Efficient Reflectance Capture With a Deep Gated Mixture-of-Experts

Xiaohue Ma, Yaxin Yu, Hongzhi Wu, and Kun Zhou, Fellow, IEEE

Abstract—We present a novel framework to efficiently acquire anisotropic reflectance in a pixel-independent fashion, using a deep gated mixture-of-experts. While existing work employs a unified network to handle all possible input, our network automatically learns to condition on the input for enhanced reconstruction. We train a gating module that takes photometric measurements as input and selects one out of a number of specialized decoders for reflectance reconstruction, essentially trading generality for quality. A common pre-trained latent-transform module is also appended to each decoder, to offset the burden of the increased number of decoders. In addition, the illumination conditions during acquisition can be jointly optimized. The effectiveness of our framework is validated against photographs, as well as rendered samples the combinations of all lighting and view directions can be performed with a spherical gantry, which exhaustively location, lighting and view direction.

High-quality digitization of physical material appearance is an important problem in computer graphics and vision, with a wide range of applications including visual effects, cultural heritage, e-commerce and computer games. The digital result, often represented as a 6D spatially-varying bidirectional reflectance distribution function (SVBRDF), can be rendered to faithfully reproduce the complex physical look that varies with location, lighting and view direction.

Directly capturing a general, near-planar reflectance sample can be performed with a spherical gantry, which exhaustively samples the combinations of all lighting and view directions [1], [2]. This results in thousands or even millions of photographs, making it prohibitively expensive both in time and storage.

To improve the acquisition efficiency, one highly successful class of methods employ illumination multiplexing: instead of using a single source at a time, multiple lights are programmed simultaneously; the corresponding photometric measurements are then processed to produce the reflectance result in a pixel-independent manner. Representative work includes the light-stage [3], [4], the linear light source reflectometry [5], [6], and setups with an LCD screen [7] or an LED array [8]. Recently, neural acquisition techniques [9], [10], [11] map both the physical acquisition and the computational reconstruction to a neural network, enabling the joint and automatic optimization of both processes. This leads to a substantially improved efficiency: 32 photographs for pixel-independent reconstruction of anisotropic reflectance from a single view [9].

Our goal is to further push the limit of physical acquisition efficiency, as it is critical for light exposure safety in digitizing delicate artifacts in cultural heritage, or scalability to mass digitization in e-commerce. We observe that state-of-the-art work is based on a unified neural network for all possible input, leading to a relatively lower processing efficiency, due to the potential interference effects. Inspired by the recent success of gated mixture-of-experts [12], [13], our key idea is to introduce deep “divide-and-conquer” to enhance reflectance acquisition.

In this article, we propose a novel framework to adaptively learn to capture and reconstruct an SVBRDF. We automatically and jointly train a gating module to select one out of a number of specialized decoders for optimal reflectance reconstruction, based on photometric measurements acquired with pre-optimized lighting conditions. Each decoder is specifically tailored to efficiently handle a subset of possible reflectance only, essentially trading generality for quality. To alleviate the burden of the increasing number of decoders, we additionally pre-train a reflectance latent-space transform and simplify all decoders to output latent vectors only. Moreover, the illumination conditions during acquisition can be optimized in conjunction with the main network to improve sampling efficiency in the angular domain.

The effectiveness of our framework is demonstrated using an illumination multiplexing setup on 6 sets of challenging near-planar samples that vary considerably in appearance (Fig. 1). We improve the acquisition efficiency of anisotropic reflectance: for results with the same number of input images, our reconstruction quality is above that of the state-of-the-art technique [9], both qualitatively and quantitatively; for equal-quality results, we reduce the number of input photographs to 12 (corresponding to 6 seconds of acquisition time), in comparison with 32 as in [9]. Our results are validated against photographs, as well as rendered

Index Terms—Anisotropic reflectance, computational illumination, SVBRDF.
with novel lighting and view conditions. To further demonstrate the generality of the framework, we apply it to boost the quality of non-planar reflectance scanning [8].

II. RELATED WORK

Below we mainly review existing work with active illumination, which is most related to this article. For a comprehensive overview of reflectance acquisition, please refer to excellent recent surveys [14], [15], [16], [17].

A. Direct Sampling

A straightforward approach to capture a general SVBRDF with high quality is to densely sample its 6D domain [1], [2]. A spherical gantry takes photographs of a sample with a moving pair of a camera and a point light, effectively enumerating different combinations of the view and lighting directions. The acquisition process is prohibitively time consuming.

To improve the physical efficiency, various priors have been introduced to properly regularize the problem, while considerably reducing the number of measurements. Isotropic reflectance of a homogeneous convex object is recovered from a single view direction [18]. Lensch et al. [19] model the appearance as a linear combination of basis materials, to constrain the reconstruction from a sparse number of flash-lit images. Wang et al. [20] exploit the spatial similarity of reflectance and the spatial variation of local frames, to complete the microfacet distributions of BRDFs from single-view measurements. The reflectance is assumed to lie on a low-dimensional manifold for reconstruction from sparse samples [21]. Hui et al. [22] propose a dictionary-based reflectance prior. Recently, Nam et al. [23] 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.

The quality of appearance reconstructed with strong-prior-based methods is usually limited, due to the lack of anisotropic reflections or intricate spatial details. In comparison, our approach does not rely on the aforementioned priors. Instead, we reconstruct complex anisotropic appearance in a pixel-independent fashion.

B. Illumination Multiplexing

Instead of using one light at a time, illumination-multiplexing-based approaches program the intensities of a number of sources simultaneously, substantially improving the acquisition efficiency and signal-to-noise-ratio. Traditional work first manually designs illumination conditions, captures corresponding responses of a material sample under such conditions and finally recovers the reflectance properties from measurements.

Lightstages take photographs of a material sample under gradient illumination [3] or spherical harmonics [4], and recover the reflectance from a manually derived inverse lookup table, which maps the observed radiance to anisotropic BRDF parameters. In [5], [6], a linear light source is regularly moved over a planar material sample, and the SVBRDF is reconstructed from the corresponding appearance variations. Irregular motion of the linear light is supported in [24] with the help of pre-calibrated physical BRDF patches that are imaged with the sample. Aittala et al. [7] employ a camera and a near-field LCD panel as a programmable light source, to capture an isotropic reflectance based on a frequency domain analysis. Nam et al. [25] propose a system that reconstructs micro-scale reflectance via an alternating optimization, with the assumption of a small number of basis materials.

Recently, neural reflectance acquisition techniques map both the physical acquisition and computational processing to a single network, enabling the joint and automatic optimization of both the hardware and software. High-quality results are demonstrated on reconstructing planar reflectance [9]/non-planar reflectance and geometry [10], [11] from structured input, as well as non-planar reflectance from unstructured input using a free-form hand-held scanner [8]. Compared with traditional methods, this leads to nearly an order of magnitude increase in the acquisition efficiency. Our work is most similar to this line of work. Instead of employing a unified network, we adaptively process each input with a most suitable network, further boosting the sampling efficiency.

C. Estimation From Highly Sparse Input

Because of its practical value, SVBRDF estimation from a very small number of photographs, often with uncontrolled illumination, has received considerable attention in academia. This challenging problem is highly ill-posed, due to the huge gap in the amount of information between the limited input and the 6D output. Therefore, strong, hand-crafted or learning-based priors must be supplied to fill in this information gap. As a result, the final quality is affected: the spatial resolution of the output is usually limited; and general reflectance, such as anisotropic one, is not supported.

The structural similarity is exploited to estimate a stationary SVBRDF from a flash-/non-flash-lit pair of images [26], or even a single flash image [27]. Li et al. [28] present a CNN-based solution for modeling SVBRDF from a single photograph of a planar sample with unknown natural illumination, using a
self-augmentation training process. Deschaintre et al. propose networks trained over a large dataset of procedural materials to predict an isotropic SVBRDF from a single [29] or multiple [30] flash-lit photographs. In [31], a latent embedding of planar SVBRDFs is learned to regularize the optimization for appearance reconstruction from an arbitrary number of input images. Adversarial frameworks [32], [33] are explored to estimate an isotropic SVBRDF from flash-lit image(s). Henzler et al. [34] propose to learn a generative model for material textures, which takes a flash-lit image of a stationary natural material as input. Guo et al. [35] introduce highlight-aware convolution to estimate the saturated highlights from the adjacent unsaturated area in a single image.

III. ACQUISITION SETUP

We build a hemicube-shaped, near-field lightstage to conduct physical experiments (Fig. 2). Its size is about 70 cm × 70 cm × 40 cm. We install a 24MP Basler a2A5328-15ucPRO vision camera to capture photographs of a near-planar material sample placed on the bottom plane of the device, from an angle of approximately 45°. The maximum size of the sample is 20 cm × 20 cm. There are 12,288 high-intensity RGB LEDs around the sample, attached with diffusers and mounted to the left, right, front, back and top sides of our setup. The LED pitch is 1 cm, and the intensity is quantized with 8 bits and controlled using Pulse Width Modulation (PWM) with house-made circuits. We calibrate the intrinsic and extrinsic parameters of the camera, as well as the positions, orientations and the angular intensity distribution of each LED. In addition, vignetting is corrected with a flat field source, and color calibration is performed with an X-Rite ColorChecker Passport.

IV. PRELIMINARIES

Following the work of LDAE [9], we first list the relationship among the image measurement \( B \) from a surface point \( p \), the reflectance \( f \) and the intensity \( I \) of each LED of the device. Below we focus on a single channel for brevity.

\[
B(I, x_p, n_p, t_p) = \sum_l I(l) \int \frac{1}{|x_1 - x_l|} \Psi(x_1, -\omega_l) V(x_1, x_p) f(\omega_l; \omega_o, p)(\omega_l \cdot n_p) (-\omega_l \cdot n_1)^+ d x_1.
\]

Here \( l \) is the index of a 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 article. Moreover, \( x_p/n_p/t_p \) is the position/normal/tangent of \( p \), while \( x_l/n_1 \) is the position/normal of a point on the light whose index is \( l \). We denote \( \omega_l/\omega_o \) as the lighting/view direction in the world space, and \( \omega_l/\omega_o' \) as the counterparts in the local frame of \( p \). \( \omega_l \) can be computed as \( \omega_l = (x_1 - x_p)/|x_1 - x_p| \). \( \Psi(x_1, \cdot) \) represents the angular distribution of the light intensity. \( V \) is a binary visibility function between \( x_1 \) and \( x_p \). The operator \((\cdot)^+\) computes the dot product between two vectors, and clamps a negative result to zero.

Our framework is not tied to any specific BRDF model. In this article, we use the anisotropic GGX model [36] to efficiently represent \( f \):

\[
f(\omega_l; \omega_o, p) = \frac{\rho_d}{\pi} + \rho_s \frac{D(\omega_l' \cdot \alpha_x, \alpha_y) F(\omega_l', \omega_o') G(\omega_l', \omega_o'; \alpha_x, \alpha_y)}{4(\omega_l \cdot n_p)(\omega_o \cdot n_p)},
\]

where \( \rho_d/\rho_s \) is the diffuse/specular albedo, \( \alpha_x/\alpha_y \) is the roughness and \( \omega_h \) 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. An index of refraction of 1.5 is used in \( F \) in all experiments. Please refer to the original paper for precise definitions of \( D, F \) and \( G \).

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

\[
B(I) = \sum_l I(l)m(l).
\]

Note that we drop \( x_p, n_p \) and \( t_p \) for brevity. Here the limutexel \( m \) is defined as the collection of virtual measurements of the BRDF \( f \) at a surface point, with one light on at a time [19]. It is a function of the light index \( l \) as follows:

\[
m(l) = B(l) = 1, \forall k \neq l F(k) = 0),
\]

which can be decomposed as the sum of a diffuse limutexel \( m_d \) and a specular one \( m_s \) [10]:

\[
m(l) = m_d(l) + m_s(l),
\]

where \( m_d/m_s \) records the reflected radiances due to diffuse/specular reflections, respectively.

V. OVERVIEW

We propose a deep gated mixture-of-experts network, to efficiently reconstruct the reflectance of a near-planar sample from single-view photographs under a set of pre-optimized lighting patterns. For each valid pixel location, the network first physically encodes the corresponding limutexel as photometric measurements. Next, they are fed to the gating module to pick a suitable decoder, tailored for similar limutexels. The decoder then transforms the same set of measurements to separately recover the diffuse/specular limutexel. We fit a 4D BRDF along with a local frame to the decoded limutexels at each pixel, which yields texture maps that represent the final 6D SVBRDF. Please refer to Fig. 3 for a graphical illustration.

VI. THE NETWORK

Our goal is to introduce a differentiable framework that automatically learns to condition on the input for improved
reflectance reconstruction quality. The idea is to first split the set of all possible input, and then process each subset separately. The reconstruction quality is expected to be improved, since each sub-space usually has a fractional size of the original space, and processing specialized to a sub-space can therefore trade generality for quality.

A. Input/Output

The input to our network is the set of # physical measurements of a point on the material sample, captured with different pre-optimized lighting patterns. During training, the output is the diffuse/specular lumitexels reconstructed with different decoders; at runtime, the output is the diffuse/specular lumitexel from a single decoder. We use # to denote the number of measurements/lighting patterns. Note that similar to [10], we separately output diffuse/specular lumitexels to reduce the complexity of subsequent processing. We use a dimension of 12288, the same as the number of LEDs, to represent the specular lumitexel. And a dimension of 192 is used for the diffuse lumitexel, due to its low-frequency nature.

B. Architecture

The main network consists of three parts: a gating module, a total of $n$ specialized decoders and a latent-transform module ($n = 128$ in most experiments). Please refer to Fig. 4 for an overview of the architecture and Fig. 5 for network details. Each decoder has an index of a $\log_2 n$-bit integer that starts from 0.

The gating module can be viewed as a continuous form of supervised hashing. It takes as input the photometric measurements at a pixel, and predicts a probability distribution over all decoders. The pre-trained latent-transform module (L-T Mod. in the figure) transforms the latent vector output from a decoder to a diffuse/specular lumitexel. The total loss is computed as the weighted average of the prediction loss of each decoder, using the aforementioned gating probability as weights.
Fig. 6. Visualization of our lighting patterns (2nd row), patterns trained with LDAE (3rd row) and computed by PCA on anisotropic samples (4th row). Each lighting pattern is parameterized on a cross, by unfolding all side faces to the top plane. The first row shows actual photographs of the sample set PAPER lit with corresponding lighting patterns in the second row. Note that only a subset of all patterns are shown due to limited space.

Fig. 7. The lumitexels averaged over all that are sent to a particular decoder by our gating module. Each image shows the average lumitexel for a different decoder. A subset of all average lumitexels are displayed due to limited space.

Specifically, the gating module consists of $\log_2 n$ single-bit gating subnets. Each subnet takes as input the photometric measurements and outputs $g(k)$, the probability of the $k$-th bit of the index of the most suitable decoder being 1. Equivalently, for a decoder with an index of $a$, its chance of being picked by the gating can be computed as a joint probability:

$$Pr(a) = \prod_{k=0}^{(\log_2 n)-1} \left[ a_k g(k) + (1 - a_k)(1 - g(k)) \right].$$  \hfill (6)

where $a_k$ denotes the $k$-th bit of $a$. Note that our framework is not tied to a specific way of gating. The current one is employed for its simplicity and $O(\log_2 n)$ space complexity, and can be replaced with other methods (see Fig. 8).

Next, each decoder takes as input the same photometric measurements and produces as output a latent vector, which is further converted to a diffuse/specular lumitexel, by a pre-trained latent-transform module. Each decoder has the same structure with 5 fc layers. While one may employ decoders that directly generate lumitexels as output without the latent-transform module, we find it more efficient to exploit a latent space of all lumitexels, as the intrinsic dimensionality of GGX BRDF is limited. This
substantially reduces the size of each decoder, allowing us to train more of them for improved quality.

Finally, the latent-transform module is pre-trained as part of an autoencoder, whose input is the physical lumitexel and the output is the corresponding diffuse/specular lumitexel. The dumbbell-shaped autoencoder has 17 fc layers. Its 128D bottleneck corresponds to a latent vector of a lumitexel. After pre-training, we discard the part of the network prior to the bottleneck, and leave the remaining as the latent-transform module. Other work on the latent representation of 4D appearance may also be explored [37, 38, 39].

Note that similar to previous work like [9], we link the lighting patterns during acquisition with the main network in a differentiable fashion: measurements of the reflected radiances under physically projected lighting patterns are essentially modeled as dot products between the physical lumitexel and the lighting patterns, according to (1). This allows the joint optimization of the active illumination conditions, the gating and the decoders, towards optimal reconstruction quality.

C. Loss Function

The loss function measures the squared difference between the predicted diffuse/specular lumitexels and their labels, for each decoder weighted by a probability determined by gating (6):

$$L = \sum_{a=0}^{n-1} Pr(a)[\lambda_d \sum_i [m_d^a(l) - \tilde{m}_d(l)]^2 + \lambda_s \sum_i [\log(1 + m_s^a(l)) - \log(1 + \tilde{m}_s(l))]^2].$$

Here $m_d^a/m_s^a$ represents the diffuse/specular lumitexel predicted by the decoder with the index $a$, respectively. The corresponding ground-truths 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 $\lambda_s = 0.05$ in all experiments. Since the gating module affects $Pr(a)$, it gets optimized in conjunction with the decoders via back-propagation.

Note that our loss is a mixture of prediction error of each decoder, not the error on the mixture of predictions as in [12]. Also we do not find it necessary to apply extra regularizations to force load balancing among decoders. This is because load balancing is not a sufficient or necessary condition for obtaining an optimal loss. In fact, our gating module and decoders are automatically and jointly trained towards the goal of minimizing the loss.

D. Training

Our network is implemented with PyTorch, and trained using the Adam optimizer with mini-batches of 50 and a momentum of 0.9. Xavier initialization is applied, except that the gating is initialized with a zero-mean Gaussian noise ($\sigma = 0.1/0.01$ for weights/biases). Both the latent autoencoder and the main network are trained for 1 million iterations with a learning rate of $1 \times 10^{-4}$. Based on the GGX BRDF model and the calibration data of the device, we generate 200 million virtual lumitexels as training data (1). Specifically, for the location on the physical sample, we randomly choose a point from the valid region of the sample plane. Similarly, for the shading frame, we randomly sample $n_p$ in the upper hemisphere of the sample plane, and then $t_p$ as a random unit vector that is orthogonal to $n_p$. For the BRDF
Fig. 11. Reconstruction quality of pre-trained latent autoencoder. The input lumitexel (top row) is compressed into a latent code, which is then decoded with the latent-transform module back to a reconstructed lumitexel (bottom).

Fig. 12. Comparisons of average prediction qualities of different networks with different parameters. The loss $L$ is computed on the validation dataset. Our networks/LDAEs with different number of input images are marked as yellow/red dots, respectively. We also show the losses of several variants in blue dots. Please refer to Section VII-B for details.

Fig. 13. Impact of lighting pattern number over lumitexel reconstruction quality. The first column are the ground-truths and the next three columns are the reconstruction results of our networks with 128 decoders but different lighting patterns ($\# = 32/20/12$). The last column are results of LDAE($\# = 32$). The numerical errors, computed using (7) with $\lambda_d = 0$, are listed at the bottom-right corner of related images. All results are direct network outputs prior to fitting.

Fig. 14. Impact of the number of decoders over lumitexel reconstruction quality. For the top row to bottom, ground-truth lumitexels, reconstruction results from our networks with different numbers of decoders ($n = 32/64/128/256/512$) and the results of LDAE(2x).

$f$, we use the anisotropic GGX model and randomly sample $\rho_d/\rho_s$ uniformly in the range of $[0, 1]$, and $\alpha_x/\alpha_y$ uniformly on the log scale in the range of $[0.006, 0.5]$.

For robustness in physical acquisition, we apply dropout regularization with a rate of 30% to most layers, and perturb the synthetic measurements as well as sampled BRDF parameters with a multiplicative Gaussian noise ($\mu = 1, \sigma = 5\%$), similar to [10]. Moreover, we multiply a Gaussian noise ($\mu = 1, \sigma = 5\%$) to the input of the softmax layer in the gating module, to make it more resilient to potential measurement noise.

E. Runtime

We first average the RGB channels of photometric measurements to a single gray-scale channel. The results are then sent to our network for gating computation, and the decoder with the highest $Pr$ is selected to produce a diffuse/specular lumitexel.
Note that we never mix the outputs of multiple decoders. Next, we nonlinearly fit a normal to the diffuse lumitexel, which serves as a good initialization for a subsequent fitting of the shading frame and roughness parameters from the specular lumitexel, using L-BFGS-B [40]. Finally, with the fixed shading frame and roughnesses, we compute the RGB diffuse/specular albedos, by solving non-negative linear least squares, constrained by the original photometric measurements, similar to [8]. An illustration is shown in the latter part of Fig. 3.

VII. RESULTS & DISCUSSIONS

We capture the reflectance of 6 sets of near-planar physical samples (40 distinct samples in total) with a wide variation in appearance. For a set of 12/32 lighting patterns, it takes 6/15 seconds in total to capture high-dynamic-range (HDR) images using exposure bracketing. Similar to [10], a lighting pattern that contains both positive and negative weights is split into two for physical realization: one containing all positive weights with others set to zero, and the other with all negative weights sign-flipped and others set to zero. Throughout this article, we report the number of physically realized lighting patterns for consistency.

All computation is done on a workstation with dual Intel Xeon 4210 CPUs, 256 GB DDR4 memory and 4 NVIDIA GeForce RTX 3090 GPUs. It takes on average 72 hours to train our network for 1 million iterations. The latent autoencoder takes 60 hours to pre-train. At runtime, it takes 5 minutes for our network to decode 1 million pairs of diffuse/specular lumitexels from measurements, and 1.5 hours for the subsequent GGX parameter fitting. The timing is comparable to existing work like LDAE. We use a spatial resolution of $1024^2$ to store all GGX parameters.

Fig. 16 visualizes our lighting patterns, the patterns trained using LDAE along with those computed by PCA on anisotropic training samples. Our patterns exhibit higher-frequency details. In Fig. 17, the gating result at each pixel (i.e., the decoder index with the highest $Pr$) is visualized. Our gating module automatically learns to cluster pixels with similar high-dimensional appearance for efficient processing.

To see what lumitexels each decoder is tuned to, we show in Fig. 7 the average lumitexel among all that are sent to a particular decoder by our gating module, computed over 100 K randomly sampled lumitexels. Moreover, the percentage of these lumitexels whose maximum predicted $Pr$ is above a threshold is plotted in Fig. 8: there is almost always a dominating decoder.

In Fig. 17, we show reflectance fitting results of 4 physical sample sets with our network ($# = 32$) as well as 2 sets using a different lighting pattern number ($# = 12$), in the form of texture maps that represent GGX parameters. Our network separates the diffuse and specular reflections, estimates challenging anisotropic reflectance and produces high-quality normal maps. The smallest estimated roughness is about 0.03 on the coins. It is interesting to observe that how the highly complex appearance on the banknotes in the PAPER set is modeled by our approach. In addition, please refer to the accompanying video for rendering results of the sample sets with novel view and lighting conditions.

A. Comparisons

We validate our results against photographs, and compare with LDAE with the same number of lighting patterns ($# = 32$) in Fig. 9. In all cases, our network produces results that more closely resemble the corresponding photographs with a novel lighting condition not used in training, compared with LDAE; superior quantitative errors in SSIM are also reported, demonstrating our improved efficiency (i.e., effective sampled information per lighting pattern).

In Fig. 10, our network ($# = 12$) is compared with LDAE ($# = 32$), both of which have similar validation losses on either of the two sample sets, according to Fig. 12. Our results are
Fig. 17. GGX model fitting and gating results using our network ($\# = 32$ for the top 4 sample sets and $\# = 12$ for the remaining two). 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 $\alpha_x / \alpha_y$ is visualized in the red/green channel. We color-code the index of the decoder with the maximum predicted probability at each pixel in the last column; on the top-right corner of each image, a histogram of decoder selection is additionally visualized: each inset has a resolution of $16 \times 8$, representing 128 decoders; each pixel indicates the number of times that the gating network selects the corresponding decoder across the current sample: the blue-to-yellow visualization represents a range from 0 to 5000.

B. Evaluations

In this section, we evaluate the impact of various parameters/factors over the prediction quality of our network.

First, we validate the reconstruction quality of our latent-transform module in Fig. 11. A wide variety of lumitexels, including highly anisotropic/specular ones, can be faithfully reconstructed. In Fig. 13, we evaluate the impact of the number of lighting patterns over lumitexel reconstruction quality. With the reduction in the number of lighting patterns, the reconstruction quality decreases. Nevertheless, the specular highlight shape is well preserved with our approach using as few as 12 lighting patterns. The quality is comparable with LDAE at a higher number of patterns ($\# = 32$). 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 different settings.

We plot the validation losses of different networks with different parameters in Fig. 12, representing the average reconstruction quality of lumitexels. The horizontal axis indicates the input number of lighting patterns ($\#$), and the vertical axis comparable to LDAE qualitatively and quantitatively, with respect to the corresponding photograph. Note that we need only about 1/3 the number of input images, showing a considerable increase in efficiency.
shows the network loss $L$ (7). Note that our $L$ is computed on the prediction with the highest $Pr$. For the vanilla version, our network consistently outperforms LDAE at the same # (cf. Fig. 13), marked as yellow and red dots. Since the size of our network is about twice that of LDAE, we double the capacity of their network and find that the validation loss stays on the same level, marked as LDAE(2x). This demonstrates the benefit of our architecture over LDAE at similar capacities. We also switch the lighting patterns in our network to fixed ones, obtained by applying principal component analysis to a large number of synthetic lumitexels, marked as PCA. The loss increases substantially, demonstrating the benefit of our jointly trained lighting patterns.

We further study the impact of the number of decoders over reconstruction quality, given a fixed network size, in Figs. 8 and 14. As the lower half of Fig. 8 indicates, more smaller decoders are preferred over fewer bigger ones, though at the cost of increased training time. That being said, if the number of decoders gets too large (>256 in our case), the capacity of each decoder may be insufficient for accurate predictions. Examples of reconstructed specular lumitexels with different networks are visualized in Fig. 14.

Finally, sensitivity tests over our gated MoE-enhanced network are performed in Fig. 15. Random Gaussian noise ($\mu = 1$, $\sigma = 30\%$) is multiplied to each synthetic measurement to account for possible noise not modeled in the acquisition process. For each row in Fig. 15, we keep a sample if its decoder with the highest $Pr$ is different from any of the previous samples. These samples are then sorted according to specular lumitexels reconstruction error (7) with $\lambda_d = 0$, and displayed alongside with the ground-truth. The results show that the output are consistent across decoders whose index is close (measured with Hamming distance) to the one with the highest $Pr$ when no noise perturbation is added. Here we do not observe undesired discontinuity in output lumitexels, when the decoder index is flipped by 1 or 2 bits.

C. Generalizations

Our framework is not coupled with near-planar reflectance, nor is it limited to our setup. To demonstrate its generality, we extend to improve a state-of-the-art free-form scanning technique for non-planar reflectance [8]. Their original network consists of two parts. The first part converts image measurements at different conditions to a 1,024D global feature vector, and the second further transforms the feature vector into a diffuse/specular lumitexel. To apply our idea, we modify the second part of their network as follows: first, a gating module that conditions on the global feature vector is added; it then selects one out of 64 decoders, each of which produces a 128D latent vector as output; finally, the latent vector is converted to a diffuse/specular lumitexel via a pre-trained latent-transform module (Section VI-A).

With this modification, we considerably reduce the validation loss from 10.3 to 8.3. As visualized in Fig. 16, our enhanced network also more precisely predicts the reflectance (i.e., no more “hallucinated” highlight on Bowser’s hat) on a physical sample.
[18] S. R. Marschner, S. H. Westin, E. P. F. Lafontaine, K. E. Torrance, and D. P. Greenberg, “Image-based BRDF measurement including human skin,” in Proc. 10th Eurographics Conf. Rendering, 1999, pp. 131–144.

[19] H. P. A. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H.-P. Seidel, “Image-based reconstruction of spatial appearance and geometric detail,” ACM Trans. Graph., vol. 22, no. 2, pp. 234–257, Apr. 2003.

[20] J. Wang, S. Zhao, X. Tong, J. Snyder, and B. Guo, “Modeling anisotropic surface reflectance with example-based microfacet synthesis,” ACM Trans. Graph., vol. 27, no. 3, pp. 41:1–41:9, Aug. 2008.

[21] Y. Dong et al., “Manifold bootstrapping for SVBRDF capture,” ACM Trans. Graph., vol. 29, no. 4, pp. 98:1–98:10, Jul. 2010.

[22] Z. Hui, K. Sunkavalli, J.-Y. Lee, S. Hadap, J. Wang, and A. C. Sankaranarayanan, “Reflectance capture using univariate sampling of BRDFs,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 5372–5380.

[23] G. Nam, J. H. Lee, D. Gutierrez, and M. H. Kim, “Practical SVBRDF acquisition of 3D objects with unstructured flash photography,” in Proc. SIGGRAPH Asia Tech. Papers, 2018, Art. no. 267.

[24] P. Ren, J. Wang, J. Snyder, X. Tong, and B. Guo, “Pocket reflectometry,” ACM Trans. Graph., vol. 30, no. 4, pp. 1–10, 2011.

[25] G. Nam, J. H. Lee, H. Wu, D. Gutierrez, and M. H. Kim, “Simultaneous acquisition of microscale reflectance and normals,” ACM Trans. Graph., vol. 35, no. 6, pp. 185:1–185:11, Nov. 2016.

[26] M. Aittala, T. Weyrich, and J. Lehtinen, “Two-shot SVBRDF capture for stationary materials,” ACM Trans. Graph., vol. 34, no. 4, pp. 110:1–110:13, Jul. 2015.

[27] M. Aittala, T. Aila, and J. Lehtinen, “Reflectance modeling by neural texture synthesis,” ACM Trans. Graph., vol. 35, no. 4, pp. 65:1–65:13, Jul. 2016.

[28] X. Li, Y. Dong, P. Peers, and X. Tong, “Modeling surface appearance from a single photograph using self-augmented convolutional neural networks,” ACM Trans. Graph., vol. 36, no. 4, pp. 45:1–45:11, Jul. 2017.

[29] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau, “Single-image SVBRDF capture with a rendering-aware deep network,” ACM Trans. Graph., vol. 37, no. 4, pp. 128:1–128:15, Jul. 2018.

[30] V. Deschaintre, M. Aittala, F. Durand, G. Drettakis, and A. Bousseau, “Flexible SVBRDF capture with a multi-image deep network,” Comput. Graph. Forum, vol. 38, no. 4, pp. 1–13, 2019.

[31] D. Guo, X. Li, Y. Dong, P. Peers, K. Xu, and X. Tong, “Deep inverse rendering for high-resolution SVBRDF estimation from an arbitrary number of images,” ACM Trans. Graph., vol. 38, no. 4, pp. 134:1–134:15, Jul. 2019.

[32] Y. Guo, C. Smith, M. Halan, K. Sunkavalli, and S. Zhao, “MaterialGAN: Reflectance capture using a generative SVBRDF model,” ACM Trans. Graph., vol. 39, no. 6, pp. 254:1–254:13, Nov. 2020.

[33] X. Zhou and N. K. Kalantari, “Adversarial single-image SVBRDF estimation with hybrid training,” Comput. Graph. Forum, vol. 40, no. 2, pp. 315–325, 2021.

[34] B. Walter, S. R. Marschner, H. Li, and K. E. Torrance, “Microfacet models for refraction through rough surfaces,” in Proc. 18th Eurographics Conf. Rendering Techn., 2007, pp. 195–206.

[35] J. Guo, Y. Guo, J. Pan, and W. Lu, “BRDF analysis with directional statistics and its applications,” IEEE Trans. Vis. Comput. Graph., vol. 26, no. 3, pp. 1476–1489, Mar. 2020.

[36] B. Hu, J. Guo, Y. Chen, M. Li, and Y. Guo, “DeepBRDF: A deep representation for manipulating measured BRDF,” Comput. Graph. Forum, vol. 39, pp. 157–166, May 2020.

[37] G. Rainer, A. Ghosh, W. Jakob, and T. Weyrich, “Unified neural encoding of BTFs,” Comput. Graph. Forum, vol. 39, no. 2, pp. 167–178, 2020.