Free-form Scanning of Non-planar Appearance with Neural Trace Photography

XIAOHE MA, KAIZHANG KANG, RUISHENG ZHU, and HONGZHI WU, State Key Lab of CAD&CG, Zhejiang University, China
KUN ZHOU, State Key Lab of CAD&CG, Zhejiang University and ZJU-FaceUnity Joint Lab of Intelligent Graphics, China

Fig. 1. Using a light-weight device consisting of a single camera and an RGB LED array as shown on the left, we acquire the complex, non-planar appearance from photographs at unstructured views with optimized per-LED intensities, for a pre-captured 3D shape. The rendering results of the collection of captured appearance with novel view and illumination conditions are shown on the right.

We propose neural trace photography, a novel framework to automatically learn high-quality scanning of non-planar, complex anisotropic appearance. Our key insight is that free-form appearance scanning can be cast as a geometry learning problem on unstructured point clouds, each of which represents an image measurement and the corresponding acquisition condition. Based on this connection, we carefully design a neural network, to jointly optimize the lighting conditions to be used in acquisition, as well as the spatially independent reconstruction of reflectance from corresponding measurements. Our framework is not tied to a specific setup, and can adapt to various factors in a data-driven manner. We demonstrate the effectiveness of our framework on a number of physical objects with a wide variation in appearance. The objects are captured with a light-weight mobile device, consisting of a single camera and an RGB LED array. We also generalize the framework to other common types of light sources, including a point, a linear and an area light.

CCS Concepts: • **Computing methodologies** → **Reflectance modeling**.

Additional Key Words and Phrases: illumination multiplexing, optimal lighting pattern, SVBRDF

*Corresponding author: Hongzhi Wu (hwu@acm.org).
Authors' addresses: Xiaohe Ma; Kaizhang Kang; Ruisheng Zhu; Hongzhi Wu, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, 310058, China; Kun Zhou, State Key Lab of CAD&CG, Zhejiang University and ZJU-FaceUnity Joint Lab of Intelligent Graphics, Hangzhou, 310058, China.

ACM Reference Format:
Xiaohe Ma, Kaizhang Kang, Ruisheng Zhu, Hongzhi Wu, and Kun Zhou. 2021. Free-form Scanning of Non-planar Appearance with Neural Trace Photography. *ACM Trans. Graph.* 40, 4, Article 124 (August 2021), 13 pages. https://doi.org/10.1145/3450626.3459679

1 INTRODUCTION

Digitization of real-world objects is one central problem in computer graphics and vision. Represented as a 3D mesh and a 6D Spatially-Varying Bidirectional Reflectance Distribution Function (SVBRDF), a digitized object can be rendered to realistically reproduce the original look with any view and lighting conditions. It has important applications in fields like cultural heritage, e-commerce, computer games and movie production.

While high-precision geometry can be conveniently captured with a commercial mobile 3D scanner nowadays [Artec 2021; Shining3D 2021], it is desirable to develop a light-weight device to conduct free-form appearance scanning for the following reasons. First, such a device is scalable to scan objects of different sizes, as long as the camera pose can be reliably estimated. Second, the mobility makes it possible to perform on-site capture, in which case the objects, e.g, precious artifacts, are not allowed to be transported. In addition, the short time and low cost to build a light-weight device makes it accessible to a wider audience. Last but not least, it offers a user-friendly experience similar to the commonly practiced geometry scanning.

In spite of the surging demand, it remains an open problem to efficiently scan non-planar appearance. At one hand, the majority of existing work on mobile reflectance scan takes photographs with a single point/directional light on, corresponding to a low sampling efficiency in the 4D domain of view and lighting directions. Priors are required to trade the spatial resolution for the angular accuracy [Nam et al. 2018; Riviere et al. 2016]. On the other hand, illumination multiplexing devices like lightstages can acquire complex appearance with high performance, by independently controlling a large number of light sources [Kang et al. 2019; Tunwattanapong et al. 2013]. However, related techniques strongly exploit the fixed view condition(s) while the illumination changes. It is not clear how to extend them to a mobile device, with unstructured, constantly varying views, and an incomplete coverage over the lighting domain owing to its small form factor.

In this paper, we observe that general free-form scanning of non-planar reflectance can be formulated as *trace photography*, originally designed for fixed-view reflectance estimation [Dong et al. 2014; Gardner et al. 2003; Ren et al. 2011] and shape reconstruction in the presence of complex light transport [Morris and Kutulakos 2007]. Specifically, for each point p on the surface of the physical object, its appearance is captured as image measurements with different view and illumination conditions. For each image measurement, we can concatenate it with the corresponding view and lighting parameters to produce a high-dimensional point. The collection of all such points form the *trace* of p. Now appearance reconstruction is equivalent to mapping each trace to its reflectance representation. While existing work (effectively) designs this mapping by hand, our key insight here is that free-form **appearance scanning** can be cast as a **geometry learning** problem on unstructured point clouds (i.e., trace), making it possible to tackle the challenges in the former field by harnessing the advances in the latter.

Based on the above connection, we develop *neural trace photography*, a novel framework to learn high-quality, free-form scanning of non-planar, anisotropic appearance. Inspired by a highly successful geometry learning technique [Qi et al. 2017], we propose a network architecture to predict the pixel-independent reflectance, by efficiently aggregating and transforming the information from a variable-length trace, taken from unstructured views with optimized lighting during free-form scanning. The input/output of the network are carefully designed, to predict accurate anisotropic appearance, which is invariant to the imprecise local frame in the input view specification. Our framework is not tied to a specific setup. It can adapt to various factors, including the geometry of the device, the number/type of light sources and the properties of appearance, in a data-driven manner.

To validate our framework in practice, we build a light-weight mobile reflectance scanner, consisting of a camera and an RGB LED array, which has a higher illumination sampling capability than a point source. Neural trace photography is applied to optimize the intensity of each LED on the array during acquisition, as well as to reconstruct non-planar reflectance from corresponding measurements at unstructured views. Its effectiveness is demonstrated on a number of physical objects with a wide variation in appearance. We validate our reconstructions against photographs, as well as the counterpart captured with a high-end lightstage [Kang et al. 2019]. The results are also compared against one state-of-the-art technique on mobile scanning of non-planar appearance [Nam et al. 2018]. Finally, extensive experiments are conducted to study the impact of various factors over the reconstruction quality. Notably, we generalize the framework to other common types of light sources, such a point, a linear and an area light, and compare with our setup in experiments.

2 RELATED WORK

Here we primarily review previous work on high-quality SVBRDF capture with *active illumination*. The approaches that handle uncontrolled lighting [Dong et al. 2014; Wu et al. 2016a; Zhou et al. 2016] are beyond the scope of this paper. Interested readers are directed to excellent surveys on acquisition techniques [Dong 2019; Guarnera et al. 2016; Weinmann and Klein 2015; Weyrich et al. 2009]. In general, densely sampling the 4D view-illumination domain with a camera produces high-quality results in manually designed [Dana et al. 1999; Lawrence et al. 2006] or even learned appearance representations [Gao et al. 2020]. But it is prohibitively time-consuming, as usually a camera and a point light need to be mechanically positioned to a large number of direction pairs. Significant research efforts has been made to improve the efficiency. Below we categorize them based on whether the captured view(s) are fixed or unstructured.

2.1 Fixed View(s)

This class of approaches fix a single or multiple views during acquisition, and typically recover appearance from the image variations with respect to different illumination conditions. We further divide them based on the type of the light source.

2.1.1 Point Light(s). Nearly flat appearance can be estimated from a single fixed view and a sparse number of lighting directions. Dong et al. [2010] compute a microfacet-based SVBRDF, from two sets of sparse measurements that focus on sampling in the angular/spatial domain, respectively. Aittala et al. exploit the structural similarity to estimate a stationary SVBRDF, from a flash-/non-flash-lit pair of images [2015], or a single flash image [2016]. In [Li et al. 2017], the reflectance is estimated from one photograph under unknown natural illumination, using a self-augmentation training process. Deschaintre et al. [2018] train a neural network on a large dataset for appearance modeling from one flash image.

2.1.2 Illumination Multiplexing. When multiple (effective) lights are available, they can be independently programmed in the temporal domain to encode the physical appearance into fixed-view measurements, which are then computationally decoded as the digital results. As more appearance information is packed into each measurement, it is possible to efficiently handle more challenging cases like anisotropic reflectance, as well as to perform pixel-independent reconstructions, leading to higher-quality results [Tunwattanapong et al. 2013].

A linear light source is first employed in [Gardner et al. 2003] to scan over a planar, isotropic sample. Temporal per-pixel measurements are mapped to BRDF parameters via precomputed look-up

tables. Chen et al. [2014] extend the idea to capture anisotropic appearance in a low-rank space, by spatially modulating the lighting intensities. Ren et al. [2011] allow irregular motions of the linear source, with the help of pre-calibrated physical BRDF patches that are imaged alongside with the sample.

A lightstage allows the programming of hundreds to tens of thousands of lights in parallel, resulting in high-precision reconstructions. Complex lighting patterns, such as gradient illumination [Ghosh et al. 2009] or spherical harmonics [Tunwattanapong et al. 2013], are manually derived for acquisition. In a similar spirit, Aittala et al. [2013] treat a near-field LCD as a programmable source, to capture isotropic reflectance with patterns derived from a frequency domain analysis. Recently, mixed-domain networks are proposed, to jointly and automatically optimize physical lighting patterns along with the computational reconstruction. Efficient capture of pixel-independent anisotropic reflectance is achieved on planar samples [Kang et al. 2018] and non-planar ones [Kang et al. 2019].

It is non-trivial to apply the work in this category to our case, for two main reasons. First, the majority of work is based on a single view. It is not clear how to extend to efficiently aggregate information across different unstructured views. Second, illumination multiplexing usually requires a fixed view when projecting multiple patterns. This condition is rarely met in free-form scanning.

2.2 Unstructured Views

This class of methods deal with free-form scanning input. In the majority of related work, images are captured with a camera and a point/directional light at a time, which corresponds a point sample in the 4D domain of lighting and view directions. Due to this inherent low sampling efficiency, various forms of priors have been proposed to regularize the reconstruction.

2.2.1 Traditional Priors. Lensch et al. [2003] model the appearance as a linear combination of basis materials, to constrain the reconstruction from a sparse number of flash-lit images. With a similar assumption, Nam et al. [2018] estimate the appearance, normals and 3D geometry in an alternating optimization. Zickler et al. [2005] trade the spatial resolution for the angular accuracy, and compute the reflectance via scattered-data interpolation. A coaxial projector-camera pair is proposed in [Holroyd et al. 2010], with a strong prior imposed on the reflectance to handle highly limited samples. Wu et al. [2015] treat the IR emitter of a Kinect sensor as a point light source, and estimate the glossiness from multi-view observations with a non-linear optimization. Additional IR point lights are employed in [Wu et al. 2016b], with custom circuits to switch one light on at a time. In [Hui et al. 2017], a dictionary-based reflectance prior is proposed, to compute a planar SVBRDF from images acquired by a collocated camera and flash. Riviere et al. [2016] introduce a mobile flash-based method and a fixed-view LCD-based one to estimate isotropic reflectance, followed by a surface detail enhancement algorithm to further improve the resolution.

2.2.2 Deep-Learning-Based Priors. Gao et el. [2019] learn the latent embedding of planar SVBRDFs, to regularize the optimization for appearance reconstruction with respect to an arbitrary number of input images. The idea is extended to non-planar appearance in [Zhang et al. 2020]. Deschaintre et al. [2019] propose a pooling-based network, to aggregate appearance information from 1 to 10 images. A GAN-based framework is introduced in [Guo et al. 2020], to seek the latent representation of planar SVBRDFs, the rendered images of which are optimized with respect to multi-view photographs. Recently, Bi et al. [2020] propose a setup of 6 wide-baseline cameras with collocated point lights. Multi-view flash-lit images are warped to the current view, based on an estimated geometry. These images are then fed to a network to generate appearance maps on a per-view basis, from which per-vertex BRDFs are optimized.

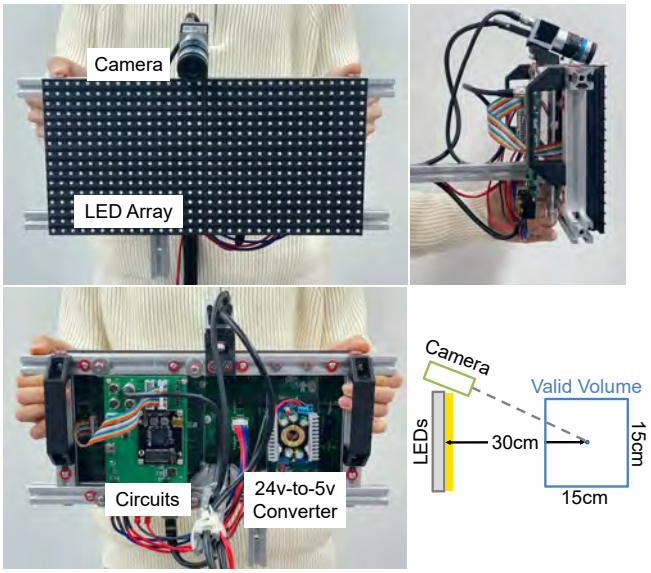

Fig. 2. Our appearance scanner. The left two images show the front-/back-view of the device, and the side-view is on the top right. The bottom-right diagram illustrates the valid volume and its spatial relationship with the scanner from a side-view.

While substantial progress has been made in the past for free-form scanning with a point/directional light, it is not clear how to extend to automatically exploit the higher angular sampling capability of a complex light source. It is also worth mentioning that in previous work like [Deschaintre et al. 2019; Gao et al. 2019], the view conditions are discarded by the neural network. In comparison, our definition of trace incorporates the view information, which is subsequently exploited by the network to produce high-quality appearance.

3 PROTOTYPE SCANNER

As illustrated in Fig. 2, our light-weight scanner is made of a rectangular RGB LED array and a single machine vision camera. The LED array has a size of 32cm×16cm, consisting of 32×16=512 RGB LEDs, with a pitch of 1cm and a maximum total power of 40W. The intensity of each LED is independently controlled, and quantized with 8 bits per channel for implementation via Power Width Modulation (PWM). The camera, a 5MP Basler acA2440-75uc, is mounted on the top edge of the LED array. An exposure time of $\frac{1}{60}$ second is used during acquisition, which is synchronized with the LED array

on the circuit level. A PC communicates with the control circuits via gigabit ethernet, and retrieves captured images via USB3. We define the volume of valid 3D points as a box of 15cm×15cm×15cm, whose center is 30cm in front of the center of the LED array. The camera is pointed towards the center of the volume.

The intrinsic/extrinsic parameters of the camera, as well as the positions, orientations, angular intensity and spectral distribution of LEDs, are carefully calibrated. Color calibration is performed with an X-Rite ColorChecker Passport. The scale ambiguity of diffuse/specular albedo is resolved with a planar diffuse patch of a uniform albedo [Gardner et al. 2003]. Please refer to the supplemental material for more details about our scanner and calibration procedures.

4 PRELIMINARIES

4.1 Assumptions

We assume that a sufficiently accurate shape of a physical object is pre-captured using, e.g., an off-the-shelf 3D scanner or a camera with multi-view stereo. Also, the appearance of interest can be described as an SVBRDF. Moreover, we assume a known camera pose for each captured photograph. The lighting intensity for each LED is set as a constant that does not change with time. We discuss the extension to temporally varying intensities in Sec. 8. Similar to the majority of related work, we do not consider global illumination effects such as inter-reflections, which is a promising direction for future research.

For brevity, we interpret the relative motion between the scanner and the object during acquisition as the movement of the object only, while fixing the scanner to a canonical pose.

4.2 Equations

The following equations are based on a gray-scale channel, which can be easily extended to the spectral domain using the color calibration data (detailed in the supplemental material). First, we describe the relationship among the image measurement B from a surface point $\mathbf{p}$, the reflectance f and the intensity I of each LED on the scanner below.

$$B(I, \mathbf{x_p}, \mathbf{n_p}, \mathbf{t_p}) = \sum_l I(l) \int \frac{1}{||\mathbf{x_l} - \mathbf{x_p}||^2} \Psi(\mathbf{x_l}, -\omega_\mathbf{i}) V(\mathbf{x_l}, \mathbf{x_p}) f(\omega_\mathbf{i}{}', \omega_\mathbf{o}{}', \mathbf{p})(\omega_\mathbf{i} \cdot \mathbf{n_p})^+ (-\omega_\mathbf{i} \cdot \mathbf{n_l})^+ d\mathbf{x_l}. \quad (1)$$

Here l is the index of a locally planar light source, and $I(l)$ is its intensity in the range of $[0, 1]$, the collection of which will be referred to as a lighting pattern in this paper. In addition, $\mathbf{x_p}/\mathbf{n_p}/\mathbf{t_p}$ is the position/normal/tangent of $\mathbf{p}$, while $\mathbf{x_l}/\mathbf{n_l}$ is the position/normal of a point on the light whose index is l. We denote $\omega_\mathbf{i}/\omega_\mathbf{o}$ as the lighting/view direction, with $\omega_\mathbf{i} = \frac{\mathbf{x_l} - \mathbf{x_p}}{||\mathbf{x_l} - \mathbf{x_p}||}$. $\Psi(\mathbf{x_l}, \cdot)$ represents the angular distribution of the light intensity. V is a binary visibility function between $\mathbf{x_l}$ and $\mathbf{x_p}$. The operator $(\cdot)^+$ computes the dot product between two vectors, and clamps a negative result to zero. $f(\cdot; \omega_\mathbf{o}{}', \mathbf{p})$ is a 2D BRDF slice, which is a function of the lighting direction. We use the anisotropic GGX model to represent f, but other models can also be employed here. Note that all above points/vectors are defined in the camera space, with the exception of $\omega_\mathbf{i}{}'/\omega_\mathbf{o}{}'$ expressed in the local frame of $\mathbf{p}$.

Next, we define the j-th entry of the trace of $\mathbf{p}$ as a high-dimensional point by concatenating the image measurement, the corresponding position $\mathbf{x_p}$ and the local frame $\mathbf{n_p}/\mathbf{t_p}$ at the j-th view:

$$\text{concat}[B(I, \mathbf{x_p^j}, \mathbf{n_p^j}, \mathbf{t_p^j}), \mathbf{x_p^j}, \mathbf{n_p^j}, \mathbf{t_p^j}]. \quad (2)$$

Here concat is a concatenation operation, which results in a 1+3+3+3 = 10D point. $\mathbf{x_p^j}$, $\mathbf{n_p^j}$ and $\mathbf{t_p^j}$ are expressed in the camera space of the j-th view. The design consideration for the above definition will be detailed in Sec. 6.1. Note that in experiments, we take RGB measurements and thus the trace is a collection of 12D points.

Finally, we follow existing work [Kang et al. 2019; Lensch et al. 2003] to define a lumitexel m as the collection of virtual measurements of the BRDF f at a surface point $\mathbf{p}$, with one light on at a time. It is a function of the index of light l:

$$m(l; \mathbf{p}) = B(\{I(l) = 1, \forall_{k \neq l} I(k) = 0\}, \mathbf{p}). \quad (3)$$

Note that we cannot directly measure m in free-form scanning, since the relative spatial relationship between the scanner and $\mathbf{p}$ may change, as the illuminating LED is switched from one to another.

5 OVERVIEW

To scan the appearance of a physical non-planar object, we first use the prototype scanner to take photographs of the object at different unstructured views with precomputed intensities for each light in the LED array. The lighting intensities are pre-optimized to more efficiently measure the information useful for appearance reconstruction. Next, we estimate the camera pose of each photograph, with the help of the pre-captured 3D geometry. For each point $\mathbf{p}$ on the surface of the object, we assemble its trace from the corresponding pixels in captured images. A neural network then takes as input this variable-length trace, and predicts anisotropic reflectance as a diffuse/specular lumitexel. Finally, the BRDF parameters along with a corresponding local frame are fitted to the network output, and stored in texture maps as the final appearance results. Please refer to Fig. 3 for an illustration of the processing pipeline.

6 OUR NETWORK

In this section, we describe our neural network in details. For each visible point $\mathbf{p}$ on the object surface, it independently predicts reflectance information from the corresponding trace, captured at unstructured views with the optimized lighting pattern. Please refer to Fig. 4 for the architecture of our network.

6.1 Input

The input to our network is a trace as defined in Eq. 2. Unlike existing work that records the measurement B only [Gardner et al. 2003; Ren et al. 2011], we augment B with the corresponding view specification, which consists of the 3D position $\mathbf{x_p^j}$ and the local frame represented by $\mathbf{n_p^j}$ and $\mathbf{t_p^j}$, as an entry in the trace. This additional information describes the spatial relationship between $\mathbf{p}$ and the camera **at the j-th view**, which is necessary for the network to predict complex appearance like anisotropic reflectance. Note that no lighting information is incorporated, as it does not change with the view.

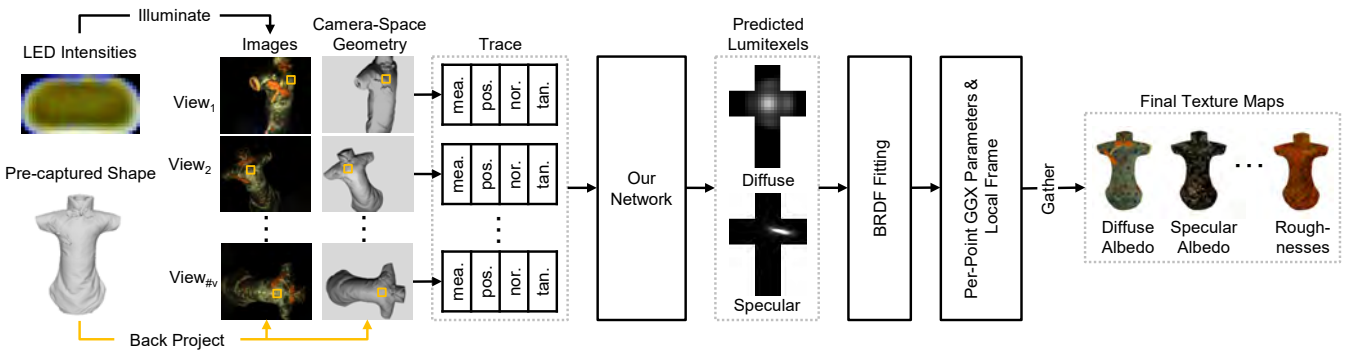

Fig. 3. Our processing pipeline. We first take photographs of the object at different views with pre-optimized intensities for each LED. Next, we estimate the camera pose of each photograph, with the help of the pre-captured 3D geometry. For each point on the surface of the object, we assemble its trace from the corresponding pixel and a consistent local frame in different images. A neural network takes as input this trace, and predicts anisotropic reflectance as a diffuse/specular lumitexel. Finally, the BRDF parameters along with a corresponding local frame are fitted to the network output, which are gathered to generate the final texture maps. Note that mea. = image measurements, pos. = position, nor. = normal and tan. = tangent.

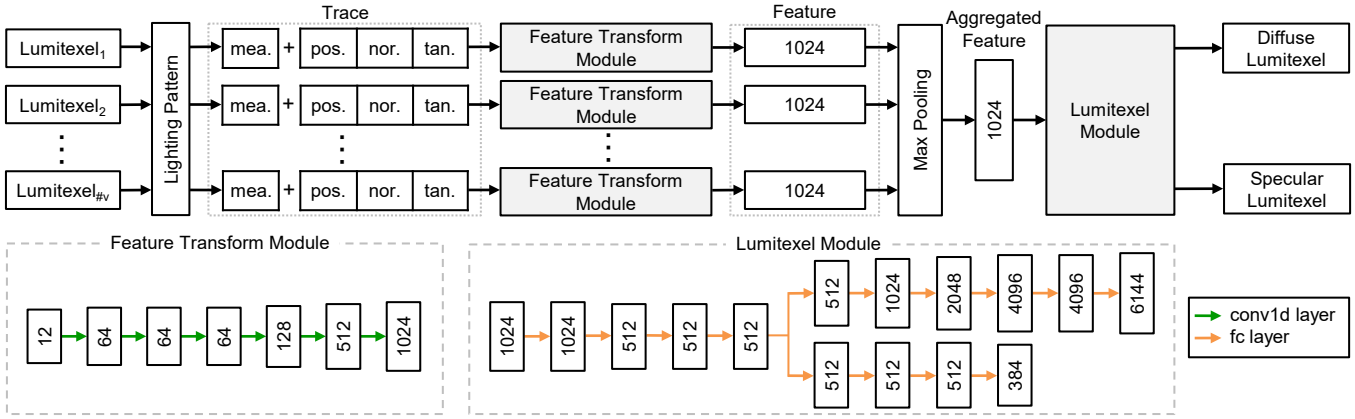

Fig. 4. Network architecture. Our network consists of a feature transform module for processing each element in the trace, a max pooling layer that aggregates the features from all unstructured views, and a subsequent module that produces a diffuse/specular lumitexel. Prior to the feature transform module, we also model the acquisition process, as a dot product between a physical lumitexel and the lighting pattern.

However, the accurate shading frame $(\mathbf{n}_\mathbf{p}^\mathbf{j}/\mathbf{t}_\mathbf{p}^\mathbf{j})$ is not known at the input stage. It is usually computed from the predicted reflectance, which is the output of our network. To break this chicken-and-egg cycle, we observe that any frame, denoted as $\hat{\mathbf{n}}_\mathbf{p}/\hat{\mathbf{t}}_\mathbf{p}$, can be used in the trace instead, as long as they are both unit vectors and orthogonal to each other. The relationship between the two frames, expressed in the camera space of each view, can be described as:

$$\mathbf{n}_\mathbf{p}^\mathbf{j} = R\hat{\mathbf{n}}_\mathbf{p}^\mathbf{j}, \mathbf{t}_\mathbf{p}^\mathbf{j} = R\hat{\mathbf{t}}_\mathbf{p}^\mathbf{j}, \forall j.$$

where R is a rotation matrix. In fact, using $\hat{\mathbf{n}}_\mathbf{p}^\mathbf{j}/\hat{\mathbf{t}}_\mathbf{p}^\mathbf{j}$ instead of $\mathbf{n}_\mathbf{p}^\mathbf{j}/\mathbf{t}_\mathbf{p}^\mathbf{j}$ can be considered as a change of the coordinate system, with no loss in the view information supplied in the trace (i.e., the relative motion between two different views stays the same). In practice, we set $\hat{\mathbf{n}}_\mathbf{p}^\mathbf{j}$ to the geometric normal of $\mathbf{p}$ from the pre-captured shape, and $\hat{\mathbf{t}}_\mathbf{p}^\mathbf{j}$ a random unit vector orthogonal to $\hat{\mathbf{n}}_\mathbf{p}^\mathbf{j}$. As a result, the final modified version of a trace is defined as follows:

$$\cup_j \{\text{concat}[B(I, \mathbf{x}_\mathbf{p}^\mathbf{j}, \mathbf{n}_\mathbf{p}^\mathbf{j}, \mathbf{t}_\mathbf{p}^\mathbf{j}), \mathbf{x}_\mathbf{p}^\mathbf{j}, \hat{\mathbf{n}}_\mathbf{p}^\mathbf{j}, \hat{\mathbf{t}}_\mathbf{p}^\mathbf{j}]\}. \quad (4)$$

6.2 Output

The output of the network is a diffuse/specular lumitexel, parameterized over a unit-sized cube map with a resolution of $6\times8^2/6\times32^2$, respectively. For each input trace, we carefully define the corresponding lumitexel space: its origin coincides with $\mathbf{x}_\mathbf{p}$, its positive z axis aligns with $\hat{\mathbf{n}}_\mathbf{p}$ and its positive x axis with $\hat{\mathbf{t}}_\mathbf{p}$. The lumitexel consists of the virtual lighting-varying measurements of the diffuse/specular reflection: we use a virtual camera whose view direction aligns with the positive z axis; a virtual point light with a unit intensity is turned on at a time, whose position is the center of the corresponding texel in the cube map. Our design considerations are explained below.

First, following previous work [Kang et al. 2018, 2019], we choose to predict lumitexels, instead of directly regressing the BRDF parameters along with its local frame, as we are not aware of a regression

method that can produce high-quality results for anisotropic reflectance. The lumitexel can be viewed as a 2D BRDF slice, which is sufficient to reconstruct the complete 4D BRDF, as commonly practiced in related work.

Moreover, unlike [Kang et al. 2018], we use a lumitexel parameterization that does not correspond to our physical LEDs, for two reasons. First, our LED array does not provide a complete sampling of the illumination domain. Important reflectance features may not be represented if we parameterize over the LEDs. Second, the content of a lumitexel parameterized over the LED array would change with the view. It is not clear which view from the input trace the output should be based on. The current parameterization based on virtual lights addresses both issues mentioned above.

In addition, it is important to define the lumitexel space based on the frame of $\hat{\mathbf{n}}_\mathbf{p}/\hat{\mathbf{t}}_\mathbf{p}$ to make lumitexel prediction amenable for learning, since the frame directly corresponds to $\hat{\mathbf{n}}^\mathbf{j}_\mathbf{p}/\hat{\mathbf{t}}^\mathbf{j}_\mathbf{p}$ stored in the input trace. Alternatively, if we use a different frame of $\tilde{\mathbf{n}}_\mathbf{p}/\tilde{\mathbf{t}}_\mathbf{p}$ in defining the lumitexel space, it would impose an extra burden on the network to implicitly learn how to precisely transform from the frame of $\hat{\mathbf{n}}^\mathbf{j}_\mathbf{p}/\hat{\mathbf{t}}^\mathbf{j}_\mathbf{p}$ in the input trace to this new frame. After fitting the output lumitexel (Sec. 7), we obtain the shading normal/tangent expressed in the local frame of $\hat{\mathbf{n}}_\mathbf{p}/\hat{\mathbf{t}}_\mathbf{p}$. They can be easily transformed to the model space as the final result. Therefore, the whole process is essentially a frame-invariant reconstruction. Please refer to Fig. 5 for an example.

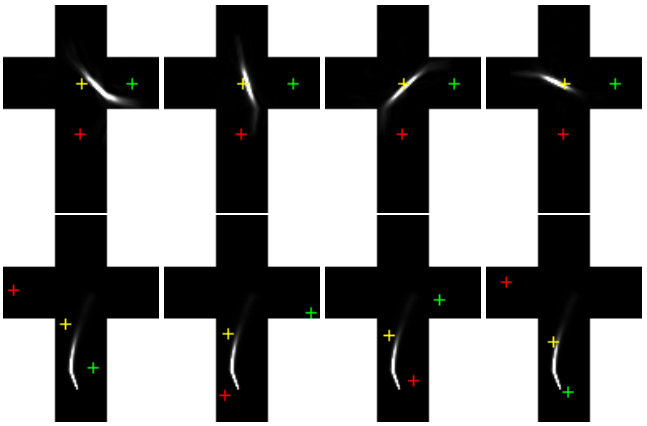

Fig. 5. Frame-invariant reflectance reconstruction. The top row shows the same BRDF expressed in the coordinate system of different $\hat{\mathbf{n}}_\mathbf{p}/\hat{\mathbf{t}}_\mathbf{p}$, which varies with the column. The bottom row is the same BRDF expressed in the frame of the ground-truth $\mathbf{n}_\mathbf{p}/\mathbf{t}_\mathbf{p}$. $\hat{\mathbf{n}}_\mathbf{p}/\hat{\mathbf{t}}_\mathbf{p}/\hat{\mathbf{b}}_\mathbf{p}$ is indicated with a cross marked with yellow/red/green in both rows. As $\mathbf{n}_\mathbf{p}/\mathbf{t}_\mathbf{p}$ is unknown before fitting (Sec. 7), our network learns to output lumitexels in an inaccurate frame, as shown in the top row. The results can be corrected after transforming the fitted frame to the model space, which is equivalent to a frame-invariant reconstruction.

6.3 Loss Function

The loss function measures the squared difference between the predicted diffuse/specular lumitexels and their labels:

$$L = \lambda_d \Sigma_l [m_d(l) - \tilde{m}_d(l)]^2 + \lambda_s \beta \Sigma_l [\log(1 + m_s(l)) - \log(1 + \tilde{m}_s(l))]^2. \quad (5)$$

Here m_d/m_s represents the predicted diffuse/specular lumitexel, respectively. The corresponding labels are denoted with a tilde. A log transform is performed to compress the high dynamic range in the specular reflectance. We use $\lambda_d = 1$ and $\lambda_s = 0.01$ in all experiments.

Note that we weigh the specular lumitexel loss with a confidence β. The idea is to direct the network away from learning to "hallucinate" a specular highlight that is barely observed in the trace, the loss of which would be much higher than without β. On one hand, if throughout the trace no measurement of the specular highlight is taken, we have no clue about the specular reflectance and therefore set the confidence to 0. On the other hand, if the peak of the specular highlight is recorded at one view, we set the confidence to 1. Specifically, for each view, we compute the maximum reflectance with respect to a single LED in the scanner over the maximum value of the BRDF across all lighting directions. Then we determine the confidence as the maximum of the above per-view ratio over all views, followed by an early-saturation non-linear mapping:

$$\beta = \min\left(\frac{1}{\epsilon} \max_j \left[\frac{\max_l \log(1 + f(\omega^{\mathbf{j}'}_\mathbf{i}(l); \omega^{\mathbf{j}'}_\mathbf{o}, \mathbf{p}))}{\max_{\omega_\mathbf{i}'} \log(1 + f(\omega_\mathbf{i}'; \omega^{\mathbf{j}'}_\mathbf{o}, \mathbf{p}))}\right], 1\right), \quad (6)$$

where $\epsilon = 50\%$ in our experiments.

6.4 Architecture

Our network consists of a feature transform module for processing each entry in the trace, a max pooling layer that aggregate the features from all unstructured views, and a subsequent module that produces a diffuse/specular lumitexel. The complete network architecture is visualized in Fig. 4.

First, the feature transform module is similar to the highly successful one proposed in PointNet [Qi et al. 2017], with no T-Net attached. It takes as input a 12D point from the trace, transforms using 6 convolution layers, and generates as output a 1024D feature vector. We refer interested readers to the original paper for more details. Note that our framework is not tied to PointNet. Other excellent techniques in geometry learning on unstructured point clouds may also be plugged in here.

The lumitexel module takes as input the 1024D feature vector after aggregating information via max-pooling, and transform it with 5 fully connected (fc) layers. The module then diverges into two branches: one with 4 fc layers to generate a diffuse lumitexel, and the other with 7 fc layers to produce a specular one.

Note that prior to the feature transform module, we also model the acquisition process, which is essentially a dot product between the physical lumitexel and the lighting pattern, with a single linear fc layer. The weights in this layer correspond to the lighting intensities, and each one is preceded by a sigmoid transform to ensure that the intensity is within the range of [0,1] for physical plausibility.

6.5 Training

We implement our network with PyTorch, using the Adam optimizer with mini-batches of 50 and a momentum of 0.9. Xavier initialization

is applied to all weights in the network. We train 1 million iterations with a learning rate of 1×10^{-4}, which takes 66 hours to finish.

The training/validation data are synthetically generated. For each synthetic point $\mathbf{p}$, we first randomly sample its BRDF parameters (as defined in the appendix): ρ_d/ρ_s is sampled uniformly in the range of $[0, 1]^3$, and α_x/α_y uniformly on the log scale in the range of [0.006, 0.5]. Next, we generate the view conditions for a trace. For the j-th view, we randomly sample $\mathbf{x}_p^j$ in the valid volume (Fig. 2), and then $\mathbf{n}_p^j$ in the visible hemisphere with respect to the camera. $\mathbf{t}_p^j$ is computed as a random vector orthogonal to $\mathbf{n}_p^j$. To mimic the geometric normal $\hat{\mathbf{n}}_\mathbf{p}^j$, we perturb $\mathbf{n}_p^j$ with a random orthogonal vector, whose length is drawn from a Gaussian distribution ($\mu = 0$, $\sigma = 0.15$). The normalized result is stored as $\hat{\mathbf{n}}_\mathbf{p}^j$. $\hat{\mathbf{t}}_\mathbf{p}^j$ is sampled as a random vector orthogonal to $\hat{\mathbf{n}}_\mathbf{p}^j$. Note again that all points/vectors are expressed in the camera space. With related material and geometry parameters sampled, we can compute the corresponding lumitexels to synthesize the image measurements in the input trace, as well as the output labeled lumitexels. We split all synthetic data into the training/validation set with a ratio of 8:2.

We add different forms of training noise to increase the robustness of the network. Each image measurement is multiplied by a sample drawn from a Gaussian ($\mu = 1, \sigma = 0.05$), to account for sensor noise as well as other factors not considered in our equation. Furthermore, we perturb each channel of ρ_d/ρ_s and α_x/α_y, by multiplying with random numbers drawn from a Gaussian ($\mu = 1, \sigma = 0.1/0.15$ for albedo/roughness), for each view in a trace. We also apply a dropout rate of 30% to all fc layers except for the last in the lumitexel module.

7 IMPLEMENTATION DETAILS

We pre-capture the geometry of a physical object with a commercial mobile 3D scanner [Shining3D 2021]. After all photographs are taken with our scanner, only the sharp images [Crete et al. 2007] are automatically selected as input to our network. A uv-parameterization with a texture resolution of 1024^2 is generated.

To compute the camera pose for each image, we first perform structure-from-motion with COLMAP [Schönberger et al. 2016], resulting in a 3D point cloud and camera poses with respect to it. Next, this point cloud is precisely aligned with the pre-captured shape via [Myronenko and Song 2010]. We then update the camera poses with this additional transform from the point cloud to the pre-captured shape. While we find the above process produces sufficiently accurate poses in experiments, standard bundle adjustment can also be added to further improve the precision.

To assemble the trace for the 3D point $\mathbf{p}$ corresponding to each texel, we test for a view j to see if (1) $\mathbf{x}_\mathbf{p}^\mathbf{j}$ is visible and in the valid volume, and (2) $(\omega_o, \hat{\mathbf{n}}_\mathbf{p}^j) > 0.3$ to prevent grazing angles, and (3) the value of each channel of the corresponding pixel is in the range of [32, 224]. If all conditions are met, we add an entry to the trace from this view. Otherwise, the measurement is unreliable and therefore discarded.

We fit the predicted grayscale specular lumitexel with L-BFGS-B, to obtain n, t, α_x and α_y. Next, the RGB ρ_d/ρ_s are obtained, by solving a bounded linear least squares problem that minimizes the ℓ-2 difference between the real measurements and synthetic ones computed from n, t, α_x, α_y and the pre-computed lighting pattern.

8 RESULTS & DISCUSSIONS

All experiments are conducted on a workstation with dual Intel Xeon 4210 CPUs, 256GB DDR4 memory and an NVIDIA GeForce RTX 3090 graphics card. All results are rendered with path tracing using NVIDIA OptiX.

We capture 7 physical objects with a wide variety in appearance. The maximum dimension of each object ranges from 9 to 32cm. The acquisition scene is augmented with AR tags [Fiala 2005]. Note that other markers can also be used here, as the idea is to add more feature points to improve the accuracy of subsequent image registration. The acquisition starts with the pre-capture and reconstruction of the object shape using the commercial 3D scanner, which takes about 20 minutes. Next, we spend about 9 minutes to free-form scan the appearance of an object, with a 0.5s interval between two consecutive acquisition, resulting in 1,000 photographs on average. In the acquisition process, we tilt the scanner and point it towards the object of interest from a variety of viewpoints. Please refer to the accompanying video for an example. Samples of the captured images under our optimized lighting pattern can be viewed in Fig. 6, as well as in the video. No high-dynamic-range images are computed, due to the changing view conditions. It takes 2 hours to preprocess the photographs, with image registration takes up the majority of time. Our network needs 6 minutes to predict lumitexels from all traces, which are fitted to produce the final texture maps in 2 hours.

8.1 Results

Our appearance reconstruction results for the 7 physical objects are shown in Fig. 17, as texture maps that represent various parameters of GGX BRDF. In Fig. 7, we further validate the results against the photographs, taken with a novel lighting condition not used in acquisition. The main reflectance features are well preserved in our reconstructions. Quantitative errors in SSIM are also reported in the figure. Please refer to the accompanying video for animated results with varying views and novel lighting. Note that we intentionally put down the Cheongsam example to better reveal its anisotropic characteristics.

8.2 Comparisons

We compare our result with one state-of-the-art technique on mobile appearance acquisition [Nam et al. 2018] in Fig. 8. To eliminate the impact of geometry, we test our network on both the high-precision geometry captured by our 3D scanner, and the one from [Nam et al. 2018], which is reconstructed by COLMAP [Schönberger et al. 2016]. In both cases, the appearance recovered by our approach closely resembles the corresponding photograph. In comparison, the result from [Nam et al. 2018] shows inaccurate diffuse/specular separation on the face of the bust.

Next, we further validate our approach against a high-end, unmovable lightstage [Kang et al. 2019] in Fig. 9, with 24,576 independently controlled LEDs covering the complete illumination domain. To focus on appearance comparisons, we feed the high-precision scanned geometry to their method, and reconstruct only the SVBRDF from

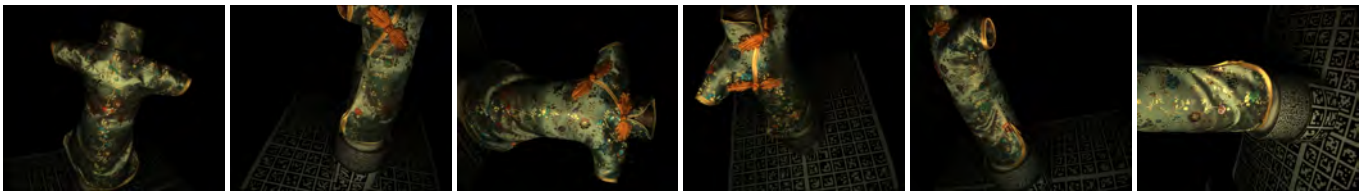

Fig. 6. Sample photographs captured from the Cheongsam object. The brightness of the original images has been doubled for a better visualization.

SSIM=0.89 SSIM=0.90 SSIM=0.93 SSIM=0.91 SSIM=0.93 SSIM=0.88 SSIM=0.90

Fig. 7. Photograph validations. The top row shows photographs of physical objects, while the next row are the rendered images of our reconstructions. Quantitative errors of our results with respect to the photographs are reported in SSIM at the bottom.

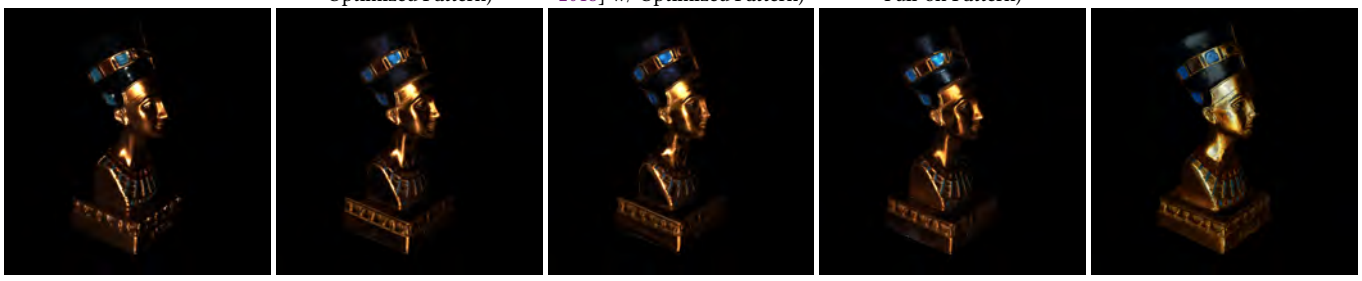

Fig. 8. Comparison with [Nam et al. 2018]. We show the appearance results reconstructed using our framework, on the accurate shape scanned with the mobile 3D scanner, and on the shape from [Nam et al. 2018]. The appearance result from [Nam et al. 2018] is shown, in addition to a photograph of the physical object. Moreover, we compare the results with optimized (2nd image) / fixed full-on lighting pattern (4th image) using our framework.

24×32×3 = 2,304 low-dynamic-range images. Our result is qualitatively similar to theirs, despite that we use a much lighter-weight device. The major difference is the higher perceived spatial resolution in their result, due to the precisely controlled view conditions with a digital turntable and a higher-resolution still camera.

8.3 Evaluations

In this section, we evaluate the impact of various factors over our approach.

First, we conduct a repeatability experiment in Fig. 10. Two graduate students not involved in this project are asked to independently use our scanner, to capture about the same number of photographs of the same bust object. The two reconstruction results are visually similar, as shown in the figure.

Next, we test the impact of geometric reconstruction quality over the recovered appearance in Fig. 11. In addition to the high-precision scanned geometry, we reconstruct the appearance with two lower-quality shapes: one is obtained via mesh filtering on the scanned shape, which removes the high-frequency surface details; the other is the direct output of COLMAP [Schönberger et al. 2016] from multi-view images. As shown in the figure, similar appearance results are obtained, demonstrating the robustness of our framework with respect to minor geometric inaccuracies.

Fig. 9. Comparison with a high-end, unmovable lightstage [Kang et al. 2019]. The top row shows our appearance reconstructions, while the bottom row theirs.

Fig. 10. Repeatability test. Each pair of images show the appearance results, reconstructed from the input photographs scanned by two different persons.

Fig. 11. Impact of the geometry over appearance reconstruction. From the left column to right, the high-quality mesh from the 3D scanner, the result after filtering out high-frequency geometric details from the scanned mesh, and the direct output using COLMAP [Schönberger et al. 2016]. The top row compares the appearance reconstructions, using the corresponding geometry visualized in the bottom.

In Fig. 12, we test the sensitivity of the network with respect to synthetic camera pose error. Note that in this and following figures, only specular lumitexels are shown, as the diffuse lumitexels are of low frequency and can be accurately recovered in experiments. For the j-th input view, $\hat{\mathbf{n}}_\mathbf{p}^\mathbf{j}$ is rotated along a random orthogonal vector with an angle drawn from a Gaussian ($\mu = 0$, $\sigma = \eta$); the frame is re-orthogonalized, then we repeat the same process to $\hat{\mathbf{t}}_\mathbf{p}^\mathbf{j}$ and finally to $\hat{\mathbf{b}}_\mathbf{p}^\mathbf{j}$. Next, we apply a random translation to each component of $\mathbf{x_p}$, the value of which is drawn from a Gaussian ($\mu = 0$, $\sigma = 10\eta$). Here η represents the magnitude of the camera pose error. As shown in the figure, the predicted lumitexels deviate more from the ground-truth, as η increases.

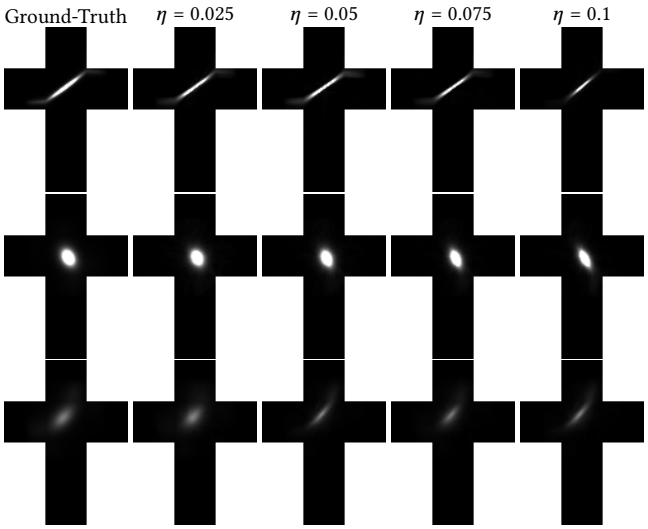

Fig. 12. Impact of camera pose error. Random errors with different magnitudes η is applied to perturb the view parameters in the traces in each column. Each row of images represent a specular lumitexel along with the predicted results from the same network with different errors in the trace. Please refer to the text for details about the perturbation.

We further evaluate the impact of specular highlight coverage over predicted specular lumitexels in Fig. 13. During synthetic trace generation, we reject a view when the half vector h between ω_i and ω_o satisfies $(h, \mathbf{n_p}) > \zeta$. The smaller the ζ, the more portion of the highlight will be excluded from the trace. In the figure, we observe that the quality of the network gracefully degrades, as ζ decreases, which is equivalent to supplying less information about the specular highlight.

In Fig. 14, we evaluate the impact of lighting pattern over the reconstructed lumitexels. First, we compare the baseline version of our network with different variants. For the second column to the third, we reduce the number of optimizable LEDs by shrinking the coverages (16×8 and 8×4) and setting the rest LEDs off. As expected, the prediction error rises as the coverage reduces. We also train our network on fixed, predefined lighting patterns, shown in the fourth column to the sixth. It can observed that fixing the lighting pattern results in a higher loss, compared with our baseline version that jointly optimizes the lighting pattern. It is worth mentioning

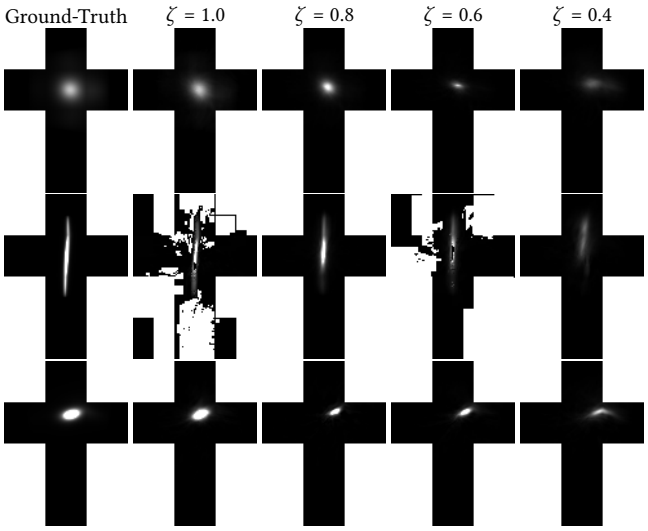

Fig. 13. Impact of specular highlight coverage. During synthetic trace generation, we reject a view when the half vector h between ω_i and ω_o satisfies $(h, \mathbf{n_p}) > \zeta$. Each row of images represent a ground-truth specular lumitexel, and the reconstructions using the same network with the input traces determined by different ζ.

that **a single point source results in the highest error** among all alternatives, due to its lowest sampling capability in the angular domain. This suggests that it will be useful to consider a light source with a larger support in future research on appearance acquisition. The last column shows a binary pattern, which is trained by gradually increasing the penalty for each intensity not to be on or off. This pattern might be useful for high-speed acquisition with LED light sources, since projecting a binary pattern does not involve the more time consuming PWM, which is needed in projecting more fine-grained intensity levels. In a pilot study, we also train multiple RGB lighting patterns, and do not observe clear quality boost despite the increase in the number of patterns. A systematic investigation into this case will be promising for future work.

Furthermore, we compare the impact of two different lighting patterns (visualized in the 2nd and 5th column, the top row in Fig. 14) over appearance reconstruction in a physical acquisition experiment. The results are shown in the 2nd and 4th image in Fig. 8, the quality of which agrees with the corresponding network loss reported in Fig. 14: using a fixed full-on pattern produces a result, whose quality is slightly lower than using a pattern jointly optimized with the reconstruction network; in particular, the tint of the highlight on the face of the reconstructed bust deviates more from the photograph and there are more visual discontinuities, when captured with a full-on pattern. This experiment demonstrates the effectiveness of our lighting pattern optimization.

Finally, we study the impact of the number of views (i.e., the length of the trace) over the reconstructed specular lumitexels. Two experiments are conducted. In one experiment, we test different number of views used during training (Fig. 15). In the other, we use a network trained with 64 views, and test its performance over different number of test views (Fig. 16). We observe both qualitatively and quantitatively that the appearance reconstruction quality improves with the increase of training/test views. It is worth noting that in Fig. 16, our network can perform even better, when the number of test views exceeds the number of training ones. This demonstrates the ability of the network to effectively aggregate multi-view information in the trace.

9 LIMITATIONS & FUTURE WORK

Our work is subject to a number of limitations. First, we do not explicitly account for global illumination effects like interreflections, similar to the majority of related work. Second, we require a relatively precise 3D shape as input, although we have demonstrated the robustness to minor geometric inaccuracies. In addition, our data-driven network cannot faithfully infer appearance that substantially deviates from the training samples.

We hope that this work could open up interesting directions for future research. Besides addressing the above limitations, it will be tempting to extend our idea to a tablet with a similar-sized screen and a front camera (e.g., iPad Pro), to benefit a broader audience. Also, we are highly interested in developing a neural mobile scanner that supports the joint acquisition of reflectance and shape. Moreover, it will be useful to compute real-time feedback, to guide the acquisition in a more active fashion.

ACKNOWLEDGMENTS

We would like to thank anonymous reviewers for their helpful comments, Minyi Gu and Yaxin Yu for building the acquisition device, Zimin Chen for calibrations and geometric alignment, Lijian Ge for professional soldering, Yang Li for the discussion that motivates this paper, Giljoo Nam and Min H. Kim for generously sharing the bust model, Yue Dong for insightful discussions and Yiruo Zhao for proofreading. This work is partially supported by the National Key Research & Development Program of China (2018YFB1004300) and NSF China (61772457, 62022072 & 61890954).

REFERENCES

Miika Aittala, Timo Aila, and Jaakko Lehtinen. 2016. Reflectance Modeling by Neural Texture Synthesis. *ACM Trans. Graph.* 35, 4, Article 65 (July 2016), 13 pages.

Miika Aittala, Tim Weyrich, and Jaakko Lehtinen. 2013. Practical SVBRDF Capture in the Frequency Domain. *ACM Trans. Graph.* 32, 4, Article 110 (July 2013), 12 pages.

Miika Aittala, Tim Weyrich, and Jaakko Lehtinen. 2015. Two-shot SVBRDF Capture for Stationary Materials. *ACM Trans. Graph.* 34, 4, Article 110 (July 2015), 13 pages.

Artec. 2021. Space Spider Portable 3D Scanner. Retrieved January, 2021 from https://www.artec3d.com/portable-3d-scanners/artec-spider

Sai Bi, Zexiang Xu, Kalyan Sunkavalli, David Kriegman, and Ravi Ramamoorthi. 2020. Deep 3D Capture: Geometry and Reflectance from Sparse Multi-View Images. In *CVPR*. 5960–5969.

Guojun Chen, Yue Dong, Pieter Peers, Jiawan Zhang, and Xin Tong. 2014. Reflectance Scanning: Estimating Shading Frame and BRDF with Generalized Linear Light Sources. *ACM Trans. Graph.* 33, 4, Article 117 (July 2014), 11 pages.

Frederique Crete, Thierry Dolmiere, Patricia Ladret, and Marina Nicolas. 2007. The blur effect: perception and estimation with a new no-reference perceptual blur metric. In *Human Vision and Electronic Imaging XII*, Vol. 6492. 196 – 206.

Kristin J. Dana, Bram van Ginneken, Shree K. Nayar, and Jan J. Koenderink. 1999. Reflectance and Texture of Real-world Surfaces. *ACM Trans. Graph.* 18, 1 (Jan. 1999), 1–34.

Valentin Deschaintre, Miika Aittala, Fredo Durand, George Drettakis, and Adrien Bousseau. 2018. Single-image SVBRDF Capture with a Rendering-aware Deep Network. *ACM Trans. Graph.* 37, 4, Article 128 (July 2018), 15 pages.

Valentin Deschaintre, Miika Aittala, Frédo Durand, George Drettakis, and Adrien Bousseau. 2019. Flexible SVBRDF Capture with a Multi-Image Deep Network. In

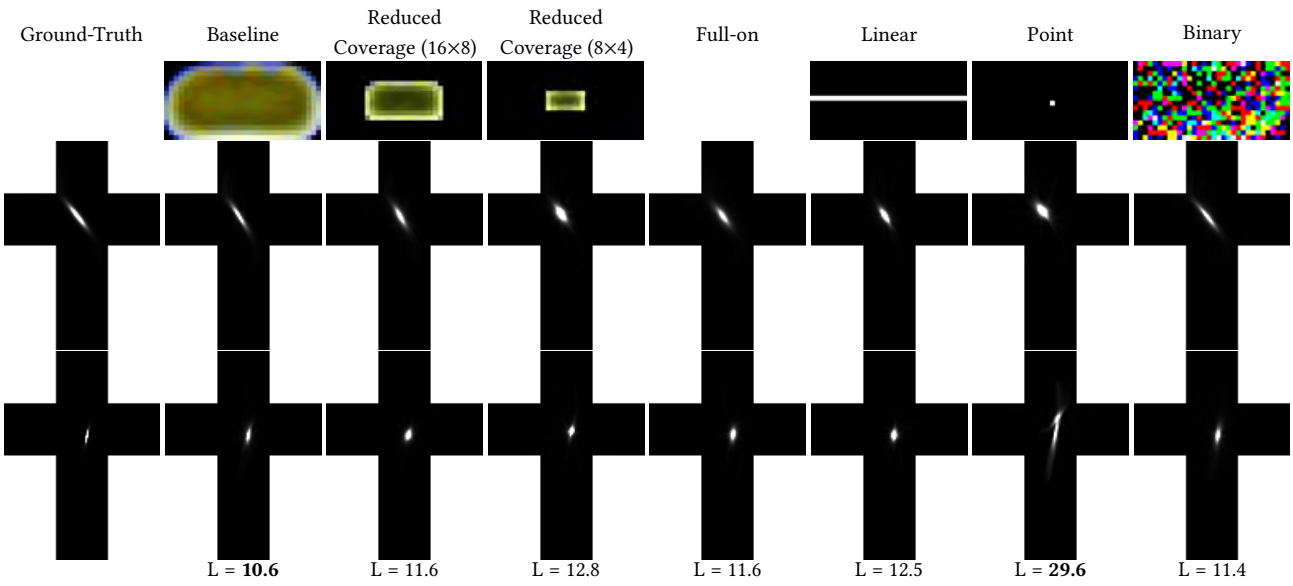

Fig. 14. Impact of lighting pattern over appearance reconstruction. The first column shows two different specular lumitexels. The top row of images visualize different lighting patterns, and the next two rows are predicted specular lumitexels using a network corresponding to the pattern shown on the top. The bottom row lists the validation loss for each network. Note that the full-on pattern is blank, as it has the same color as the background.

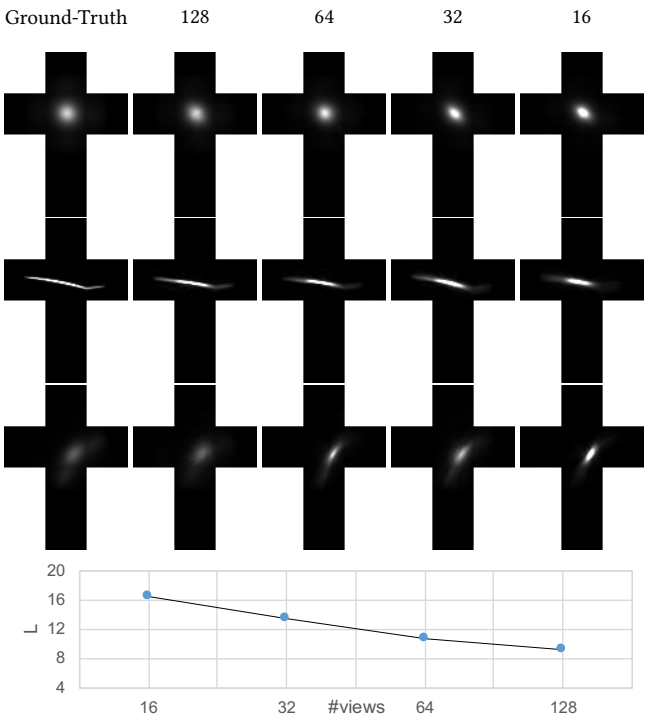

Fig. 15. Impact of the number of training views. A different specular lumitexel and its reconstruction with a network trained with different number of input views, are shown in each of the first three rows of images. The validation loss with respect to the number of training views is plotted at the bottom.

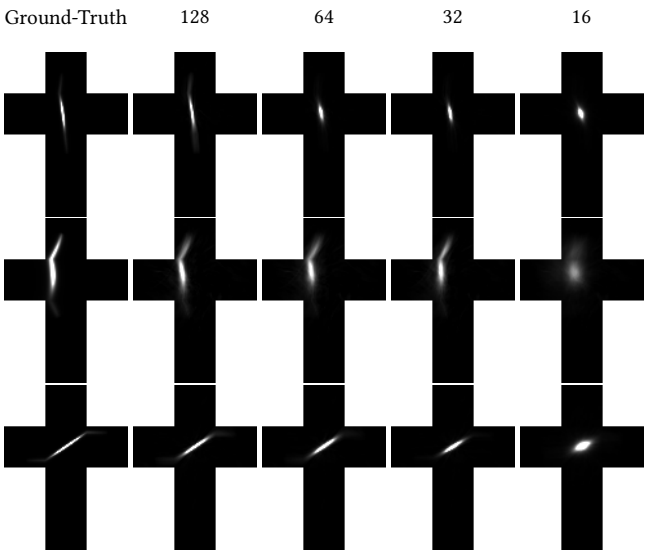

Fig. 16. Impact of the number of test views. A specular lumitexel and its reconstruction with the same network and different number of input views during test, are shown in each row of images.

CGF, Vol. 38. 1–13.

Yue Dong. 2019. Deep appearance modeling: A survey. *Visual Informatics* 3, 2 (2019), 59–68.

Yue Dong, Guojun Chen, Pieter Peers, Jiawan Zhang, and Xin Tong. 2014. Appearance-from-motion: Recovering Spatially Varying Surface Reflectance Under Unknown Lighting. *ACM Trans. Graph.* 33, 6, Article 193 (Nov. 2014), 12 pages.

Yue Dong, Jiaping Wang, Xin Tong, John Snyder, Yanxiang Lan, Moshe Ben-Ezra, and Baining Guo. 2010. Manifold Bootstrapping for SVBRDF Capture. *ACM Trans. Graph.* 29, 4, Article 98 (July 2010), 10 pages.

Mark Fiala. 2005. ARTag, a fiducial marker system using digital techniques. In *CVPR*.

Fig. 17. Reflectance reconstruction results with our network. Each normal/tangent is added with (1, 1, 1) and then divided by 2 to fit to the range of $[0, 1]^3$ for visualization. The roughness α_x/α_y is visualized in the red/green channel. Pre-captured shapes are also shown on the rightmost column for reference.

Duan Gao, Guojun Chen, Yue Dong, Pieter Peers, Kun Xu, and Xin Tong. 2020. Deferred Neural Lighting: Free-Viewpoint Relighting from Unstructured Photographs. *ACM Trans. Graph.* 39, 6, Article 258 (Nov. 2020), 15 pages.

Duan Gao, Xiao Li, Yue Dong, Pieter Peers, Kun Xu, and Xin Tong. 2019. Deep Inverse Rendering for High-resolution SVBRDF Estimation from an Arbitrary Number of Images. *ACM Trans. Graph.* 38, 4, Article 134 (July 2019), 15 pages.

Andrew Gardner, Chris Tchou, Tim Hawkins, and Paul Debevec. 2003. Linear light source reflectometry. *ACM Trans. Graph.* 22, 3 (2003), 749–758.

Abhijeet Ghosh, Tongbo Chen, Pieter Peers, Cyrus A. Wilson, and Paul Debevec. 2009. Estimating Specular Roughness and Anisotropy from Second Order Spherical Gradient Illumination. *CGF* 28, 4 (2009), 1161–1170.

Darya Guarnera, Giuseppe C. Guarnera, Abhijeet Ghosh, Cornelia Denk, and Mashhuda Glencross. 2016. BRDF Representation and Acquisition. *Computer Graphics Forum* 35, 2 (2016), 625–650.

Yu Guo, Cameron Smith, Miloš Hašan, Kalyan Sunkavalli, and Shuang Zhao. 2020. MaterialGAN: reflectance capture using a generative SVBRDF model. *ACM Trans. Graph.* 39, 6 (2020), 1–13.

Michael Holroyd, Jason Lawrence, and Todd Zickler. 2010. A Coaxial Optical Scanner for Synchronous Acquisition of 3D Geometry and Surface Reflectance. *ACM Trans. Graph.* 29, 4, Article 99 (July 2010), 12 pages.

Zhuo Hui, Kalyan Sunkavalli, Joon-Young Lee, Sunil Hadap, Jian Wang, and Aswin C. Sankaranarayanan. 2017. Reflectance Capture Using Univariate Sampling of BRDFs. In *ICCV*.

Kaizhang Kang, Zimin Chen, Jiaping Wang, Kun Zhou, and Hongzhi Wu. 2018. Efficient Reflectance Capture Using an Autoencoder. *ACM Trans. Graph.* 37, 4, Article 127 (July 2018), 10 pages.

Kaizhang Kang, Cihui Xie, Chengan He, Mingqi Yi, Minyi Gu, Zimin Chen, Kun Zhou, and Hongzhi Wu. 2019. Learning Efficient Illumination Multiplexing for Joint Capture of Reflectance and Shape. *ACM Trans. Graph.* 38, 6, Article 165 (Nov. 2019), 12 pages.

Jason Lawrence, Aner Ben-Artzi, Christopher DeCoro, Wojciech Matusik, Hanspeter Pfister, Ravi Ramamoorthi, and Szymon Rusinkiewicz. 2006. Inverse Shade Trees for Non-parametric Material Representation and Editing. *ACM Trans. Graph.* 25, 3 (July 2006), 735–745.

Hendrik P. A. Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. 2003. Image-based Reconstruction of Spatial Appearance and Geometric Detail. *ACM Trans. Graph.* 22, 2 (April 2003), 234–257.

Xiao Li, Yue Dong, Pieter Peers, and Xin Tong. 2017. Modeling Surface Appearance from a Single Photograph Using Self-augmented Convolutional Neural Networks. *ACM Trans. Graph.* 36, 4, Article 45 (July 2017), 11 pages.

N. J. W. Morris and K. N. Kutulakos. 2007. Reconstructing the Surface of Inhomogeneous Transparent Scenes by Scatter-Trace Photography. In *ICCV*.

A. Myronenko and X. Song. 2010. Point Set Registration: Coherent Point Drift. *IEEE PAMI* 32, 12 (2010), 2262–2275.

Giljoo Nam, Joo Ho Lee, Diego Gutierrez, and Min H Kim. 2018. Practical SVBRDF acquisition of 3D objects with unstructured flash photography. In *SIGGRAPH Asia Technical Papers*. 267.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*. 652–660.

Peiran Ren, Jiaping Wang, John Snyder, Xin Tong, and Baining Guo. 2011. Pocket reflectometry. *ACM Trans. Graph.* 30, 4 (2011), 1–10.

Jérémy Riviere, Pieter Peers, and Abhijeet Ghosh. 2016. Mobile surface reflectometry. In *CGF*, Vol. 35. 191–202.

Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In *ECCV*.

Shining3D. 2021. EinScan Pro 2X Plus Handheld Industrial Scanner. Retrieved January, 2021 from https://www.einscan.com/handheld-3d-scanner/2x-plus/

Borom Tunwattanapong, Graham Fyffe, Paul Graham, Jay Busch, Xueming Yu, Abhijeet Ghosh, and Paul Debevec. 2013. Acquiring Reflectance and Shape from Continuous Spherical Harmonic Illumination. *ACM Trans. Graph.* 32, 4, Article 109 (July 2013), 12 pages.

Bruce Walter, Stephen R. Marschner, Hongsong Li, and Kenneth E. Torrance. 2007. Microfacet Models for Refraction through Rough Surfaces. In *Rendering Techniques (Proc. EGWR)*.

Michael Weinmann and Reinhard Klein. 2015. Advances in Geometry and Reflectance Acquisition. In *SIGGRAPH Asia Courses*. Article 1, 71 pages.

Tim Weyrich, Jason Lawrence, Hendrik P. A. Lensch, Szymon Rusinkiewicz, and Todd Zickler. 2009. Principles of Appearance Acquisition and Representation. *Found. Trends. Comput. Graph. Vis.* 4, 2 (2009), 75–191.

Hongzhi Wu, Zhaotian Wang, and Kun Zhou. 2016a. Simultaneous Localization and Appearance Estimation with a Consumer RGB-D Camera. *IEEE TVCG* 22, 8 (Aug 2016), 2012–2023.

Hongzhi Wu and Kun Zhou. 2015. AppFusion: Interactive Appearance Acquisition Using a Kinect Sensor. *CGF* 34, 6 (2015), 289–298.

Zhe Wu, Sai-Kit Yeung, and Ping Tan. 2016b. Towards Building an RGBD-M Scanner. *CoRR* abs/1603.03875 (2016).

Jianzhao Zhang, Guojun Chen, Yue Dong, Jian Shi, Bob Zhang, and Enhua Wu. 2020. Deep Inverse Rendering for Practical Object Appearance Scan with Uncalibrated Illumination. In *Advances in Computer Graphics*. 71–82.

Zhiming Zhou, Guojun Chen, Yue Dong, David Wipf, Yong Yu, John Snyder, and Xin Tong. 2016. Sparse-as-Possible SVBRDF Acquisition. *ACM Trans. Graph.* 35, 6, Article Article 189 (Nov. 2016), 12 pages.

Todd Zickler, Sebastian Enrique, Ravi Ramamoorthi, and Peter Belhumeur. 2005. Reflectance Sharing: Image-based Rendering from a Sparse Set of Images. In *Proc. EGSR*. 253–264.

A GGX BRDF MODEL

The reflectance f is represented with the anisotropic GGX model [Walter et al. 2007]:

$$f(\omega_\mathbf{i}; \omega_\mathbf{o}, \mathbf{p}) = \frac{\rho_d}{\pi} + \rho_s \frac{D(\omega_\mathbf{h}; \alpha_x, \alpha_y) F(\omega_\mathbf{i}, \omega_\mathbf{h}) G(\omega_\mathbf{i}, \omega_\mathbf{o}; \alpha_\mathbf{x}, \alpha_\mathbf{y})}{4(\omega_\mathbf{i} \cdot \mathbf{n})(\omega_\mathbf{o} \cdot \mathbf{n})}.$$

Here ρ_d/ρ_s is the diffuse/specular albedo, α_x/α_y is the roughness, and ω_h is the half vector. In addition, D is the microfacet distribution function, F is the Fresnel term, and G is the geometry term for shadowing/masking effects. Please refer to the original paper for the precise definition of D, F and G. In addition, an index of refraction of 1.5 is used in all experiments.