Learning Efficient Illumination Multiplexing for Joint Capture of Reflectance and Shape

KAIZHANG KANG and CIHUI XIE, State Key Lab of CAD&CG, Zhejiang University, China
CHENGAN HE, Yale University, USA
MINGQI YI, MINYI GU, and ZIMIN CHEN, 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
HONGZHI WU, State Key Lab of CAD&CG, Zhejiang University, China

Fig. 1. Using as few as 16 ~ 32 automatically learned lighting patterns, we efficiently take multi-view photographs of a physical object in a high-performance near-field lightstage, and simultaneously reconstruct its reflectance and shape. Here we show the captured results of a wide variety of real-world objects under novel lighting and view conditions. Background texture courtesy of Design Connected EOOD.

We propose a novel framework that automatically learns the lighting patterns for efficient, joint acquisition of unknown reflectance and shape. The core of our framework is a deep neural network, with a shared linear encoder that directly corresponds to the lighting patterns used in physical acquisition, as well as non-linear decoders that output per-pixel normal and diffuse / specular information from photographs. We exploit the diffuse and normal information from multiple views to reconstruct a detailed 3D shape, and then fit BRDF parameters to the diffuse / specular information, producing texture maps as reflectance results. We demonstrate the effectiveness of the framework with physical objects that vary considerably in reflectance and shape, acquired with as few as 16 ~ 32 lighting patterns that correspond to 7 ~ 15 seconds of per-view acquisition time. Our framework is useful for optimizing the efficiency in both novel and existing setups, as it can automatically adapt to various factors, including the geometry / the lighting layout of the device and the properties of appearance.

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

Additional Key Words and Phrases: optimal sampling, multi-view stereo, lighting patterns, SVBRDF

ACM Reference Format:
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 (November 2019), 12 pages. https://doi.org/10.1145/3355089.3356492

*Hongzhi Wu (hwu@acm.org) is the first corresponding author, and Kun Zhou (kunzhou@acm.org) is the second.
Authors' addresses: Kaizhang Kang; Cihui Xie, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, 310058, China; Chengan He, Yale University, New Haven, CT, 06511, USA; Mingqi Yi; Minyi Gu; Zimin Chen, 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; Hongzhi Wu, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, 310058, China.

1 INTRODUCTION

Digitally acquiring the appearance of real-world objects is a long-standing problem in computer graphics and vision, with applications in cultural heritage, e-commerce, visual effects and electronic games. A high-quality digitalized object, represented as a 3D mesh and a 6D Spatially-Varying Bidirectional Reflectance Distribution Function (SVBRDF), can be rendered to faithfully reproduce the look of the object in the virtual world, from any view and lighting conditions.

However, efficient capture of both reflectance and shape is fundamentally challenging. At one hand, the unknowns are of high dimensionality, and at the same time tightly coupled in image-based

measurements, as modeled by the rendering equation [Kajiya 1986]. Therefore, one would like to take as many measurements as possible, to gain sufficient information to recover the complex reflectance and shape separately. On the other hand, in real-world applications like e-commerce and visual inspection, the number of samples can be strictly limited in practice, as the physical capture time is critical in the digitization of a large number of different products. The key to address the above issue is to optimize the **acquisition efficiency**.

Significant research efforts have been made towards efficient capture of reflectance and shape. Geometry acquisition of objects with simple reflectance characteristics is a mature field. Highly accurate geometry can be obtained using techniques like structured lighting [Scharstein and Szeliski 2003] or structure-from-motion [Schönberger et al. 2016]. Reflectance capture on a simple or known geometry can realistically reproduce complex appearance, such as sharp specular reflections, using techniques like illumination multiplexing that efficiently samples the domain of lighting directions [Gardner et al. 2003; Kang et al. 2018].

Joint estimation of reflectance and shape is substantially more challenging than capturing either factor alone. Existing approaches make assumptions like distant lighting [Tunwattanapong et al. 2013], isotropic reflectance [Xia et al. 2016; Zhou et al. 2013], or a small number of basis materials [Holroyd et al. 2010; Nam et al. 2018] to reduce the uncertainty in the solution. While these assumptions make the problem more tractable by limiting its scope, no previous work explicitly considers the optimization of physical acquisition efficiency in the general setting, thus considerably hinders wider applications in practice.

In this paper, we present a general framework to learn illumination multiplexing for high-quality, efficient capture of both reflectance and shape. We map the physical acquisition and the computational reconstruction to a deep neural network. This allows the automatic optimization of lighting patterns with respect to the joint acquisition efficiency, and breaks the complex mutual dependency between reflectance and shape in image measurements. We also carefully design the network structure to share information between reflectance and shape reconstruction, as well as to combine existing domain-specific knowledge on digitization with deep learning. Furthermore, our framework is flexible and can adapt to various factors, including the configuration of the physical setup and the properties of appearance. This is in contrast with the majority of existing work, where the lighting patterns require sophisticated manual derivations, and cannot be easily adapted to other setups.

We build a high-performance near-field lightstage to demonstrate the effectiveness of our framework. A number of objects with considerable variations in reflectance and shape (Fig. 1) are captured, using as few as 16 $\sim$ 32 lighting patterns that correspond to 7 $\sim$ 15 seconds of per-view acquisition time. In comparison, 44 distant lighting patterns are used in [Tunwattanapong et al. 2013], a technique most similar to ours. We also validate our results with the photographs under the same lighting and view condition.

2 RELATED WORK

To digitally reconstruct the reflectance and/or the shape of a physical object is a central problem in computer graphics and vision. Existing work can be roughly divided into two categories, based on whether the incident illumination is controlled or not. For the sake of brevity, we mainly review acquisition approaches under *controlled lighting*, which are most related to this paper. Interested readers are referred to excellent recent surveys [Dong 2019; Guarnera et al. 2016; Weinmann and Klein 2015; Weyrich et al. 2009].

2.1 Geometry Reconstruction with the Diffuse Assumption

Highly accurate geometry can be reconstructed with active illumination methods such as structured lighting [Scharstein and Szeliski 2003]. On the other hand, passive approaches like structure-from-motion [Schönberger et al. 2016] achieve huge successes in recovering shapes with rich surface textures. However, both classes of methods assume a diffuse-dominant reflectance that is invariant with view conditions, to establish multi-view correspondences. For objects whose appearance can be represented by a general SVBRDF, this assumption no longer holds: the reflectance that changes with the view is often treated as outliers, or physically modified via means like powder coating.

Another line of work is photometric stereo [Woodham 1980]. Assuming a diffuse reflectance, it estimates a normal field that can be subsequently integrated into a 3D surface, from appearance variations under different illuminations. However, even one latest technique [Ikehata 2018] is limited to handle isotropic specular reflections, under as many as 96 distant lighting conditions.

2.2 Spatially-Varying Reflectance Capture on a Known Shape

Direct sampling the 6D domain of SVBRDF by mechanically positioning a camera and a light source is prohibitively expensive [Dana et al. 1999; Lawrence et al. 2006]. Priors over the reflectance data are introduced to reduce the acquisition cost, including a linear combination of basis materials [Lensch et al. 2003; Wu et al. 2016], a low-dimensional reflectance manifold [Dong et al. 2010], and stochastic-texture-like materials [Aittala et al. 2015].

Illumination-multiplexing-based approaches can capture high-quality results efficiently, as a number of light sources are programmed simultaneously. The lightstage system [Ghosh et al. 2009] captures the photographs of a material sample under spherical harmonics lighting patterns, and recovers the reflectance from a manually derived inverse lookup table, which maps the observed radiance to BRDF parameters. The linear light source reflectometry [Chen et al. 2014; Gardner et al. 2003] moves a linear light source over a planar material sample, and reconstructs the SVBRDF from the corresponding appearance variations. Aittala et al. [2013] use a camera and a near-field LCD panel as the light source, to capture an isotropic reflectance based on a frequency domain analysis.

Kang et al. [2018] learn an autoencoder that jointly optimizes a small number of the lighting patterns and the corresponding decoding network, to efficiently acquire a general reflectance. It is not straightforward to extend their paper to our case, due to the extra complexity of unknown geometry, and the complicated interplay between reflectance and shape in image measurements.

2.3 Joint Acquisition of Reflectance and Shape

Tunwattanapong et al. [2013] build a rotating LED arc to project continuous spherical harmonics patterns to a sample object. With the distant lighting assumption, per-pixel reflectance maps are first computed for each view, which are then used as input to a multi-view stereo algorithm for shape reconstruction. The technique cannot be easily extended to our case, as an accurate 3D position at each pixel is required to eliminate the near-field effects for reflectance reconstruction. Zhou et al. [2013] capture different views of an object with a number of circular LED lights turned on one at a time. Multi-view photometric stereo is applied to estimate the geometry, followed by an isotropic reflectance computation. The sparse number of lights prevents per-pixel reflectance estimation. The geometry reconstruction heavily relies on the isotropic reflectance assumption and cannot be easily extended to handle anisotropic one. Holroyd et al. [2010] build a gantry with a projector-camera pair and use phase-shift patterns for geometry reconstruction. However, a strong prior is imposed on the recovered reflectance, due to the sparse sampling in the angular domain.

Xia et al. [2016] recover the shape and isotropic reflectance from a video sequence of rotating object, exploiting the discontinuities in the unknown illumination. Recently, Nam et al. [2018] take hundreds of flash photographs from multiple views, to compute a 3D geometry and isotropic reflectance expressed as a linear combination of basis materials, via an involved alternating optimization.

2.4 Deep-Learning-Assisted Modeling

Considerable research progress has been made in applying the deep learning techniques to reflectance modeling [Deschaintre et al. 2018; Li et al. 2017] and shape modeling [Kendall et al. 2017; Yao et al. 2018]. Recently, Li et al. [2018] regress the isotropic reflectance and a depth map directly from a single image under unknown environment illumination and flash lighting. Wu et al. [2018] transfer multi-view images of a 3D shape with a homogeneous isotropic reflectance into a diffuse one via a generative adversarial network, to improve the geometry reconstruction.

While the majority of existing work in this category focuses on recovering shape / reflectance from highly sparse input, we take a more active step further by optimizing both the measurement and the reconstruction processes using deep learning, for high-quality, simultaneous acquisition of reflectance and geometry.

3 ACQUISITION SETUP

Our acquisition device is a near-field lightstage in the shape of a cube of 80cm^3 (Fig. 2). It is designed to illuminate a physical object placed on a digital turntable at the center, with different lighting patterns. A single machine vision camera, Basler acA2440-35uc is installed at the center of the edge between the top and the front face, taking photographs of the sample at about 45 degrees above the horizontal plane, at the resolution of $2,448 \times 2,048$. The camera has a narrow field of view and is focused on the sample object, whose maximum size is 20cm × 20cm × 20cm. The turntable can be programmed to rotate the object. Please refer to Fig. 3 for an illustration.

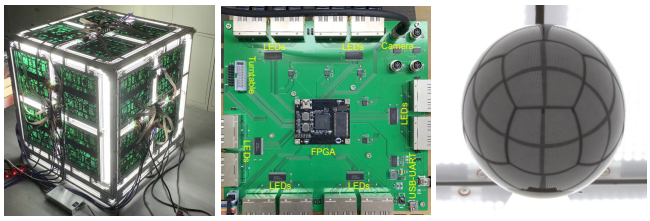

Fig. 2. Our acquisition setup. From the left to right, the exterior of the setup, the main circuit board, and a black calibration sphere inside the setup with all LEDs on.

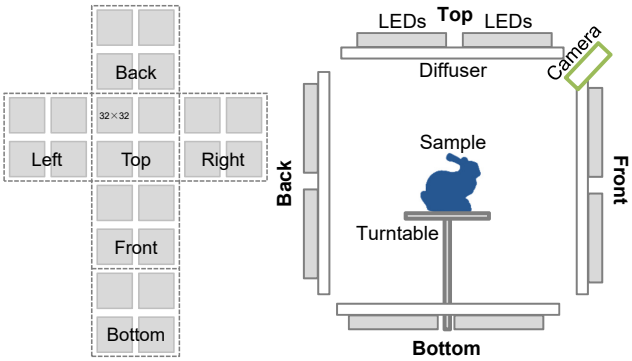

Fig. 3. Our LED layout (left) and acquisition device (right). Each face of our acquisition box has 4 LED boards, each of which consists of 32×32 LEDs. All LEDs are unfolded to a 2D plane in a vertical cross way. A side view of our device is illustrated on the right.

Our setup is equipped with 24, 576 white LEDs, which are grouped as 24 boards of 32×32 LEDs, attached with polycarbonate diffusers and mounted to all six faces of the cube. We intentionally leave small gaps between light boards, for additional camera installations in the future. The distance between two adjacent LEDs is 1cm. The LED intensity is quantized with 8 bits, and independently controlled via Pulse Width Modulation (PWM) with an Intel Cyclone 10 FPGA and circuits we designed. The device can display over 20, 000 binary lighting patterns, each of which contains 24, 576 bits, in 1 second. We employ Low-Voltage Differential Signaling (LVDS) to reliably transmit the high frequency LED control signals from the FPGA to LED boards, which can be as distant as 200cm away.

Before acquisition, we calibrate the intrinsic and extrinsic parameters of the camera, as well as the positions, orientations and the angular intensity distribution of LEDs. Color calibration is performed with an X-Rite ColorChecker Passport. The scale ambiguity of diffuse / specular albedo is resolved using a planar diffuse patch of a uniform albedo [Gardner et al. 2003]. The rotation of the turntable is estimated from printed markers [Fiala 2005] on its surface.

4 PRELIMINARIES

Without loss of generality, we assume a single-camera acquisition setup with independently controlled, near-field or distant light sources. We also assume an opaque object of interest, whose geometry can be modeled as a 3D mesh and surface appearance as an

anisotropic SVBRDF. No polarization filter is used. Moreover, the reflectance at each point is reconstructed **independently**, with no assumption on its spatial coherence.

The radiance B reflected to the camera can be modeled as:

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

Here each light l is viewed as a locally planar source. $\mathbf{x_p}/\mathbf{n_p}$ is the position / normal of a point $\mathbf{p}$ on the physical sample, and $\mathbf{x_l}/\mathbf{n_l}$ is the position / normal of a point on the light source l. $\omega_\mathbf{i}/\omega_\mathbf{o}$ denotes the lighting / view directions in the world space, while $\omega_\mathbf{i}'/\omega_\mathbf{o}'$ is the counterpart in the local frame of $\mathbf{p}$. The lighting direction is computed as $\omega_\mathbf{i} = \frac{\mathbf{x_l}-\mathbf{x_p}}{||\mathbf{x_l}-\mathbf{x_p}||}$. $I(l)$ is the programmable intensity for the light l, and the array $\{I(l)\}$ corresponds to a lighting pattern. $\Psi(\mathbf{x_l}, \cdot)$ describes the angular distribution of the light intensity. V is a binary function that tests the visibility between $\mathbf{x_l}$ and $\mathbf{x_p}$. $f(\cdot; \omega_\mathbf{o}', \mathbf{p})$ is a 2D BRDF slice, a function of the lighting direction.

While our approach is not tied to any specific BRDF model, we use the anisotropic GGX model [Walter et al. 2007], the de-facto industry standard [McAuley et al. 2012], to efficiently represent f:

$$f(\omega_\mathbf{i}; \omega_\mathbf{o}, \mathbf{p}) = \frac{\rho_d}{\pi} + \rho_s \frac{D_{\mathrm{GGX}}(\omega_\mathbf{h}; \alpha_x, \alpha_y) F(\omega_\mathbf{i}, \omega_\mathbf{h}) G_{\mathrm{GGX}}(\omega_\mathbf{i}, \omega_\mathbf{o}; \alpha_\mathbf{x}, \alpha_\mathbf{y})}{4(\omega_\mathbf{i} \cdot \mathbf{n})(\omega_\mathbf{o} \cdot \mathbf{n})}. \quad (2)$$

Here ρ_d/ρ_s is the diffuse / specular albedo, α_x/α_y is the roughness, and ω_h is the half vector. D_{GGX} is the microfacet distribution function, F is the Fresnel term and G_{GGX} accounts for shadowing / masking effects (see Sec. A for details).

Due to the linearity of B with respect to I (Eq. 1), B can be expressed as the dot product between I and a lumitexel m:

$$B(I,\mathbf{p}) = \Sigma_l I(l) m(l;\mathbf{p}). \quad (3)$$

Here m is a function of the light source l, defined on the surface point $\mathbf{p}$ of the sample object [Lensch et al. 2003]:

$$m(l;\mathbf{p}) = B(\{I(l) = 1, \forall_{j \neq l} I(j) = 0\}, \mathbf{p}). \quad (4)$$

Each element of m records the reflected radiance B from $\mathbf{p}$ to the camera, with only one light source turned on and set to its maximum intensity, and the remaining lights off. For brevity, we drop $\mathbf{p}$ from m in the remaining text.

Furthermore, a lumitexel m can be expressed as the sum of a diffuse lumitexel m_d and a specular one m_s:

$$m(l) = m_d(l) + m_s(l). \quad (5)$$

Here m_d / m_s records the reflected radiances due to the diffuse / specular reflections, respectively.

5 OVERVIEW

We propose a mixed-domain neural network to capture the reflectance and shape of a physical object, from multi-view photographs under the same, small set of lighting patterns. For each valid pixel location from each view, the network physically encodes the lumitexel at the corresponding visible point $\mathbf{p}$ on the object surface into a small number of measured values, by projecting different lighting patterns; it then computationally decodes the measurements as the diffuse / specular lumitexels, the normal and the approximate position (Sec. 6). From the decoded diffuse / normal / position information at different views, we compute a detailed 3D mesh with multi-view stereo (Sec. 7). Once the shape is determined, we fit a 4D BRDF along with a local frame to the diffuse / specular lumitexels at every surface point (Sec. 8), which yields texture maps that represent the final 6D SVBRDF. Fig. 4 illustrates our processing pipeline.

Note that we use the terms "encoder / decoder" as the output of our network can be viewed as different components of the input, despite not being exactly the same.

5.1 Design Considerations

Here we briefly discuss the major design considerations of the neural network. First, we do not directly use the end-to-end learned approximate position p for the final geometry. The reason is that although the lumitexel of p contains position-dependent information (e.g., the form factor $\frac{(-\omega_\mathbf{i} \cdot \mathbf{n_l})}{||\mathbf{x_l}-\mathbf{x_p}||^2}$ in Eq. 1), such information is not sensitive to small changes in p, which prevents the determination of high-precision 3D positions. Nevertheless, our predicted p is sufficiently accurate to help eliminate the near-field effects in the decoded diffuse lumitexel for estimating ρ_d (Sec. 7).

Second, instead of using the output from the reflectance reconstruction as in existing work [Kang et al. 2018; Nam et al. 2018], we learn to predict the per-pixel normal directly. The reason is that in our near-field case, the reflectance and geometry are highly coupled: the reflectance reconstruction requires a position at the current pixel, while the shape reconstruction takes normals as input. We leverage the deep neural network to break this mutual dependency.

Similar to previous work [Kang et al. 2018], we choose to learn the lumitexels, rather than directly regressing the BRDF parameters, due to the simple spatially invariant, linear relationship among the lighting pattern, the lumitexel and the measurements (Eq. 3). In comparison, the mapping from an input lumitexel to its BRDF parameters is more complicated and challenging for learning.

Moreover, the majority of existing work on multi-view stereo spatially aggregates information to establish reliable correspondences across different views, a crucial step for geometry reconstruction. However, our network takes as input the lumitexel of a single point only. The reason is that this makes the network simple and circumvents the possible combinatorial explosion in synthesizing training data of varied reflectance and shape. As a result, we exploit the state-of-the-art existing work for spatial aggregation, instead of placing the burden to our network.

6 OUR NETWORK

6.1 Input / Output

The input to our network is a physical grayscale lumitexel at a visible point on the object surface from a particular view. The output is the corresponding diffuse / specular lumitexel, the normal and the approximate position (Fig. 5). The extension to RGB channels is detailed in Sec. 9.

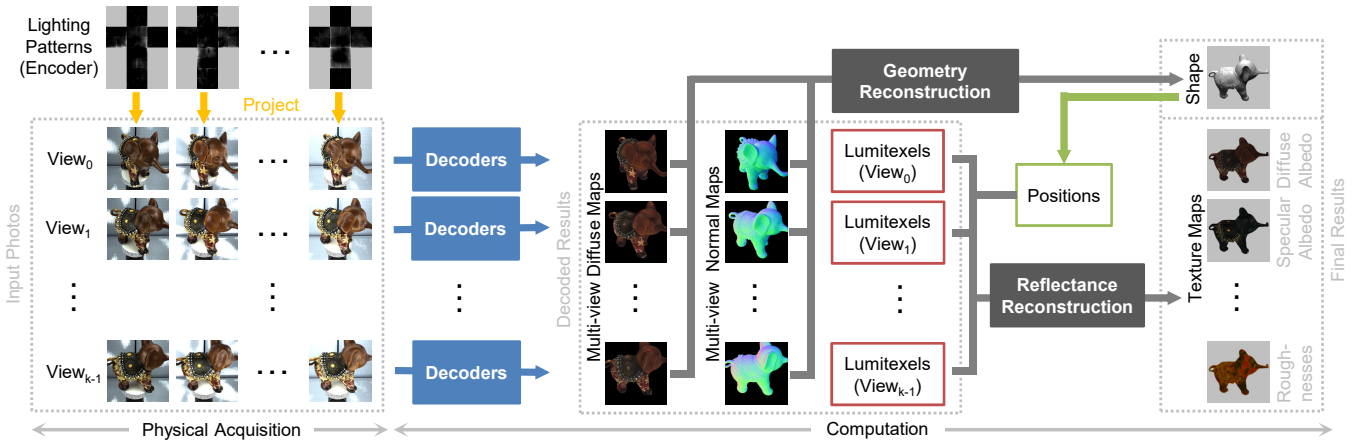

Fig. 4. Our pipeline. First, for each view, we physically encode the lumitexel of a visible point on the object surface by projecting different learned lighting patterns and taking corresponding photographs. Next, for each valid pixel location at each view, we computationally transform the image measurements into the diffuse / specular lumitexel, the normal and the approximation position (not shown in this figure), using the same decoders. From the diffuse lumitexels and the approximate positions at each view, we compute a diffuse map. With the diffuse and normal maps from different views, multi-view stereo is applied to reconstruct a detailed 3D mesh. Finally, using the more precise positions from the mesh and the decoded diffuse / specular lumitexels, we reconstruct the spatially-varying reflectance as texture maps.

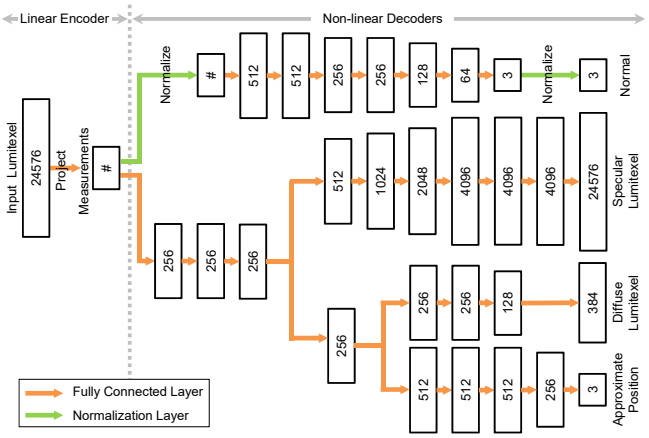

Fig. 5. The architecture of our mixed-domain neural network. The physical input lumitexel is transformed into a small number of measurements by a shared, linear encoder, implemented as a single fully connected layer that represents all lighting patterns. Four nonlinear decoders then recover the diffuse / specular lumitexel, the normal and the approximate position from the measurements. Here # indicates the number of lighting patterns.

We employ slightly different parameterizations for the output diffuse / specular lumitexels, compared with the input physical one (see Fig. 10 for illustration). We uniformly resample each face of the cube where all lights are located, and use a cube map of 6×8^2 / 6×64^2 for the diffuse / specular lumitexels. The idea is to produce outputs that are more uniformly sampled in the angular domain for reflectance and shape reconstruction. In comparison, the parameterization of the input lumitexel (Eq. 4) is strictly determined by the physical lighting layout, subject to multiple practical factors (e.g., we need to leave gaps between LED boards for camera installation in our setup). The difference in the cube map resolution is due to the different frequency natures in diffuse / specular lumitexels.

6.2 Loss Function

The loss function L of our network consists of four terms, which measure the differences between the predicted diffuse / specular lumitexels / normal / position and their ground-truths:

$$L = \lambda_d L_d(m_d) + \lambda_s L_s(m_s) + \lambda_n L_n(\mathbf{n}) + \lambda_p L_p(\mathbf{p}). \tag{6}$$

The terms are defined as follows:

$$L_d(m_d) = \Sigma_l [m_d(l) - \tilde{m}_d(l)]^2, \tag{7}$$

$$L_s(m_s) = \Sigma_l [\log(1 + m_s(l)) - \log(1 + \tilde{m}_s(l))]^2, \tag{8}$$

$$L_n(\mathbf{n}) = ||\mathbf{n} - \tilde{\mathbf{n}}||_2, \tag{9}$$

$$L_p(\mathbf{p}) = ||\mathbf{p} - \tilde{\mathbf{p}}||_2, \tag{10}$$

where m_d / m_s is the diffuse / specular lumitexel, $\mathbf{n}$ is the normal and $\mathbf{p}$ is the position, all predicted by our network. The corresponding ground-truths are denoted with a tilde. We use $\lambda_d = 5$, $\lambda_s = 0.01$, $\lambda_n = 1$ and $\lambda_p = 0.001$ in our experiments. Note that in L_s, we apply a log transform to compress the high dynamic range in specular lumitexels, similar to previous work [Kang et al. 2018; Nielsen et al. 2015].

6.3 Architecture

The network consists of one shared linear encoder, whose weights correspond to lighting patterns, and four nonlinear decoders (Fig. 5).

Specifically, the shared encoder is a single fully connected (fc) layer that transforms a physical lumitexel into a number of measurements by multiplying with the lighting patterns. It has no bias and $h \times c$ linear weights, where h is the dimension of a lumitexel and $2c$ is the number of physical patterns: for physical realization, each $h \times 1$ weights correspond to two lighting patterns, one containing

the positive weights and the other negative ones, similar to previous work [Tunwattanapong et al. 2013]. To bound the weights and prevent degeneration in the presence of noise (Sec. 6.4), there is a normalization step for every group of $h \times 1$ weights, to enforce its l^2-norm to be 1. In physical acquisition, we convert each $h \times 1$ weights into two lighting patterns, quantize each pattern, project onto the sample, and finally combine the measurements as if computing a dot product between the physical lumitexel and the weights (Eq. 3).

For the decoders, their shared input is a number of measured pixel values under different lighting patterns. To avoid making assumptions about the relationships among components of the measurements, the majority part of all decoders are made of fc layers of different sizes, as illustrated in Fig. 5. Each fc layer, except for the final one that outputs the result, is followed by a leaky ReLU activation layer. As shown in the figure, the first few layers are shared by the decoders for diffuse / specular lumitexels and positions, as their tasks are correlated. For the normal decoder, we incorporate *a-priori* knowledge with two additional normalization layers. One layer follows the input measurements, and the other preceding the final result: the former is to directly eliminate the impact of reflectance intensity, which is irrelevant to the normal; the latter is to explicitly produce a unit vector as output.

6.4 Training

Our neural network is implemented with the TensorFlow framework. The Adam optimizer [Kingma and Ba 2015] is employed with mini-batches of 50 and a momentum of 0.9. For the encoder, the initial weights are drawn i.i.d. from a normal distribution ($\mu = 0, \sigma = 1$). For all weights in the decoder, Xavier initialization is applied. To train the network, we run 5 million iterations with a learning rate of 1×10^{-4}.

A large number of high-quality, varied data are critical for training a good network. To handle the wide range of possible reflectance and shape in the real world, we synthetically generate 200 million lumitexels, by evaluating Eq. 4 with randomly sampled location p, f_r and its the local frame. Among all the samples, 80% are used for training and 20% for validation.

Specifically, we first randomly sample p from a valid volume inside the lightstage (detailed in Sec. 3). For the local frame, we sample $\mathbf{n}$ in an upper hemisphere whose apex aligns with the view direction $\omega_\mathbf{o}$, and then $\mathbf{t}$ as a random unit vector that is orthogonal to $\mathbf{n}$. For the BRDF f_r, we use the anisotropic GGX model and randomly sample ρ_d/ρ_s uniformly in the range of $[0, 1]$, and α_x/α_y uniformly on the log scale in the range of $[0.006, 0.5]$. The calibration data of the acquisition setup (Sec. 3) are applied when evaluating Eq. 4 for lumitexel synthesis.

To increase the robustness of our network to physical measurement noise, we add to each component η of the encoding result a Gaussian noise, with a zero mean and a standard deviation of $\frac{1}{100}|\eta|$. Furthermore, dropout regularization with a rate of 10% is applied to all fc layers, except the ones right preceding the outputs.

7 GEOMETRY RECONSTRUCTION

The shape reconstruction consists of two steps: a rough mesh is first computed with multi-view stereo, based on estimated diffuse albedos and normals; then it is refined with the densely decoded normals to obtain the final detailed geometry. Please refer to Fig. 6 for an example.

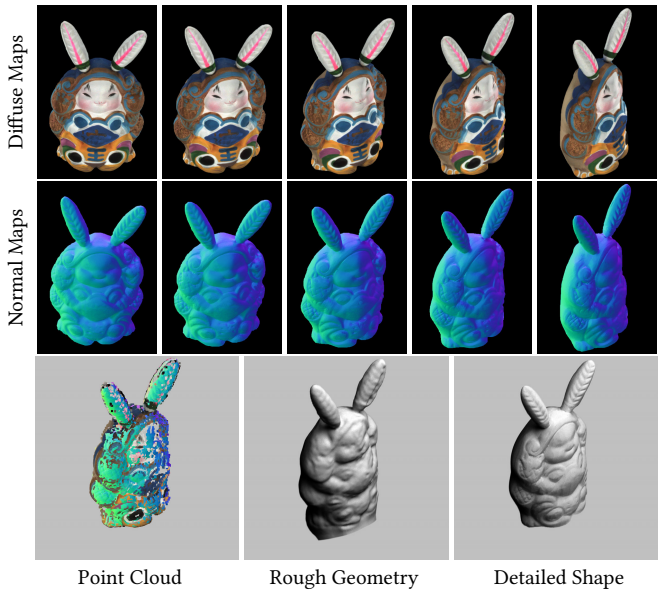

Fig. 6. Geometry reconstruction. Using multi-view diffuse and normal maps from our network, we compute a point cloud with multi-view stereo. Then a rough geometry is obtained by screened Poisson surface reconstruction. We further optimize for a detailed shape, based on the normal maps. Not all input maps are shown due to the space limit.

Specifically, for rough geometry computation, we first transform the decoded normals for each view to a common coordinate system, resulting in a set of normal maps. We also efficiently estimate ρ_d from each decoded diffuse lumitexel m_d, yielding a set of diffuse maps at different views: with decoded $\mathbf{p}$ and $\mathbf{n}$, we synthetically generate a diffuse lumitexel m_0 with a unit diffuse albedo and a zero specular one, according to Eq. 2 & 4; then ρ_d is computed as $\rho_d = \frac{(m_d \cdot m_0)}{(m_0 \cdot m_0)}$ that minimizes $\sum_l [m_d(l) - \rho_d m_0(l)]^2$.

The multi-view diffuse / normal maps are respectively used as input to a state-of-the-art multi-view stereo technique [Schönberger et al. 2016], resulting in two point clouds. We then combine them into a single point cloud, and apply screened Poisson surface reconstruction [Kazhdan and Hoppe 2013] to compute a rough 3D mesh. Note that we use diffuse albedos / normals due to their invariance to view and lighting changes, which is critical for establishing reliable multi-view correspondences in stereo vision. Moreover, as shown in Fig. 7, diffuse and normal maps produce feature points that cover the object surface more completely, leading to a higher-quality mesh than using either set of maps alone.

Next, we refine the rough shape with details in the multi-view normal maps to obtain the final geometry. First, a remeshing is performed to approximately match the vertex density with the normal map resolution. Then, for each vertex, we select an un-occluded view with the smallest angle between the view direction and the current normal; the normal is updated with the value in the normal

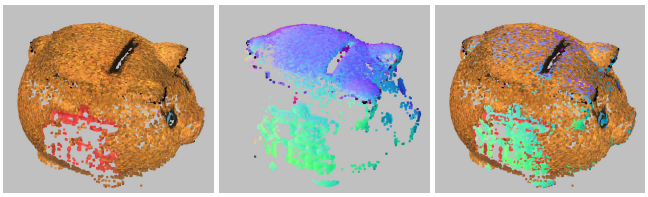

Fig. 7. The point cloud computed with multi-view stereo, using diffuse maps (left) / normal maps (center) generated by our neural network. Combining both point clouds result in a more complete coverage over the object surfaces (right).

map from the corresponding view, at the projected 2D location of the current vertex. Finally, the position of each vertex is refined with the new normal using [Nehab et al. 2005]. The procedure is repeated twice in our experiments. We find that the above closest-view strategy works well in experiments, though advanced techniques (e.g., [Bi et al. 2017]) can also be adopted.

8 REFLECTANCE RECONSTRUCTION

Once a detailed shape is reconstructed, for each point on its surface, we fit parameters of the GGX model (Eq. 2) along with a local frame to the decoded lumitexels. Specifically, we perform nonlinear least squares optimizations using L-BFGS-B [Morales and Nocedal 2011], by minimizing the squared differences between the diffuse and specular lumitexels computed with current estimates of parameters (Eq. 2 & 4) and the corresponding predictions from our network. The spatially-varying results, $\{\rho_d, \rho_s, \alpha_x, \alpha_y, \mathbf{n}, \mathbf{t}\}$, are stored as texture maps.

Note that we recompute the diffuse albedo here, as a more precise estimate of the position is available from the geometry result. Moreover, our framework is not limited to fitting the GGX model. Any BRDF model that preserves the reflectance features of interest can be adopted.

9 IMPLEMENTATION DETAILS

We run our network on the measurements of each color channel separately, to decode the diffuse / specular lumitexels in the R, G and B channel. Then we fit the corresponding grayscale lumitexels to obtain α_x, α_y, $\mathbf{n}$ and $\mathbf{t}$, and discard the grayscale ρ_d / ρ_s. With the remaining parameters fixed, the chromatic ρ_d and ρ_s are computed by fitting the decoded RGB lumitexels using linear least squares.

For each view, we compute a binary mask of the object from two back-lit photographs, one with the object and one without, similar to [Gardner et al. 2003]. Occasional imperfections in the results may be further refined with user assistance. After geometry reconstruction, we build a uv-parameterization for the 3D mesh using [Zhou et al. 2004].

10 RESULTS & DISCUSSIONS

We conduct experiments on a workstation with an Intel Core i9-9900K CPU, 64GB memory, and a GeForce GTX 2080 Ti video card. In acquisition, we merge 3 low-dynamic-range (LDR) photographs of the physical sample with different exposures into an HDR one using bracketing. The typical capture time per view using 32 learned lighting patterns plus 1 back-lit pattern for mask computation (Sec. 9) is about 15 seconds. This time scales linearly with respect to the number of patterns. Please refer to the accompanying video for a demonstration of the process. We rotate the turntable 24 times, each with an angle of $\frac{2\pi}{24}$, to capture different views of the sample object. The size of all photographs is about 10GB. Note that currently there is only one camera in our setup, so the total acquisition time is (the per-view capture time + the turntable rotation time) $\times$ the number of views. We may further reduce it by deploying more cameras to take the photographs from multiple views simultaneously.

The neural network training takes 70 hours to complete. One of our main results is a deep neural network with 32 lighting patterns, trained using general, anisotropic lumitexel samples. For this network, the average error of the decoded normal is 3.8°, and the error of the decoded position is 18mm, both computed on the validation set. For the reflectance and shape reconstruction of a sample object in Fig. 8, the average decoding time is 15 minutes, and the SVBRDF fitting time with our unoptimized code is 2 hours. All texture maps have a resolution of 1024^2. The reconstruction results are rendered with path tracing using NVIDIA OptiX.

We visualize the learned lighting patterns with anisotropic training samples (# = 32) / isotropic ones (# = 16) in Fig. 9. The corresponding photographs of a physical sample under the former set of lighting patterns are also shown: rich variations in the angular domain are revealed under our optimized patterns, which is helpful for the subsequent reflectance and shape reconstruction.

10.1 Results

We show examples of reflectance reconstruction in Fig. 10. For the input lumitexels with considerable variations, our network clearly decouples the diffuse and specular lumitexels, which even have a slightly different parameterization than the input (Sec. 6.1). Moreover, reflectance results along with normals computed after BRDF fitting are demonstrated, closely matching the ground-truths. For fairness, the input lumitexels are not used in the training.

We demonstrate the effectiveness and generality of our framework over 7 non-planar objects, which cover a wide variety of materials and geometries. In Fig. 11, we validate our reconstruction results by qualitatively comparing with the photographs of the physical samples under a novel lighting condition not used in the acquisition: the main appearance features are well preserved in our reconstructions, with quantitative errors reported in structural similarity index (SSIM). We also show the rendering results under novel lighting and view conditions. In Fig. 8, we show various texture maps representing the reflectance results, along with the geometry reconstruction results. Please refer to the accompanying video for animated results.

10.2 Evaluations

We first evaluate the impact of the number of lighting patterns over the decoding quality in Fig. 12. We plot L (Eq. 6) as a function of the pattern number. As more patterns are used, the decoding error L decreases. This is because more information about the reflectance

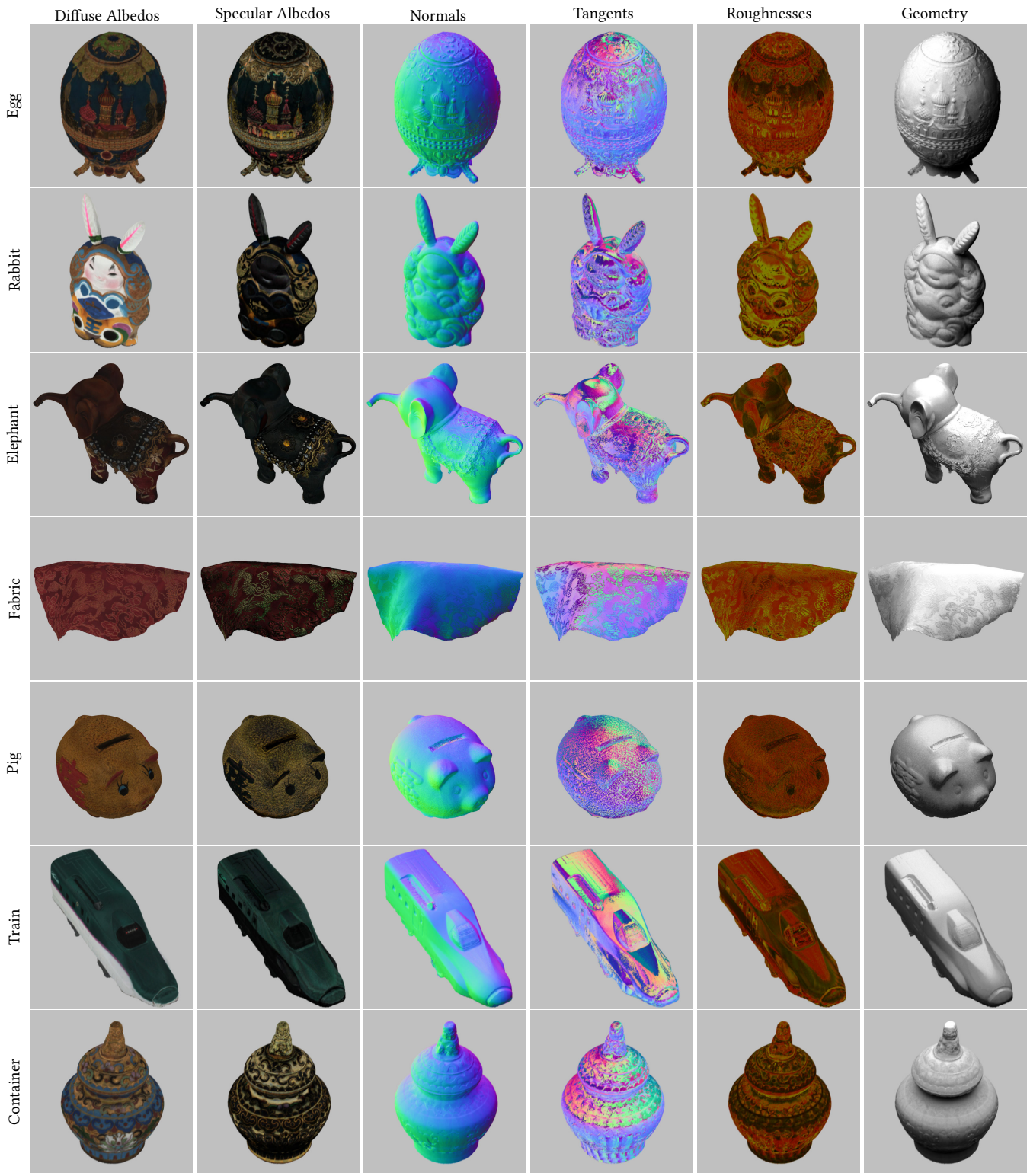

Fig. 8. Reflectance and shape modeling results. Each normal is added with (1, 1, 1) and then divided by 2 to fit to the range of $[0, 1]^3$ for visualization. The tangents are visualized in the same way. For roughnesses, α_x/α_y are visualized in the red / green channel.

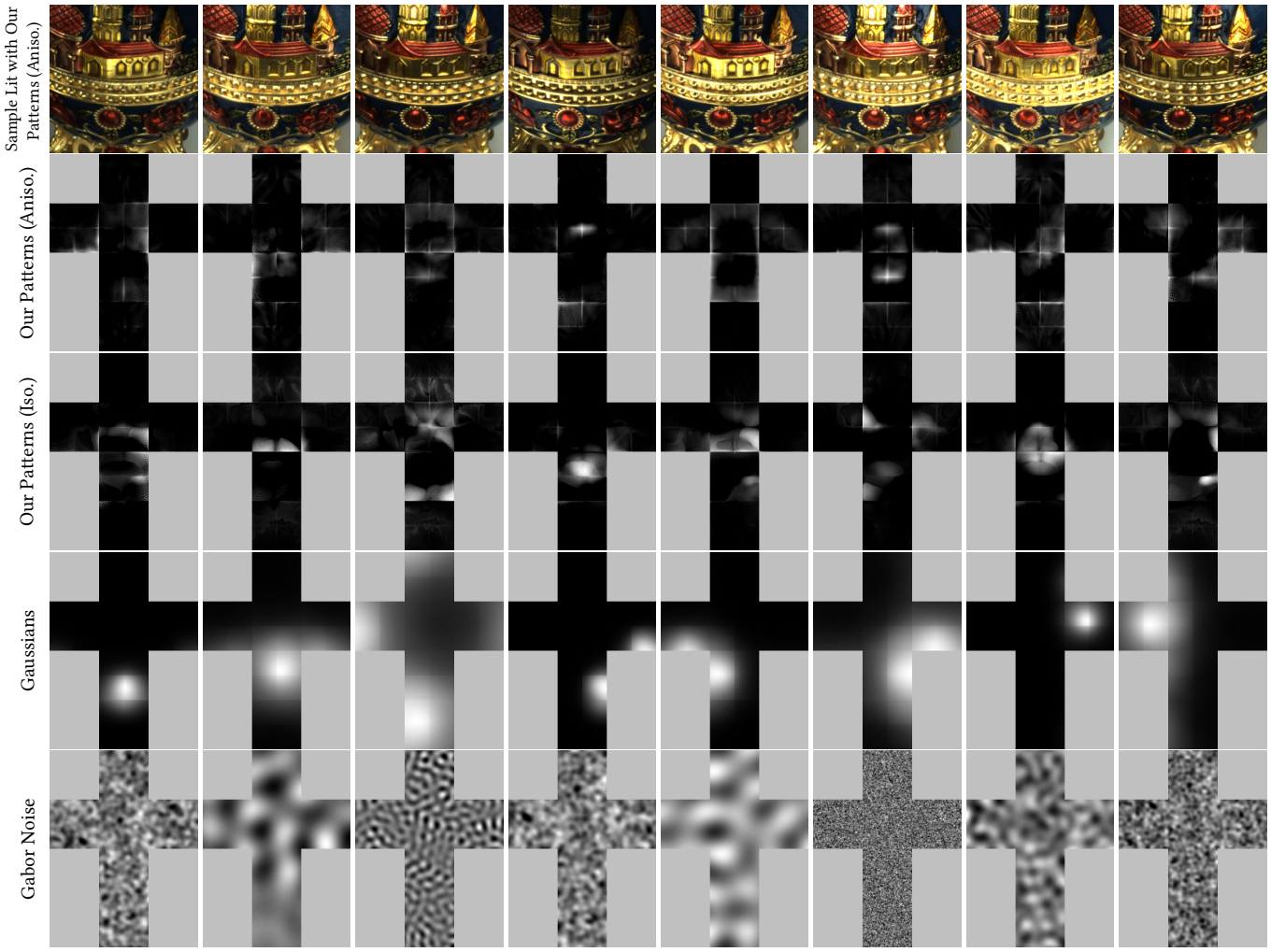

Fig. 9. Different lighting patterns. From the top row to the bottom: the photographs of a physical sample lit with the corresponding lighting patterns in the second row, our patterns learned from anisotropic training samples (# = 32), our patterns from isotropic samples (# = 16), and the patterns using randomly sampled spherical Gaussians / Gabor noise (# = 32). Only a subset of all patterns are shown due to the limited space.

and shape is measured and passed down to the decoder, which makes it possible to produce a more accurate output.

Next, we evaluate the effectiveness of our learned patterns (# = 32) against the same number of fixed ones. We compare with two sets of patterns: one generated using spherical Gaussians with randomly sampled means and standard deviations, the other using Gabor noise with randomly sampled parameters [Lagae et al. 2009]. A visualization of the patterns can be found in Fig. 9. As shown in Fig. 13, although the corresponding decoders are trained to adapt to the fixed lighting patterns, the decoding quality is below ours, where the lighting patterns (i.e., encoder) are optimized in conjunction with the decoders.

Furthermore, we perform sensitivity tests on our network in Fig. 14, by adding a Gaussian noise to each component of the encoding, with a zero mean and a standard deviation proportional to the magnitude of the component, to simulate measurement noise / factors not modeled. The results demonstrate that our decoders are considerably robust to the measurement noise, which is explicitly handled in the training process (Sec. 6.4).

Finally, we study the impact of the training data distribution over the number of lighting patterns. Two networks, one trained with anisotropic samples (#=32) and the other with isotropic samples (#=16), are used to reconstruct the same physical object. As more knowledge about the SVBRDF of interest is exploited in the training, the amount of information needed from the measurements reduces, resulting in a decrease in the number of lighting patterns for reconstructions of similar quality, as shown in Fig. 15.

11 LIMITATIONS & FUTURE WORK

Our work is subject to a number of limitations. We do not explicitly model inter-reflections or self-shadowing for lumitexel reconstruction, similar to related work [Nam et al. 2018; Tunwattanapong

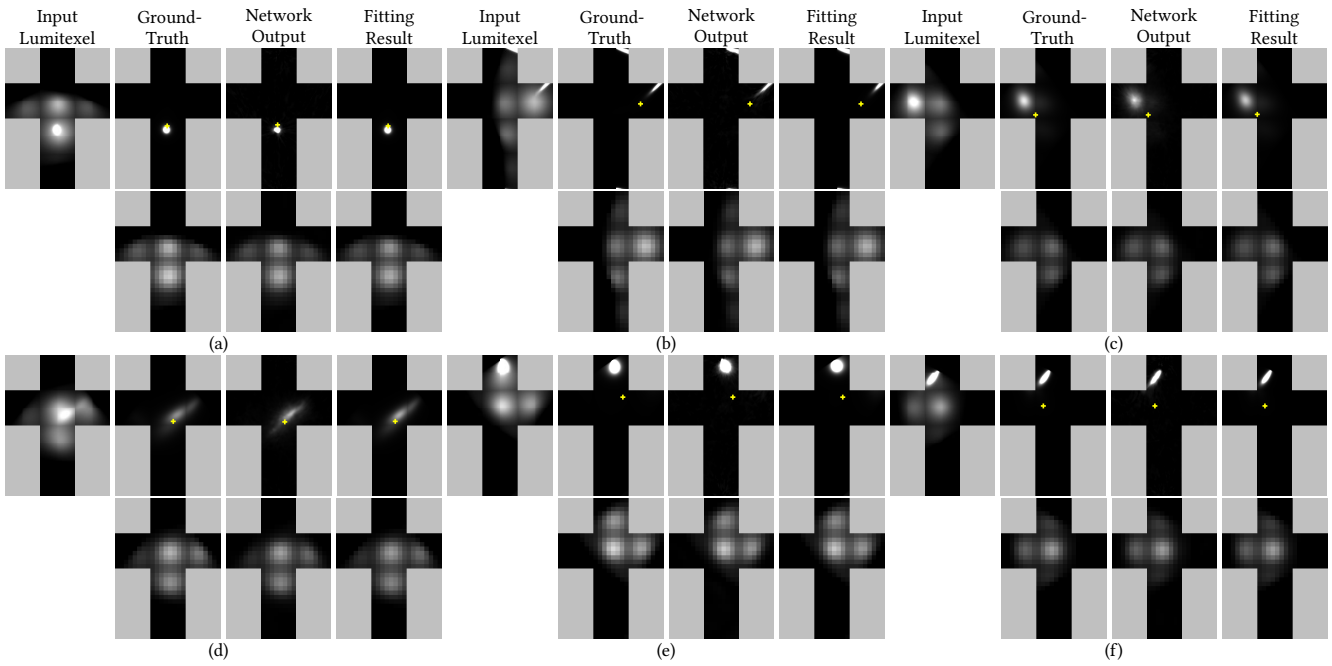

Fig. 10. Reflectance reconstruction. Except for those marked as the input lumitexels, the odd row shows the specular lumitexels, and the even row is the diffuse lumitexels. The normal is indicated as a yellow cross. Six examples (a)-(f) are shown. Note that the input lumitexels have a different parameterization than the output specular / diffuse lumitexels (Sec. 6.1).

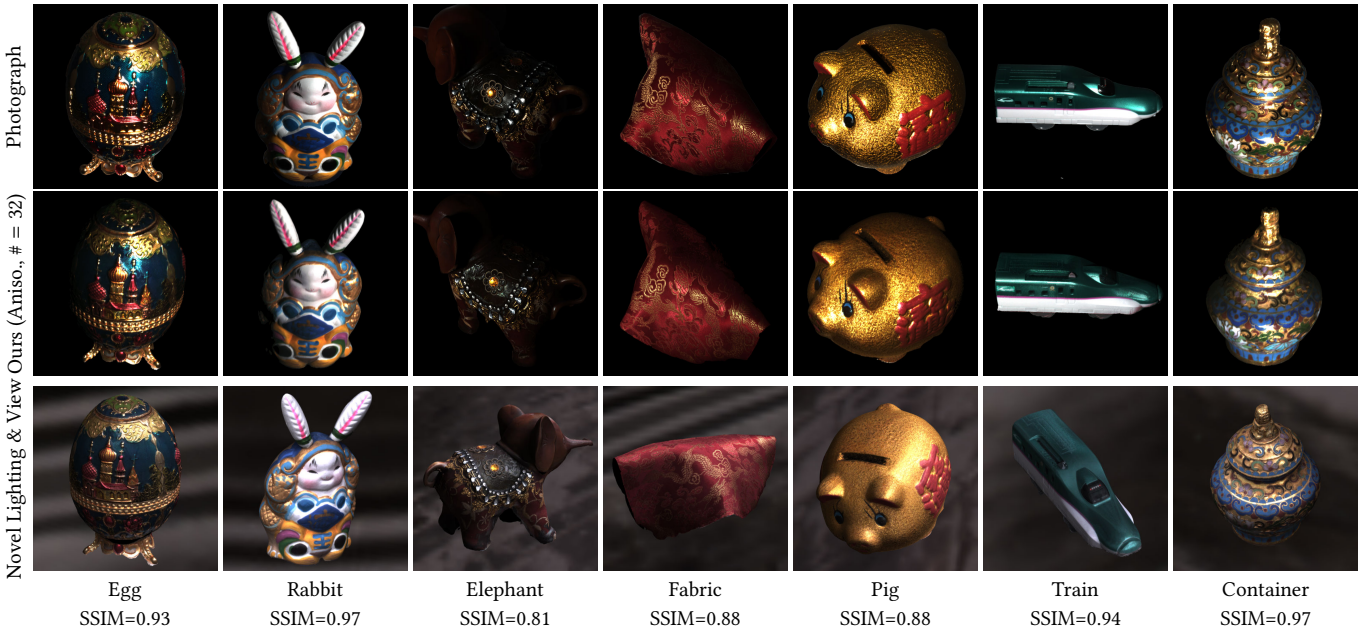

Fig. 11. Validation results. For images from the top row to the bottom in each column: a photograph of the physical object, the rendering of our result captured using learned lighting patterns (anisotropic training samples, # = 32), the rendering of our result with novel lighting and view conditions. The last row reports quantitative errors of our results with respect to the photographs, measured in SSIM. Please refer to the accompanying video for animated results.

et al. 2013]. Following previous work [Kang et al. 2018], our framework cannot faithfully recover lumitexels that substantially deviate from training samples, due to the data-driven nature. Moreover, we

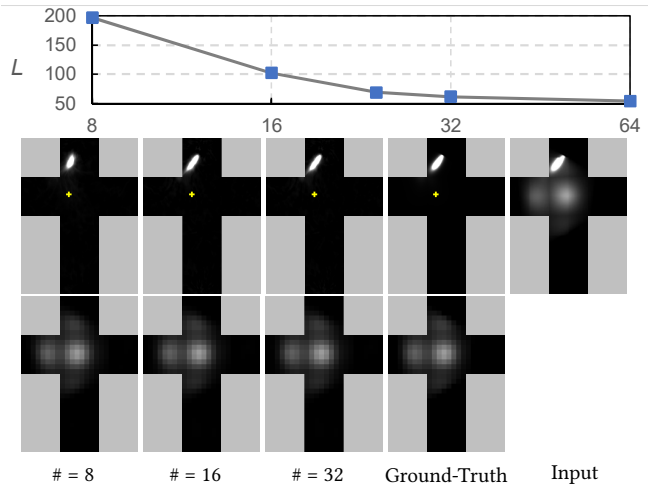

Fig. 12. The impact of the number of lighting patterns over the decoding quality. Top chart: the loss L as a function of the number of lighting patterns. Bottom rows: the outputs from our networks, trained with different number of patterns, for the rightmost input lumitexel.

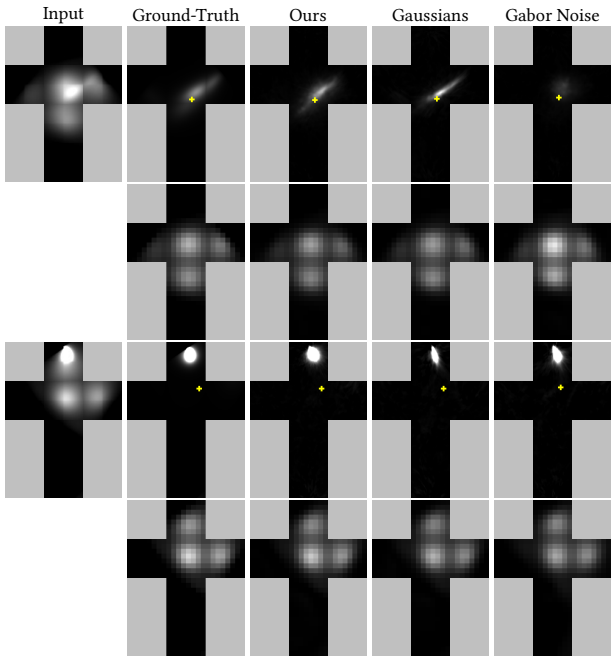

Fig. 13. The impact of different encoders over the decoding quality. The decoded diffuse / specular lumitexels, using our network with learned lighting patterns and networks trained with fixed patterns of randomly sampled spherical Gaussians / Gabor noise (cf. Fig. 9), are shown along with the ground-truths.

cannot reconstruct details that are not observed from the sampled views.

We hope that this work will inspire future research on the broader topic of **differentiable acquisition**, to jointly and automatically optimize different components of the acquisition pipeline. It will also be interesting to apply our framework to improve the physical efficiency of existing setups (e.g., [Aittala et al. 2013; Tunwattanapong et al. 2013]), as well as to guide the design of novel illumination-multiplexing devices. Moreover, it will be intriguing to handle other types of appearance like subsurface scattering.

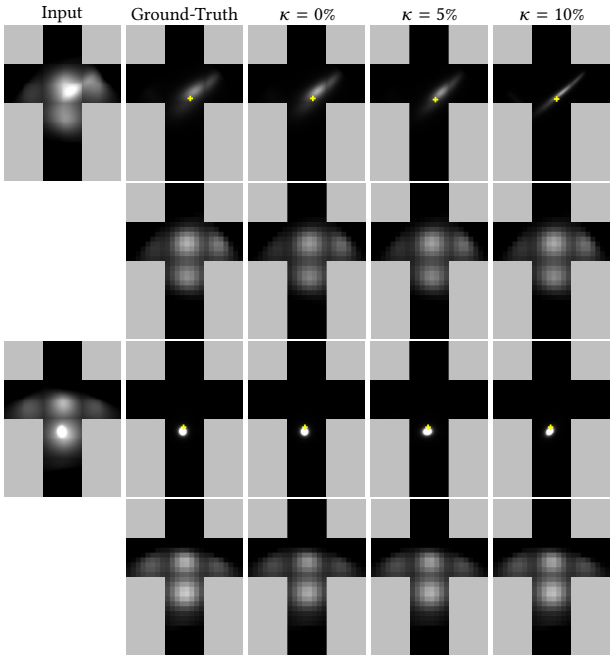

Fig. 14. The impact of the simulated measurement noise over our network (anisotropic samples, # = 32). For each component η of the encoding result, we add a Gaussian noise with a zero mean and a standard deviation of $\kappa|\eta|$. The decoding results are shown in the right three columns.

Fig. 15. The impact of the training data distribution over the number of lighting patterns. The number can be reduced with more certainty about the object appearance (e.g., isotropic reflectance), for comparable quality reconstructions. The reconstruction results with 32 patterns (anisotropic samples) are shown in (a) and (c), while those with 16 patterns (isotropic samples) in (b) and (d).

ACKNOWLEDGMENTS

We would like to thank anonymous reviewers, Xiaohe Ma, Lijian Ge, Jingke Wang, Tong Yang and Design Connected EOOD (https://www.designconnected.com/) for their help. This work is partially supported by the National Key Research & Development Program of China (2018YFB1004300) and NSF China (61772457 & U1609215).

REFERENCES

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.

Sai Bi, Nima Khademi Kalantari, and Ravi Ramamoorthi. 2017. Patch-based Optimization for Image-based Texture Mapping. *ACM Trans. Graph.* 36, 4, Article 106 (July 2017), 11 pages.

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.

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.

Yue Dong. 2019. Deep appearance modeling: A survey. *Visual Informatics* (2019).

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*.

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. *Computer Graphics Forum* 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.

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.

Satoshi Ikehata. 2018. CNN-PS: CNN-based photometric stereo for general non-convex surfaces. In *ECCV*.

James T. Kajiya. 1986. The Rendering Equation *(SIGGRAPH '86)*. 143–150.

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.

Michael Kazhdan and Hugues Hoppe. 2013. Screened Poisson Surface Reconstruction. *ACM Trans. Graph.* 32, 3, Article 29 (July 2013), 13 pages.

Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. 2017. End-To-End Learning of Geometry and Context for Deep Stereo Regression. In *ICCV*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

Ares Lagae, Sylvain Lefebvre, George Drettakis, and Philip Dutré. 2009. Procedural Noise Using Sparse Gabor Convolution. *ACM Trans. Graph.* 28, 3, Article 54 (July 2009), 10 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.

Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. 2018. Learning to Reconstruct Shape and Spatially-varying Reflectance from a Single Image. *ACM Trans. Graph.* 37, 6, Article 269 (Dec. 2018), 11 pages.

Stephen McAuley, Stephen Hill, Naty Hoffman, Yoshiharu Gotanda, Brian Smits, Brent Burley, and Adam Martinez. 2012. Practical Physically-based Shading in Film and Game Production. In *ACM SIGGRAPH 2012 Courses*. Article 10, 7 pages.

José Luis Morales and Jorge Nocedal. 2011. Remark on "Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-scale Bound Constrained Optimization". *ACM Trans. Math. Softw.* 38, 1, Article 7 (Dec. 2011), 4 pages.

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.

Diego Nehab, Szymon Rusinkiewicz, James Davis, and Ravi Ramamoorthi. 2005. Efficiently Combining Positions and Normals for Precise 3D Geometry. *ACM Trans. Graph.* 24, 3 (July 2005), 536–543.

Jannik Boll Nielsen, Henrik Wann Jensen, and Ravi Ramamoorthi. 2015. On Optimal, Minimal BRDF Sampling for Reflectance Acquisition. *ACM Trans. Graph.* 34, 6, Article 186 (Oct. 2015), 11 pages.

Daniel Scharstein and Richard Szeliski. 2003. High-accuracy stereo depth maps using structured light. In *CVPR*.

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

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.

Robert J Woodham. 1980. Photometric method for determining surface orientation from multiple images. *Optical engineering* 19, 1 (1980), 191139.

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

Shihao Wu, Hui Huang, Tiziano Portenier, Matan Sela, Daniel Cohen-Or, Ron Kimmel, and Matthias Zwicker. 2018. Specular-to-Diffuse Translation for Multi-View Reconstruction. In *ECCV*.

Rui Xia, Yue Dong, Pieter Peers, and Xin Tong. 2016. Recovering Shape and Spatially-varying Surface Reflectance Under Unknown Illumination. *ACM Trans. Graph.* 35, 6, Article 187 (Nov. 2016), 12 pages.

Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. 2018. MVSNet: Depth Inference for Unstructured Multi-view Stereo. In *ECCV*.

Kun Zhou, John Synder, Baining Guo, and Heung-Yeung Shum. 2004. Iso-charts: Stretch-driven Mesh Parameterization Using Spectral Analysis. In *SGP*.

Zhenglong Zhou, Zhe Wu, and Ping Tan. 2013. Multi-view photometric stereo with spatially varying isotropic materials. In *CVPR*.

A GGX BRDF MODEL

The functions involved in the anisotropic GGX model are listed below:

$$D_{\mathrm{GGX}}(\omega_{\mathbf{h}}; \alpha_x, \alpha_y) = \frac{1}{\pi \alpha_x \alpha_y [(\frac{\omega_{\mathbf{h}} \cdot \mathbf{t}}{\alpha_x})^2 + (\frac{\omega_{\mathbf{h}} \cdot \mathbf{b}}{\alpha_y})^2 + (\omega_{\mathbf{h}} \cdot \mathbf{n})^2]^2},$$

$$F(\omega_{\mathbf{i}}, \omega_{\mathbf{h}}) = F_0 + (1 - F_0)(1 - \omega_{\mathbf{i}} \cdot \omega_{\mathbf{h}})^5,$$

$$G_{\mathrm{GGX}}(\omega_{\mathbf{i}}, \omega_{\mathbf{o}}; \alpha_x, \alpha_y) = G(\omega_{\mathbf{i}}; \alpha_{\mathbf{x}}, \alpha_{\mathbf{y}}) G(\omega_{\mathbf{o}}; \alpha_x, \alpha_y),$$

where

$$G(\omega; \alpha_x, \alpha_y) = \frac{2(\omega \cdot \mathbf{n})}{(\omega \cdot \mathbf{n}) + \sqrt{[(\omega \cdot \mathbf{t})\alpha_x]^2 + [(\omega \cdot \mathbf{b})\alpha_y]^2 + (\omega \cdot \mathbf{n})^2}}.$$

Here $\mathbf{t}/\mathbf{b}$ represents the tangent / binormal. For the Fresnel term F, we use an index of refraction of 1.5 in all experiments.