concatenate the normal map estimated from the rough depth input as an additional channel to the input. We also split the shading branch into two sibling branches, one for diffuse shading and the other for specular shading, each with the same loss function used for the shading component in [28]. The exact network structure is illustrated in the supplementary material.

Diffuse shading translation The relationship between the angular light distribution and the spatial diffuse shading map is non-local, as light from one angular

direction will influence the entire image according to the rendering equation:

$$R(x, y) = v(x, y)n(x, y) \cdot l \tag{1}$$

where $v(x, y)$ denotes the visibility between the surface point at pixel (x, y) and the light direction l, $n(x, y)$ represents the surface normal, and $R(x, y)$ is the irradiance map for lighting direction l.

Diffuse shading can be expressed as an integral of the radiance over all the angular directions. For discrete samples of the angular directions, the diffuse shading map will be a weighted sum of the irradiance maps for the sampled directions, with the light intensity as the weight. As a result, with an accurate shading map and irradiance maps, we can directly solve for the lighting via optimization [32, 4]. However, optimization based solutions suffer from inaccurate shading and geometry and slow computation. Our translation also follows such a physically-based design. The neural network takes the same input as the numerical optimization, namely the diffuse shading map and the irradiance maps of sparsely sampled lighting directions. Then the network outputs the intensity values of those sampled lighting directions. In practice, we sample the pixel centers of a $8 \times 8 \times 6$ cube-map as the lighting directions to compute a 32×32 resolution auxiliary irradiance map. The diffuse shading map and each irradiance map is separately passed through one layer of convolution, then their feature maps are concatenated together. The concatenated features are sent through a series of convolution and pooling layers to produce a set of feature maps corresponding to $8 \times 8 \times 6$ angular directions. With the feature maps now defined in the angular domain, we rearrange them into a latitude and longitude representation and perform a series of angular domain convolutions to output an estimated environment map of 256×128 resolution.

Specular shading translation Unlike the lowpass filtering effects of diffuse shading, specular reflections are generally of much higher frequency and cannot be adequately represented by the sparse low-resolution sampling used for diffuse shading translation, making the irradiance based solution inefficient. Since specular reflection has a strong response along the mirror direction, we thus map the decomposed specular shading to the angular domain according to the mirror direction, computed from the viewing direction and surface normal as $o = (2n - v)$. Each pixel in the specular shading map is translated individually, and the average value is computed among all the pixels that are mapped to the same angular map location, represented in terms of latitude and longitude.

In practice, depending on the scene geometry, some angular directions may not have any corresponding pixels in the specular shading map. To take this into account, we maintain an additional mask map that indicates whether an angular direction has a valid intensity value or not. After mapping to the 256×128 latitude and longitude map, the maps are processed by four convolution layers to produce angular feature maps.

Angular fusion network After the diffuse and specular translation network, both the diffuse and specular feature maps are defined in the angular domain. Those

feature maps will be concatenated together for the fusion network to determine the final lighting estimates.

The angular fusion network is a standard U-net structure, with a series of convolutions and downsampling for encoding, followed by convolution and upsampling layers for decoding. We also include skip links to map the feature maps from the encoder to the decoder, to better preserve angular details. Instead of having the estimated lighting from the diffuse spatial-angular translation network as input, we use the feature map (just before the output layer) of the translation network as the diffuse input to the fusion network. The feature map can preserve more information processed by the translation network (e.g. confidence of the current estimation / transformation), which may help with fusion but is lost in the lighting output.

Recurrent convolution layers Due to the ill-conditioned nature of lighting estimation, there may exist ambiguity in the solution for a video frame. Moreover, this ambiguity may differ from frame to frame because of the different information that the frames contain (e.g., specular objects reflect light from different directions depending on the viewing angle, and cast shadows might be occluded by the objects in certain views). To help resolve these ambiguities and obtain a more coherent solution, we make use of temporal information from multiple frames in determining light predictions. This is done by introducing recurrent convolution layers into the diffuse shading translation network as well as the angular fusion network.

Depth estimation for RGB only input Our system is built with RGBD input in mind. However, scene depth could instead be estimated by another neural network, which would allow our system to run on RGB images without measured depth. We trained a depth estimation network adapted from [25] but with the viewpoint information as an additional input, as this may facilitate estimation over the large range of pitch directions encountered in AR scenarios. The estimated depth is then included as part of our input. Figure 7 presents real measured examples with light estimated using the predicted depth.

5 Training

5.1 Supervision and training losses

We train all the networks using synthetically generated data to provide supervision for each network individually. For the reflectance decomposition network, ground truth maps are used to supervise its training. Following previous networks for intrinsic image decomposition [28], we employ the L2 loss of the maps as well as the L2 loss of the gradients of the albedo and specular maps. The fusion network is trained with the ground truth full-resolution environment map, using the Huber loss.

Recurrent convolution training We train the recurrent convolution layers with sequential data and expand the recurrent layers. In practice, we expand the recurrent layers to accommodate 10 consecutive frames during training. When training the recurrent layers, in addition to the Huber loss that minimizes the difference between the estimated lighting and the ground truth for each frame, we also include a temporal smoothness loss to promote smoothness between consecutively estimated lights L_i, L_{i+1}, defined as

$$\mathcal{L}_t(L_i, L_{i+1}) = ||L_i - \mathcal{W}_{i+1\to i}(L_{i+1})||^2 \tag{2}$$

where $W_{i+1\to i}$ is the ground truth optical flow from the lighting of the $(i+1)$-th frame to the i-th frame.

5.2 Training data preparation

The training data is composed of many elements such as reflectance maps and angular environment lighting which are difficult to collect in large volumes from real scenes. As a result, we choose to train our neural networks with synthetically generated data. For better network generality to real-world inputs, this synthetic data is prepared to better reflect the properties of real scenes. The environment lights consist of 500 HDR environment maps collected from the Internet, 500 HDR environment map captured by a 360 camera and 14K randomly generated multiple area light sources. We collected over 600 artist modeled objects with realistic texture details and material variations and organize those objects into local scenes that include about 1 - 6 objects. The scene is then lit by randomly selected environment lights. The viewpoint is uniformly sampled over the upper hemisphere to reflect most AR scenarios. Please refer to the supplementary materials for more details about training data preparation.

5.3 Implementation

We implement the neural networks in Tensorflow [1]. We train our system using the Adam [23] optimizer, with a 10^{-5} learning rate and the default settings for other hyper-parameters. The decomposition is separately trained using 2×10^6 iterations; the diffuse-translation network and the fusion network are first trained with 8×10^5 iterations without recurrent layers, and then finetuned over 10^5 iterations for the recurrent layer training. Finally, the whole system is finetuned end-to-end with 10^5 iterations. Training the decomposition network takes about 1 week on a single NVidia Titan X GPU. The diffuse-translation and fusion network training takes about 10 hours, and the final end-to-end finetuning takes about 20 hours.

6 Results

6.1 Validations

Single-image inputs Beside video frames, our network does support single-image input by feeding ten copies of the single static image into the network to get the

lighting estimation result. This allows us to perform validation and comparsion to existing works on single-image inputs.

Error metric To measure the accuracy of the estimated light, we directly measure the RMSE compared to the reference environment map. In addition, since the goal of our lighting estimation is rendering virtual objects for AR applications, we also measure the accuracy of our system using the rendering error of a fixed set of objects with the estimated light. All the objects in this set are shown as virtual inserts in our real measured results.

Dataset To evaluate the performance of our method on real inputs, we captured 180 real input images with the ground truth environment map, providing both numerical and visual results. To systematically analyze how our method works on inputs with different light, layout and materials, we also designed a comprehensive synthetic test set with full control of each individual factor. Specifically, we collected eight artist-designed scenes to ensure plausible layouts, various object shapes, and a range of objects. Ten random viewpoints are generated for each scene, under selected environment maps with random rotations.

Real measured environment maps For testing the performance of our method under different environment lights, we collected 20 indoor environment maps which are not used for neural network training. These environment maps can be classified into several categories that represent different kinds of lighting conditions. Specifically, we have five environment maps that contain a single dominant light source, five with multiple dominant lights, five with very large area light sources, and five environment maps with near ambient lighting.

Table 1 lists the average error for each category of environment light. Intuitively, lights with high frequency details, such as a dominant light, are more challenging than low frequency cases, like near-ambient lighting. Visual results for one typical example from each category are displayed in Figure 2, which shows that our method can correctly estimate environment lighting and produce consistent re-rendering results across all the types of environment lighting.

Synthetic environment maps We also synthetically generate a set of lighting conditions that provide more continuous coverage over area light source size and number of light sources. For each set of lighting conditions, we render each scene

Fig. 2. We test our system with different classes of environment maps. For each class, the left column shows the ground truth, and the right column shows our results.

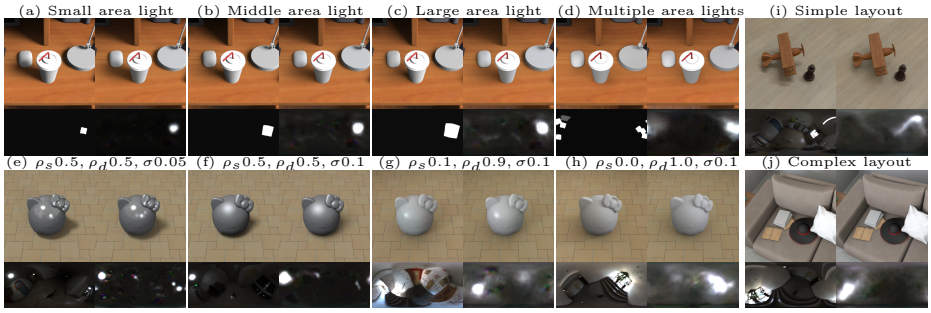

Fig. 3. (a-d) Light estimation results for the same scene lit by different-sized area light sources and multiple light sources. (e-h) Light estimation results for the same shape with varying material properties. (i,j) Light estimation results for scenes with simple or complex layouts. In each group, the left column shows the ground truth, and the right column shows our results.

	Render	Light
Environment light		
One dominant	0.057	2.163
Multiple lights	0.057	1.653
Large area light	0.045	1.222
Near ambient	0.038	0.994
Layout robustness		
Simple layout	0.046	0.915
Complex layout	0.049	1.508
Real captured test set		
Real captured	0.052	0.742

	Synthetic		Real	
Comparasion	Render	Light	Render	Light
Gardner et al. 17[11]	0.095	3.056	0.088	3.462
Gardner et al. 19[10]	0.100	1.614	0.093	1.303
Sato[32] + our decomposition	0.073	1.851	0.081	1.257
Sato[32] + ground truth shading	0.064	1.944	-	-
Ablation				
Diffuse only	0.050	1.447	0.055	0.768
Specualr only	0.056	1.440	0.061	0.756
Without decomposition	0.052	1.434	0.059	0.794
Direct regression	0.049	1.432	0.057	0.759
Without recurrent layer	0.048	1.429	0.056	0.760
Our results	**0.046**	**1.419**	**0.052**	**0.742**
Our results (estimated depth)	0.053	1.437	0.064	0.773

Table 1. Average RMSE of estimated lights and re-rendered images.

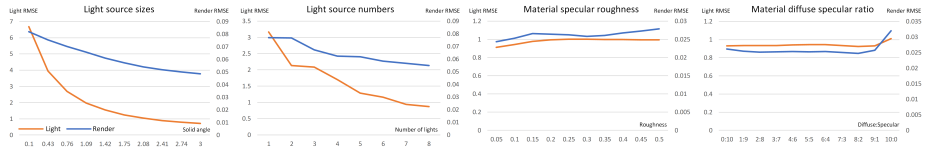

Fig. 4. Lighting estimation and re-rendering error with respect to area light source sizes, number of light sources and different surface materials.

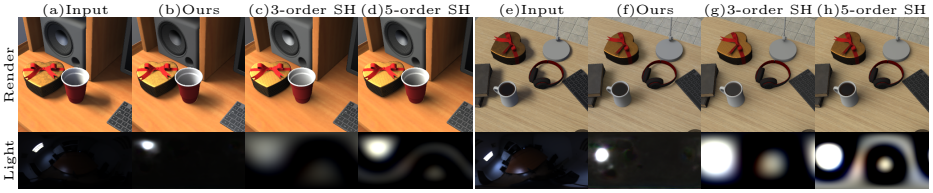

Fig. 5. Comparison to SH-based representation. Our method recovers an all-frequency environment map and produces sharp shadows similar to the ground truth. Fitting the ground truth light with 3- or 5-order SH still cannot reproduce such effects.

and each view under those lighting conditions with ten random rotations. We then plot the average error of our system on the different lighting conditions in the set.

Figure 4 plots the lighting estimates and re-rendering error for each dataset. Inputs with smaller area light sources are more challenging for lighting estimation, since small changes of light position leads to large error in both the angular lighting domain as well as the re-rendering results. Figure 3 (a-d) exhibits a selected scene under various lighting conditions. For a near-diffuse scene lit by multiple area light sources, even the shadows of objects do not provide sufficient information to fully recover the rich details of each individual light source. However, the re-rendering results with our estimated lighting matches well with that rendered with ground truth light, making them plausible for most AR applications.

Object materials To analyze the how surface material affects our estimation results, we prepared a set of objects with the same shape but varying materials. We use a homogeneous specular BRDF with varying roughness and varying diffuse-specular ratios. The results, shown in Figure 3 (e-h), indicate that lighting estimation is stable to a wide range of object materials, from highly specular to purely diffuse. Since our method uses both shading and shadow cues in the scene for lighting estimation, the lighting estimation error is not sensitive to the material of the object, with only a slightly larger error for a purely diffuse scene.

Layouts We test the performance of our method on scenes with different layouts by classifying our scenes based on complexity. The numerical results for the two layout categories are listed in Table 1. As expected, complex scene layouts lead to more difficult lighting estimation. However, as shown in Figure 3 (i,j), our method produces high quality results in both cases.

Spatially-varying illumination Our object-based lighting estimation can be easily extended to support spatially-varying illumination effects, such as near-field illumination, by taking different local regions of the input image. Please find example results in the supplementary material.

6.2 Comparisons

Here, we compare our method to existing lighting estimation methods. For systematic evaluation with ground truth data, we compared with scene-based methods [11, 10] as well as an inverse rendering method [32], on both synthetic and real measured test sets. Our method takes advantage of physical knowledge to facilitate inference from various cues in the input data, and outperforms all the existing methods, as shown in Table 1. In Figure 6, we show one example of a real captured example. Note that our method estimates an area light source with the right size at the correct location, while other methods either over-estimate the ambient light [11] or result in an incorrect light position [10, 32].

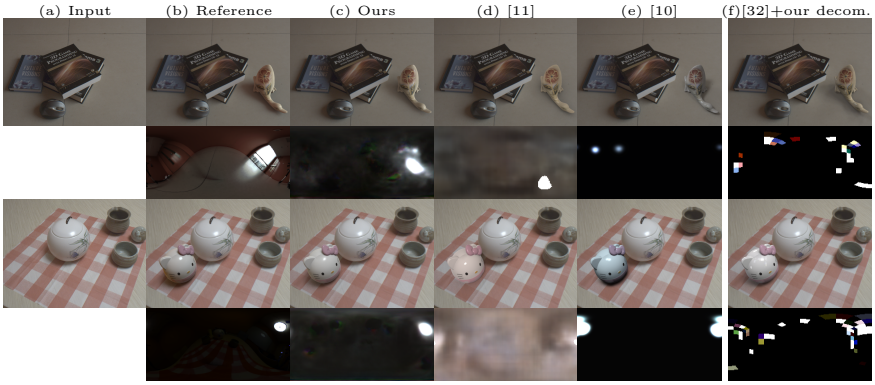

Fig. 6. We compare our method to existing lighting estimation methods on our real captured test set. (c) Our method correctly estimates the size and position of the light sources and produces correct shadow and specular highlights. (d) [11] overestimates the ambient light, the position of the dominant light is off. (e) [10] estimates a incorrect number of lights at incorrect locations. (f)[32] results in many incorrect light sources.

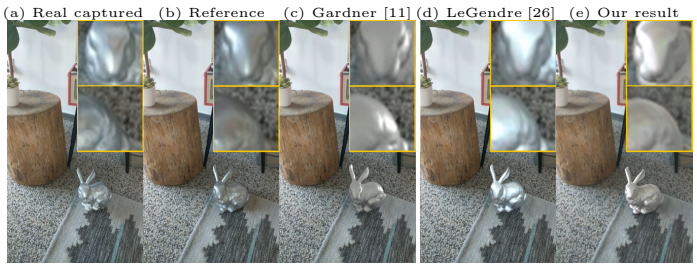

Fig. 7. We compare our method (with estimated depth) to existing lighting estimation methods on a real image from [26] . (a) The photograph of a real 3D-printed bunny placed in the scene. Rendering results of a virtual bunny under (b) captured ground truth environment map, (c) environment map estimated by [11], (d) [26] and (e) our method with estimated depth. Note that our result successfully reproduces specular highlight over the left side of the bunny (closed-up view inserted), similar to the ground-truth. [11] produces wrong specular highlights on the right ear and body of the bunny, and [26] results in wrong highlights all over the bunny.

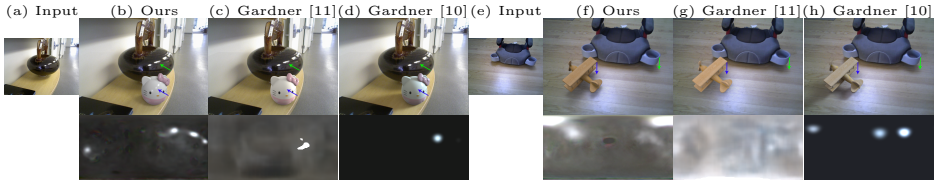

Fig. 8. Comparison on the redwood dataset [8]. Left: our method faithfully reproduces the four specular highlights (blue arrow) visible from the input object (green arrow), which are missing from other methods' results. Right: our method produces shadow with direction consistent to the input (blue and green arrows). Previous methods either fail to reproduce the shadow [11] or generate shadow in the wrong direction [10].

We also compare our method on the redwood RGBD dataset [8], as illustrated in Figure 8. Although without a ground truth reference, virtual objects rendered with our estimated lights produce consistent specular highlights and shadows, while existing methods fail to generate consistent renderings.

Scene based methods usually need input with a wider field of view, thus we compare our method with [26, 11] on their ideal input. Their methods use the full image as input, while our method uses only a small local crop around the target object. The depth of the crop is estimated for input to our system. Figure 7 illustrates one example result. Note that the rendering result with our estimated light matches well to the reference (the highlight should be at the left side of the bunny). More comparisons can be found in the supplementary material.

To compare with methods based on low-order spherical harmonics (SH) [12], we fit the ground-truth light with 3- and 5-order SH and compare the light and rendering results with our method. As shown in Figure 5, a 5-order SH (as used by [12]) is incapable of representing small area light sources and generates blurred shadow; our method estimates a full environment map and produces consistent shadow in the rendering.

6.3 Ablation studies

We conduct ablation studies to justify the components of our system, including the separation into diffuse and specular reflections, and to verify the effectiveness of the translation networks.

We first test our system without the diffuse or specular translation network, but with the remaining networks unchanged. Empirical comparisons are shown in Figure 9. Without the specular translation network, the system fails to estimate the high-frequency lighting details and produces inconsistent specular highlights in the re-rendering results. Without the diffuse translation network, the system found difficulties estimating light from a diffuse scene.

We then remove the decomposition network and feed the RGBD input directly to the spatial-angular translation networks, followed by the fusion network. It is also possible to train a neural network to regress the angular environment lighting directly from the RGBD input. Such a neural network structure shares a design similar to [14, 26] but is trained on our dataset. As shown in Table 1 (numerical) and Figure 9 (visual), compared to those alternative solutions, our method produces the best quality results by combining both diffuse and specular information with our physics-aware network design. With only RGB input, our method can also predict environment maps with estimated depth, and outperforms existing methods. Larger estimation error is found due to inaccuracy in the depth input, which could be improved by training our network together with the depth estimation network.

Please refer to our supplementary video to see how the recurrent convolution layers increase the temporal coherence of the estimated illumination.

Finally, we captured sequences of indoor scenes with dynamic lighting effects, and continuously changing viewpoints, demonstrating the advantages of having

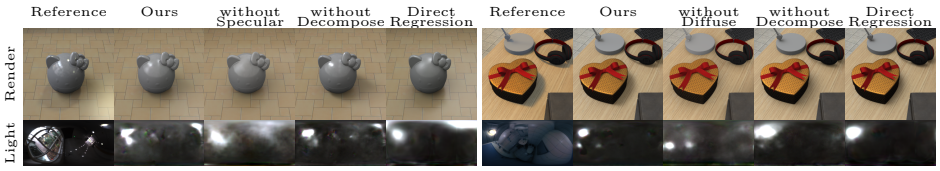

Fig. 9. Ablation study of our physically-based estimation. For a scene containing specular objects (left), our specular network can help to infer high-frequency lighting details from the decomposed specular reflections. For diffuse dominated scenes (right), the diffuse network plays an important role in estimating light from the diffuse shading and shadows. Without having the corresponding network or the decomposition, the system fails to produce accurate estimations. Direct regression also produces blurred light estimation. The inaccurate estimations lead to degraded rendering results, such as missing highlights (left) and shadows (right).

real-time lighting estimation for AR applications. Please refer to the supplementary material for the implementation details and real video sequence results. For all the real video results, we crop the central 384×384 region as the input of our method.

6.4 Performance

We test the runtime performance of our method on a workstation with an Intel 7920x CPU and a single NVidia RTX 2080 Ti GPU. For the full system, the processing time per frame is within 26 ms, of which 14 ms is needed for preparing the input and 12 ms is used for neural network inference. We regard adopting to mobile GPUs as future work.

7 Conclusion

We presented a scheme for realtime environment light estimation from the RGBD appearance of individual objects, rather than from broad scene views. By designing the neural networks to follow physical reflectance laws and infusing rendering knowledge with additional input, our method can robustly estimate environment lighting for scenes with arbitrary objects and various illumination conditions. Our recurrent convolution design also offers temporal and spatial smoothness which is critical for many AR applications.

Although our method supports near-field illumination effects by estimating light at different local regions of the input, a potential improvement would be to estimate a near-field illumination model, combining inferences from multiple different patches, to yield a more robust solution for near-field illumination.

Currently, our method estimates environment light only based on the shading information of objects and does not require the input contains contents of the environment map. Combining scene-based method [26, 35, 10] with our object-based method would be a potential future direction, yielding better quality results.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015), https://www.tensorflow.org/, software available from tensorflow.org
2. Azinovic, D., Li, T.M., Kaplanyan, A., Niessner, M.: Inverse path tracing for joint material and lighting estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
3. Barron, J.T., Malik, J.: Intrinsic Scene Properties from a Single RGB-D Image. In: CVPR. pp. 17–24. IEEE (jun 2013). https://doi.org/10.1109/CVPR.2013.10
4. Barron, J.T., Malik, J.: Shape, Illumination, and Reflectance from Shading. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**(8), 1670–1687 (aug 2015), http://ieeexplore.ieee.org/document/6975182/
5. Calian, D.A., Lalonde, J.F., Gotardo, P., Simon, T., Matthews, I., Mitchell, K.: From Faces to Outdoor Light Probes. Computer Graphics Forum (2018)
6. Chaitanya, C.R.A., Kaplanyan, A.S., Schied, C., Salvi, M., Lefohn, A., Nowrouzezahrai, D., Aila, T.: Interactive reconstruction of monte carlo image sequences using a recurrent denoising autoencoder. ACM Trans. Graph. **36**(4), 98:1–98:12 (Jul 2017)
7. Cheng, D., Shi, J., Chen, Y., Deng, X., Zhang, X.: Learning Scene Illumination by Pairwise Photos from Rear and Front Mobile Cameras. Computer Graphics Forum (2018)
8. Choi, S., Zhou, Q.Y., Miller, S., Koltun, V.: A large dataset of object scans. arXiv:1602.02481 (2016)
9. Debevec, P.: Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques. pp. 189–198. SIGGRAPH '98, ACM, New York, NY, USA (1998)
10. Gardner, M.A., Hold-Geoffroy, Y., Sunkavalli, K., Gagne, C., Lalonde, J.F.: Deep parametric indoor lighting estimation. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
11. 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 Transactions on Graphics **36**(6), 1–14 (nov 2017)
12. 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) (June 2019)
13. Georgoulis, S., Rematas, K., Ritschel, T., Fritz, M., Tuytelaars, T., Gool, L.V.: What is around the camera? In: ICCV (2017)
14. Georgoulis, S., Rematas, K., Ritschel, T., Gavves, E., Fritz, M., Gool, L.V., Tuytelaars, T.: Reflectance and natural illumination from single-material specular objects using deep learning. PAMI (2017)
15. Gruber, L., Langlotz, T., Sen, P., Höherer, T., Schmalstieg, D.: Efficient and robust radiance transfer for probeless photorealistic augmented reality. In: 2014 IEEE Virtual Reality (VR). pp. 15–20 (March 2014)

16. Gruber, L., Richter-Trummer, T., Schmalstieg, D.: Real-time photometric registration from arbitrary geometry. In: 2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). pp. 119–128 (Nov 2012)
17. Hold-Geoffroy, Y., Athawale, A., Lalonde, J.F.: Deep sky modeling for single image outdoor lighting estimation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
18. Hold-Geoffroy, Y., Sunkavalli, K., Hadap, S., Gambaretto, E., Lalonde, J.F.: Deep outdoor illumination estimation. In: IEEE International Conference on Computer Vision and Pattern Recognition (2017)
19. Huang, Y., Wang, W., Wang, L.: Bidirectional recurrent convolutional networks for multi-frame super-resolution. In: Advances in Neural Information Processing Systems 28 (2015)
20. Jiddi, S., Robert, P., Marchand, E.: Illumination estimation using cast shadows for realistic augmented reality applications. In: 2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct). pp. 192–193 (Oct 2017)
21. Karsch, K., Sunkavalli, K., Hadap, S., Carr, N., Jin, H., Fonte, R., Sittig, M., Forsyth, D.: Automatic scene inference for 3d object compositing. ACM Trans. Graph. **33**(3), 32:1–32:15 (Jun 2014)
22. Khan, E.A., Reinhard, E., Fleming, R.W., Bülthoff, H.H.: Image-based material editing. ACM Trans. Graph. **25**(3), 654–663 (jul 2006). https://doi.org/10.1145/1141911.1141937, http://doi.acm.org/10.1145/1141911.1141937
23. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. In: ICLR (may 2015)
24. Kronander, J., Banterle, F., Gardner, A., Miandji, E., Unger, J.: Photorealistic rendering of mixed reality scenes. Comput. Graph. Forum **34**(2), 643–665 (May 2015)
25. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. 3DV (2016)
26. LeGendre, C., Ma, W., Fyffe, G., Flynn, J., Charbonnel, L., Busch, J., Debevec, P.E.: Deeplight: Learning illumination for unconstrained mobile mixed reality. In: CVPR (2019)
27. LeGendre, C., Yu, X., Liu, D., Busch, J., Jones, A., Pattanaik, S., Debevec, P.: Practical multispectral lighting reproduction. ACM Trans. Graph. **35**(4), 32:1–32:11 (Jul 2016)
28. Li, Z., Snavely, N.: Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In: European Conference on Computer Vision (ECCV) (2018)
29. Lombardi, S., Nishino, K.: Reflectance and Illumination Recovery in the Wild. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(1), 129–141 (jan 2016). https://doi.org/10.1109/TPAMI.2015.2430318
30. Nishino, K., Nayar, S.K.: Eyes for relighting. ACM Trans. Graph. **23**(3), 704–711 (Aug 2004)
31. Romeiro, F., Zickler, T.: Blind reflectometry. In: Proceedings of the 11th European Conference on Computer Vision: Part I. pp. 45–58. ECCV'10, Springer-Verlag, Berlin, Heidelberg (2010)
32. Sato, I., Sato, Y., Ikeuchi, K.: Illumination from shadows. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(3), 290–300 (March 2003)
33. Sengupta, S., Gu, J., Kim, K., Liu, G., Jacobs, D.W., Kautz, J.: Neural inverse rendering of an indoor scene from a single image. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 8598–8607 (2019)

34. Sengupta, S., Kanazawa, A., Castillo, C.D., Jacobs, D.W.: Sfsnet: Learning shape, reflectance and illuminance of faces 'in the wild'. In: CVPR (2018)
35. Song, S., Funkhouser, T.: Neural illumination: Lighting prediction for indoor environments. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
36. Sun, T., Barron, J.T., Tsai, Y.T., Xu, Z., Yu, X., Fyffe, G., Rhemann, C., Busch, J., Debevec, P., Ramamoorthi, R.: Single image portrait relighting. ACM Trans. Graph. **38** (2019)
37. Tewari, A., Zollhöfer, M., Garrido, P., Bernard, F., Kim, H., Pérez, P., Theobalt, C.: Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In: CVPR (2018)
38. Tewari, A., Zollöfer, M., Kim, H., Garrido, P., Bernard, F., Perez, P., Christian, T.: MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction. In: ICCV (2017)
39. Unger, J., Gustavson, S., Ynnerman, A.: Densely sampled light probe sequences for spatially variant image based lighting. In: Proceedings of GRAPHITE 06 (2006)
40. Waese, J., Debevec, P.: A real-time high dynamic range light probe. In: Proceedings of the 27th annual conference on Computer graphics and interactive techniques: Conference Abstracts and Applications (2002)
41. 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)
42. Wu, C., Wilburn, B., Matsushita, Y., Theobalt, C.: High-quality Shape from Multi-view Stereo and Shading under General Illumination. In: CVPR (2011)
43. Yi, R., Zhu, C., Tan, P., Lin, S.: Faces as lighting probes via unsupervised deep highlight extraction. In: ECCV (2018)
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) (June 2019)
45. Zhou, H., Sun, J., Yacoob, Y., Jacobs, D.W.: Label denoising adversarial network (ldan) for inverse lighting of faces. In: CVPR (2018)