Reconstructing Reflection Maps using a Stacked-CNN for Mixed Reality Rendering

Andrew Chalmers, Junhong Zhao, Daniel Medeiros, and Taehyun Rhee, *Members, IEEE*

Abstract—Corresponding lighting and reflectance between real and virtual objects is important for spatial presence in augmented and mixed reality (AR and MR) applications. We present a method to reconstruct real-world environmental lighting, encoded as a reflection map (RM), from a conventional photograph. To achieve this, we propose a stacked convolutional neural network (SCNN) that predicts high dynamic range (HDR) 360° RMs with varying roughness from a limited field of view, low dynamic range photograph. The SCNN is progressively trained from high to low roughness to predict RMs at varying roughness levels, where each roughness level corresponds to a virtual object's roughness (from diffuse to glossy) for rendering. The predicted RM provides high-fidelity rendering of virtual objects to match with the background photograph. We illustrate the use of our method with indoor and outdoor scenes trained on separate indoor/outdoor SCNNs showing plausible rendering and composition of virtual objects in AR/MR. We show that our method has improved quality over previous methods with a comparative user study and error metrics.

Index Terms—Light estimation, reflection map, environment map, image-based lighting, deep learning, mixed reality

1 Introduction

Augmented and mixed reality (AR and MR) is becoming ubiquitous for both mobile and tethered use. In this field, the real world blends with the virtual, which can be observed through a mobile display or dedicated device such as a head-mounted display. One of the core challenges in AR/MR is the seamless blending of virtual objects such that they appear to be physically situated in the real world environment. This plays an important role for users to feel spatially present with the inserted object. Scene understanding and inverse rendering is often used to solve this problem, which involves digitizing real world elements (e.g., lighting, material models, geometry, etc.) from a photograph. These elements are then used to render virtual objects such that they seamlessly blend into the photograph. Of particular interest in this paper is predicting the environmental lighting conditions, encoded as a reflection map (RM), which exists outside the field of view of a conventional photograph.

The environmental lighting is often stored as an environment map (EM) - a 360° high dynamic range (HDR) image used for illuminating and compositing virtual objects into photographs for film and mixed reality. Capturing an EM involves taking photographs on site, in multiple directions with varying exposure levels, and stitching them together to form the HDR 360° texture. While the EM can be used for lighting directly, a common technique in real-time graphics (including AR/MR) is to encode the EM as reflectance. This is done by pre-convolving the EM with a material, and storing the convolved result in another 360° texture, referred to as an RM [1]. The RM is then used as a lookup table to efficiently render virtual objects. Multiple RMs can be encoded to accommodate varying material roughness (specular, glossy, and diffuse). Predicting RMs covering 360° using a conventional photograph is challenging as it requires hallucinating environmental details from a limited field of view while also estimating the dynamic range required for HDR lighting.

Recent methods have used deep learning to predict parametric [2][3][4] and non-parametric [5][6][7][8] EMs from a photograph. The parametric models are able to model outdoor day time lighting or diffuse indoor lighting, but have limited ability to produce natural reflections for low roughness materials. The non-parametric models are able to regress EMs with textural details for reflections, however we found that predicting a non-parametric EM is challenging to model directly and is prone to error.

Instead, we propose a novel stacked convolutional neural network (SCNN) structure that progressively predicts HDR 360° RMs from a standard limited field of view, low dynamic range (LDR) photograph (Figure 1). Each CNN in the stack minimizes the error of a specific material reflectance *roughness* level (including high roughness diffuse to low roughness glossy). This allows for the CNN to accurately predict RMs corresponding to the roughness level of the virtual object being rendered. The stack is progressively trained from high to low roughness. The progressive approach allows the higher accuracy of the high roughness RM prediction to propagate through to lower roughness networks, improving the prediction accuracy across all roughness levels. We show that predicting the RMs performs better than predicting the EM with a comparative user study as well as error metrics against recent EM prediction methods [4][8]. Since we are able to predict the RM directly, this also contributes towards real-time rendering in AR/MR [9].

To the best of our knowledge, this is the first time that lighting, encoded as RMs, has been predicted from conventional photographs with no exemplar object or expected known geometry. The main contributions of our paper are summarized as follows:

- A progressively trained SCNN that predicts HDR RMs from a conventional, limited field of view, LDR photograph.
- We predict RMs from each CNN in the stack, which then can be used directly for rendering virtual objects with the

• *A. Chalmers, J. Zhao, D. Medeiros, and T. Rhee are with Computational Media Innovation Centre (CMIC), Victoria University of Wellington, New Zealand.*
E-mail: andrew.chalmers@ecs.vuw.ac.nz, taehyun.rhee@ecs.vuw.ac.nz

Manuscript received February 20, 2020; revised June 2, 2020.

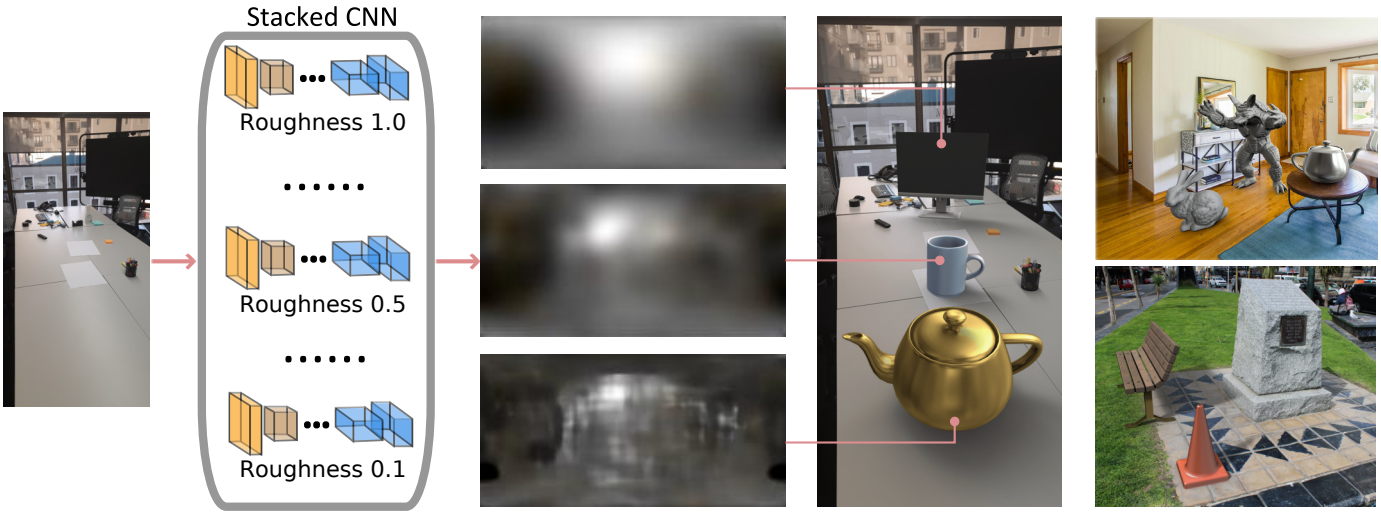


Fig. 1: Left to right: an input photograph taken by a conventional camera, the trained SCNN, the predicted 360° reflection maps (RMs) from three roughness levels in the SCNN, rendering and composition using the predicted RMs, and two more examples of an indoor and outdoor scene. Top-right virtual objects: teapot, bunny, and armadillo. Bottom-right virtual objects: park bench and traffic cone.

corresponding roughness value. We implement a server-client system, where the server manages the RM prediction, rendering, and composition, and the client handles the display and user interface.

- We evaluate our SCNN with both indoor/outdoor scenes with a comparative user study and error metrics, showing that our method outperforms the previous work.

2 Related Work

Inverse lighting is the fundamental technique underpinning the research aims of this paper. Our literature survey focuses on environmental lighting, and predicting environment maps from photographs.

2.1 Environmental Lighting for AR/MR

AR/MR rendering involves lighting and composition of virtual objects into photographs. A conventional, limited field of view, LDR photograph on its own does not provide environmental lighting information required for high quality illumination and reflections. Debevec et al. [10] proposed to address this by storing the surrounding environment into a 360° HDR EM. Image-based lighting (IBL) is able to use EMs to realistically render virtual objects such that they match the illumination of the real-world environment. Differential rendering [11] is then used to composite virtual objects, including their shadows and colour bleeding, into photographs. While this process is computationally expensive, various optimisations can be made to accommodate for real-time graphics (required for AR/MR). Environment maps can be pre-convolved and stored as RMs by pre-computing diffuse and glossy reflections and storing the result in a 360° HDR texture [1]. The RMs are then used as a lookup table for efficient rendering. Rhee et al. [9][12] used light detection in combination with differential rendering and reflection mapping to achieve real-time AR/MR rendering. Mobile friendly implementations capture RGB-D images by moving the mobile camera around while storing the pixels into an environment map [13][14]. Alternatively, a client/server setup where multiple cameras stream the environmental lighting to a mobile device [14]. While EMs or RMs provide realistic real-world lighting in offline and real-time computer graphics, they are often not available with a single conventional RGB photograph.

2.2 Inverse Lighting

The process of inverse lighting involves working backwards from a photograph to the lighting that induced the illumination conditions in the photograph [15]. Going from a conventional limited field-of-view photograph to a full panoramic image is a challenging problem as it requires hallucinating details that are not observable within the photograph. Early work created an EM from a photograph by stretching the photograph into a 360° texture [16]. Since the photograph is LDR, it also requires inverse tone-mapping [17] to recover the HDR light information. Karsch et al. [18] estimated HDR EMs by using texture information in the photograph and finding a texturally similar EM from a database. While these methods are able to recover HDR EMs, they are highly prone to error as they do not use illumination cues. To recover high quality HDR EMs from a conventional photograph, a specular chrome ball [1] can be used. When a chrome ball is not present in the photograph, alternate illumination cues (e.g., shadows, highlights) within the photograph need to be used instead.

Prior work has used machine learning approaches that use a foreground object exhibiting unknown reflectance as input to estimate the environmental lighting [19], the object's appearance [20][21], or decomposes it into reflectance and illumination [22]. Lalonde et al. [23] and Liu et al. [24] were able to recover outdoor lighting using a combination of illumination cues within the photograph. Outdoor lighting was later improved [3][4] using convolutional neural networks (CNNs) by estimating parameterized outdoor EMs [25]. While the parameterized model provides sufficient directional lighting data, it does not contain textured reflection information of the surrounding environment, such as buildings or trees. This was partially addressed by considering weather conditions [6]. Gardner et al. [8] were able to predict indoor EMs using a CNN by training on a large LDR and HDR EM dataset. They used spatial warping to increase the size of the dataset and to also account for spatially varying

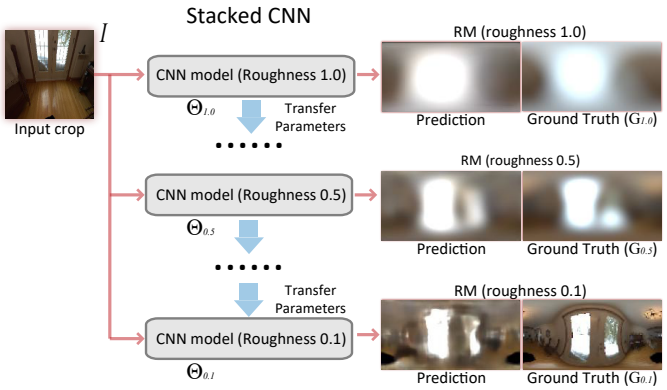


Fig. 2: Overview of Stacked CNN architecture.

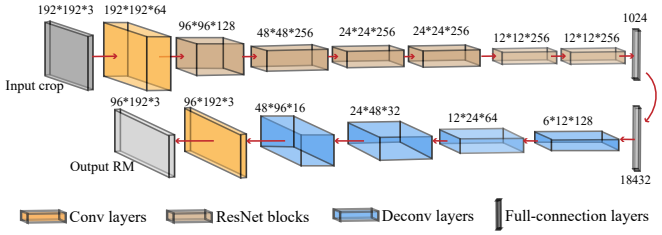


Fig. 3: Overview of a single CNN architecture.

lighting within indoor scenes. This work also used a progressive learning technique, though it is focused on the EM's high frequency lighting, as opposed to the environmental lighting convolved with a material. Indoor lighting was then extended to predicting spherical harmonics for efficient rendering [2] but was limited in producing high frequency information. Song et al. [7] hallucinated complete high resolution texture information for indoor EMs while also considering lighting conditions, enabling mirror-like reflections with a natural appearance. Legendre et al. [5] trained on both indoor and outdoor data using LDR video. By inserting three lighting calibration balls (specular, glossy, diffuse), they were able to infer HDR information in the training process. Our goal is similar to these methods, except we focus on obtaining the reflections directly by predicting RMs instead of EMs. This idea is motivated by Ramamoorthi et al. [26], which showed that high frequency light information is dependent on the available material properties in the photograph. By predicting the roughness levels directly, we are able to ensure the highest prediction quality possible for a given roughness level. Furthermore, we take advantage of using RMs with a progressive structure to improve the quality.

3 Progressively Learning Reflection Maps

The objective of our method is to recover HDR RMs from a limited field of view LDR photograph. Previous approaches predict the EM, then convolve it to the specified material roughness during the rendering process. Instead, we progressively predict RMs using a stack of multiple CNNs (SCNN), moving from high to low roughness predictions (Figure 2). This follows the general idea of curriculum learning [27][28], which is the process of solving easier problems first and gradually increasing the difficulty. To this end, we demonstrate that recovering a high roughness RM directly produces more accurate results than predicting the EM, which is then mathematically convolved. This supports the idea that predicting a low roughness RM is an easier problem to solve, motivating a curriculum learning strategy. We refer to Hacohen and Weinshall [29] which provides an empirical investigation and theoretical analysis behind this strategy. This idea also provides the SCNN with more data (rather than learning from just the EM, but also the corresponding RMs).

The progressive SCNN setup has three distinct advantages. First, since we train our CNNs directly for each roughness level, the network is able to minimize the error for each roughness level, providing higher accuracy results than predicting the EM which is then convolved. Second, the progressive structure allows subsequent CNNs in the stack to benefit from the improved accuracy from the previous higher roughness CNN, thus propagating the improved accuracy throughout stack. Third, the predicted RM for each CNN can be used directly for efficient rendering with accurate reflectance for each roughness level. In summary, directly predicting RMs, as opposed to predicting an EM which is later convolved, produces higher accuracy results and requires less computation time in the rendering process.

We formulate our learning process in Equation 1, where I is the input limited field of view, low dynamic range photograph, $G_{HDR,r}$ is the 360° HDR RM having the roughness level r, and Θ_r is the parameter set of the single CNN for the roughness level r. We optimize $\{\Theta_r\}$ for the whole SCNN in a progressive way, from the highest roughness value to the lowest roughness value. Here, Δr is the step between neighboring roughness levels. To optimize the parameters Θ_r for a single CNN, the parameters from its upper roughness level $\Theta_{r+\Delta r}$ are used to initialize the mapping g from I to a predicted HDR RM, which is implemented by a single CNN model (Figure 3)

$$\Theta_r^* = \arg\min_{\Theta_r} \mathscr{L}(G_{HDR,r};\ g((I,\Theta_{r+\Delta r});\Theta_r)), \tag{1}$$

where $\mathscr{L}$ is the loss function defined in equation 4. The SCNN architecture is able to train with less difficulty on high roughness RMs and propagate the improvements through to lower roughness RMs.

3.1 Progressively Trained Stacked CNNs

Our SCNN consists of a series of CNNs in which a higher roughness network is stacked upon its neighboring lower roughness network (Figure 2). The RMs at varying roughness levels are intrinsically related to one another, where higher roughness RMs contain information about the ambient tones and directionality of the light sources, which gradually converge to more specific tones and directions at lower roughness levels.

Taking the dependencies among RMs with varying roughness into consideration, we design our SCNN to train using a reflection model which has a roughness parameter in the range $[0,1]$. In our tests we use the Phong reflection model [30], where the specular power exponent is $\frac{2}{\alpha^2} - 1$, and $\alpha = roughness^2$ [31]. This remapping ensures that the specular exponent is at least 1. When computing RMs, the view vector is the same as the normal, as such the above is equivalent to Lambertian diffuse when $roughness = 1.0$. Note that other reflection models could be used in place of the Phong model, as long as it supports a way of progressively changing its roughness (e.g., GGX [32]). The SCNN trains from the highest roughness level ($roughness = 1.0$), as we found it is easier to learn the low frequency lighting and textures. After the higher roughness model has trained, we transfer its network parameters to

the adjacent lower roughness network. Then the following lower roughness network fine-tunes the initialized model toward higher frequency RM. This allows the lower roughness model to focus on additional high frequency information rather than modeling the data from scratch. We continue the progressive pattern until the lowest roughness level (roughness=0.1). After training each CNN in the stack, each CNN can be used separately to predict RMs at different roughness levels. We refer to this training scheme as "progressive", while the networks which train without the progressive parameter transfer as "isolated". Our experiments show that the SCNN has better accuracy than isolated CNNs across all roughness levels.

We could theoretically model our SCNN to predict any arbitrary roughness level with a roughness value in the range $[0,1]$. But in practice, we have a finite memory footprint, so we generate a discrete number of models. In real-time graphics, interpolating RMs only requires a few levels (e.g., six was used by Rhee et al. [9]), where continuous roughness values are supported by interpolating between the discrete levels. In this paper, we experimented with a roughness step $\Delta r = 0.1$, from 1.0 to 0.1. We tested that these 10 models are sufficient for smooth interpolation, though this could be optimized further with fewer levels.

3.2 Network Architecture for each CNN

For each CNN in the stacked architecture, the input is a limited field of view image. In out tests, we experimented with two different resolutions (width×height): 192×192, and 135×240 portrait image to match the input from mobile devices). After going through a series of convolutional layers, ResNet blocks and a full-connection layer, the input is encoded into a 1024-dimensional latent vector which serves as the compressed lighting feature vector. The decoder takes this latent vector as the input, upsampling these features using several deconvolution layers and expanding them into the RM.

The sizes and depths of the feature maps for each layer is shown, for a single CNN architecture, in Figure 3. A stride step of 2 is used to downsample/upsample the feature maps where the neighboring layers have different sizes. For the encoder, the first layer is a standard full-conv (in order to distinguish between "convolution" used for both the RM and the CNN operation, we use "conv" for the latter) layer, with a kernel size of 7 and a filter number of 64. The following 12 layers have residual connections [33] with a kernel size of 5 for the first residual block and a kernel size of 3 for the other 5 residual blocks. For the decoder, the encoder output will first go through one full-connection layer, and through 4 deconvolution layers followed by one conv layer with 3 channels to generate the final RM with a resolution of 192×96. The ELU activation function [34] and batch normalization [35] are used on all encoder and decoder layers.

3.3 Data Preparation

To generate input-output pairs for training, a photograph is paired with an RM that is filtered to a roughness level. To do this, photographs are extracted from a 360° panoramic image in 8 different directions, uniformly distributed along the azimuth angle (horizon line). The zenith angle is chosen randomly between 90° and 135°, which is biased toward looking toward the floor - a typical angle for mixed reality applications where objects are composited onto the ground. For each of the 8 photographs, we rotate the panorama such that the corresponding extracted photograph is centred, removing any complexity in rotational variation during the training process. Finally, to generate the RMs, the centred panorama is filtered at 10 different roughness levels. This produces a set of 8 photographs, where each photograph has 10 corresponding RMs. Our network structure can support any material model which is able to convolve from low to high frequency reflectance.

We apply our networks to both indoor and outdoor datasets separately. This is due to the large differences in lighting between the two scene types. These include differences in positional and directional lighting, contrast, as well as different tones. Some previous work train on one or the other (e.g., [8], and [4]), however, we show that our network structure can model both scene types. To obtain indoor data, we use the Laval HDR indoor dataset using a warping operator on the environment map to account for indoor spatial variation of the lighting [8]. For outdoor data, we found there was limited HDR data. As such, we use the low dynamic range (LDR) SUN360 dataset [36] and convert it to HDR using a CNN based inverse tone mapping operator [37]. In total, we have $17,000$ samples for our indoor dataset and $24,600$ samples for our outdoor dataset.

3.4 Training Details

To train each CNN model the HDR data needs to be normalized to a reasonable range. The loss function also needs to be specifically designed for the equirectangular format which we use for the RM.

Dynamic range normalization: Due to the high variation between bright and dark values in HDR data, normalization is required for efficient training. The normalization needs special consideration to avoid crushing small values to 0 while reducing large values. We normalize into exponent-space

$$\mathbf{G}^*_{\mathbf{HDR}} = \alpha(\mathbf{G}_{\mathbf{HDR}})^{\beta} \tag{2}$$

where $\mathbf{G}_{\mathbf{HDR}}$ is the ground truth HDR RM that we aim to model. Here we use $\alpha = 0.2$ and $\beta = 2.2$ to compress the HDR into a reasonable interval to help network modeling. The inverse of Equation 2 is

$$\mathbf{P}_{\mathbf{HDR}} = \frac{1}{\alpha}(\mathbf{P}_{\mathbf{HDR}}{}^*)^{\frac{1}{\beta}} \tag{3}$$

which will be used to recover the predicted RMs $\mathbf{P}_{\mathbf{HDR}}{}^*$ into full HDR.

Loss Function: The predicted RM, P_{HDR}, is in equirectangular format, where each pixel should have different weights to account for the distortion towards the poles. Thus, similar to [38], we train the CNN to minimize a solid angle weighted loss function. For both indoor and outdoor scenes, we use the L2 distance on the pixel output:

$$\mathscr{L}(\mathbf{G}_{\mathbf{HDR}}{}^*, \mathbf{P}_{\mathbf{HDR}}{}^*) = \frac{1}{N}\sum_{1}^{N} \|\mathbf{w} \odot (\mathbf{G}_{\mathbf{HDR}}{}^* - \mathbf{P}_{\mathbf{HDR}}{}^*)\|_2 \tag{4}$$

where $\mathbf{w}$ is the solid angle matrix and $\odot$ represent pixel-wise multiplication.

Training schedules: For both the indoor and outdoor datasets, we randomly split the entire dataset for the training and testing sets respectively. To train the CNNs, we use the ADAM optimizer [39] with a mini-batch size of 64. The learning rate starts at 0.01 and decays exponentially by a factor of 0.998 in every 30 steps. The process typically converges in around 5 epochs using this dataset. During fine-tuning, we initialize all the trainable parameters in both encoder and decoder by corresponding parameters from prior models. The training time was 4 hours per CNN using a NVidia GeForce GTX 1080 Ti GPU which has 16GB of memory.

4 AR/MR Rendering using Predicted Reflection Maps

Once the SCNN is trained, we are able to use it to generate RMs. In this section, we detail the how to use the predicted RMs for rendering and composition, followed by system specific implementation details.

4.1 Rendering and Composition

Three RMs are required for rendering and composition: one to produce shadows, and another for the diffuse and specular component (Figure 4). A set of RMs is generated from the SCNN (e.g., using all 10 roughness levels, or a subset for optimisation). Then during the rendering process, a virtual object is assigned a RM for the specular component using its roughness parameter. If the roughness parameter is between two roughness levels, we interpolate between the two as commonly done in mipmapping [40]. The specular value is obtained by looking up the pixel value in the selected RM using the reflection vector. The diffuse value is obtained by sampling the highest roughness (roughness= 1.0) RM using the normal vector. Finally, the lowest roughness (roughness= 0.1) RM is used for shadow casting. The virtual object and its shadow is composited into the photograph using differential rendering [11]. We assume geometry is provided for shadow casting (e.g., using ARCore or ARKit's real-time plane detection). To render the shadows, this can either be done offline using ray tracing for static images, or optimised using importance sampling or image processing [9] for real-time rendering.

4.2 Implementation

We propose two ways to utilize our method for AR/MR: a server-client or system-deployed setup. We used the server-client setup in our tests. This removes the processing time and memory overhead from the client, running the SCNN and rendering engine on a remote server. We used a mobile phone as the client, and a desktop machine with an NVidia GeForce GTX 1080 Ti GPU running an instance of TensorFlow [41] as the server. The phone, using its back facing camera, transmits a photograph to the server. The server crops and resizes the photograph to match the input dimensions to the SCNN, which then predicts 10 RMs. The client also sends touch input data to manipulate an object's position/rotation and material roughness with a slider. The server, using the touch data and predicted RMs, renders and composites the virtual objects into the photograph, and transmits the result back to the phone. The phone then displays the photograph with the composited virtual objects to the user on the phone. The data-flow diagram is shown in Figure 5. In our experiments, the SCNN took 0.2 seconds per layer in the stack. The network latency was approximately 2 seconds. Since we are using reflection maps, the rendering and composition time is negligible, and could be moved onto the client side. The total time from client to server and back to client is approximately 3 seconds. While transmission is not real-time, we did not find it was necessary in most use cases where the lighting does not fluctuate greatly between frames. Improvements to network latency and reducing stack layers will move our method towards real-time. Note that RMs across all roughness levels are transmitted together, so users can change the material's roughness in real-time. For the system-deployed setup, the server processes are moved to the client as well. This will introduce overhead in memory usage from the SCNN, where a single CNN in the stack uses 240MB. This is feasible on mobile devices if fewer levels in the stack are used.

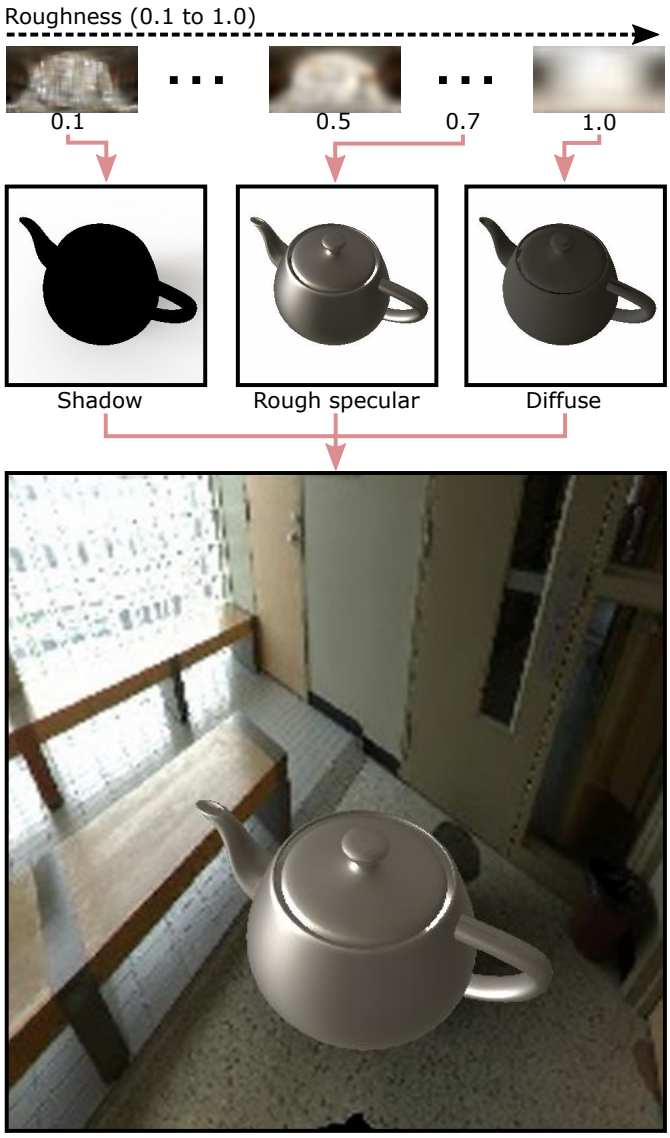


Fig. 4: Rendering and composition using the reflection maps generated from the SCNN. The shadow is generated using the roughness=0.1 RM, the diffuse component uses the roughness=1.0 RM, and the specular component uses its roughness parameter, in this case 0.7, to select a RM in the range 0.1 to 1.0.

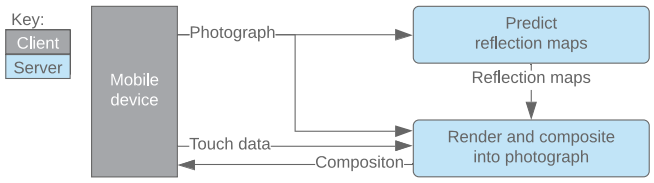


Fig. 5: Server-client data-flow diagram.

5 Results

In this section, we evaluate our results with a comparative user study and error metrics against previous light estimation methods for both indoor and outdoor scenes. This is followed by analysis and discussion of the results. Both the user study and error metric take into consideration various qualities of environmental lighting, such as ambient tones and high frequency light sources.

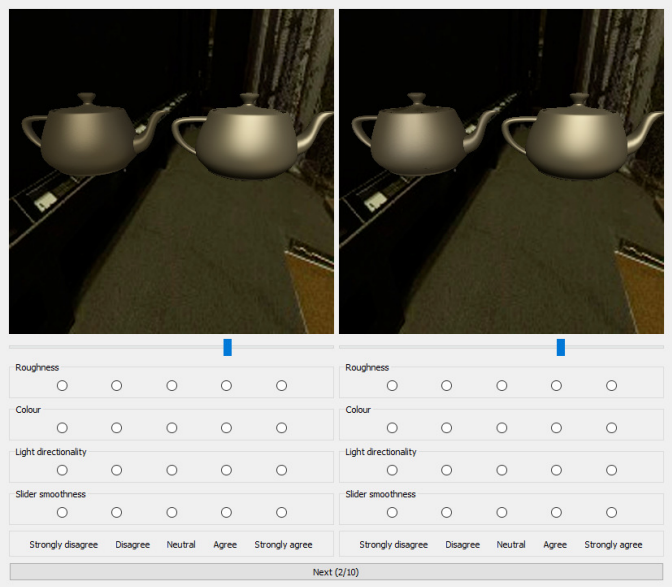


Fig. 6: The user study interface we used to compare predicting the RM directly against predicting the EM which is then convolved. Within each image, the left teapot uses a prediction, the right uses the ground truth as a reference. We can also observe how, in this case, the RM (right image, left teapot) better matches the reference than using the EM (left image, left teapot).

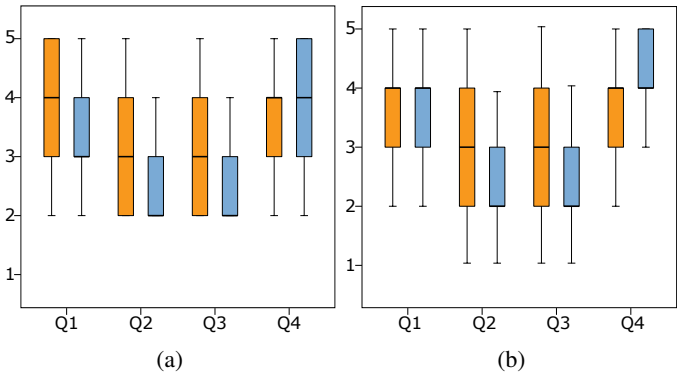


Fig. 7: Boxplot summarizing the results for the four questions in the user study regarding indoor (left) and outdoor (right) images. Orange uses the RM prediction, blue uses EM prediction which is then convolved.

	Q1	Q2	Q3	Q4
Indoor	Z=-6.454 **p<0.001**	Z=-8.919 **p<0.001**	Z=-4.805 **p<0.001**	Z=-6.063 **p<0.001**
Outdoor	Z=-2.288 **p=0.022**	Z=-3.931 **p<0.001**	Z=-3.931 **p<0.001**	Z=-3.580 **p<0.001**

TABLE 1: Statistical results for both indoor and outdoor images. p values in bold indicate statistical significance.

5.1 User Study

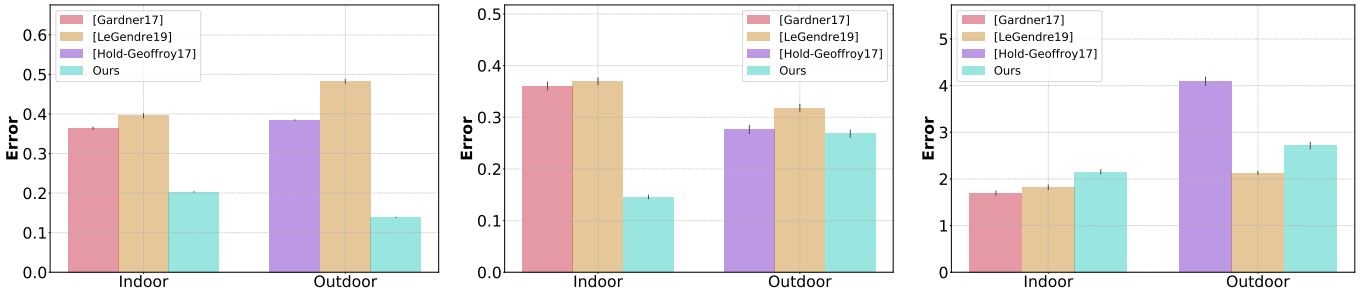
The goal of this user study was to compare the subjective visual quality of our method of predicting RMs at multiple roughness levels directly against previous methods [3][4][5][6][7][8] that predict the EM and then convolve it to a roughness level (or multiple roughness levels). The hypothesis is that predicting the RM at a specific roughness level yields visually higher quality results than predicting the EM which is then convolved to the specific roughness level. To evaluate this with a fair comparison, we use our same network structure to predict the EM (roughness=0.1), and convolve it to 10 different roughness levels. Then we use our SCNN to generate the 10 RMs to compare against.

To evaluate the visual quality we implemented a user interface (Figure 6) that presented two images side-by-side, each of which contain two virtual teapots. One of the images uses the predicted RM, the other uses the convolved EM. The order of methods (left, right) was randomized during the study to avoid bias. For both images, the teapot on the right is the reference teapot using the ground truth EM, and the teapot on the left uses a predicted RM or convolved EM. To test our hypothesis, we render a ground truth virtual object (teapot) at a high roughness level (roughness=0.7), and have users use a continuous roughness slider to change the prediction to visually match the ground truth as close as possible based on how similar the roughness of the surfaces appear. Since there are 10 discrete roughness levels, we interpolate between levels based on the continuous roughness slider.

Once they had matched the prediction to the reference we asked the participants to answer four 5-Likert scale questions. The first three questions evaluated the visual quality of our method regarding: 1. roughness, 2. colour, and 3. light directionality. The specific statements are as follows: 1. *The material's roughness matches the reference well*, 2. *The colour matches the reference well*, and 3. *The light directionality matches the reference well*. One possible limitation of our method is that we are predicting 10 discrete roughness levels which do not ensure any consistency between levels. As such, we ask a fourth question, 4. slider smoothness, with the specific statement: 4. *When moving the roughness slider, it smoothly changes the material's roughness*. All four questions were scored on the following scale: *1. Strongly disagree, 2. Disagree, 3. Neutral, 4. Agree, and 5. Strongly agree* (Note: The user interface in Figure 6 contains an abbreviation of the question for convenience, while the participants had the full question on separate sheet). Finally, we also included a fifth metric to evaluate how close the user set the roughness slider value to the reference's roughness value. Given that even the lowest roughness RM is already blurred, the hypothesis is that the user would set the roughness slider below the reference's roughness value.

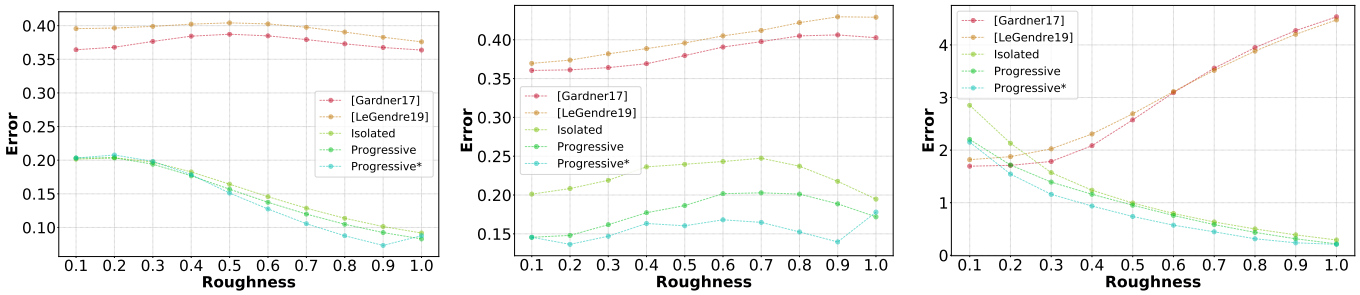
We performed the user study with 10 images for both indoor and outdoor scenes (20 images total), each under different lighting conditions (e.g., clear sky, overcast, complex interiors, dark areas, etc.). We had a total of 36 participants (18 for indoor, 18 for outdoor). The participants are students and staff from the university, and were recruited through our university mailing list. The task was conducted using a Dell Monitor (Dell 24 inch SE2419HR), to ensure similar colour calibration for all participants.

To evaluate the proposed metrics we conducted a statistical analysis of the results. We used the Wilcoxon signed ranks test to indicate statistical significance in both indoor and outdoor data. For the distance variable, because of its continuous characteristic, we conducted a Shapiro-Wilk test to test for data normality. Since both data inputs from the indoor and outdoor follow a normal distribution, we also conducted a Wilcoxon Signed ranks test for these variables. The result from the statistical analysis is summarised in Figure 7 and Table 1. The detailed analysis is in Section 5.3.



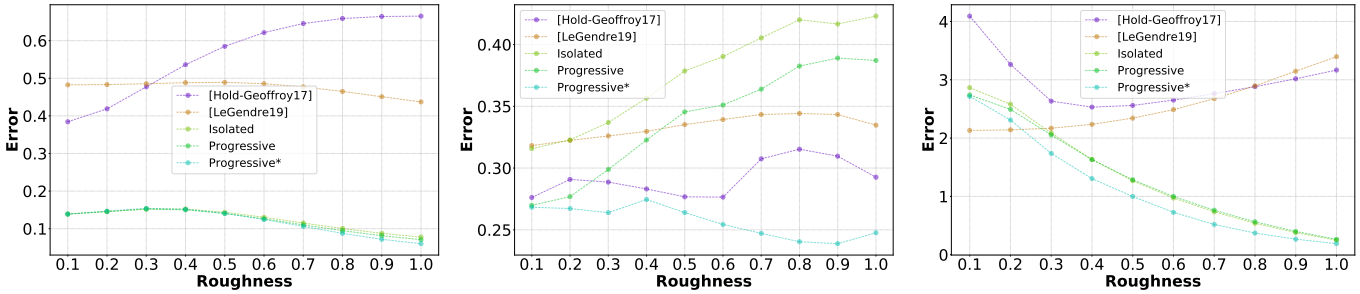
(a) Ambient Lighting. (b) Directional Lighting. (c) Dynamic Range.

Fig. 8: Comparison between our low roughness RM predictions (roughness=0.1) and EM predictions from prior work.



(a) Ambient lighting. (b) Directional lighting. (c) Dynamic range.

Fig. 9: Indoor comparison across all roughness levels.



(a) Ambient lighting. (b) Directional lighting. (c) Dynamic range.

Fig. 10: Outdoor comparison across all roughness levels.

5.2 Metric Evaluation

We also evaluate the accuracy of our RM predictions with three metrics based on ambient lighting, directional lighting, and dynamic range. We compare our indoor results with LeGendre et al. [5] and Gardner et al. [8] and outdoor results with LeGendre et al. [5] and Hold-Geoffroy et al. [4]. We compute these metrics on the test dataset, which has 198 indoor and 198 outdoor RMs. We also do this for each of the 10 roughness levels for a total of 3,960 comparisons. For a fair comparison, we retrain our SCNN with a small FOV input image when comparing against LeGendre et al. and a wide FOV image when comparing against Gardner et al. and Hold-Geoffroy et al. Figure 8 shows the lowest roughness (environment map) results for both indoor and outdoor scenes. We can observe how progressive learning improves the lowest roughness level. Figure 9 and 10 show the results across all roughness levels for indoor and outdoor scenes respectively. Accuracy is improved across all levels in most cases. The detailed analysis is in Section 5.3. We calculate the error between the ground truth RM and the prediction using the metrics, and aggregate the result across the test data set. The metrics are defined as follows:

Ambient lighting: We use the root mean squared logarithmic error (RMSLE) to measure the low frequency environmental colour and tone of the scene. The logarithmic scale places more emphasis on the ambient lighting rather than bright light sources.

Directional lighting: We measure the angular accuracy of the high frequency lights in the RM by first detecting the directions of the high frequency light sources [9] in the RM. We then compute the average angular error between the nearest pairs of lights in the prediction and ground truth.

Dynamic range: This is the relationship between high and low frequency lighting, which impacts how bright glossy highlights or how dark shadows appear in the rendered scene. We use the following to compute contrast:

$$\frac{\max(\mathrm{RM}) - \mu}{\mu}, \tag{5}$$

where max returns the maximum intensity value and μ is the

Fig. 11: Indoor, scene 1.

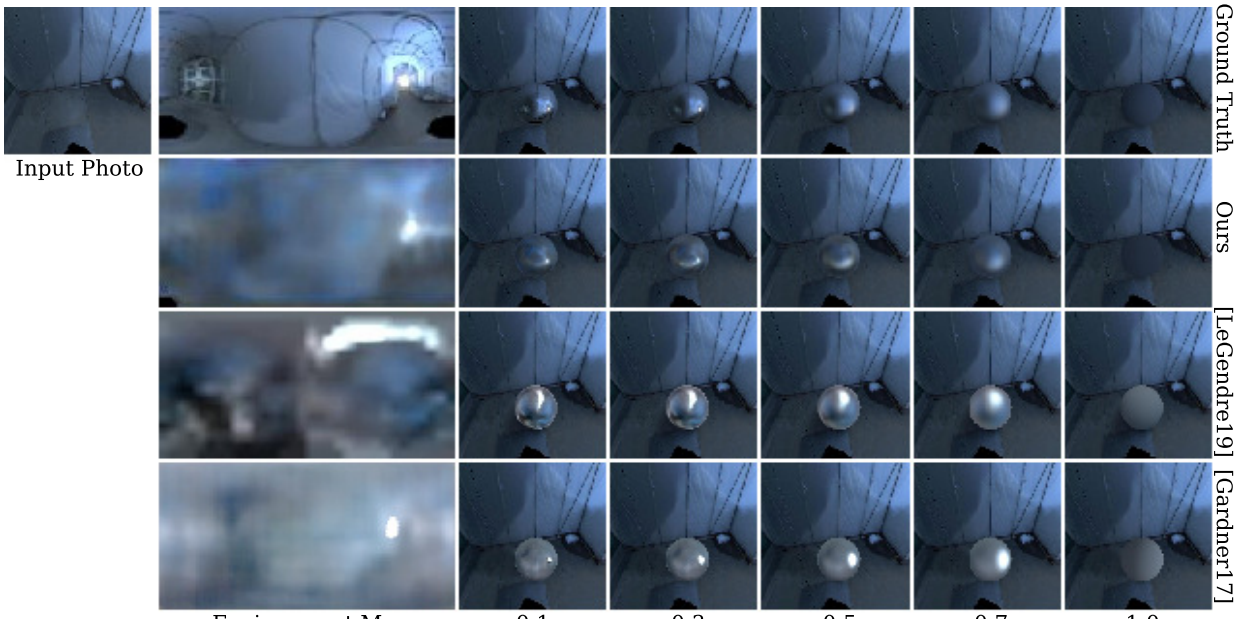


Fig. 12: Indoor, scene 2.

mean value in the RM. We use the log of this equation to follow perceptual uniformity of brightness. We take the squared difference between contrast measurements when computing a distance between two RMs.

For all three metrics, the error is computed using the ground truth EM which is mathematically convolved to a RM, against each model's predicted EM which is also mathematically convolved in all cases except one, which is our progressive method (annotated with a "*"). In this case, we use our specific roughness level in our SCNN to predict the RM directly.

Qualitative results: We also show qualitative results with objects rendered at various roughness levels for indoor scenes in Figure 11, 12, and 13, as well as outdoor scenes in Figure 14, 15, and 16. Across each example we can observe improved overall lighting and blending quality with the background photograph. Note: The input photo is used as a background for compositing into, however our model as well as LeGendre et al.'s used a smaller FOV with a 16:9 aspect ratio (portrait).

5.3 Analysis and Discussion

Through both the error metric and user study evaluation, we show that there is improved accuracy when predicting the RM directly ("progressive*") as opposed to predicting the EM and convolving afterward ("progressive" and the other approaches).

The results of the user study are summarised in Figure 7a and Figure 7b for indoor and outdoor scenes respectively. All of these results have statistical significance, as summarised in Table 1. Statistical analysis found that the RM method was the best in terms of visual quality regarding roughness, colour, and light directionality for both indoor and outdoor scenes. In terms

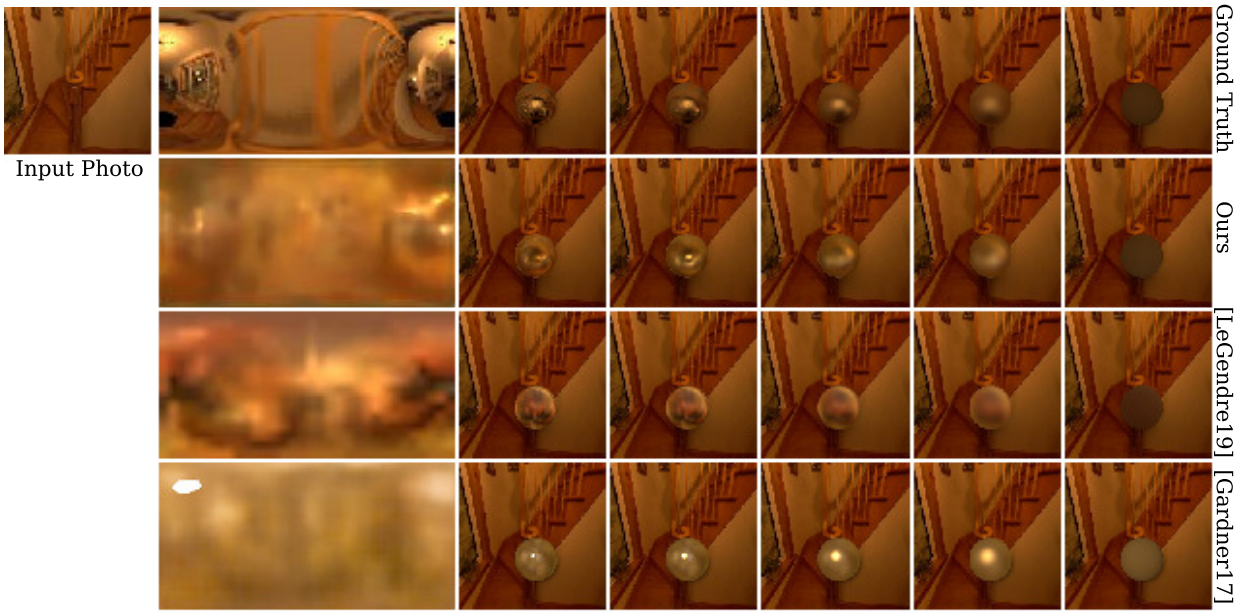


Fig. 13: Indoor, scene 3.

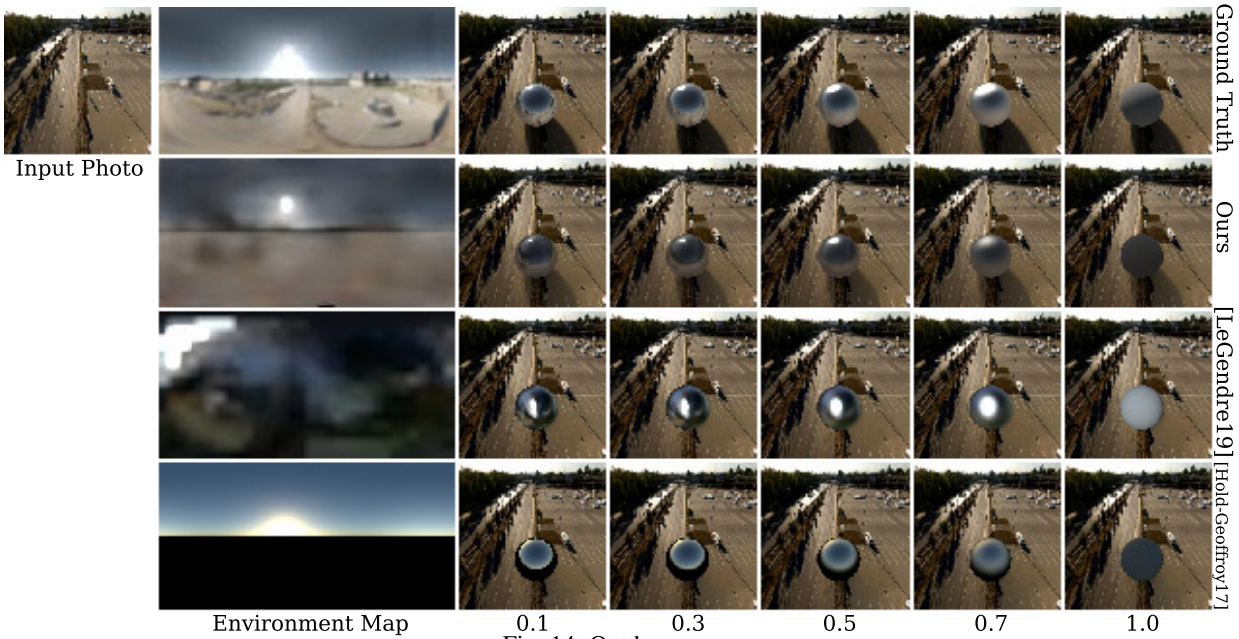


Fig. 14: Outdoor, sunny.

of slider smoothness, we expected to find the RM to be worse since the EM method is mathematically convolved, though we were interested in how much worse. We found that generally the RM method was sufficiently smooth enough, with the median score above the neutral line in both indoor ($Z=-6.3$ $p<0.001$) and outdoor conditions ($Z=-8.551$ $p<0.001$). Interestingly, the EM method did not score perfect results - which is effectively a measurement of the error introduced by interpolating between discrete levels. For the roughness slider distance, we found that users undershot the slider value for both the RM and EM method. The RM was below the reference's roughness values by 0.073889 and 0.059636 on average for indoor and outdoor respectively, and for the EM it was below the reference by 0.121667 and 0.055758 on average for indoor and outdoor respectively.

The error metric evaluation is summarised in Figure 8, 9, and 10. Figure 8a shows that our progressive structure only slightly improves ambient lighting for both indoor and outdoor scenes. We can also observe slight improvements across all roughness levels in Figure 9a and Figure 10a. Figure 8b shows that our method is able to predict light directions more accurately. Figure 9b and Figure 10b show that the RMs benefits from the progressive training (note that the y-axis is normalized, where an angular error of 180° is mapped to 1.0). The progressive nature of our SCNN is able to find strong candidate directional light sources at the high roughness levels, which then propagate through to the low roughness network. This is particularly important when using the low roughness RMs for shadow casting. Finally, Figure 8c shows the dynamic range results, where our method is not as accurate as prior work. Hold-Geoffroy et al. [4] is able to accomplish this as they use a sky model which can reliably place a sun into the prediction. Gardner et

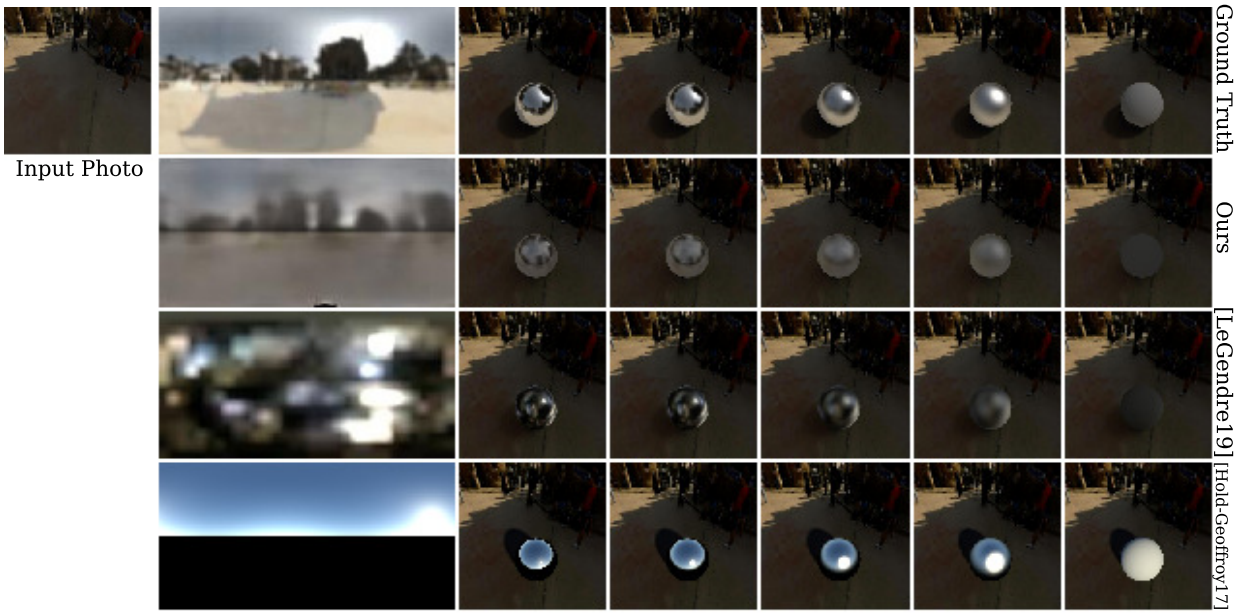


Fig. 15: Outdoor, in shadow.

Fig. 16: Outdoor, overcast.

Fig. 17: Failure cases where our model predicts more than one light where there should only be one (the sun).

al. [8] on the other hand use two CNNs in a two-stage scheme. One focusing on predicting light locations from LDR data combined with a light detector, and the other focusing on light intensity using HDR data to fine-tune the pre-localized light sources.

6 Conclusion

We have presented a SCNN structure for recovering HDR RMs from LDR limited field of view indoor or outdoor photographs. The SCNN is able to produce RMs with varying material roughness. Using a progressive training scheme from high to low roughness, the SCNN is able to obtain higher accuracy predictions. This enables high quality rendering of virtual objects when compositing into photographs. Using the SCNN, we are able to query CNNs at each roughness level, allowing for efficient rendering using RMs, without the need for convolving environment maps at rendering time.

Limitations and Future Work: While our RM predictions improve upon the state of the art, we found that there are still more improvements to be made. The SCNN is comprised of 10

CNN layers, which has a high memory overhead. Future work could consider using fewer roughness layers, or smaller network models. We also found that the overall accuracy in this field should be improved, where the directional metric shows the average angular error is over 15°, which is an angle that would produce obvious incorrect shadow angles and highlights. The contrast of our lowest roughness prediction is not as optimal as previous approaches (e.g., Figure 13), 0.1 roughness, where Gardner et al. is able to obtain a clear specular highlight), which indicates that a hybrid approach using our RM prediction and light detection could be considered. In this paper, we used the Phong reflectance model, which could trivially be changed to the GGX model since there is a corresponding roughness parameter. Future work could consider extending this further to take into account more complex models, such as layered or anisotropic materials. While we produce ambient tones that are plausible for most material types, pure specular reflections will produce obvious artifacts as there is no high resolution detail. Our SCNN produces RMs with various low and high frequency peaks, however, outdoor lighting usually contains only a singular light source (the sun), whereas our results can on occasion produce multiple lights (Figure 17). We tested our system with a server-client setup, however to deploy on mobile devices, optimisations such fewer layers in the SCNN and using MobileNetV2 [42] could be considered for future work.

REFERENCES

[1] G. Miller, "Illumination and reflection maps: Simulated objects in simulated and real environments," *ACM SIGGRAPH'84 course notes*, 1984.

[2] M. Garon, K. Sunkavalli, S. Hadap, N. Carr, and J.-F. Lalonde, "Fast spatially-varying indoor lighting estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6908–6917.

[3] Y. Hold-Geoffroy, A. Athawale, and J.-F. Lalonde, "Deep sky modeling for single image outdoor lighting estimation," *arXiv preprint arXiv:1905.03897*, 2019.

[4] Y. Hold-Geoffroy, K. Sunkavalli, S. Hadap, E. Gambaretto, and J.-F. Lalonde, "Deep outdoor illumination estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7312–7321.

[5] C. LeGendre, W.-C. Ma, G. Fyffe, J. Flynn, L. Charbonnel, J. Busch, and P. Debevec, "Deeplight: Learning illumination for unconstrained mobile mixed reality," *arXiv preprint arXiv:1904.01175*, 2019.

[6] J. Zhang, K. Sunkavalli, Y. Hold-Geoffroy, S. Hadap, J. Eisenman, and J.-F. Lalonde, "All-weather deep outdoor lighting estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 158–10 166.

[7] S. Song and T. Funkhouser, "Neural illumination: Lighting prediction for indoor environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6918–6926.

[8] 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 Transactions on Graphics (SIGGRAPH Asia)*, vol. 9, no. 4, 2017.

[9] T. Rhee, L. Petikam, B. Allen, and A. Chalmers, "Mr360: Mixed reality rendering for 360 panoramic videos," *IEEE Transactions on Visualization & Computer Graphics*, no. 4, pp. 1379–1388, 2017.

[10] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '97. ACM Press/Addison-Wesley Publishing Co., 1997, pp. 369–378.

[11] P. Debevec, "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*, ser. SIGGRAPH '98. ACM, 1998, pp. 189–198.

[12] T. Rhee, S. Thompson, D. Medeiros, R. Dos Anjos, and A. Chalmers, "Augmented virtual teleportation for high-fidelity telecollaboration." *IEEE Transactions on Visualization & Computer Graphics*, vol. 26, no. 5, pp. 1923–1933, 2020.

[13] K. Rohmer, J. Jendersie, and T. Grosch, "Natural environment illumination: Coherent interactive augmented reality for mobile and non-mobile devices," *IEEE transactions on visualization and computer graphics*, vol. 23, no. 11, pp. 2474–2484, 2017.

[14] K. Rohmer, W. Büschel, R. Dachselt, and T. Grosch, "Interactive near-field illumination for photorealistic augmented reality with varying materials on mobile devices," *IEEE transactions on visualization and computer graphics*, vol. 21, no. 12, pp. 1349–1362, 2015.

[15] S. R. Marschner and D. P. Greenberg, *Inverse rendering for computer graphics*. Citeseer, 1998.

[16] E. A. Khan, E. Reinhard, R. W. Fleming, and H. H. Bülthoff, "Image-based material editing," in *ACM Transactions on Graphics (TOG)*, vol. 25, no. 3. ACM, 2006, pp. 654–663.

[17] H. Landis, "Production-ready global illumination," *Siggraph course notes*, vol. 16, no. 2002, p. 11, 2002.

[18] K. Karsch, K. Sunkavalli, S. Hadap, N. Carr, H. Jin, R. Fonte, M. Sittig, and D. Forsyth, "Automatic scene inference for 3d object compositing," *ACM Trans. Graph.*, vol. 33, no. 3, June 2014.

[19] S. Georgoulis, K. Rematas, T. Ritschel, M. Fritz, T. Tuytelaars, and L. Van Gool, "What is around the camera?" in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5170–5178.

[20] K. Rematas, T. Ritschel, M. Fritz, E. Gavves, and T. Tuytelaars, "Deep reflectance maps," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4508–4516.

[21] M. Maximov, L. Leal-Taixe, M. Fritz, and T. Ritschel, "Deep appearance maps," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 8729–8738.

[22] 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 Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 8, pp. 1932–1947, 2017.

[23] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan, "Estimating natural illumination from a single outdoor image," in *IEEE International Conference on Computer Vision*, 2009.

[24] Y. Liu, X. Qin, S. Xu, E. Nakamae, and Q. Peng, "Light source estimation of outdoor scenes for mixed reality," *The Visual Computer*, vol. 25, no. 5-7, pp. 637–646, 2009.

[25] L. Hosek and A. Wilkie, "An analytic model for full spectral sky-dome radiance," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, p. 95, 2012.

[26] R. Ramamoorthi and P. Hanrahan, "On the relationship between radiance and irradiance: determining the illumination from images of a convex lambertian object," *JOSA A*, vol. 18, no. 10, pp. 2448–2459, 2001.

[27] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, no. 1, pp. 71–99, 1993.

[28] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 41–48.

[29] G. Hacohen and D. Weinshall, "On the power of curriculum learning in training deep networks," *arXiv preprint arXiv:1904.03626*, 2019.

[30] B. T. Phong, "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no. 6, pp. 311–317, 1975.

[31] B. Burley and W. D. A. Studios, "Physically-based shading at disney," in *ACM SIGGRAPH*, vol. 2012, 2012, pp. 1–7.

[32] B. Walter, S. R. Marschner, H. Li, and K. E. Torrance, "Microfacet models for refraction through rough surfaces," in *Proceedings of the 18th Eurographics conference on Rendering Techniques*. Eurographics Association, 2007, pp. 195–206.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[34] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[36] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba, "Recognizing scene viewpoint using panoramic place representation," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2695–2702.

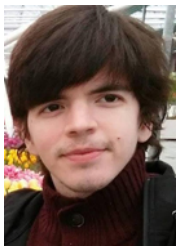
[37] J. Zhang and J.-F. Lalonde, "Learning high dynamic range from outdoor panoramas," in *IEEE International Conference on Computer Vision*, 2017.

[38] H. Weber, D. Prévost, and J.-F. Lalonde, "Learning to estimate indoor lighting from 3d objects," *arXiv preprint arXiv:1806.03994*, 2018.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[40] L. Williams, "Pyramidal parametrics," in *Proceedings of the 10th Annual conference on Computer graphics and interactive techniques (ACM Siggraph)*, vol. 17, no. 3. ACM, 1983, pp. 1–11.
[41] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 2016, pp. 265–283.
[42] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

Andrew Chalmers completed his PhD at Victoria University of Wellington in 2017, advised by Associate Professor Taehyun Rhee and Associate Professor John Lewis, and is currently conducting research at the Computational Media Innovation Centre (CMIC). His interests are a combination of global illumination and computer vision and is a co-founder of Wellington-based startup DreamFlux. Previously, he has taught computer graphics and computer science courses at Victoria University of Wellington and has been credited in feature films for his research with Weta Digital.

Junhong Zhao works as a postdoctoral researcher at the Computational Media Innovation Centre of Victoria University of Wellington, New Zealand. She completed her doctoral degree in 2015 at the Institute of Electronics, Chinese Academy of Sciences (CAS). She finished her Ph.D. project at the Human-Computer Speech Interaction Lab of Tsinghua University, where she researched on computer-assisted language learning using machine learning-based speech recognition and computer graphics techniques (2011-2015). She then worked at CAS' Institution of Information Engineering as an assistant researcher (2015-2017). She is now contributing her background in machine learning and computer graphics to the CMIC's research projects.



Daniel Medeiros is a postdoctoral research fellow at the Computational Media Innovation Centre of Victoria University of Wellington, New Zealand. Daniel holds a PhD in Computer Engineering from Instituto Superior Técnico at University of Lisbon, an MSc in Informatics from PUC-Rio/Brazil and a BSc in Computer Science from UFPB/Brazil. His current interests lie in virtual reality, human-computer interaction, collaboration and 3D user interfaces.

Taehyun James (TJ) Rhee is an associate professor at Victoria University of Wellington, New Zealand, a director of the Computational Medial Innovation Centre (CMIC), and a founder/director of the Victoria Computer Graphics Research Lab. Before joining Victoria in 2012, he was a principal researcher in the Mixed Reality Group, Future IT Centre at Samsung (2008-2012). He was also a senior researcher and researcher of Research Innovation Center at Samsung Electronics (1996-2003). His current research activities are focused on cinematic real-time lighting and rendering, mixed reality, immersive visualization and interaction, and human digital content interaction.