

# A Non-parametric Sparse BRDF Model

ANONYMOUS AUTHOR(S)  
SUBMISSION ID: PAPERS\_632S1

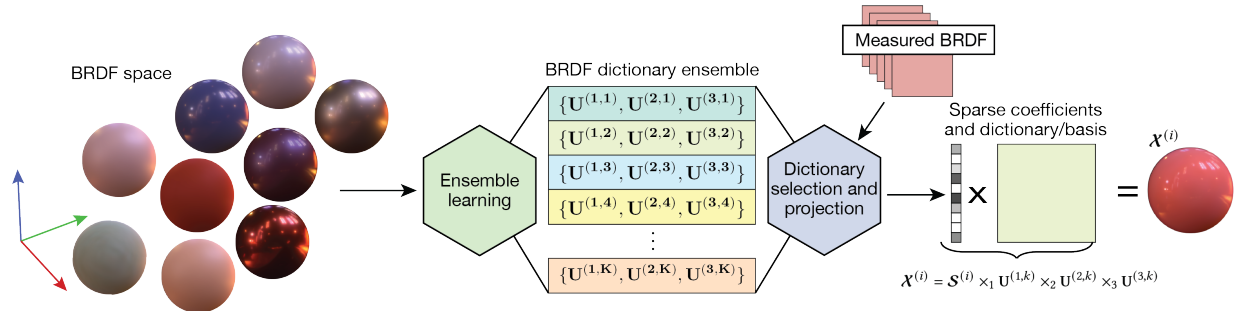


Fig. 1. shows an overview of the proposed framework for learning accurate representations and sparse data-driven BRDF models through analysis of the space of BRDFs. The BRDF dictionary ensemble is trained once and can accurately represent a wide range of previously unseen materials.

Accurate modeling of measured material properties described by the bidirectional reflectance distribution function (BRDF) is a key component in photo-realistic and physically-based rendering. Current data-driven models are based on either analytical basis functions or tensor decompositions. Analytical representations are usually efficient in terms of memory footprint and computational complexity but typically lead to larger approximation errors. Most decomposition methods operate on individual BRDFs and come at a larger computational cost and require larger number of coefficients to achieve high quality results.

This paper presents a novel non-parametric BRDF model derived using a machine learning approach to explore the space of possible BRDFs and to span this space with a set of sub-spaces, or dictionaries. By training the dictionaries under sparsity constraints, the model guarantees high quality representations with minimal storage requirements and an inherent clustering of the BRDF-space. The model can be trained once and then reused to represent a wide variety of measured BRDFs. Moreover, the proposed method is robust to BRDF transformations, and is flexible to incorporate new unseen data sets, parameterizations, and transformations. The proposed sparse BRDF model is evaluated using the MERL, DTU and RGL-EPFL BRDF databases. Experimental results show that the proposed approach results in about 9.7dB higher SNR on average for rendered images as compared to current state-of-the-art models.

CCS Concepts: • **Computing methodologies** → **Reflectance modeling; Machine learning approaches.**

Additional Key Words and Phrases: Rendering, Reflectance and shading models

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2021 Association for Computing Machinery.  
0730-0301/2021/1-ART \$15.00

<https://doi.org/10.1145/nmnnmnnn.nmnnmnnn>

## ACM Reference Format:

Anonymous Author(s). 2021. A Non-parametric Sparse BRDF Model. *ACM Trans. Graph.* 1, 1 (January 2021), 12 pages. <https://doi.org/10.1145/nmnnmnnn.nmnnmnnn>

## 1 INTRODUCTION

The bidirectional reflectance distribution function [Nicodemus et al. 1992] describes how light scatters at the surfaces of a scene, depending on their material characteristics. The BRDF is a 4D function parameterized by the incident and exitant scattering angles and can be described using either parametric models [Ashikhmin and Shirley 2000; Blinn 1977; Cook and Torrance 1982; Löw et al. 2012; Walter et al. 2007] or data-driven models [Bagher et al. 2016; Bilgili et al. 2011; Lawrence et al. 2004; Tongbuasirilai et al. 2019]. Parametric models present great artistic freedom and the possibility to interactively tweak parameters to achieve the desired look and feel. However, most analytical models are not designed for efficient and accurate representation of the scattering properties of measured real-world materials. Data-driven models on the other hand enable the use of measured BRDFs and real-world materials directly in the rendering pipeline, and are commonly used in computer vision applications [Romeiro et al. 2008]. Here we will focus on data-driven models and learning accurate representations describing the space of possible BRDFs.

Data-driven models can represent BRDFs in many different ways. Iterative-factored representations approximate BRDFs with multiple low-rank components [Bilgili et al. 2011; Lawrence et al. 2004; Tongbuasirilai et al. 2019], while hybrid analytical data-driven models [Bagher et al. 2016; Sun et al. 2018] rely on non-parametric components or basis functions computed using specific weighting and optimization schemes.

The efficiency, or performance, of a non-parametric model is typically measured in terms of the number of variables/coefficients required to represent a BRDF at a given quality and the efficacy of the underlying basis representation. Most, if not all, existing methods either sacrifice the model accuracy to achieve fast reconstruction

for real-time applications, or aim for high image fidelity leading to increasing storage and computational requirements. Another important aspect is the complexity of the basis functions used in the representation. At one end of the spectrum, we have analytical basis functions such as spherical harmonics and wavelets [Clausen et al. 2003; Ramamoorthi and Hanrahan 2001], which provide compact and computationally efficient representations but suffer from low approximation accuracy. On the other end, we have decomposition based methods [Bilgili et al. 2011] that model the BRDF as a multiplication of a set of coefficients and a basis matrix/tensor computed from data. Unfortunately, these approaches require a computationally expensive decomposition, e.g. PCA or SVD, for each BRDF individually and suffer from a high storage cost for the basis itself. Another problem is that the expressiveness of existing bases/decomposition methods is limited. Except for a few, they are in most cases also not designed for BRDF data, hence requiring high numbers of coefficients for accurate BRDF representation.

The goal in this paper is to develop a new data-driven BRDF model that enables high accuracy representation with a minimal number of coefficients, as well as a basis representation that can be trained once and is expressive enough to represent any BRDF. To solve this challenge, we derive a model that in essence relies on decomposing BRDFs into a coefficient – basis pair but uses machine learning to adapt the basis to the space of BRDFs and minimize its memory footprint while providing maximally sparse coefficients. Sparse BRDF modeling is achieved using a novel BRDF dictionary ensemble and a novel model selection algorithm to efficiently represent a wide range of real-world materials. The learned dictionary ensemble consists of a set of basis functions trained such that they guarantee a very sparse BRDF representation and near optimal signal reconstruction. Moreover, our model takes into account the multidimensional structure of measured BRDFs (e.g. 3D or 4D depending on the parameterization) and can exploit the information redundancy in the entire BRDF space to reduce the number of coefficients.

The learned ensemble is versatile and can be trained only once to be reused for representing a wide range of previously unseen materials. Additionally, the dictionary ensemble is not restricted to a single BRDF transformation as previous models. Instead multiple BRDF transformations can be included in the ensemble training, such that for each individual BRDF, the best representation can be selected automatically and used. We also develop a novel model selection method to pick a dictionary in the ensemble that leads to the sparsest solution, the smallest reconstruction error, and the most suitable transformation with respect to rendering quality. For the experiments and evaluations presented here, we use the MERL [Matusik et al. 2003] and RGL-EPFL [Dupuy and Jakob 2018] databases, which are divided into a training set and a test set used for evaluation. The main contributions of this paper can be summarized as follows:

- A novel non-parametric BRDF model using sparse representations that significantly outperforms existing decomposition-based methods with respect to both model error and rendering quality.
- A multidimensional dictionary ensemble learning method tailored to measured BRDFs.

- A novel BRDF model selection method that chooses the best dictionary for efficient BRDF modeling, as well as the most suitable BRDF normalization function. This enables a unified non-parametric BRDF model regardless of the characteristics of the material.

We compare the proposed non-parametric BRDF to current state-of-the-art models and demonstrate that it performs significantly better in terms of both rendering SNR and visual quality. To the authors' knowledge this is the first BRDF model based on sparse representations and dictionary learning.

**Notations-** Throughout the paper, we use the following notational convention. Vectors and matrices are denoted by boldface lower-case (**a**) and bold-face upper-case letters (**A**), respectively. Tensors are denoted by calligraphic letters, e.g.  $\mathcal{A}$ . A finite set of objects is indexed by superscripts, e.g.  $\{\mathbf{A}^{(i)}\}_{i=1}^N$ , whereas individual elements of **a**, **A**, and  $\mathcal{A}$  are denoted  $a_i$ ,  $A_{i_1, i_2}$ ,  $\mathcal{A}_{i_1, \dots, i_n}$ , respectively. The  $\ell_p$  norm of a vector **s**, for  $1 \leq p \leq \infty$ , is denoted by  $\|\mathbf{s}\|_p$ . Frobenius norm is denoted  $\|\mathbf{s}\|_F$ . The  $\ell_0$  pseudo-norm of this vector,  $\|\mathbf{s}\|_0$ , defines the number of non-zero elements.

## 2 BACKGROUND AND RELATED WORK

Measured BRDFs have proven to be an important tool in achieving photo-realism during rendering [Dong et al. 2016; Dupuy and Jakob 2018; Matusik et al. 2003]. Even highly-complex surfaces such as layered materials require multiple components of measured data to construct novel complex materials [Jakob et al. 2014]. Measured materials, however, are high-dimensional signals with large memory footprint and a key challenge is that small approximation errors can lead to visual artifacts during rendering. To efficiently represent such high-dimensional measured BRDF data, one can use parametric models, or data-driven models, since densely-sampled BRDF data imposes a large memory footprint, making it impractical to use in many applications.

**Parametric models.** By careful modeling, BRDFs can be encoded with only a few parameters. The components or factors of such models are based on either assumptions describing by the physics of light – surface interactions using e.g. microfacet theory [Cook and Torrance 1982; Holzschuch and Pacanowski 2017; Walter et al. 2007], or empirical observations of BRDF behaviors [Ashikhmin and Shirley 2000; Blinn 1977; Löw et al. 2012; Ward 1992]. However, in many practical cases and applications, parametric models cannot accurately fit measured real-world data [Bagher et al. 2016].

**Data-driven models.** Due to their non-parametric property, data-driven models are superior to parametric models in that the number of degrees of freedom, or implicit model parameters, is much higher. This means that the representative power is higher and the expected approximation error is lower. Factored BRDF models use decomposition techniques to factorize BRDF into several components. Matrix and tensor decompositions have been used by Lawrence et al. [2004], Bilgili et al. [2011], and Tongbuasirilai et al. [2019]. Moreover, factored-based models for interactive BRDF editing have been presented in [Ben-Artzi et al. 2006; Kautz and McCool 1999].

A problem with existing factored models is that rank-1 approximations in most cases lead to inferior results. Accurate modeling

requires iterative solutions with many layered factors. Analytic-data-driven BRDF models [Bagher et al. 2016; Sun et al. 2018] employ analytical models extended to higher number of parameters fitted with measured data to achieve higher accuracy. The recent advancement of machine learning algorithms, in particular deep learning, brings new research paths on BRDF-related topics [Dong 2019]. Deep learning has been used for BRDF editing [Hu et al. 2020] and BRDF acquisition [Deschaintre et al. 2018, 2019; Li et al. 2018]. To the best of our knowledge, deep learning has not been applied to BRDF modeling.

**Dictionary Learning.** One of the most commonly used dictionary learning methods is K-SVD [Aharon et al. 2006], and its many variants [Marsousi et al. 2014; Mazhar and Gader 2008; Mukherjee et al. 2016; Rusu and Dumitrescu 2012], where a 1D signal (i.e. a vector) is represented as a linear combination of a set of basis vectors, called atoms. A clear disadvantage of K-SVD for BRDF representation is signal dimensionality. For instance, a measured BRDF in the MERL data set, excluding the spectral information, is a  $90 \times 90 \times 180 = 1,458,000$  dimensional vector. In practice, the number of data points needed for K-SVD dictionary training should be a multitude of the signal dimensionality to achieve a high quality dictionary. In addition to unfeasible computational power required for training, the limited number of available measured BRDF data sets renders the utilization of K-SVD impractical.

In contrast to 1D dictionary learning methods, multidimensional dictionary learning has received only little attention in the literature [Ding et al. 2017; Hawe et al. 2013; Roemer et al. 2014]. In multidimensional dictionary learning, a data point is treated as a tensor, and a dictionary is trained along each mode. For instance, given our example above, instead of training one 1,458,000 dimensional dictionary for the MERL data set, one can train three dictionaries (i.e. one for each mode), where the atom size for these dictionaries are 90, 90 and 180, corresponding to the dimensionality of each mode. To the best of our knowledge, there exists only a few multidimensional dictionary learning algorithms. Our sparse BRDF model in this paper is inspired by the multidimensional dictionary ensemble training proposed in [Miaudji et al. 2019], which has been shown to perform well for high dimensional signals such as light fields and light field videos. We will elaborate on our training scheme for BRDFs in Section 3.2.

### 3 SPARSE DATA DRIVEN BRDF MODEL

Our non-parametric model is based on learning a set of multidimensional dictionaries, a dictionary ensemble, spanning the space of BRDFs, i.e. the space in which each BRDF is a single multidimensional point. Each dictionary in the ensemble consists of a set of basis functions, representing each dimension of the BRDF space, that admit sparse representation of any measured BRDF using only a small number of coefficients as illustrated in Figure 1. The dictionary ensemble is trained only once on a given training set of measured BRDFs and can then be reused to represent a wide range of different BRDFs. This is in contrast to previous models that use tensor or matrix decomposition techniques, where the basis and the coefficients are calculated for each BRDF individually.

A major challenge when using machine learning methods, and in particular dictionary learning, on BRDFs is the high dynamic range inherent to the data. In Section 3.1, we describe two data transformations that when applied on measured BRDFs, they improve the fitting to our non-parametric model, see Section 4. The training of the multidimensional dictionaries is described in sections 3.2 and 3.3, followed by our model selection technique in Section 3.4, where we describe a method to select the most suitable dictionary (among the ensemble of dictionaries) for any unseen BRDF such that the coefficients are maximally sparse, the modeling error is minimal, and that the data transformation used is one that leads to a better rendering quality.

A BRDF can be parameterized in many different ways [Barla et al. 2015; Löw et al. 2012; Rusinkiewicz 1998; Stark et al. 2005]. Our dictionary learning approach does not rely on the parameterization of given BRDFs as long as the resolution of these BRDFs is the same. For simplicity, all the data sets we use here are based on the Rusinkiewicz’s parameterization [Rusinkiewicz 1998] at a resolution of  $90 \times 90 \times 180$ .

#### 3.1 BRDF data transformation

Measured BRDF tensors often exhibit a very high dynamic range, which introduces many difficulties during parameter fitting and optimization. It is therefore necessary to apply a transformation of the BRDF values using e.g. a log-mapping as suggested by [Löw et al. 2012; Tongbuasirilai et al. 2019] and [Nielsen et al. 2015; Sun et al. 2018]. In this paper we use two data transformation functions to improve the performance of our model during training and testing. The first transformation is based on *log-plus transformation* proposed by Löw et al., [Löw et al. 2012]:

$$\rho_{l1}(\omega_h, \omega_d) = \log(\rho(\omega_h, \omega_d) + 1) \quad (1)$$

where  $\rho$  is the original BRDF value, and  $\rho_{l1}$  is the transformed BRDF value. For the second transformation, we use the *log-relative mapping* proposed by Nielsen et al. [Nielsen et al. 2015]; however, we exclude the denominator. We call this transformation *log-plus-cosine transformation*:

$$\rho_{l2}(\omega_h, \omega_d) = \log(\rho(\omega_h, \omega_d) * \cosMap(\omega_h, \omega_d) + 1) \quad (2)$$

where  $\cosMap()$  is a function mapping the input  $(\omega_h, \omega_d)$  directions to  $\cos(\theta_i) * \cos(\theta_o)$  to suppress noise in grazing and near grazing angles.

Using the proposed non-parametric model, we have conducted experiments using both transformations, see Table 1. The log-plus transformation in Equation 1 yields better results when compared to the log-plus-cosine transformation in Equation 2 for specular materials. The log-plus-cosine is in most cases a better choice for diffuse BRDFs.

While we use the two most commonly used BRDF transformations, our sparse BRDF model is not limited to the choice of the transformation function. Indeed, given any new such function, the previously trained dictionary ensemble can be directly applied. However, to further improve the model accuracy, one can train a small set of dictionaries given a training set obtained with the new BRDF transformation. We then add this set to the previously trained ensemble of dictionaries. The expansion of the dictionary ensemble

Table 1. SNR of rendered images using the BRDF dictionaries trained with different dictionary sparsity levels: 32, 64, 128, and 256. Each dictionary has two transformations,  $\rho_{t1}$  and  $\rho_{t2}$ . The test set consists of 15 MERL materials (not included in the training). The bottom row shows the average SNR over the test set. The underlined numbers are best SNR values for  $\rho_{t1}$  and the bold numbers are the best SNR values for  $\rho_{t2}$ .

Material	Ensemble with $\tau_l = 32$		Ensemble with $\tau_l = 64$		Ensemble with $\tau_l = 128$		Ensemble with $\tau_l = 256$	
	$\rho_{t1}$ SNR(dB)	$\rho_{t2}$ SNR(dB)	$\rho_{t1}$ SNR(dB)	$\rho_{t2}$ SNR(dB)	$\rho_{t1}$ SNR(dB)	$\rho_{t2}$ SNR(dB)	$\rho_{t1}$ SNR(dB)	$\rho_{t2}$ SNR(dB)
blue-fabric	53.9003	58.4393	57.0890	61.1925	56.7419	62.4704	58.7229	<b>62.9932</b>
blue-metallic-paint	<u>54.8105</u>	56.8017	52.4779	59.7930	54.2249	<b>61.0643</b>	52.5073	60.5738
dark-red-paint	44.1094	51.9677	45.8218	52.4098	<u>48.4695</u>	<b>54.7743</b>	46.3005	54.4020
gold-metallic-paint2	<u>46.9514</u>	38.6907	45.6783	36.0324	46.1956	37.4564	42.4208	<b>41.1227</b>
green-metallic-paint2	50.7108	41.8161	49.4635	39.3023	<u>52.8230</u>	43.0459	49.6811	<b>50.2204</b>
light-red-paint	41.4139	49.0550	43.7449	48.7451	<u>47.7306</u>	<b>52.1905</b>	45.1613	50.6002
pink-fabric2	44.8244	49.3862	48.5446	52.5484	<u>52.6230</u>	53.5701	52.5405	<b>54.4938</b>
purple-paint	43.8932	38.8859	42.2491	47.5648	<u>48.2735</u>	<b>48.7324</b>	45.3798	47.1568
red-fabric	47.5606	52.3038	50.9287	54.7831	53.9668	56.7085	<u>55.3687</u>	<b>58.5863</b>
red-metallic-paint	47.2351	40.3386	46.9943	38.4251	<u>49.1860</u>	42.1207	48.6971	<b>42.6229</b>
silver-metallic-paint2	40.3291	42.9256	44.0442	43.2292	44.0323	<b>46.8208</b>	46.4961	44.1504
specular-green-phenolic	48.4841	41.6432	47.3226	36.5157	<u>49.4785</u>	<b>48.8586</b>	49.2522	45.9519
specular-violet-phenolic	48.2384	42.7994	47.4994	37.9801	47.4863	<b>44.5840</b>	<u>48.3638</u>	41.3332
specular-yellow-phenolic	<u>46.4907</u>	<b>39.1758</b>	44.5259	36.1666	45.4146	35.4231	43.1846	36.4720
violet-acrylic	<u>48.7179</u>	<b>44.0610</b>	48.7112	38.9536	47.6828	42.0749	48.1322	36.7368
<b>Average</b>	47.1779	45.8860	47.6730	45.5761	49.6219	48.6596	48.8139	48.4944

is a unique characteristic of our model. We utilize this property in Section 3.3 to combine different sets of dictionaries, each trained with a distinct training sparsity. The same approach can be used here for improving the model accuracy when a new measured BRDF data set, that requires a more sophisticated transformation, is given.

### 3.2 Multidimensional dictionary learning for BRDFs

To build the non-parametric BRDF model, we seek to accurately model the space of BRDFs using basis functions leading to a high degree of sparsity for the coefficients while maintaining the visual fidelity of each BRDF in the training set. To achieve this, the training algorithm needs to take into account the multidimensional nature of BRDF objects, typically 3D or 4D, depending on the parameterization. Let  $\{\mathcal{X}^{(i)}\}_{i=1}^N$  be a set of  $N$  BRDFs, where  $\mathcal{X} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ . Here we do not assume any specific parameterization and only require that all the BRDFs in  $\{\mathcal{X}^{(i)}\}_{i=1}^N$  have the same resolution. Moreover, as discussed in Section 3.1, we utilize two BRDF transformations,  $\rho_{t1}$  and  $\rho_{t2}$ . As a result, the training set consists of two versions of each BRDF. In other words, the dictionary ensemble is trained on both transformations only once.

To achieve a sparse three-dimensional representation of  $\{\mathcal{X}^{(i)}\}_{i=1}^N$ , we train an ensemble of  $K$  three-dimensional dictionaries, denoted  $\{\mathbf{U}^{(1,k)}, \mathbf{U}^{(2,k)}, \mathbf{U}^{(3,k)}\}_{k=1}^K$ , such that each BRDF,  $\mathcal{X}^{(i)}$ , can be decomposed as

$$\mathcal{X}^{(i)} = \mathbf{S}^{(i)} \times_1 \mathbf{U}^{(1,k)} \times_2 \mathbf{U}^{(2,k)} \times_3 \mathbf{U}^{(3,k)}, \quad (3)$$

where  $\mathbf{U}^{(1,k)} \in \mathbb{R}^{m_1 \times m_1}$ ,  $\mathbf{U}^{(2,k)} \in \mathbb{R}^{m_2 \times m_2}$ ,  $\mathbf{U}^{(3,k)} \in \mathbb{R}^{m_3 \times m_3}$ , and  $k \in \{1, \dots, K\}$ . Moreover, we have  $\|\mathbf{S}^{(i)}\|_0 \leq \tau$ , where  $\tau$  is a user-defined sparsity parameter. It is evident from (3) that each BRDF is represented using *one* dictionary in the ensemble, in this case  $\{\mathbf{U}^{(1,k)}, \mathbf{U}^{(2,k)}, \mathbf{U}^{(3,k)}\}$ .

The ensemble training is performed by solving the following optimization problem

$$\min_{\mathbf{U}^{(j,k)}, \mathbf{S}^{(i,k)}, \mathbf{M}_{i,k}} \sum_{i=1}^N \sum_{k=1}^K \mathbf{M}_{i,k} \left\| \mathcal{X}^{(i)} - \mathbf{S}^{(i,k)} \times_1 \mathbf{U}^{(1,k)} \times_2 \mathbf{U}^{(2,k)} \times_3 \mathbf{U}^{(3,k)} \right\|_F^2 \quad (4a)$$

subject to

$$\left( \mathbf{U}^{(j,k)} \right)^T \mathbf{U}^{(j,k)} = \mathbf{I}, \quad \forall k = 1, \dots, K, \quad \forall j = 1, \dots, 3, \quad (4b)$$

$$\|\mathbf{S}^{(i,k)}\|_0 \leq \tau, \quad (4c)$$

$$\sum_{k=1}^K \mathbf{M}_{i,k} = 1, \quad \forall i = 1, \dots, N, \quad (4d)$$

where the matrix  $\mathbf{M} \in \mathbb{R}^{N \times K}$  is a clustering matrix associating each BRDF in the training set to one multidimensional dictionary in the ensemble; moreover, Equation (4b) ensures the orthogonality of the dictionary, that the sparsity of the coefficients is enforced by (4c), and that the representation of each BRDF with *one* dictionary is achieved by (4c). The user-defined parameter  $\tau_l$  defines the training sparsity. It should be noted that the clustering matrix  $\mathbf{M}$  divides the BRDFs in the training set into a set of clusters such that optimal sparse representation is achieved with respect to the number of model parameters (or coefficients) and the representation error. This clustering is an integral part of our model and improves the accuracy of BRDF representations.

Our sparse BRDF modeling is inspired by the Aggregate Multidimensional Dictionary Ensemble (AMDE) proposed by Miandji et al. [Miandji et al. 2019]. However we do not perform pre-clustering of data points, in this case BRDFs, for the following two reasons:



First, the number of existing measured BRDF data sets is very limited. Hence, if we apply pre-clustering, the number of available BRDFs to train a dictionary ensemble becomes inadequate. Second, since we use each BRDF as a data point, the size of each data point is  $90 * 90 * 180 = 1458000$ , hence rendering the proposed pre-clustering method in [Miandji et al. 2019] impractical. Indeed, the two BRDF transformations discussed in Section 3.1 can be seen as a pre-clustering of the training set. These transformations divide the training set into diffuse and glossy BRDFs. Moreover, as it will be described in Section 3.3, and unlike the method of Miandji et al. [Miandji et al. 2019], we perform multiple trainings of the same training set but with a different training sparsity  $\tau_l$ . The obtained ensembles are combined to form an ensemble that can efficiently represent BRDFs with less reconstruction error.

### 3.3 BRDF Dictionary ensemble with multiple sparsities

Measured BRDFs exhibit a variable degree of sparsity. Indeed given a suitable dictionary, a diffuse material requires only a small number of coefficients while a highly glossy BRDF needs a significantly higher number of coefficients for an accurate representation. This phenomenon has been observed by previous work on non-parametric modeling of BRDFs based on factorization or using commonly known basis functions such as spherical harmonics [Lawrence et al. 2004; Nielsen et al. 2015; Sun et al. 2018; Tunwattanapong et al. 2013]. A shortcoming of the dictionary ensemble learning method described in Section 3.2 is that we do not take into account the intrinsic sparsity of various materials in the training set. In other words, since the training sparsity  $\tau_l$  is fixed for all the BRDFs in the training set, a small values for  $\tau_l$  will steer the optimization algorithm to more efficiently model low frequency (or diffuse-like) materials, while neglecting high frequency materials. Indeed if a large value for  $\tau_l$  is used, the opposite happens, leading to degradation of quality for diffuse materials due to over-fitting.

In Table 1, we present rendering SNR results obtained from ensembles trained with different values for the training sparsity,  $\tau_l$ ; in particular, we use four ensembles with  $\tau_l = 32$ ,  $\tau_l = 64$ ,  $\tau_l = 128$ , and  $\tau_l = 256$ . Note that the set of 15 materials we consider here were not used in the training set, which consists of 85 materials from the MERL data set. As it can be seen, there is a relatively large gap in SNR for each material when we compare different ensembles, e.g. for  $\tau_l = 32$  and  $\tau_l = 256$ . Moreover, we also observe that most BRDFs in this set favor ensembles trained with  $\tau_l = 128$  and  $\tau_l = 256$ . This is because we set the testing sparsity to  $\tau_t = 262$ , see Section 3.4 for the definition of the testing sparsity. The relation between the training and testing sparsity is analyzed in [Miandji et al. 2019].

To address the problem mentioned above, we train multiple ensembles of dictionaries, each with a different value for  $\tau_l$ , so that we can model both low and high frequency details of the training BRDFs more efficiently, while lowering the risk of over-fitting. After training each ensemble according to the method described in Section 3.2, we combine them all to form one ensemble that includes all the dictionaries. In this paper, we train 4 ensembles, each with 8 dictionaries, which are trained with  $\tau_l = 32$ ,  $\tau_l = 64$ ,  $\tau_l = 128$ , and  $\tau_l = 256$ ; hence, the final ensemble consists of 32 dictionaries. In Section 3.4, we describe our model selection method to find a

dictionary in the combined ensemble that leads to the most sparse coefficients and the least reconstruction error.

### 3.4 BRDF model selection

Once the ensemble of dictionaries is trained, the next step is to use it for the sparse representation of BRDFs. We call this stage *model selection*, since out of the dictionaries in the ensemble and the transformations used on the BRDF, we need to find one dictionary that leads to the most sparse coefficients with the least error, as well as the best performing transformation between  $\rho_{t1}$  and  $\rho_{t2}$ . Indeed, as mentioned in Section 3.1, our method is not limited to the number of transformations.

We begin by describing our method for selecting the most suitable dictionary in the ensemble for BRDF reconstruction. This can be achieved by projecting each BRDF onto all the dictionaries in the ensemble. The projection step is formulated as

$$\hat{\mathbf{S}}^{(i,k)} = \mathbf{Y}^{(i)} \times_1 \left( \mathbf{U}^{(1,k)} \right)^T \times_2 \left( \mathbf{U}^{(2,k)} \right)^T \times_3 \left( \mathbf{U}^{(3,k)} \right)^T, \quad (5)$$

where  $\mathbf{Y}^{(i)}$  is a BRDF in the testing set that we like to obtain a sparse representation of. The smallest components in the coefficient tensors  $\hat{\mathbf{S}}^{(i,k)}$  are progressively nullified until we reach a user defined sparsity level, called the testing sparsity,  $\tau_t$ , or when the representation error becomes larger than a user defined threshold. The testing sparsity, which defines the model complexity, is different than the training sparsity  $\tau_l$  and we typically require  $\tau_t \geq \tau_l$ . For instance, a higher value for  $\tau_t$  is required for glossy materials than for diffuse to achieve an accurate BRDF representation. However, if storage cost is important, e.g. for real-time rendering applications, one can reduce  $\tau_t$  at the cost of degrading the rendering quality. Indeed, this provides a trade-off between quality and performance, making our model flexible enough to be applied in a variety of applications.

After sparsifying  $\hat{\mathbf{S}}^{(i,k)}$ ,  $\forall k \in \{1, \dots, K\}$ , we pick the dictionary corresponding to the sparsest coefficient tensor  $\hat{\mathbf{S}}^{(i,k)}$ ,  $\forall k \in \{1, \dots, K\}$ . If all the coefficient tensors  $\hat{\mathbf{S}}^{(i,k)}$ ,  $\forall k \in \{1, \dots, K\}$  achieve the same sparsity, we pick the dictionary corresponding to the least reconstruction error. The reconstruction error for a BRDF in the test set,  $\mathbf{Y}^{(i)}$ , modeled using a dictionary  $\{\mathbf{U}^{(1,k)}, \mathbf{U}^{(2,k)}, \mathbf{U}^{(3,k)}\}$ ,  $k \in \{1, \dots, K\}$ , is simply calculated as

$$\left\| \mathbf{Y}^{(i)} - \hat{\mathbf{S}}^{(i,k)} \times_1 \mathbf{U}^{(1,k)} \times_2 \mathbf{U}^{(2,k)} \times_3 \mathbf{U}^{(3,k)} \right\|_2^2. \quad (6)$$

Because the BRDF dictionary ensemble is trained once and can be used for the sparse representation of unobserved BRDFs, the storage cost of the model in Equation 3 is defined by the storage complexity of the sparse coefficient tensor  $\hat{\mathbf{S}}^{(i)}$  in Equation 5. We store the nonzero elements in  $\hat{\mathbf{S}}^{(i)}$  as the tuples of nonzero element location and value, denoted  $\{l_t^1, l_t^2, l_t^3, \hat{\mathbf{S}}_{l_t^1, l_t^2, l_t^3}^{(i)}\}_{t=1}^{\tau_t}$ , where the indices  $l_t^1$ ,  $l_t^2$ , and  $l_t^3$  store the location of the  $t$ th nonzero element of  $\hat{\mathbf{S}}^{(i)}$ , while the corresponding value is  $\hat{\mathbf{S}}_{l_t^1, l_t^2, l_t^3}^{(i)}$ .

The reconstruction of a given BRDF,  $\mathbf{Y}^{(i)}$ , using our model is computed by multiplying the sparse coefficient tensor  $\hat{\mathbf{S}}^{(i,k)}$ , where  $k$  is the index of the dictionary chosen by the model selection method

Table 2. Rendering SNR, Gamma-mapped-MSE, and MSE, obtained using our sparse BRDF model for  $\rho_{t1}$  and  $\rho_{t2}$ . For each quality metric, the best result between  $\rho_{t1}$  and  $\rho_{t2}$  is shown by bold numbers. Comparing the chosen transformation based on rendering SNR with Gamma-mapped-MSE and MSE in the BRDF space, we see that the Gamma-mapped-MSE can well distinguish the suitable transformation for 13 out of 15 materials. It can also be seen that MSE only selects the correct transformation for 3 out of 15 materials. For results generated using Gamma-mapped-MSE, we set  $\gamma = 2.0$ .

Material	Rendering SNR (dB)		Gamma-mapped-MSE		MSE	
	Our $\rho_{t1}$	Our $\rho_{t2}$	Our $\rho_{t1}$	Our $\rho_{t2}$	Our $\rho_{t1}$	Our $\rho_{t2}$
blue-fabric	53.99	<b>62.16</b>	0.0233	<b>0.0038</b>	<b>0.0002</b>	0.0062
blue-metallic-paint	51.65	<b>60.53</b>	0.0448	<b>0.0375</b>	<b>0.0011</b>	0.0317
dark-red-paint	49.16	<b>54.80</b>	0.0616	<b>0.0242</b>	<b>0.0295</b>	0.1209
gold-metallic-paint2	<b>48.29</b>	37.68	<b>0.9248</b>	0.9350	72.0330	<b>38.7850</b>
green-metallic-paint2	<b>57.48</b>	43.36	<b>0.8767</b>	0.8939	31.5660	<b>11.9140</b>
light-red-paint	46.51	<b>51.68</b>	0.0552	<b>0.0312</b>	<b>0.0567</b>	0.1699
pink-fabric2	52.66	<b>52.71</b>	0.0230	<b>0.0125</b>	<b>0.0003</b>	0.0323
purple-paint	45.81	<b>47.24</b>	0.1991	<b>0.1725</b>	2.8225	<b>2.1342</b>
red-fabric	<b>56.26</b>	55.05	0.0177	<b>0.0078</b>	<b>0.0002</b>	0.0172
red-metallic-paint	<b>52.70</b>	42.62	<b>1.2910</b>	1.3086	45.0140	<b>19.3610</b>
silver-metallic-paint2	<b>44.70</b>	44.55	0.0988	<b>0.0895</b>	<b>0.0029</b>	0.1217
specular-green-phenolic	<b>53.09</b>	36.67	<b>0.9889</b>	1.0161	27.1120	<b>14.0560</b>
specular-violet-phenolic	<b>50.51</b>	38.21	<b>0.9722</b>	0.9925	22.4920	<b>14.34100</b>
specular-yellow-phenolic	<b>46.81</b>	36.40	<b>0.9454</b>	0.9686	18.8780	<b>10.6280</b>
violet-acrylic	<b>50.07</b>	42.61	<b>0.7770</b>	0.7849	20.2560	<b>12.7940</b>

described above, with the corresponding dictionary as follows

$$\hat{\mathbf{Y}}^{(i)} = \hat{\mathbf{S}}^{(i,k)} \times_1 \mathbf{U}^{(1,k)} \times_2 \mathbf{U}^{(2,k)} \times_3 \mathbf{U}^{(3,k)}. \quad (7)$$

Thanks to the fact that the coefficient tensor  $\hat{\mathbf{S}}^{(i,k)}$  is sparse, Equation (7) is computationally tractable even for real-time applications. Indeed, we can evaluate (7) by only considering nonzero elements of  $\hat{\mathbf{S}}^{(i,k)}$ .

Since our dictionary is trained with two sets of transformed BRDFs, i.e.  $\rho_{t1}$  and  $\rho_{t2}$ , we can obtain two reconstructed BRDFs from an unseen BRDF by employing the algorithm described above. This still leaves us with the problem of selecting the best reconstructed BRDF between  $\rho_{t1}$  and  $\rho_{t2}$ . Due to the discrepancy between quantitative quality metrics computed over the BRDF space (such as MSE) and the rendering quality [Bieron and Peers 2020], model selection is a difficult task for BRDF fitting, as well as learning based methods such as ours. For instance, log-based metrics [L6w et al. 2012; Sun et al. 2018] have been used to improve efficiency of fitting measured BRDFs to parametric functions. Indeed the most reliable technique is to render a collection of images for all possible variations of the model and select one that is closest to an image rendered using the reference BRDF. This approach has been used by Bieron et al. [Bieron and Peers 2020] for BRDF fitting. To reduce the number of renderings, multiple BRDF parameter fitting are performed using a power function with different inputs. The model selection is then performed by rendering a test scene and choosing the best model based on image quality metrics.

We propose a model selection approach that does not require rendering the reconstructed  $\rho_{t1}$  and  $\rho_{t2}$  BRDFs. From our observations, we found that using MSE to select the final reconstructed BRDF from  $\rho_{t1}$  and  $\rho_{t2}$  does not match a selection method based rendering quality. To address this problem, we use a Gamma mapping function,  $\Gamma(\rho, \gamma) = \rho^{1/\gamma}$ , on the reference,  $\rho_{t1}$ , and  $\rho_{t2}$ , prior to computing the MSE. We call this error metric *Gamma-mapped-MSE*.

Note that since the reference BRDF is in linear BRDF domain, i.e. it is not transformed, we invert  $\rho_{t1}$  and  $\rho_{t2}$  according to (1) and (2), respectively, prior to computing the Gamma-mapped-MSE.

In Table 2 we report reconstruction quality measured with rendering SNR, Gamma-mapped-MSE, and MSE for both  $\rho_{t1}$  and  $\rho_{t2}$ . For these results we used 15 test materials from the MERL data set, while the remaining 85 materials were used for training. For each error metric, the best result is highlighted in bold-face characters. It can be seen that Gamma-mapped-MSE can well distinguish the best transformation among  $\rho_{t1}$  and  $\rho_{t2}$  with respect to rendering SNR for 13 out of 15 materials. The two exceptions are *red-fabric* and *silver-metallic-paint2*. It can also be seen that MSE only selects the correct transformation for 3 out of 15 materials. To obtain Gamma-mapped-MSE results we used  $\gamma = 2.0$ . Indeed, this parameter can be tuned per-BRDF to further improve our results; however, we found that fixed value of  $\gamma = 2.0$  is adequate to achieve a significant advantage over previous methods.

## 4 RESULTS AND DISCUSSION

This section presents an evaluation of the proposed BRDF model and comparisons to the current state-of-the-art models in terms of BRDF reconstruction error and rendering quality. The rendering results were generated using PBRT [Pharr and Humphreys 2010] with the Grace Cathedral environment map. The images were rendered at a resolution of  $512 \times 512$  pixels using 512 pixel samples in PBRT with the *directlighting* surface integrator and 256 *infinite* light-source samples.

The BRDF dictionary was trained using materials from the MERL database [Matusik et al. 2003] and RGL-EPFL isotropic BRDF database [Dupuy and Jakob 2018]. We split the MERL and RGL-EPFL materials into a training set and a test set. The training set contains 136 materials, where 85 materials are from the MERL dataset and 51

Table 3. Average, standard deviation, minimum, and maximum rendering SNR values of each BRDF model obtained from 15 materials in the MERL dataset. None of these materials were included in our training set. Yet, our method significantly outperforms state-of-the-art decomposition based methods, such as [Bagher et al. 2016], where the basis and coefficients should be computed for each given BRDF (i.e. the training and testing sets are not distinct).

BRDF Model	Average SNR (dB)	Standard Deviation	Minimum SNR (dB)	Maximum SNR (dB)
Ours, $\tau_t = 262$ , (log-plus)	50.65	3.8106	44.70	57.49
Ours, $\tau_t = 262$ , (log-plus-cosine)	47.08	8.5939	36.40	62.16
Ours, $\tau_t = 262$ , (using Gamma-mapped-MSE)	52.51	4.9752	44.55	62.16
Bagher et al.	42.76	11.6323	27.11	63.88
Bilgili et al.	32.63	5.8724	22.86	43.17
Tongbuasirilai et al.[CPD-PDV rank-1 (L=1)]	33.83	5.5236	22.22	42.71
Tongbuasirilai et al.[CPD-HD rank-1 (L=1)]	32.51	8.4995	22.97	52.27

materials are from the EPFL dataset. The test set contains 28 materials with 15 materials from the MERL dataset, 8 materials from the DTU data set [Nielsen et al. 2015] and 5 materials from RGL-EPFL [Dupuy and Jakob 2018]. The training and test sets cover a wide range of material classes. None of the materials in the test set appear in the training set.

Each BRDF color channel is processed independently for the training and model selection. We use the Rusinkiewicz parameterization [Rusinkiewicz 1998], at a resolution of  $90 \times 90 \times 180$ , i.e. we have  $m_1 = 90$ ,  $m_2 = 90$ , and  $m_3 = 180$ . For our experiments, we trained four ensembles, each with  $K = 8$  dictionaries and with training sparsities of  $\tau_l = 32$ ,  $\tau_l = 64$ ,  $\tau_l = 128$ , and  $\tau_l = 256$ . We then construct one ensemble by taking the union of the dictionaries in the four ensembles that were trained, as described in Section 3.3. The training BRDFs were transformed using *log-plus* ( $\rho_{t1}$ ) and *log-plus-cosine* ( $\rho_{t2}$ ) functions before starting the training, hence resulting in 272 materials. Once the ensemble is trained, we use the model selection algorithm, described in Section 3.4, to obtain the reconstruction of each BRDF in the test set. Note that for rendering, we invert equations (2) and (1) to convert the BRDFs to lie in the original linear domain.

To evaluate our sparse BRDF model, we use two quality metrics: Signal-to-Noise Ratio (SNR) that is calculated on the rendered images (floating-point images) and Relative Absolute Error (RAE), which computed on linear BRDF values. The RAE is defined as

$$RAE = \sqrt{\frac{\sum(\rho_{ref} - \rho_{recon})^2}{\sum(\rho_{ref}^2)}}, \quad (8)$$

where  $\rho_{ref}$  is the reference BRDF,  $\rho_{recon}$  is reconstructed BRDF. The linear BRDF values are obtained by inverting the transformations described in Section 3.1 for both the reference and reconstructed BRDFs. Even though rendering SNR (or PSNR) is used to evaluate BRDF models in many publications, RAE is very useful to capture the model accuracy of the entire BRDF space without relying on a specific rendering setup.

We compare our results to Bagher et al. [2016] (Naive model), Bilgili et al. [2011] (Tucker decomposition) and Tongbuasirilai et al. [2019] (rank-1 CPD decomposition with  $L = 1$ ) on 15 MERL test materials. The naive model stores  $(90+90+180+2) = 362$  coefficients per channel, Bilgili et al. uses  $(128+16+16+64+2) = 226$  coefficients, and the CPD decompositions from Tongbuasirilai et al. uses  $(90 +$

$90 + 180) = 360$  coefficients per channel. Since the Tucker and CPD methods use an iterative approach, we limit our comparisons to  $L = 1$ , i.e. a single factorization was performed so that the number of coefficients used for all models were roughly the same. The CPD method was tested using two different parameterizations: the PDV [Löw et al. 2012; Tongbuasirilai et al. 2019] and HD [Rusinkiewicz 1998] parameterizations.

To the best of our knowledge, the model of Bagher et al. is the current state-of-the-art. Since our representation is sparse, we only store nonzero locations,  $1 + 1 + 2$  bytes, and values, 8 bytes. Simple calculations show that by using  $\tau_t = 262$  coefficients for our model, we can match the storage complexity of [Bagher et al. 2016], which uses 362 coefficients to model each color channel of a BRDF.

For the rendered images, shown in tables 3 and 4, and Fig. 5, we apply gamma-corrected tone-mapping. The difference images, also known as false-color, produced by normalizing the error image of each BRDF and for all models in the range  $[0,1]$  and applying a *jet* color map using MATLAB. All the difference images display normalized linear errors multiplied by 10 for visualization.

For a more comprehensive overview of the results, we refer the reader to the supplementary material which includes a large number of additional rendering SNR values, error metrics and rendering results.

#### 4.1 Quantitative evaluations

Table 3 reports Signal-to-Noise Ratio (SNR) statistics for 15 test materials in the MERL database. The average SNR of our model is about 8dB, 5dB and 10dB higher for *log-plus*, *log-plus-cosine* and the proposed model selection based on Gamma-mapped-MSE, respectively, when compared to Bagher et al.; moreover, our results show a smaller standard deviation on SNR. Our model with our proposed selection method can achieve higher SNR on average compared to both our model of *log-plus* and *log-plus-cosine*. It can also be seen that the Tucker and CPD methods perform poorly without the power of iterative terms [Bilgili et al. 2011; Lawrence et al. 2004; Tongbuasirilai et al. 2019]. Even though our model has lower maximum SNR than Bagher et al., our minimum SNR is around 9dB higher for *log-plus-cosine*, 17dB higher for *log-plus* and 18dB higher for our selection. The lower standard deviation indicates that the proposed model can represent the MERL materials more faithfully. Table 4 shows a direct comparison of our model to that of Bagher et al. for each BRDF in the MERL test set using rendering SNR and

Table 4. Rendering SNR and BRDF-space RAE values obtained with our BRDF model and that of Bagher et al., on 15 test materials of the MERL dataset. These materials were not used in our training set. Higher rendering SNR is highlighted in bold.

Material	SNR (dB)		RAE	
	Our	Bagher	Our	Bagher
blue-fabric	62.16	<b>63.88</b>	0.8695	0.3596
blue-metallic-paint	<b>60.53</b>	44.86	0.4287	0.2736
dark-red-paint	54.80	<b>57.89</b>	0.2717	0.4955
gold-metallic-paint2	<b>48.29</b>	29.77	0.0811	0.7007
green-metallic-paint2	<b>57.48</b>	51.04	0.0726	0.4209
light-red-paint	<b>51.68</b>	51.36	0.2583	0.5639
pink-fabric2	<b>52.71</b>	52.44	0.8581	0.3388
purple-paint	<b>47.22</b>	44.29	0.1144	0.4428
red-fabric	<b>55.05</b>	51.44	0.7160	0.4163
red-metallic-paint	<b>52.70</b>	34.06	0.0752	0.7181
silver-metallic-paint2	<b>44.55</b>	27.11	0.6256	0.3626
specular-green-phenolic	<b>53.09</b>	35.69	0.0673	0.6414
specular-violet-phenolic	<b>50.51</b>	37.76	0.0592	0.6592
specular-yellow-phenolic	<b>46.81</b>	28.14	0.0683	0.7370
violet-acrylic	<b>50.07</b>	31.68	0.06536	0.5156

Table 5. Rendering SNR and BRDF-space RAE values obtained with our BRDF model, on 8 unseen materials from the DTU data set [Nielsen et al. 2015]. The bottom row shows means of each column. The last column present SNR results of our model selection method based on Gamma-mapped-MSE described in Section 3.4.

Material	Our $\rho_{t1}$		Our $\rho_{t2}$		Sel. SNR (dB)
	SNR (dB)	RAE	SNR (dB)	RAE	
binder-cover	45.70	0.0611	<b>46.12</b>	0.0303	46.12
blue-book	<b>47.24</b>	0.0574	45.07	0.0258	45.07
cardboard	44.72	0.1468	<b>48.90</b>	0.3779	48.90
glossy-red-paper	<b>45.16</b>	0.0436	41.41	0.0288	45.16
green-cloth	<b>51.60</b>	0.1145	51.58	0.7713	51.58
notebook	43.39	0.1838	<b>47.15</b>	0.2805	47.15
painted-metal	46.95	0.0817	<b>51.84</b>	0.1240	51.84
yellow-paper	47.14	0.1289	<b>49.04</b>	0.4873	49.04
<b>Average</b>	46.49	0.1022	47.64	0.2658	48.11

Table 6. Rendering SNR and BRDF-space RAE values obtained with our BRDF model, on 5 test materials of the RGL-EPFL dataset. The bottom row shows means of each column. The last column present SNR results of our model selection method based on Gamma-mapped-MSE described in Section 3.4.

Material	Our $\rho_{t1}$		Our $\rho_{t2}$		Sel. SNR (dB)
	SNR (dB)	RAE	SNR (dB)	RAE	
acrylic-felt-green-rgb	43.23	0.9922	<b>45.91</b>	0.3104	45.91
cc-amber-citrine-rgb	<b>26.33</b>	0.5394	26.24	0.8420	26.24
ilm-l3-37-dark-green-rgb	38.92	0.9558	<b>43.45</b>	0.6060	43.45
paper-blue-rgb	38.92	0.9871	<b>40.23</b>	0.4416	40.23
vch-dragon-eye-red-rgb	<b>40.48</b>	0.8973	38.16	0.7989	38.16
<b>Average</b>	37.57	0.8744	38.80	0.5998	38.80

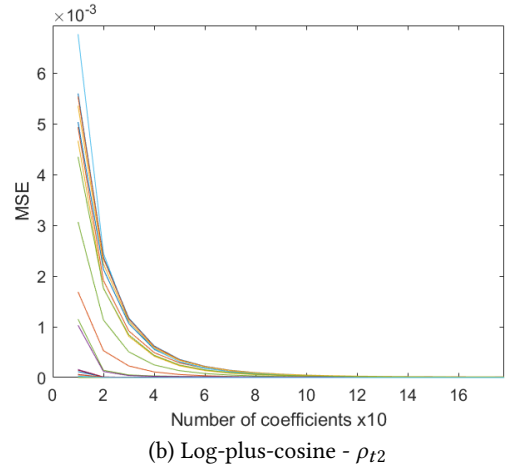
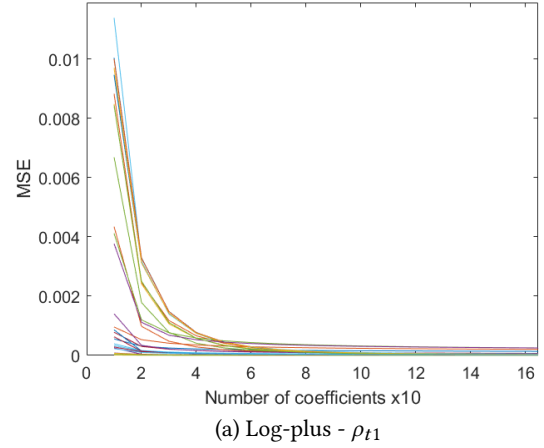


Fig. 2. BRDF error plots of all test materials from MERL, EPFL, and DTU data sets when reconstructed with increasing number of coefficients: (a) Log-plus transformation ( $\rho_{t1}$ ) and (b) Log-plus-cosine transformation ( $\rho_{t2}$ ).

Table 7. Rendering SNR obtained from reconstructions of our BRDF model and PCA with 40 coefficients on RGL-EPFL test set.

Material	Our $\rho_{t1}$	Our $\rho_{t2}$	PCA
acrylic-felt-green-rgb	39.37	40.65	25.14
cc-amber-citrine-rgb	14.12	19.90	6.46
ilm-l3-37-dark-green-rgb	36.88	40.82	19.12
paper-blue-rgb	27.16	34.19	14.80
vch-dragon-eye-red-rgb	33.20	33.86	13.34

BRDF-space RAE. Here we use our Gamma-mapped-MSE metric to choose between the transformations. Compared to the model of Bagher et al., our approach achieves significantly higher visual quality on 13 out of 15 materials, see Fig. 3.

To demonstrate the robustness of our sparse non-parametric model for representing unseen BRDFs, we also evaluate it using 8 test samples provided by Nielsen et al. [Nielsen et al. 2015]. Note that we use the same dictionary described above and that none of the



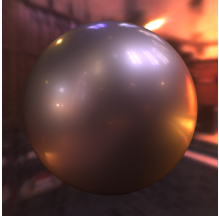

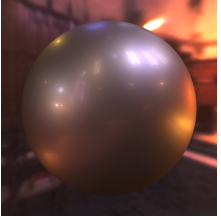
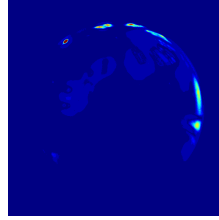
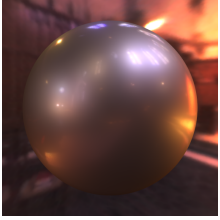

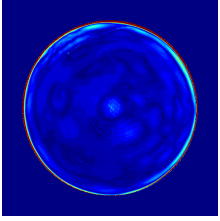

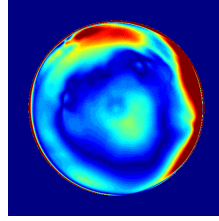

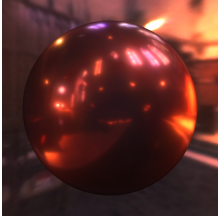
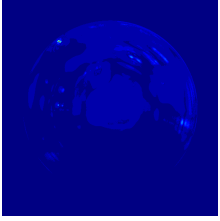
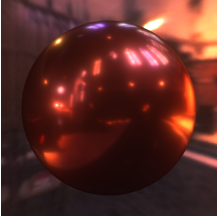
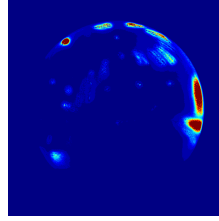
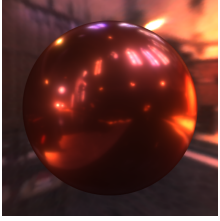
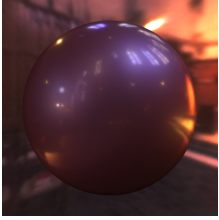
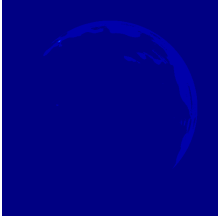
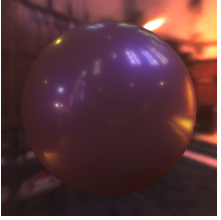
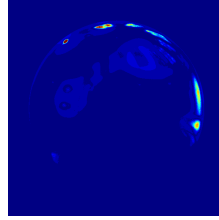
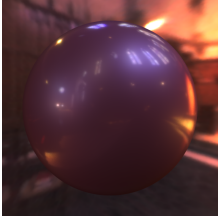
Material	Our Model	Our Model difference	Bagher et al.	Bagher et al. difference	reference
gold-metallic-paint2	 48.29dB		 29.77dB		
red-fabric	 55.05dB		 51.44dB		
red-metallic-paint	 52.70dB		 34.06dB		
violet-acrylic	 50.07dB		 31.68dB		

Fig. 3. Reconstructions of unseen materials from MERL. The reconstructed BRDFs are modeled using our BRDF models with  $\tau = 262$  coefficients compared to the model of Bagher et al. [Bagher et al. 2016]. The error images are placed on the right column of each reconstruction. All rendered images have been gamma corrected for visual representation. The error images have been multiplied by 10.0 for visual comparisons. All images have been rendered using the Grace Cathedral environment [Debevec 1998] using PBRT [Pharr and Humphreys 2010].

materials from the DTU data set were used in the training set. The results are summarized in Table 5, where we report rendering SNR and BRDF-space RAE for  $\rho_{t1}$ ,  $\rho_{t2}$ , and our model selection based on Gamma-mapped-MSE. Our BRDF model and selection method can reproduce the DTU data set with average SNR of more than 48dB. Our model selection algorithm on the DTU test set missed on 2 out of 8 materials, which are *blue-book* and *green-cloth*. Visual quality examples of the rendered images are presented in Fig. 4. The difference between  $\rho_{t1}$  and  $\rho_{t2}$  is evident in this figure. We can see that  $\rho_{t1}$  is favored by glossy materials, while  $\rho_{t2}$  is more effective in modeling low-frequency or diffuse-like materials.

Table 6 shows rendering SNR and BRDF-space RAE values for the RGL-EPFL test set using both  $\rho_{t1}$  and  $\rho_{t2}$ . The rendering SNR values of the RGL-EPFL test sets are above 35dB except *cc-amber-citrine-rgb* which is mainly due to the rendering noise. Our BRDF model and selection method can efficiently represent the RGL-EPFL data set with an average SNR of more than 38dB. Our model selection method on the RGL-EPFL test set missed on 2 out of 5 materials, which are *cc-amber-citrine-rgb* and *vch-dragon-eye-red-rgb*. The SNR values demonstrate that our data-driven model can accurately represent and faithfully reconstruct the unseen samples. See the supplementary materials for rendered images obtained using our model applied on the RGL-EPFL data set.

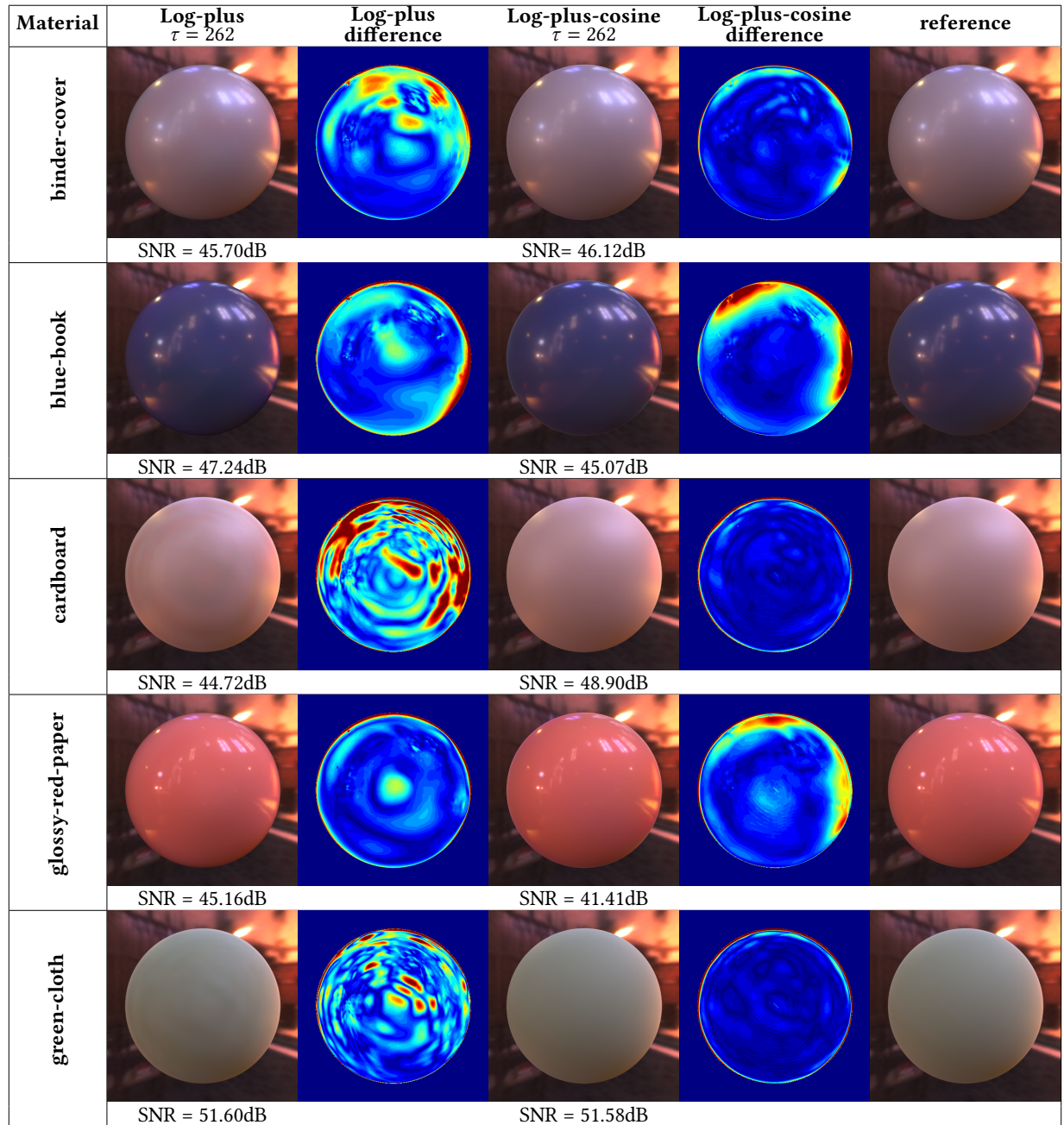


Fig. 4. Reconstructions of unseen materials from Nielsen et al. [Nielsen et al. 2015]. The reconstructed BRDFs are modeled using our BRDF models with  $\tau = 262$  coefficients with log-plus transformation and log-plus-cosine transformation. The error images are placed on the right column of each reconstruction. All rendered images have been gamma corrected for visual representation. The error images have been multiplied by 10.0 for visual comparisons. All images have been rendered using the Grace Cathedral environment [Debevec 1998] using PBRT [Pharr and Humphreys 2010].

Our results also confirm the discrepancy between BRDF-space error metrics (such as RAE) and rendering quality using SNR. For example, *blue-metallic-paint* in Table 4, *cardboard* in Table 5 and *vch-dragon-eye-red-rgb* in Table 6 demonstrate how RAE contradicts

the rendering SNR. The lower BRDF-space RAE is, the more accurate the model represents a BRDF. However, a rendered image is dependent on a variety of additional factors such as geometry of objects, lighting environment, and viewing position. As a result, the

BRDF-space RAE and rendering SNR have to be considered together for the evaluation of a BRDF model.

We also evaluated our BRDF model with fewer coefficients and compared to the PCA-based method presented in [Nielsen et al. 2015], see Table 7. Indeed our model significantly outperforms PCA. It should be noted that the PCA dictionary exhibit a very high storage cost compared to our BRDF dictionary ensemble. The size of the PCA dictionary is  $1458000 \times 300 = 437,400,000$  elements, while our dictionary consists of  $(90 \times 90 + 90 \times 90 + 180 \times 180) \times 32 = 1,555,200$  elements, i.e. the PCA dictionary is more than 280 times larger.

Figure 2 demonstrates the behavior of our BRDF model for both  $\rho_{t1}$  and  $\rho_{t2}$  for a large range of coefficients,  $\tau$ , during the model selection phase. We observe that  $\rho_{t1}$  exhibits a much higher MSE when compared to  $\rho_{t2}$ . This is expected since  $\rho_{t2}$  is typically chosen by the model selection algorithm to represent diffuse or low-frequency BRDFs. In terms of the decline of error with respect to the number of coefficients, both transformations show a similar behavior.

## 4.2 Visual results

In Figure 3, we present example renderings of four BRDFs in the MERL test set modeled using our method and [Bagher et al. 2016]. Our results are obtained using the proposed Gamma-mapped-MSE for model selection. Figure 4 shows renderings of five test BRDFs from the DTU data set, where we compare the results of both BRDF transformations,  $\rho_{t1}$  and  $\rho_{t2}$ , with the reference rendering. For both figures we used the *Grace Cathedral* HDRi environment map. In Fig. 4, we observe that the artifacts seen on the *cardboard* and *green-cloth* renderings for *log-plus* ( $\rho_{t1}$ ) do not appear in *log-plus-cosine* ( $\rho_{t2}$ ) renderings. The *log-plus-cosine* transformation suppresses the grazing angle BRDF values. For diffuse materials, this leads to better visual results and significantly higher rendering SNR. It is evident from Fig. 3 that the *log-plus* transformation is better for glossy materials as the *log-plus-cosine* transformation leads to color artifacts for some materials, e.g. *gold-metallic-paint2*, *red-metallic-paint*, and *violet-acrylic*. More results for further analysis is available in the supplementary material.

In Figure 5, we evaluate our BRDF model using the *Princeton* scene with the following materials: *blue-metallic-paint*, *gold-metallic-paint2*, *pink-fabric2*, *silver-metallic-paint2*, and *specular-yellow-phenolic*. We rendered the scene with path tracing in PBRT [Pharr and Humphreys 2010] using the *uffizi* environment map and with  $2^{17}$  samples-per-pixel. Figures 5(a) and 5(c) present rendered images from our model and Bagher et al., respectively. Our model achieves an 8.03dB advantage in SNR over the model of Bagher et al.

## 5 CONCLUSION AND FUTURE WORK

This paper presented a novel non-parametric sparse BRDF model in which a measured BRDF is represented using a trained multidimensional dictionary ensemble and a set of sparse coefficients. We showed that with careful model selection over the space of multidimensional dictionaries and various BRDF transformations, we achieve significantly higher rendering quality and model accuracy compared to current state-of-the-art. We evaluated the performance of our model and algorithm using three different data sets, MERL, RGL-EPFL, and one provided by Nielsen et al. [2015]. For the vast

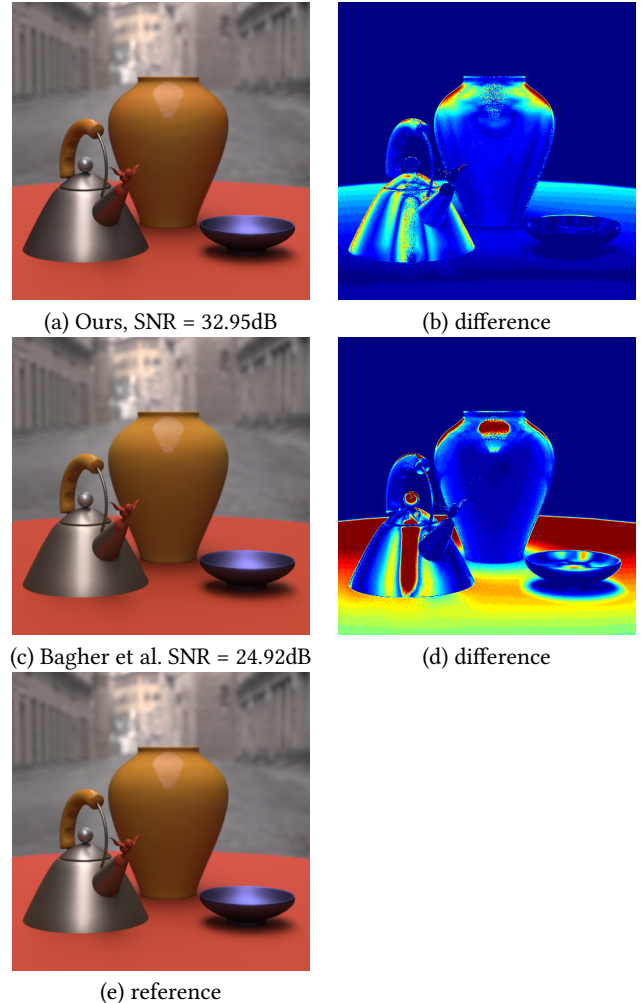


Fig. 5. Renderings of the *Princeton* scene using (a) our BRDF model and (c) the model of Bagher et al. All images were rendered at 131072 samples/pixel using the path tracing algorithm of PBRT.

majority of the BRDFs used in the test set we achieve a significant advantage over previous models.

In the future, we aim to extend our sparse BRDF model to efficiently represent anisotropic materials. Moreover, we acknowledge the fact that the discrepancy between BRDF-space error metrics and the rendering quality is still an open problem. Although we showed significant improvements using our Gamma-mapped-MSE, we believe that a more sophisticated metric that takes into account the support of the BRDF function can improve our results. Our model is relatively robust to noise. However, we believe that an application of a denoising pass that is tailored to measured BRDFs, prior to training and model selection, can greatly improve our results. This is expected since it is well-known that even a moderate amount of noise in measured BRDFs translates to lower rendering quality; and



that noise reduces the sparsity of the representation, hence increasing the model complexity. An alternative to applying a denoiser is to modify the training and model selection methods to be noise-aware.

## 6 ACKNOWLEDGEMENTS

\*- Left blank for the double blind review \*-

## REFERENCES

- M. Aharon, M. Elad, and A. Bruckstein. 2006. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Transactions on Signal Processing* 54, 11 (Nov 2006), 4311–4322. <https://doi.org/10.1109/TSP.2006.881199>
- Michael Ashikhmin and Peter Shirley. 2000. An Anisotropic Phong BRDF Model. *J. Graph. Tools* 5, 2 (Feb. 2000), 25–32. <https://doi.org/10.1080/10867651.2000.10487522>
- Mahdi M. Bagher, John Snyder, and Derek Nowrouzezahrai. 2016. A Non-Parametric Factor Microfacet Model for Isotropic BRDFs. *ACM Trans. Graph.* 35, 5, Article 159 (July 2016), 16 pages. <https://doi.org/10.1145/2907941>
- P. Barla, L. Belcour, and R. Pacanowski. 2015. In Praise of an Alternative BRDF Parameterization. In *Workshop on Material Appearance Modeling*, Reinhard Klein and Holly Rushmeier (Eds.). The Eurographics Association. <https://doi.org/10.2312/mam.20151197>
- Aner Ben-Artzi, Ryan Overbeck, and Ravi Ramamoorthi. 2006. Real-time BRDF Editing in Complex Lighting. *ACM Trans. Graph.* 25, 3 (July 2006), 945–954. <https://doi.org/10.1145/1141911.1141979>
- James C. Bieron and Pieter Peers. 2020. An Adaptive Brdf Fitting Metric. *Computer Graphics Forum* 39, 4 (July 2020). <https://doi.org/10.1111/cgf.14054>
- Ahmet Bilgili, Aydn Öztürk, and Murat Kurt. 2011. A General BRDF Representation Based on Tensor Decomposition. *Computer Graphics Forum* 30, 8 (2011), 2427–2439.
- James F. Blinn. 1977. Models of Light Reflection for Computer Synthesized Pictures. *SIGGRAPH Comput. Graph.* 11, 2 (July 1977), 192–198. <https://doi.org/10.1145/965141.563893>
- L. Claustres, M. Paulin, and Y. Boucher. 2003. BRDF Measurement Modelling using Wavelets for Efficient Path Tracing. *Computer Graphics Forum* 22, 4 (2003). <https://doi.org/10.1111/j.1467-8659.00718.x>
- R. L. Cook and K. E. Torrance. 1982. A Reflectance Model for Computer Graphics. *ACM Trans. Graph.* 1, 1 (Jan. 1982), 7–24.
- Paul Debevec. 1998. Rendering Synthetic Objects into Real Scenes: Bridging Traditional and Image-based Graphics with Global Illumination and High Dynamic Range Photography. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. ACM, New York, NY, USA, 189–198. <https://doi.org/10.1145/280814.280864>
- 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. <https://doi.org/10.1145/3197517.3201378>
- Valentin Deschaintre, Miika Aittala, Frédo Durand, George Drettakis, and Adrien Bousseau. 2019. Flexible SVBRDF Capture with a Multi-Image Deep Network. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 38, 4 (July 2019). <http://www-sop.inria.fr/revues/Basilic/2019/DADB19>
- X. Ding, W. Chen, and I. J. Wassell. 2017. Joint Sensing Matrix and Sparsifying Dictionary Optimization for Tensor Compressive Sensing. *IEEE Transactions on Signal Processing* 65, 14 (July 2017), 3632–3646. <https://doi.org/10.1109/TSP.2017.2699639>
- Yue Dong. 2019. Deep appearance modeling: A survey. *Visual Informatics* 3, 2, Article 59 (2019), 9 pages. <https://doi.org/10.1016/j.visinf.2019.07.003>
- Zhao Dong, Bruce Walter, Steve Marschner, and Donald P. Greenberg. 2016. Predicting Appearance from Measured Microgeometry of Metal Surfaces. *ACM Trans. Graph.* 35, 1, Article 9 (Dec. 2016), 13 pages. <https://doi.org/10.1145/2815618>
- Jonathan Dupuy and Wenzel Jakob. 2018. An Adaptive Parameterization for Efficient Material Acquisition and Rendering. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 37, 6 (Nov. 2018), 274:1–274:18. <https://doi.org/10.1145/3272127.3275059>
- S. Hawe, M. Seibert, and M. Kleinstüber. 2013. Separable Dictionary Learning. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*. 438–445. <https://doi.org/10.1109/CVPR.2013.63>
- Nicolas Holzschuch and Romain Pacanowski. 2017. A Two-scale Microfacet Reflectance Model Combining Reflection and Diffraction. *ACM Trans. Graph.* 36, 4, Article 66 (July 2017), 12 pages. <https://doi.org/10.1145/3072959.3073621>
- Bingyang Hu, Jie Guo, Yanjun Chen, Mengtian Li, and Yanwen Guo. 2020. DeepBRDF: A Deep Representation for Manipulating Measured BRDF. *Computer Graphics Forum* (2020). <https://doi.org/10.1111/cgf.13920>
- Wenzel Jakob, Eugene d'Eon, Otto Jakob, and Steve Marschner. 2014. A Comprehensive Framework for Rendering Layered Materials. *ACM Trans. Graph.* 33, 4, Article 118 (July 2014), 14 pages. <https://doi.org/10.1145/2601097.2601139>
- Jan Kautz and Michael D. McCool. 1999. Interactive Rendering with Arbitrary BRDFs Using Separable Approximations. In *Proceedings of the 10th Eurographics Conference*

- on Rendering (Granada, Spain) (EGWR'99). Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 247–260. <https://doi.org/10.2312/EGWR/EGWR99/247-260>
- Jason Lawrence, Szymon Rusinkiewicz, and Ravi Ramamoorthi. 2004. Efficient BRDF Importance Sampling Using a Factored Representation. *ACM Trans. Graph.* 23, 3 (Aug. 2004), 496–505. <https://doi.org/10.1145/1015706.1015751>
- Zhengqin Li, Kalyan Sunkavalli, and Manmohan Chandraker. 2018. Materials for Masses: SVBRDF Acquisition with a Single Mobile Phone Image. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Joakim Löw, Joel Kronander, Anders Ynnerman, and Jonas Unger. 2012. BRDF Models for Accurate and Efficient Rendering of Glossy Surfaces. *ACM Trans. Graph.* 31, 1, Article 9 (Feb. 2012), 14 pages. <https://doi.org/10.1145/2077341.2077350>
- M. Marsousi, K. Abhari, P. Babyn, and J. Alirezaie. 2014. An Adaptive Approach to Learn Overcomplete Dictionaries With Efficient Numbers of Elements. *IEEE Transactions on Signal Processing* 62, 12 (June 2014), 3272–3283. <https://doi.org/10.1109/TSP.2014.2324994>
- Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. 2003. A Data-driven Reflectance Model. *ACM Trans. Graph.* 22, 3 (July 2003), 759–769. <https://doi.org/10.1145/882262.882343>
- R. Mazhar and P. D. Gader. 2008. EK-SVD: Optimized Dictionary Design for Sparse Representations. In *2008 19th International Conference on Pattern Recognition*. 1–4. <https://doi.org/10.1109/ICPR.2008.4761362>
- Ehsan Mianji, Saghi Hajisharif, and Jonas Unger. 2019. A Unified Framework for Compression and Compressed Sensing of Light Fields and Light Field Videos. *ACM Trans. Graph.* 38, 3, Article 23 (May 2019), 18 pages. <https://doi.org/10.1145/3269980>
- Subhadip Mukherjee, Rupam Basu, and Chandra Sekhar Seelamantula. 2016.  $\ell_1$ -K-SVD: A robust dictionary learning algorithm with simultaneous update. *Signal Processing* 123 (June 2016), 42–52. <https://doi.org/10.1016/j.sigpro.2015.12.008>
- F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis. 1992. Radiometry. Jones and Bartlett Publishers, Inc., USA, Chapter Geometrical Considerations and Nomenclature for Reflectance, 94–145. <http://dl.acm.org/citation.cfm?id=136913.136929>
- Jannik Boll Nielsen, Henrik Wann Jensen, and Ravi Ramamoorthi. 2015. On Optimal, Minimal BRDF Sampling for Reflectance Acquisition. *ACM Transactions on Graphics (TOG)* 34, 6 (November 2015), 186:1–186:11. <https://doi.org/10.1145/2816795.2818085>
- Matt Pharr and Greg Humphreys. 2010. *Physically Based Rendering, Second Edition: From Theory To Implementation* (2nd ed.). Morgan Kaufmann Publishers Inc, San Francisco, CA, USA.
- Ravi Ramamoorthi and Pat Hanrahan. 2001. An Efficient Representation for Irradiance Environment Maps. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*. Association for Computing Machinery, New York, NY, USA, 497–500. <https://doi.org/10.1145/383259.383317>
- F. Roemer, G. Del Galdo, and M. Haardt. 2014. Tensor-based Algorithms for Learning Multidimensional Separable Dictionaries. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 3963–3967. <https://doi.org/10.1109/ICASSP.2014.6854345>
- Fabiano Romeiro, Yuriy Vasilyev, and Todd Zickler. 2008. Passive Reflectometry. In *Proceedings of the 10th European Conference on Computer Vision: Part IV (Marseille, France) (ECCV '08)*. Springer-Verlag, Berlin, Heidelberg, 859–872.
- Szymon Rusinkiewicz. 1998. A New Change of Variables for Efficient BRDF Representation. In *Rendering Techniques (Eurographics)*, George Drettakis and Nelson L. Max (Eds.). Springer, 11–22.
- C. Rusu and B. Dumitrescu. 2012. Stagewise K-SVD to Design Efficient Dictionaries for Sparse Representations. *IEEE Signal Processing Letters* 19, 10 (Oct 2012), 631–634. <https://doi.org/10.1109/LSP.2012.2209871>
- M. M. Stark, J. Arvo, and B. Smits. 2005. Barycentric parameterizations for isotropic BRDFs. *IEEE Transactions on Visualization and Computer Graphics* 11, 2 (March 2005), 126–138. <https://doi.org/10.1109/TVCG.2005.26>
- Tiancheng Sun, Henrik Wann Jensen, and Ravi Ramamoorthi. 2018. Connecting Measured BRDFs to Analytic BRDFs by Data-driven Diffuse-specular Separation. *ACM Trans. Graph.* 37, 6, Article 273 (Dec. 2018), 15 pages. <https://doi.org/10.1145/3272127.3275026>
- Tanaboon Tongbuasirilai, Jonas Unger, Joel Kronander, and Murat Kurt. 2019. Compact and intuitive data-driven BRDF models. *The Visual Computer* 36 (May 2019), 855–872.
- 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. <https://doi.org/10.1145/2461912.2461944>
- Bruce Walter, Stephen R. Marschner, Hongsong Li, and Kenneth E. Torrance. 2007. Microfacet Models for Refraction Through Rough Surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques (Grenoble, France) (EGSR'07)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 195–206. <https://doi.org/10.2312/EGWR/EGSR07/195-206>
- Gregory J. Ward. 1992. Measuring and Modeling Anisotropic Reflection. *SIGGRAPH Comput. Graph.* 26, 2 (July 1992), 265–272.