

# Representing BRDFs Using SOMs and MANs

by westman — last modified 09 August 2008 01:37 PM

**Authors: Murat Kurt\*, Muhammed Gökhan Cinsdikici**  
**Ege University, International Computer Institute, Bornova, 35100 Izmir, Turkey**

*\* Corresponding author. Tel.: +90 232 342 32 32; fax: + 90 232 388 72 30.*

*E-mail addresses: murat.kurt@ege.edu.tr (M. Kurt), muhammed.cinsdikici@ege.edu.tr (M. Cinsdikici).*

## Abstract

High-quality Monte Carlo image synthesis needs to use realistic bidirectional reflectance distribution functions (BRDFs) based on acquired reflectance data. In this study, we propose a novel model that contributes to solving memory problems and measurement noise problems of BRDFs. The model is a composite associative memory model constructed by integration of Kohonen's Self Organizing Feature Map (SOM), Mass Attraction Network (MAN), and it has been tested on data set acquired by Matusik et al.. Because it is based on table-lookups, it can't be categorized in analytical reflectance models. We show that our approach has acceptable compactness. We also show that it has more accuracy than both analytical BRDF models and Lawrence et al.'s table-based model. In this work, we are able to create images efficiently under global illumination with less memory requirements.

**Keywords:** SOM; MAN; BRDF representation; Reflection model; Table-based model.

## 1. Introduction

To be able to produce photo-realistic images in computer graphics, the interactions between light and surface must be described well. In this paper, we focus on Bidirectional Reflectance Distribution Functions (BRDFs) [1], which characterizes this process.

When we look at the BRDF history, we are able to see that a lot of analytical BRDF models have been introduced over the past 35 years. These models can be divided into two categories; empirical (phenomenological) analytical BRDF models [2, 3, 4, 5, 6, 7, 8, 9] and physically-based analytical BRDF models [10, 11, 12, 13, 14].

Both the empirical and the physically-based analytical BRDF models are only approximations of reflectance of real materials. Many of these models are based on material parameters that in principle could be measured, but in practice are difficult to acquire [15].

Alternative to directly measuring model parameters is to acquire actual samples from BRDFs using some version of a gonireflectometer [16, 17, 18, 4, 19, 20, 15] and then fit the measured data to a selected analytical model using various optimization techniques [4, 7, 21] [15]. So, the physical derivation of a physically-based BRDF model serves primarily to inspire greater confidence, and is not necessarily a practical advantage when it comes to fitting measured data [4].

There are several problems in model fitting when a BRDF is represented by an analytical function [15]. First, BRDF represented by the analytical function with the computed parameters is only an approximation of real reflectance; measured values of the BRDF are usually not exactly equal to the values of the analytical model [15]. Another major problem is that depending on the number of lobes used in modeling the BRDF, the corresponding number of parameters to be estimated usually is large. For example, when Lafortune et al. [7] model is chosen to fit for an anisotropic material, there would be at least 4 and 8 non-linear parameters should be estimated for two-lobe (1 diffuse lobe + 1 specular lobe) and three-lobe (1 diffuse lobe + 2 specular lobes) representations respectively. As Lawrence et al. [22] noted, fitting a three-lobe Lafortune et al. model can be unstable, often taking minutes to hours to converge in a non-linear minimizer, and can require manual tuning to find a good fit. Nygan et al. [21] found that in practice, fitting with four-lobe (1 diffuse lobe + 3 specular lobes) is very unstable, and hence they omitted results from the four-lobe Lafortune et al. model.

In non-linear estimation, optimization results also closely depend on the choice of initial values and a global minimum usually is not guaranteed. However, computational cost become high when large data set is used. Another problem involved in the fitting process is that the choice of the objective function for optimization is not clear. Some researchers also applied logarithmic or cubic root transformations on BRDFs before starting fitting [21, 23]. The pure least squares method assumes that the variances at sample points are the same. It is a common experience that for the BRDF data, the variances are not homogeneous and some weighting functions have been proposed to stabilize the corresponding variances [7, 21].

Unfortunately, real BRDF data have long been hard to acquire and measurements are often limited in angular resolution. Realism of both empirical and physically-based analytical models increases, if angular resolution is high. So, for efficient measure-and-fit approach, densely acquired data is very important but finding such data is very hard.

Another method for BRDF modeling is to create a model and use dense set of measurements directly in the rendering process. For example, Matusik et al. [15]'s data-driven approach and Lawrence et al. [22]'s importance sampling method have been developed for this purpose. These approaches preserve those subtleties of the measured data that are lost in a data-fitting approach and these approaches have the advantage that the produced BRDFs look very realistic since they are based on the measured BRDFs [15]. But Matusik et al. [15]'s data-driven model has a disadvantage that each BRDF is stored separately as a tabulated data structure which requires about 17 MB of memory. Using hundreds or thousands of materials in the same scene would be impossible because of memory requirements [23].

In this paper, we introduce a novel model based on Matusik et al.'s measured BRDFs. When we analyze this data set, we see that some parts of measured BRDFs are never used in the rendering process, and also different index values return the same BRDFs or almost the same BRDFs. So, we aimed to eliminate these BRDFs without losing the realism (see Fig. 1). To be able to do that, we use our composite model of neural networks: Kohonen's Self Organizing Feature Map (SOM) and Mass Attraction Network (MAN).



**Fig. 1 - Monte Carlo renderings of a buddha with “metallic-blue” BRDF under global illumination. These images were generated according to three table-based BRDF representations. From left to right: Reference image rendered using Matusik et al.’s measured data, using Lawrence et al.’s factored BRDF (PSNR = 44.53), and using our representation with 256 clusters and 50 epochs (PSNR = 52.52). Insets show a difference between Matusik et al.’s representation and the rendered images. As you can see, our proposed representation is more accurate than Lawrence et al.’s representation.**

Because of its neural network nature, our model behaves as a brain such that it remembers (stores) most useful BRDFs meanwhile forgetting the BRDFs which are not used. So, it reduces the memory requirements. Proposed approach also regularizes some minor measurement noise. Our model is as realistic as original measured BRDFs, and it requires smaller memory area than original measured BRDFs. Since the proposed method inherits from Matusik et al.’s measured data set, it carries all of the properties Matusik et al.’s measured BRDFs have. So, our model is for isotropic materials.

The next section gives a review of previous work. In Section 3, and 4 the table-based approach based on integration of Kohonen’s Self Organizing Feature Map (SOM), Mass Attraction Network (MAN) are explained respectively. Some empirical results are presented in Section 5. Section 6 is devoted for conclusions and future work.

## 2. Previous Work

The most well known and one of the oldest reflectance model developed to simulate the effects of the specular reflection is the Phong model [2]. For specular surfaces, this model assumes that the incoming light tends to bounce off in the reflection direction. For the fast computation purposes, [3] developed a model based on the Phong model, but this model uses halfway direction instead of reflection direction. So, Phong model is based on (V.R) lobe, but Blinn variation of Phong model and the most microfacet based models are based on (N.H) lobe. Nygan et al. [21] have shown that the (N.H) representation is a more accurate way to model specular reflections. However, both models are empirical, mathematically simple and this mathematical simplicity has been made them become very popular.

Fresnel effect and micro geometry of a surface are important factors in BRDF modeling. Considering the fact that a surface can be modeled by using small and flat surfaces (microfacets) with random angle and size, Torrance and Sparrow [10], Cook and Torrance [11] and Ashikhmin et al. [8] have presented more sophisticated models. Torrance-Sparrow and Cook-Torrance models are physically-based models and they are based on V-cavity, though Ashikhmin et al. is an empirical model. These models account for masking

and self-shadowing effects and predict off-specular reflection. A comprehensive but computationally expensive model based on physical theory was developed by He et al. [13]. Oren and Nayar [14] presented a non-Lambertian diffuse model to simulate rough and diffuse surfaces such as sand and plaster. This model is based on Lambertian diffuse microfacets and exhibits backscattering phenomena. For anisotropic surfaces, Poulin and Fournier [12] introduced a reflection and refraction model. Their model assumes that a surface consists of microscopic cylinders. All of three previous models are physically-based models.

On the other hand, empirical models which do not necessarily reflect the physical mechanism of the light material interaction have been developed to capture the reflectance effects. Ward [4] developed a model that, while not strictly physically based, was capable of describing most significant reflection phenomena [15]. His model obeys the most basic of physical laws (reciprocity and energy conservation) and it is relatively simple over the most of the analytical reflectance models. This model is also for both isotropic and anisotropic reflections. Duer variation of the Ward model [9] and the Ward model are almost same, but Duer variation of the Ward model [9] has a different normalization factor from Ward model. Schröder and Sweldens [5] represented reflectance functions using spherical wavelets. Koenderink et al. [6] used Zernike polynomials for representing BRDFs. Lafortune et al. [7] introduced an empirical model which can capture important phenomena such as off-specular reflection, increasing reflectance and retro-reflection, and this model is also based on (V.R) lobe.

The potential benefit of using measurements of BRDFs has also gained recent attention [4, 24, 19, 25, 20]. The measurements of Matusik et al. [15] provide a dense (90 x 90 x 180 for values) sampling of many isotropic BRDFs. The main drawback of these representations is their size, since they typically represent the full 3D isotropic BRDFs in tabular form. In Matusik [23]'s thesis, he also describes one approach for sampling these measured BRDFs, but this representation requires as much storage as the original BRDFs, making it difficult to use for scenes containing many materials [22].

In an effort to reduce the size of measured BRDFs while maintaining an accurate representation of their effects, several researchers have investigated techniques for factoring these large datasets into more compact, manageable form [26, 27, 28]. In all cases the 4D BRDF is factored into products of 2-dimensional functions that can be represented as texture maps and used to shade a model in real-time. However, in most cases these factorizations allow only a single term approximation. More important, there are no techniques for importance sampling these representations [22]. On the other hand, Lawrence et al. [22] have developed a model based on factorization of BRDF data which is convenient for importance sampling, but this representation is not as accurate as original measured BRDF data (see Fig. 1).

When we look at the BRDF history from neural networks side, we see that Gargan and Neelamkavil [29] have presented a model which uses neural networks for approximating reflectance functions. They used standard backpropagation networks with two or three weight layers, and it is a non-linear model. Their neural network structure is used during rendering. However, for reducing the size of measured BRDFs with using a neural network structure only in data preparation phase, neural networks have never been used.

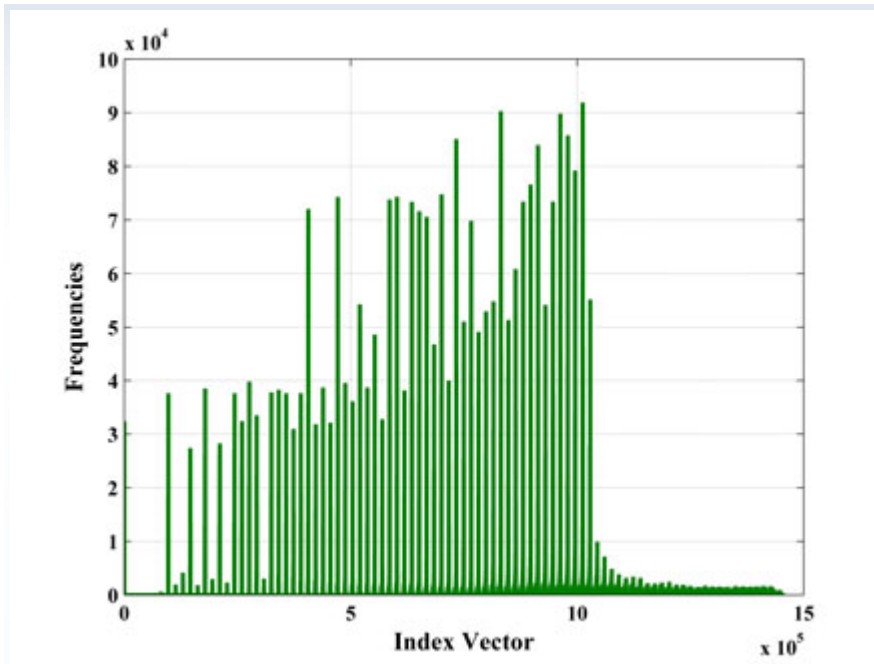
### 3. Proposed Architecture

To be able to implement our approach; first Matusik et al. [15]'s measured BRDF data must be understood well. In this representation,  $(\theta_{in}, \phi_{in}, \theta_{out}, \phi_{out})$  values are transformed to  $(\theta_{half}, \phi_{half}, \theta_{diff}, \phi_{diff})$  values. This transformation is done according to the [30]. This coordinate system allows varying the sampling density near the specular highlight which is very useful during rendering. For isotropic case,  $\phi_{half}$  values aren't used. So, full BRDF space is described with  $(\theta_{half}, \theta_{diff}, \phi_{diff}) = 90 \times 90 \times 180 = 1\,458\,000$  different values. Reciprocity has an important role on this quantity. Every one of these 1 458 000 values has a unique index value. Matusik et al. [15] also applied to an exponential mapping to half angle to represent specular highlight more efficiently. For a through review of this BRDF representation the reader is referred to [15, 23].

After we handle these data, we see that some parts of BRDF look up table are never used by the model because of this reparameterization and exponential mapping of the half angle. Rusinkiewicz [30]'s new coordinate frame and exponential mapping of the half angle, which are used by Matusik et al.'s representation, makes specular BRDFs and its corresponding index values become more useful and makes diffuse BRDFs and its corresponding index values become less useful. As you see from Fig. 2, we plot histogram of index values of Matusik et al.'s representation according to the natural coordinate system  $(\theta_{in}, \phi_{in}, \theta_{out}, \phi_{out})$ . For this figure, we sample Matusik et al.'s BRDF data according to the standard coordinate frame (using full BRDF space  $(\theta_{in}, \phi_{in}, \theta_{out}, \phi_{out}) = (90 \times 180 \times 90 \times 180)$ ) and we calculate sums of repetitions of index numbers. Then, we plot these values as a histogram. This histogram is independent from materials, so it is the same for full material set.

As you see from Fig. 2, although some index values are never used by the model, they are still stored. Because a different coordinate system and exponential mapping of half angle which are used by the representation make some index values become unused. It is also observed that some neighboring index values are same or too close to each other. Our motivation is eliminating these not used index values and representing same/closer BRDFs in a single cluster.

Our proposed composite associative memory model architecture has three sub layers, namely, input layer, cluster layer (SOM), and storage layer (MAN). The overall system architecture is shown in Fig. 3. Input layer has three neurons whose main responsibility are to accept index vector (1 458 000 values) which is got from Matusik et al.'s representation and to interact with SOM sub layer. The SOM sub layer classifies the input index vector into one of M possible clusters. The operation is a many-to-one mapping which involves mapping a set of 2N possible index combinations to one of M available combinations.



**Fig. 2 - Index histogram of Matusik et al.'s representation according to the natural coordinate system.**

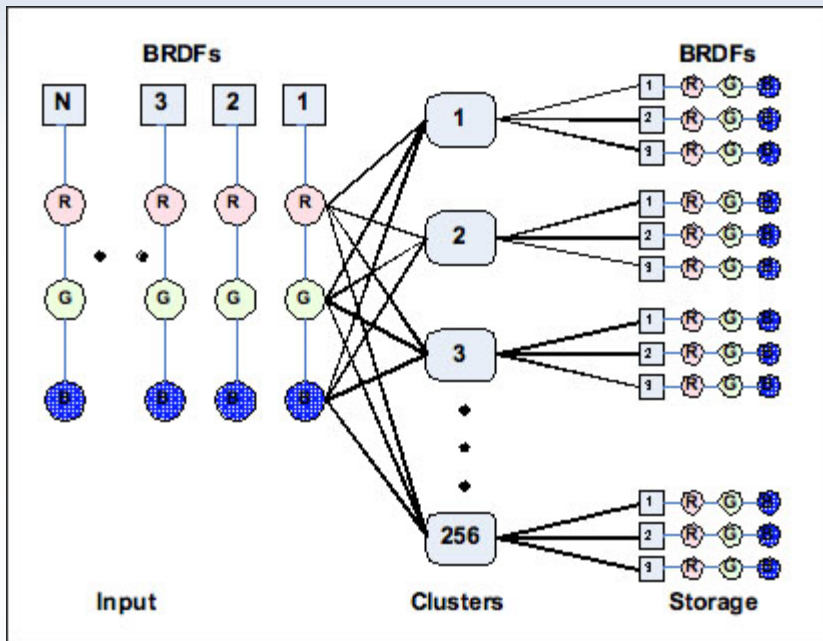
Kohonen SOM imitates the topological and physical structure of the brain. These models are also called as topology preserving maps where the geometrical arrangements of competitive output units are very important. Assuming that the output units are placed on a plane, the two cluster centers get closer and closer as the corresponding input vectors get more and more similar.

The distance between two units is typically calculated as the square of the Euclidean distance ( $L_2$ ). The weight vector for a cluster unit serves as an exemplar of the input patterns associated with that cluster. During the self organization process, the cluster unit whose weight vector matches the input pattern most closely is chosen as the winner. The winning unit and its neighboring units, which are topologically located near this unit, update their weights. For a through review of SOM the reader is referred to [31, 32].

SOM is an excellent unsupervised classifier although it suffers from the local minimum problem as in many competitive approaches. The index cluster centers determined by SOM will be topologically meaningful (close indexes will be represented by neighboring neurons), but SOM will also produce index exemplars (false indexes) which did not actually appear in the input vector space.

To eliminate these false indexes generated through SOM learning process, we have added another layer to the system. This layer –MAN- is responsible for obtaining and keeping the true indexes from the original index vector space. They are replaced with the false index clusters formed over SOM sub layer at the final step of learning.

Mass Attraction Network (MAN) is a neural network architecture proposed as an associative memory, inspired from Newton's mass attraction theory [33, 34]. The model has a similar structure with Hamming Net, where the incoming patterns are stored on the connection weights. It has been reported to be superior to Hamming combination in that it finds the winner at a single iteration, whereas Hamming Net requires Maxnet which iterates a number of times to get the winning neuron [33, 35, 36].



**Fig. 3 - Overall system architecture for  $M=256$**

## 4. Learning

The learning in the system is achieved using a combination of SOM and MAN learning algorithms. This learning is done through three phases. These phases are explained in the following sections.

### 4.1. Phase1: Training SOM

Training starts by introducing index vector (1 458 000 values from Matusik et al.'s BRDF data) into the input layer of the architecture seen in Fig. 3. The input vector is classified into one of the  $M$  possible SOM cluster centers and best matching one is found using the Eq. (1).

$$\Phi = I \left( \min \left\{ \sum_{j=1}^3 (x_p^j - c_i^j)^2 \right\}_{i=1}^{clusters} \right) \quad (1)$$

In Eq. (1), "I" is a function determines the index of the cluster center having minimum Euclidian distance between input vector and cluster weight vector. "x" is input vector of index values which have unique r, g, and b BRDF values. "c" denotes clusters itself. "c<sup>j</sup>" is weight vector components of the i.th cluster carries not used (false) indexes during training. "p" denotes an index from the original index vector. "Φ" is the index of the winner.

After winning cluster is found, the cluster center weight vector is going to be updated by the Eq. (2).

$$c_{\Phi}^j(new) = c_{\Phi}^j(old) + \alpha * x_p^j * (x_p^j - c_i^j)$$

(2)

In the Eq. (2), winner clusters are updated according to the SOM learning rule based on gradient descent. The definitions of the symbols are the same as in Eq. (1). “ $\alpha$ ” is the learning coefficient used for stimulating the learning speed. As can be seen from Eq. (2), the winner cluster weights are updated so that false index vector is created on the weights of SOM.

During the training, these false indexes are continuously replaced by new false indexes. This process is completed by limited number of iterations (epochs). Although the indexes are adjusted for being best representative of the input indexes during the training, they are false and the resultant index spectrum is going to be different from the original. This means that the index reduced form of the original index vector is composed by non-existing LUT (look up table) entries.

## 4.2. Phase2: Adaptive Learning with True Indexes

Because of the learning strategy, false indexes are located on the weights of the SOM. In order not to miss the original index values, another mechanism is used: MAN [33]. MAN, stores the original index vector found efficiently near to the cluster center. Efficiency is constituted by the storage usage on the MAN network. This is the new approach used in proposed work for MAN.

Every cluster center stores “k” neighbor index vector which is stored as closer pattern to the cluster. This allows us to decide whether the input pattern is worth to be stored or not. This is working as if the brain forgets the unused information and replaces with recently obtained valuable data.

A candidate ignorable neighbor of the “k” neighbors of the cluster center on MAN sub layer is found by just calculating the distance between winner cluster center weight vector and its stored “k” vectors. The new input vector is exchanged by winner cluster center having farthest distance from cluster center.

Finding candidate ignorable neighbor is done through the formula shown in Eq. (3).

$$\Psi = I \left( \max \left\{ \sum_{j=1}^3 (c_{\Phi}^j - s_k^j)^2 \right\} \right)^{neighbors}$$

(3)

The Eq. (3) presents the index “ $\Psi$ ” of the ignorable neighbor. “s” symbolizes stored vector of the winner cluster. “k” is the number of stored vector which is three in our case.

After the farthest candidate ignorable neighbor found, it is checked with the input vector. This checking is necessary to prove that candidate ignorable neighbor is worth to be replaced by input vector through the Eq. (4).

$$s_{\Psi}^j(new) = |sign(z - 1)| * s_{\Psi}^j(old) + (1 - |sign(z - 1)|) * x_p^j$$

(4)

In the Eq. (4), “z” is the elementary factor to conclude the replacement decision. If the input vector is closer to the cluster than the candidate ignorable neighbor, “z” concludes replacement should be done. If the



input vector is not closer to the cluster center as compared with ignorable neighbor, replacement is canceled. “z” is defined in Eq. (5). In Eq. (5), distance between ignorable neighbor and cluster center by input vector and cluster center is compared.

$$z = \left| \text{sign} \left( \text{sign} \left( \sum_{j=1}^3 (x_p^j - c_{winner}^j)^2 - \sum_{j=1}^3 (s_{qj}^j - c_{winner}^j)^2 \right) - 1 \right) \right| \quad (5)$$

After replacement is determined, the weight vector of the winning cluster center is updated adaptively. In other words, the learning of the SOM is continuing. This is done through the weight update in Eq. (6).

$$c_{\Phi}^j(\text{new}) = c_{\Phi}^j(\text{old}) + \alpha * s_{qj}^j * (s_{qj}^j - c_i^j) \quad (6)$$

The learning coefficient has been changed at each epoch convenient to the adaptive learning rate as given in Eq. (7)

$$\alpha = \alpha * 0.6 \quad (7)$$

### 4.3. Phase3: LUT&Index Reduced Image construction

After all cluster centers are constructed with its stored vector (which are called as neighbors in this work), LUT is going to be constructed and then new index reduced data can be created.

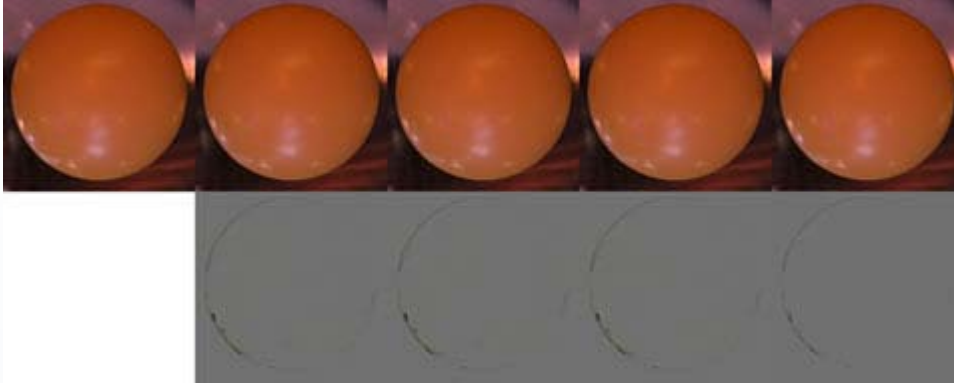
For this purpose, all clusters weights are compared with related stored neighbor vectors. Only the stored vectors having minimum Euclidean distance are taken into LUT table. The other neighbors are ignored. As a result, “M” cluster centers build new LUT with “M” entries that can also be found in old LUT.

The last operation is querying the input vector. All indexes are queried through our composite associative memory model. The model gives the new LUT correspondence for input vector.

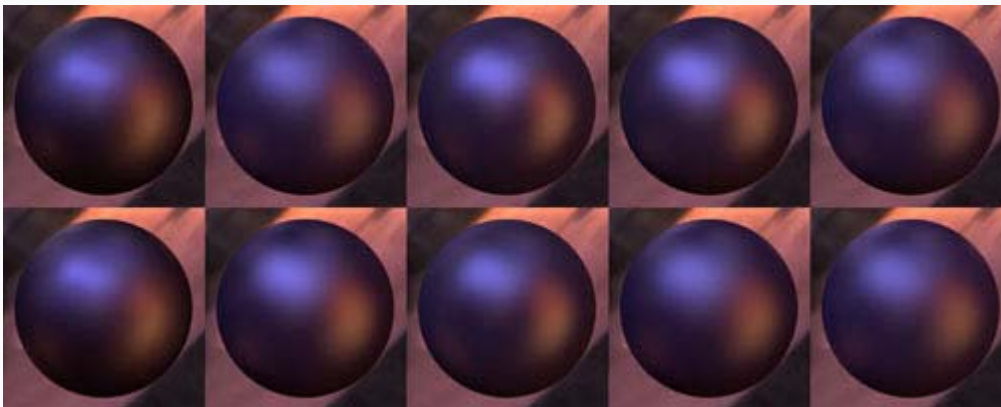
## 5. Results

We now present our table-based data representation, describing the accuracy and the compactness of our representation. In our implementation, there are two important factors which have very important impacts on results; final number of clusters and number of epochs of our algorithm. In original Matusik et al. [15]’s BRDF data, size of index vector is 1 458 000 which must be stored with corresponding three BRDFs. This number can be thought as final number of clusters in our model. In our implementation, this number can be changed according to the visual quality. Number of epochs is another important factor in our implementation. During our investigation, we got accurate results more than 10 epochs. Selecting the number of clusters for our implementation is a manual process. In most cases, general appearance for

measured BRDFs provides enough information for an accurate estimate of how many numbers of clusters are sufficient and optimal. One example scene is shown in Fig. 4 with spheres rendered under environment map. As you see from Fig. 4, numbers of epochs and especially numbers of clusters have very important roles in our representation. In some cases, our representation may have weak highlight compared to original BRDF. In such a case, number of clusters might be increased for getting the same quality specular highlight. In Fig. 4, we also represented the difference images and the Peak Signal-to-Noise Ratios (PSNR).



**Fig. 4 - Representations of measured “yellow-matte-plastic” BRDF with different cases, and the difference images. Left to right: Reference image rendered using Matusik et al.’s measured BRDF representation, using our representation with  $M=64$  clusters and 10 epochs (required data size is 5.56 MB, PSNR = 41.19), using our representation with  $M=64$  clusters and 50 epochs (required data size is 5.56 MB, PSNR = 41.19), using our representation with  $M=64$  clusters and 30 epochs (required data size is 5.56 MB, PSNR = 41.19), using our representation with  $M=1000$  clusters and 30 epochs (required data size is 5.58 MB, PSNR= 44.60).**



**Fig. 5 - Fitting the measured “metallic-blue” BRDF with the eight analytical models, Matusik et al.’s measured BRDF representation and our representation. Clockwise from upper left: Reference image rendered using Matusik et al.’s measured BRDF representation, using Torrance-Sparrow (PSNR = 33.63), using Phong (PSNR = 32.09), using Blinn-Phong (PSNR = 30.97), using Cook-Torrance (PSNR = 32.98), using Ward-Duer (PSNR = 32.86), using Ashikhmin-Shirley (PSNR = 36.09), using Lafortune (PSNR = 34.12), using Ward (PSNR = 33.99), using Our representation (PSNR = 52.12).**

We also would like to show realism and accuracy of our implementation. We sampled BRDFs from Matusik et al.’s BRDF representation. During sampling BRDFs, we used rendered scene for this purpose and we got samples from rendered sphere. Then we fitted these BRDFs which are about 1 500 000 for each color

channel to analytical isotropic models. We chose eight analytical isotropic models for our analysis: Torrance-Sparrow [10], Phong [2], Blinn-Phong [3], Cook-Torrance [11], Ward [4], Lafortune et al. [7], Ashikhmin-Shirley [37], and Ward-Duer [9]. In our fitting process, we used the same way, as [21] have done. So, we used two-lobe version of each model (1 diffuse lobe + 1 specular lobe), and we enforced diffuse contribution to Lambertian effect.

We used the same objective function, and the same non-linear fitting technique which name is Sequential Quadratic Programming (SQP). This technique has an advantage over other non-linear fitting techniques such as Levenberg-Marquardt. With SQP, user can put some constraints on non-linear parameters easily. We saw that these constraints increase fitting performance, since program searches global minimum where it should search. During fitting process, we tried to find best fit parameters for each model. So, we repeated fitting process with different initial values until we found the lowest error for each material.

One example fit, to “metallic-blue” is shown in Fig. 5 with spheres rendered under an environment map. In Fig. 5, we also represented PSNR values for rendered images. Relative quality of the fits from the different models for this material is typical of the Matusik et al.’s data set [15]. As you see from Fig. 5, our novel model and Matusik et al.’s measured BRDFs representation are more realistic than all of the analytical reflectance models. Because the analytical reflectance models aren’t based on real BRDFs data during rendering and their returned BRDFs aren’t exactly equal to the values of the measured BRDFs. The Phong, Blinn-Phong, Ward-Duer, and Ward images are all noticeably different from original data. The highlights near the center of the sphere are significantly brighter than original, and near grazing angle the highlights are much less pronounced. The reason for that is none of them have Fresnel effect. Of the four models, Ward-Duer has the strongest highlight near grazing angle, but it is still noticeably dimmer than measured data. The images from the Ashikhmin-Shirley, Cook-Torrance and Torrance-Sparrow fits are very similar to each other and are mostly faithful to the direct rendering. The Lafortune fit shows an abrupt increase in intensity near grazing angle; however values closer to normal incidence are too low [21]. But our representation exactly matches the Matusik et al.’s measured BRDFs, and in all cases our table-based approach produces an accurate result, in many cases significantly more accurate than fitting an analytical BRDF model.

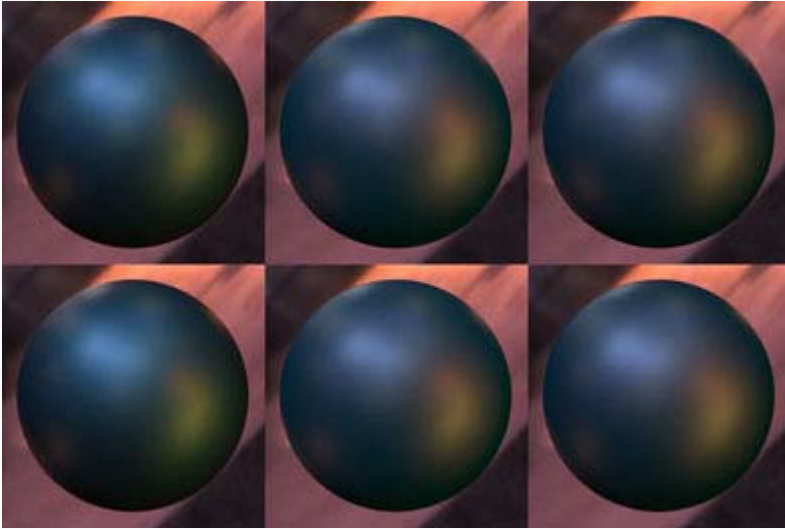
However, our implementation regularizes some minor errors which occur during BRDF measurement phase. (Matusik et al. observe errors, such as deviation from reciprocity, of 10-15% at normal angles, ranging to 60-70% at grazing angles [22]). We conclude that, for measured data, our representation appears to produce results comparable with measurement error.

Since both our representation and Matusik et al.’s measured BRDFs are generative representations, we would like to show their performance under fitting analytical models. So, we generated BRDFs from both of them, and we fitted these data to eight analytical models, namely; Torrance-Sparrow [10], Blinn-Phong [3], Cook-Torrance [11], Ward [4], Oren-Nayar [14], Lafortune et al. [7], Ashikhmin-Shirley [37], and Ward-Duer [9]. In this process, we generated BRDFs which are about 1 500 000 for each color channel from rendered sphere. In our fitting process, we used the same way, as [21] have done. So, we used two-lobe version of each model (1 diffuse lobe + 1 specular lobe), and we enforced diffuse contribution to Lambertian effect, and we used the same metric which is L2 metric, but for Oren-Nayar model, we used its approximation formula which is described in [14], and we used only this approximation formula (just 1 lobe) for fitting this model. For our representation, we used 1000 clusters version of our approach. Table 1 show these results for four materials of Matusik et al.’s dataset; “purple-paint”, “green-metallic-paint”, “yellow-matte-plastic”, and “fabric-beige”. Results show that generated BRDFs of our representation are more proper for analytical models in almost all cases. Because our neural network architecture eliminates BRDFs which don’t represent actual shape of BRDF lobe.

Analytical BRDF Models	Purple-Paint		Metallic-Blue		Yellow-Matte-Plastic		Fabric-Beige	
	Matusik Data	Our Data	Matusik Data	Our Data	Matusik Data	Our Data	Matusik Data	Our Data
Blinn-Phong	0.06421	0.06408	0.02278	0.02288	0.16684	0.08161	0.01043	0.01040
Ward	0.05824	0.05812	0.01458	0.01466	0.15956	0.07562	0.01007	0.01005
Ward-Duer	0.04351	0.04343	0.01711	0.01713	0.13588	0.06421	0.00928	0.00925
Lafortune	0.03982	0.03977	0.01324	0.01330	0.12250	0.05985	0.00752	0.00748
Cook-Torrance	0.01481	0.01465	0.01716	0.01718	0.06915	0.05859	0.00801	0.00797
Torrance-Sparrow	0.01474	0.01459	0.01685	0.01687	0.06915	0.05858	0.00741	0.00738
Ashikhmin-Shirley	0.01622	0.01603	0.01504	0.01507	0.07063	0.05888	0.00531	0.00529

**Table 1 - Fitting performance of our representation and Matusik et al.’s representation under analytical models. In this table, metric is L2 metric. We generate BRDF data from both our representation and Matusik et al.’s representation, and we fit these data to eight analytical models. We list resulting errors for four BRDF data: “purple-paint”, “green-metallic-paint”, “yellow-matte-plastic”, “fabric-beige”**

Fig. 6 is an extension of Table 1. In Fig. 6, we generated BRDFs from both our representation and Matusik et al.’s measured BRDFs representation and we fitted these BRDFs with multi-lobe Cook-Torrance [11] and Ashikhmin-Shirley [37] models. We used the same procedure as we have done in Table 1, but this time, we used three-lobe version of analytic models (1 diffuse lobe + 2 specular lobes). There are two reasons for this; first, when we increment specular lobe number, realism of analytical models increases. Second, difference between fitting an analytical model to generated BRDFs which are from our representation and fitting an analytical model to generated BRDFs which are from Matusik et al.’s measured BRDFs representation becomes more distinctive. In this figure, we used “green-metallic-paint” BRDF. As you see from Fig. 6, BRDFs generated from our representation is more proper for analytical models, and our representation is more realistic than fitting an analytical model, although it contains multi-lobe.



**Fig. 6 - Green-Metallic-Paint: upper row left to right; Rendered using Matusik et al.’s BRDF representation, using Cook-Torrance 3-lobe fitting to BRDFs which are generated by Matusik et al.’s representation (its L2 error is 0.009277), using Ashikhmin-Shirley 3-lobe fitting to BRDFs which are generated by Matusik et al.’s representation (its L2 error is 0.011025), lower row left to right; Rendered using our representation with 1000 clusters, using Cook-Torrance 3-lobe fitting to BRDFs which are generated by our representation (its L2 error is 0.009264), using Ashikhmin-Shirley 3-lobe fitting to BRDFs which are generated by our representation (its L2 error is 0.011005).**

Table 2 lists comparisons of rendering times of BRDF models for two materials of Matusik et al.’s dataset;

“metallic-blue”, and “yellow-matte-plastic”. In this table, we used nine BRDF representations and we used both two-lobe (1 diffuse lobe + 1 specular lobe) and three-lobe (1 diffuse lobe + 2 specular lobes) cases of some analytical models. For parameters of analytical models, we used the same fitting procedure as we have done previously. For Lawrence et al. [22]’s model, we got material parameters from their paper’s website. For all of the analytical models and Lawrence et al.’s model, rendering time vary according to their parameters. So, their rendering time varies from material to material. However, for both our representation and Matusik et al.’s representation rendering times are constant for that scene and sample numbers which are used in Monte Carlo renderings and it doesn’t vary from material to material. In Table 2, we calculated rendering time ratios. In other words, we first calculated rendering times of BRDF models, and then, we divided these values with rendering time of our representation. We did that because of two reasons. First, these ratios are platform independent. Second, with these ratios, trade-off between our representation and other BRDF models can be seen easily. Our representation’s rendering time is always less than Matusik et al.’s representation, because our representation requires less memory. In some situations, rendering time difference between our representation and three-lobe analytical BRDF model representation can be ignorable.

BRDF Models	Metallic-Blue	Yellow-Matte-Plastic
Blinn-Phong (2-lobe)	0.70189	0.77746
Phong (2-lobe)	0.69507	0.80946
Cook-Torrance (2-lobe)	0.72577	0.75071
Ward (2-lobe)	0.71415	0.73596
Ward-Duer (2-lobe)	0.70466	0.73006
Lafortune (2-lobe)	0.70622	0.76944
Ward (3-lobe)	0.73748	0.78244
Cook-Torrance (3-lobe)	0.81300	0.84921
Lawrence et al.	0.77326	0.77903
Matusik et al.	1.00766	1.00830

**Table 2 - Comparisons of Monte Carlo rendering times of BRDF models. In these comparisons, we used the same scene which is the same as our previous figures and the scenes were generated with a path tracer that selected 1024 reflected rays per pixel according to each model separately. This table lists rendering times of BRDF models which are divided by rendering time of our representation for two BRDF data: “metallic-blue”, and “yellow-matte-plastic”**

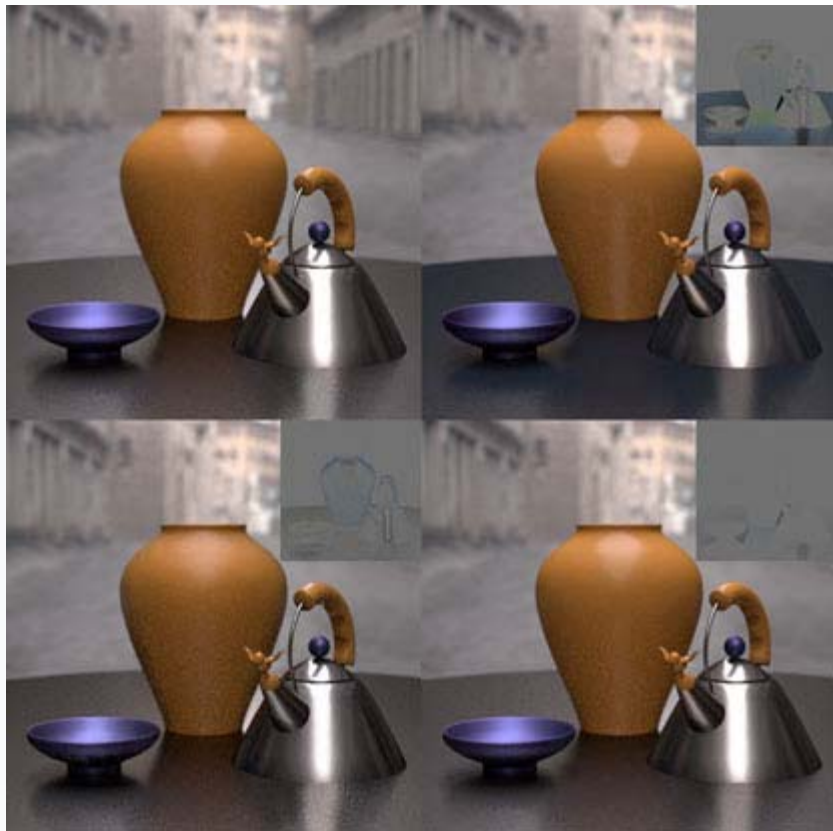
For the BRDFs presented in this paper, we used about 30 MB binary data for Matusik et al.’s representation for each material. However, we used about 5.6 MB binary data for our representation. This shows that our representation is more compact than Matusik et al.’s approach. So, with our representation, scenes which have hundreds or thousands materials are able to be rendered, but with Matusik et al.’s representation, this can't happen.

Binary data of our implementation also has low entropy. Although our approach minimizes redundancy, this low entropy occurs because our approach has to cover full BRDF space. We get about 100 KB data, when we compress our binary data which is about 5.6 MB. This compression can be done during rendering, so materials can be rendered with this memory requirement. On the other hand, we get about 14 MB data, when we compress binary data of Matusik et al.’s representation which is about 30 MB.

The other BRDF representation which may be seen as competitive with our representation is Lawrence et al. [22]’s factored representation (see Fig. 1, and Fig. 7). This model is based on factorization of BRDF and it aims to do efficient importance sampling. This model requires about 200 KB data during rendering, and in compressed case this value becomes about 50 KB. In compressed case, difference between memory requirement of our representation and memory requirement of this representation can be ignorable. This model has an advantage over our representation; it is for both isotropic and anisotropic case, but our

representation is only for isotropic case. This model is more accurate than analytical models, but it is less accurate than our representation. On the other hand, this model doesn't enforce reciprocity, and this situation isn't theoretically desirable, but our representation is reciprocal.

Fig. 7 compares the accuracy of three BRDF representations; Ward [4] BRDF model, Lawrence et al. [22]'s factored BRDF model, and our BRDF representation. We got the scene description from Dave et al. [38]. So, we used Matusik et al.'s three measured data; "nickel", "yellow-matte-plastic", and "metallic-blue" in this scene. For our representation, we used sampling technique of Blinn-Phong [3] BRDF model. We used 1000 clusters for our model's "nickel" and "yellow-matte-plastic" representations, and 256 clusters for our model's "blue-metallic-paint" representation. For Ward BRDF model, we fitted 3 lobes version (1 diffuse lobe + 2 specular lobes) to get lowest errors for that model. As you see from Fig. 7, our representation has higher accuracy than both analytical models and Lawrence et al.'s table-based model.



**Fig. 7 - Reference image for the Princeton scene rendered using Matusik et al.'s measured data (#samples = 16384). Top right: Using Ward's analytical BRDF model (1 diffuse lobe + 2 specular lobes versions) (#samples = 16384). Bottom left: Using Lawrence et al.'s factored BRDF (#samples = 4096). Bottom right: Using our representation (#samples = 16384). Insets show a difference between the reference image and rendered image. The scene is based on the same image from [38].**

Gargan and Neelamkavil's [29] neural network solution for approximating reflectance functions uses neural networks in different way comparing to our representation. They train their neural network architecture with BRDF data, and get network weights as this training result. They also use different coordinate space which is XY parameter space or projected hemispherical space. During rendering, their model uses this parameter space, a scaling constant, and network weights to compute resulting radiance.

On the other hand, our neural network architecture is used during data preparation phase. So, our neural network architecture is independent from rendering. In our representation, Rusinkiewicz [30]'s coordinate

frame is used, and this frame is known to be better than other frames.

## 6. Conclusions and future work

This work addresses solving storage problems of Matusik et al. [15]’s measured BRDFs data representation. For our representation to be implemented, first Matusik et al.’s measured BRDFs representation must be understood well. Then its index vector (and its corresponding BRDFs) which size is 1 458 000 is used in our algorithm as an input vector. So, our representation is built on Matusik et al.’s measured BRDFs data.

Our representation is more compact than Matusik et al.’s representation. It requires at least 5 times less memory. At compressed case, it requires 140 times less memory than Matusik et al.’s representation. It is very realistic and more accurate than both analytical BRDF models and Lawrence et al. [22]’s factored BRDF model.

Our representation also regularizes some minor measurement noise. In other words, our representation decreases variance. Its neural network architecture discards BRDFs which don’t represent actual shape of BRDF well.

On algorithmic side, the representation is efficient and easy to use in both local and global illumination algorithms. In Monte Carlo algorithms, reflection directions for a given incident direction can be sampled according to the [23]. In Matusik thesis, he represented an importance sampling method for his representation, and this importance sampling method can be used for our representation. In this work, we used the sampling of Blinn-Phong [3] BRDF model for our representation.

As future work, we will look into details of more efficient and more compact importance sampling algorithm for our representation. We will also look into details of representing other BRDF representations with less memory requirements.

## 7. Acknowledgements

We would thank Gregory J. Ward for his continuing support, for his constructive comments, and for helpful discussions; Szymon Rusinkiewicz for his invaluable feedbacks, and for discussions.

## References

[1] Nicodemus FE, Richmond JC, Hsia JJ, Ginsberg IW, Limperis T. *Geometric considerations and nomenclature for reflectance*. U.S. Dept. of Commerce, National Bureau of Standards, 1977.

[2] Phong BT. *Illumination for computer generated pictures*. *Communications of the ACM* 1975;18(6):311-17.

[3] Blinn JF. *Models of light reflection for computer synthesized pictures*. *Proceedings of SIGGRAPH’77*, 1977. p. 192-98.

- [4] Ward GJ. *Measuring and modeling anisotropic reflection*. *Proceedings of SIGGRAPH'92*, 1992. p. 265-72.
- [5] Schroder P, Sweldens W. *Spherical wavelets: efficiently representing functions on the sphere*. *Proceedings of SIGGRAPH'95*, 1995. p. 161-72.
- [6] Koenderink JJ, van Doorn AJ, Stavridi M. *Bidirectional reflection distribution function expressed in terms of surface scattering modes*. *Computer Vision -ECCV'96 (Proceedings of the Fourth European Conference on Computer Vision-Volume II)*, Cambridge: Springer, 1996. p. 28-39.
- [7] Lafortune EPF, Foo S-C, Torrance KE, Greenberg DP. *Non-linear approximation of reflectance functions*. *Proceedings of SIGGRAPH'97*, 1997. p. 117-26.
- [8] Ashikhmin M, Premoze S, Shirley P. *A microfacet based BRDF generator*. *Proceedings of SIGGRAPH'00*, 2000. p. 65-74.
- [9] Duer A. *On the Ward model for global illumination*. Submitted for review, 2004.
- [10] Torrance KE, Sparrow EM. *Theory for off-specular reflection from roughened surfaces*. *Journal of the Optical Society of America* 1967; 57(9): 1105–114.
- [11] Cook RL, Torrance KE. *A reflectance model for computer graphics*. In *Computer Graphics (SIGGRAPH'81 Proceedings)* 1981; 15(3): 307–16.
- [12] Poulin P, Fournier A. *A model for anisotropic reflection*. *Proceedings of SIGGRAPH'90*, 1990. p. 273-82.
- [13] He XD, Torrance KE, Sillion FX, Greenberg DP. *A comprehensive physical model for light reflection*. *Proceedings of SIGGRAPH'91*, 1991. p. 175-86.
- [14] Oren M, Nayar SK. *Generalization of Lambert's reflection model*. *Proceedings of SIGGRAPH'94*, 1994. p. 239-46.
- [15] Matusik W, Pfister H, Brand M, McMillian L. *A data driven reflectance model*. *ACM Transactions on Graphics (TOG)* 2003;22(3):759-69.
- [16] CURET: Columbia-Utrecht reflectance and texture database. (<http://www.cs.columbia.edu/CAVE/curet/>).
- [17] Cornell University. *Program of computer graphics online resource: Reflectance data*. (<http://www.graphics.cornell.edu/online/measurements/reflectance/index.html>).
- [18] NIST reference reflectometer: STARR facility. (<http://physics.nist.gov/>).
- [19] Dana KJ, van Ginneken B, Nayar SK, Koenderink JJ. *Reflectance and texture of real-world surfaces*. *ACM Trans. Graph. (TOG)* 1999;18(1):1-34.
- [20] Marschner SR, Westin SH, Lafortune EPF, Torrance KE. *Image-based bidirectional reflectance distribution function measurement*. *Applied Optics* 2000;39(16):2592-600.
- [21] Ngan A, Durand F, Matusik W. *Experimental analysis of BRDF models*. *Rendering Techniques '05 (Proceedings of the Eurographics Symposium on Rendering)*, Konstanz: Eurographics Association, 2005. p. 117-26.
- [22] Lawrence J, Rusinkiewicz S, Ramamoorthi R. *Efficient BRDF importance sampling using a factored representation*, *ACM Transactions on Graphics (TOG)* 2004;23(3):496-505.



- [23] Matusik W. *A data-driven reflectance model*. PhD. thesis, Massachusetts Institute of Technology, 2003.
- [24] Greenberg DP, Torrance KE, Shirley P, Arvo J, Lafortune EPF, Ferwerda JA, et al. *A framework for realistic image synthesis*. *Proceedings of SIGGRAPH'97*, 1997. p. 477-94.
- [25] Marschner S, Westin S, Lafortune EPF, Torrance KE, Greenberg D. *Image-based BRDF measurement including human skin*. *Rendering Techniques'99*, 1999. p. 139-52.
- [26] Kautz J, McCool MD. *Interactive rendering with arbitrary BRDFs using separable approximations*. *Rendering Techniques '99 (Proceedings of the Tenth Eurographics Workshop on Rendering)*, Granada: Springer, 1999. p. 247-60.
- [27] McCool MD, Ang J, Ahmad A. *Homomorphic factorization of BRDFs for high-performance rendering*. *Proceedings of SIGGRAPH'01*, 2001. p. 185-94.
- [28] Suykens F, vom Berge K, Lagae A, Dutre P. *Interactive rendering with bidirectional texture functions*. *Comp. Graphics Forum (Proc. Eurographics) 2003;22(3):463-72*.
- [29] Gargan D, Neelamkavil F. *Approximating reflectance functions using neural networks*. *Rendering Techniques'98*, 1998. p. 23-34.
- [30] Rusinkiewicz S. *New change of variables for efficient BRDF representation*. *Rendering Techniques'98*, 1998. p. 11-22.
- [31] Kohonen T. *Self-organization and associative memory*. 3rd ed. Berlin: Springer-Verlag; 1989.
- [32] Kohonen T. *The self-organizing map*. In *Proceedings of IEEE 1990;78(9):1464-80*.
- [33] Erdem MH, Ozturk Y. *MAN: mass attraction network*. *IEEE Conference on Circuits and Systems (ISCAS 1994)*, 1994. p. 455-58.
- [34] Baskomurcu G, Erdem MH, Ozturk Y. *Parallel implementation of mass attraction network*. In *Proceedings of International Conference on Artificial Neural Networks (ICANN 1994)*, 1994.
- [35] Lippmann RP. *An introduction to computing with neural nets*. *IEEE ASSP Magazine*, 1987. p. 4-22.
- [36] Lippmann RP, Gold B, Malpass ML. *A comparison of hamming and hopfield neural nets for pattern classification*. *MIT Lincoln Laboratory Technical Report TR-769*, 1988.
- [37] Ashikhmin M, Shirley P. *An anisotropic phong BRDF model*. *Journal of Graphics Tool*, 2000;5(2):25-32.
- [38] Edwards D, Boulos S, Johnson J, Shirley P, Ashikhmin M, Stark M, et al. *The halfway vector disk for BRDF modeling*. *ACM Transactions on Graphics (TOG) 2006;25(1):1-18*.

## About the Authors



**Murat Kurt**

*was born in Izmir, Turkey, in 1981. He received B.Sc. degree in the civil engineering from Engineering Department, Dokuz Eylul University in 2002 and M.Sc. degree in computer science from International Computer Institute, Ege University in 2007. He has been a Research Assistant with the International Computer Institute since 2005. His research interests are algorithms, computer graphics with focuses on BRDF models, appearance modeling, global illumination algorithms, photorealistic rendering and digital image processing with focuses on image registration, wavelets, LPR. He is also currently a Ph.D. student in the International Computer Institute, Ege University.*



**Muhammed G. Cinsdikici**

*was born in Izmir, Turkey, in 1973. He received B.Sc., M.Sc. degrees in the computer engineering from Computer Engineering Department and Ph.D. degree in information technologies from International Computer Institute, Ege University, Turkey in 1995, 1997 and 2004 respectively. He has been an Assistant Professor with the International Computer Institute since 2004. His research interests are classification and clustering with neural networks, computer vision algorithms with their applications, image processing and biometrics. He is a member of IEEE and IEEE Computer Society.*