applied optics

Bidirectional texture function image super-resolution using singular value decomposition

WEI DONG,[1] HUI-LIANG SHEN,[1,*] ZHI-WEI PAN,[1] AND JOHN H. XIN[2]

[1]College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China
[2]Institute of Textiles and Clothing, Hong Kong Polytechnic University, Hong Kong, China
*Corresponding author: shenhl@zju.edu.cn

Received 10 January 2017; revised 2 March 2017; accepted 2 March 2017; posted 3 March 2017 (Doc. ID 284450); published 24 March 2017

The bidirectional texture function (BTF) is widely employed to achieve realistic digital reproduction of real-world material appearance. In practice, a BTF measurement device usually does not use high-resolution (HR) cameras in data collection, considering the high equipment cost and huge data space required. The limited image resolution consequently leads to the loss of texture details in BTF data. This paper proposes a fast BTF image super-resolution (SR) algorithm to deal with this issue. The algorithm uses singular value decomposition (SVD) to separate the collected low-resolution (LR) BTF data into intrinsic textures and eigen-apparent bidirectional reflectance distribution functions (eigen-ABRDFs) and then improves the resolution of the intrinsic textures via image SR. The HR BTFs can be finally obtained by fusing the reconstructed HR intrinsic textures with the LR eigen-ABRDFs. Experimental results show that the proposed algorithm outperforms the state-of-the-art single-image SR algorithms in terms of reconstruction accuracy. In addition, thanks to the employment of SVD, the proposed algorithm is computationally efficient and robust to noise corruption.

OCIS codes: (100.3020) Image reconstruction-restoration; (100.6640) Superresolution; (290.1483) BSDF, BRDF, and BTDF.

https://doi.org/10.1364/AO.56.002745

1. INTRODUCTION

With the continuously increasing capabilities of computer hardware, more and more attention is being paid to synthesize photorealistic images. Many industrial applications require realistic reproduction of material behavior under complex viewing and lighting conditions. This behavior can be described by the bidirectional reflectance distribution function (BRDF), which was first introduced in [1]. The BRDF can be represented as a physically based analytic model, such as the Cook–Torrance [2], Ward [3], and He [4] models. These representations make the BRDF convenient to be applied in practical applications, such as computer graphics rendering. Despite its convenient usage, the BRDF is suitable only for homogeneous materials that do not contain textures. In the real world, however, a large number of rough surfaces with complicated spatial structures exist. These rough surfaces cause various effects, such as shadowing, masking, inter-reflection, and subsurface scattering, which cannot be described by BRDF.

The bidirectional texture function (BTF) was introduced by Dana *et al.* [5] to capture at least some of the above effects. BTF is essentially a six-dimensional function parameterized by position p, viewing direction ω_o, and lighting direction ω_i, as illustrated in Fig. 1. As various effects are merged in this representation, the BTF can make the rendering extremely realistic.

The wide range of surface effects usually makes the BTF too complicated to be synthesized through analytic models or simple simulations. Hence, some image-based methods have been developed to represent BTF data [6–8]. These methods rely on measured BTFs (acquired real-world data) in combination with appropriate synthesis algorithms. Generally, the appearances of the real-world samples are captured by color camera(s) [9], so the BTFs can be regarded as a collection of RGB texture images. In the existing measurement systems, the image resolutions of the BTFs are relatively low [5,10,11]. The employment of low-resolution (LR) cameras is mainly due to three reasons. First, high-resolution (HR) cameras are usually expensive. Second, the measurement time of HR images, which can be dozens of hours, is too long for practical applications. Third, the acquired raw HR BTF data require a huge storage space that can be hundreds of gigabytes.

An issue incurred by the employment of LR cameras is the loss of texture details, as illustrated in Fig. 2. The solution is to computationally reconstruct the HR BTF images from the acquired LR ones using the image super-resolution (SR) technique. However, as the BTFs of a single material have a large number of images (thousands to tens of thousands), applying image SR directly on BTFs requires a long computational time. Besides, the reconstructed HR BTFs will still take a huge

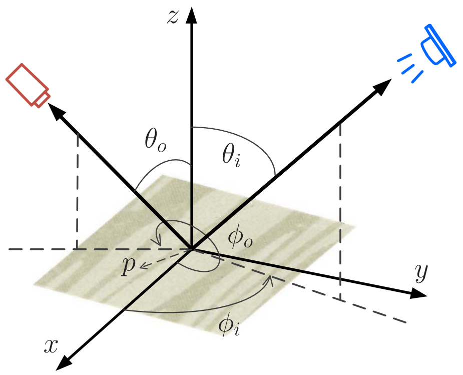

Fig. 1. Parameters describing BTF geometry in the sample coordinate system, including the viewing direction ω_o:=(θ_o, ϕ_o), lighting direction ω_i:=(θ_i, ϕ_i), and pixel position p:=(x, y).

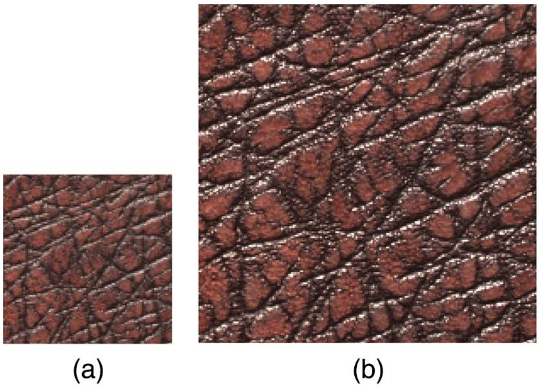

Fig. 2. Example of (a) a LR BTF image and (b) a HR BTF image of the same material. Texture details are lost in the LR image.

storage space. These two limitations need to be resolved for practical applications.

This paper proposes a fast image SR algorithm for BTF reconstruction using singular value decomposition (SVD). The employment of SVD is based on the key observation that BTFs are highly relevant and thus each can be represented by the linear combinations of a few basis functions. The flow chart of the proposed algorithm (see Fig. 3) is depicted as follows. First, all the LR BTFs of the material sample are acquired and consequently transformed to a BTF matrix. The BTF matrix is decomposed into eigen-textures, singular values, and eigen-ABRDFs (apparent BRDFs [12]) using SVD. By keeping only a few principal components, the BTF data can be compressed and the noise can be removed. Second, intrinsic textures are obtained by weighting the eigen-textures with corresponding singular values. Image SR is then applied on the intrinsic textures instead of the original BTF images. Finally, HR BTFs are obtained by fusing the HR intrinsic textures and LR eigen-ABRDFs. As the number of intrinsic textures is far less than that of BTF images, the computational time can be greatly reduced when compared with the direct SR algorithms.

To summarize, the contributions of the proposed algorithm are as follows:

- An image super-resolution algorithm is proposed for reconstructing high-resolution BTF data based on singular value decomposition.
- High computational efficiency is achieved by applying super-resolution on the decomposed intrinsic textures instead of the raw BTF images.
- By keeping a limited number of intrinsic textures, the proposed algorithm is robust to noise corruption in HR BTF reconstruction.

The rest of the paper is organized as follows. Section 2 briefly reviews the works related to BTF acquisition and compression, as well as image SR. Section 3 presents the proposed algorithm. Experimental results are demonstrated in Section 4. Finally, Section 5 concludes this paper.

2. PRIOR ARTS

In this section, we provide a brief review of the works related to BTF acquisition and compression, as well as image super-resolution.

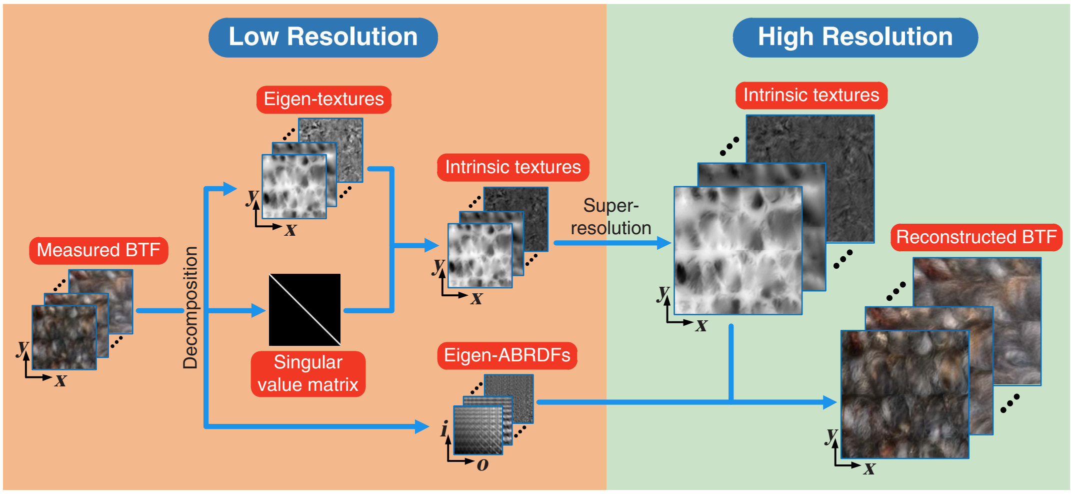

Fig. 3. Flow chart of the proposed BTF image super-resolution algorithm.

A. BTF Acquisition and Compression

In the literature, the most common BTF measurement devices are the ones with gonioreflectometer-like structures [5,10,13]. These devices are usually equipped with a camera and a light source, both of which can move across the hemisphere above the planar sample to capture the material appearance under different viewing and lighting directions. The shortage is that the measurement time is very long, with most spent on data transfer from the camera to the computer. In [14], the authors used an array of 151 commodity digital still cameras mounted on a hemispherical gantry for BTF collection. The measurement device can simultaneously acquire the material appearance of 151 different viewing directions, and thus the data acquisition efficiency is much improved. However, the cost of this measurement device is quite expensive.

The image resolutions of acquired BTFs are 256 × 256 pixels in [10] and 400 × 400 pixels in [15]. Although the image resolution is relatively low, the total size of BTF data is still huge because of the large number of viewing and lighting directions. For convenient use of BTFs in practical applications, several BTF compression algorithms have been introduced in the literature, among which SVD is about the simplest one [16]. With SVD, the transformed raw BTF matrix is decomposed to an eigen-texture matrix relating to spatial positions and an eigen-ABRDF matrix relating to viewing/lighting directions. Since the SVD can reduce the amount of data while retaining the integrity of relevant dimension information, it is widely used in many BTF extended applications. For example, in [17], the authors propose a method to generate BTFs for large-scale material samples by applying a texture optimization on the eigen-texture matrix. In [18], the authors use the eigen-ABRDF matrix to select sparse acquisition positions and reconstructs the entire BTFs from the angularly sparse measurements.

B. Image Super-Resolution

The goal of single-image SR is to estimate a HR image from a LR observed one [19]. There are mainly three categories of algorithms dedicated to the image SR problem, i.e., interpolation-based ones [20–22], reconstruction-based ones [23–25], and learning-based ones [26–28].

The interpolation-based algorithms are widely used for producing zoom-in images because of their simplicity and easy implementation. However, these algorithms tend to blur the high-frequency details and often introduce noticeable aliasing artifacts the along edges.

The reconstruction-based algorithms estimate the HR image by requiring its blurred and down-sampled version to be consistent with the original LR image [23] and incorporating prior knowledge into the ill-posed inverse problem. The reconstruction accuracy depends on the prior and its compatibility with the given image.

The learning-based algorithms learn the relations between LR and HR image patches from a given database. Adopting the philosophy of LLE [29] from manifold learning, the work in [26] maps the local geometry of the LR patch space to that of the HR patch space by assuming similarity between the two manifolds. In [28], the authors construct a set of mapping relationships between the LR and HR patches by grouping correlative neighbors using a learned LR–HR dictionary pair and efficiently reduce the runtime. The effectiveness of learning-based algorithms depends highly on the supporting image database, and the reconstruction results will degrade when similar patches do not exist in the database.

Instead of learning a relationship between LR–HR image patches, the work [30] first uses SVD to decompose the images and then learns a mapping of the singular vectors between the LR and HR images. We note that although our proposed algorithm also employs SVD, the fundamentals of these two algorithms are totally different. More specifically, the proposed algorithm uses SVD to decompose the entire BTF matrix but not a single image as in [30]. Besides, the proposed algorithm belongs to the reconstruction category and does not require any training database for image SR.

3. PROPOSED BTF IMAGE SUPER-RESOLUTION

In the proposed algorithm, only one HR reference image at a specified viewing/lighting direction is acquired for each material. Based on this LR–HR image pair, the HR versions of all the other BTF images are reconstructed from the corresponding LR ones using optimization. In the following, we first give the fundamentals of imaging model and image SR and then introduce the proposed SVD-based BTF image SR algorithm. The consideration of image boundary in the practical computation will also be discussed.

A. Fundamentals

Suppose that a BTF measurement system has m_o viewing directions and m_i lighting directions, and then a total number of $N_s = m_o \times m_i$ LR images of a sample are acquired in the measurement process. After the geometric rectification of the acquired images, the resolution of each LR image is supposed to be $m_l \times n_l$ pixels, where m_l and n_l are the numbers of image rows and columns, respectively. The objective of BTF image super-resolution is to reconstruct all the LR images to the HR images with resolution $m_h \times n_h$ pixels. Here, $m_h = m_l \times d_r$ and $n_h = n_l \times d_r$, with d_r being an integer factor not less than 2.

In the common single-image case, the relationship between the LR and HR images can be formulated as

$$\mathbf{y} = \mathbf{DHx} + \mathbf{n} = \mathbf{Px} + \mathbf{n}, \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^{N_l}$, with $N_l = m_l \times n_l$, denotes the observed LR image, and $\mathbf{x} \in \mathbb{R}^{N_h}$, with $N_h = m_h \times n_h$ denotes the HR image to be estimated. The vectors $\mathbf{y}$ and $\mathbf{x}$ are obtained by stacking the corresponding images into column vectors in a lexicographic order. In Eq. (1), $\mathbf{n} \in \mathbb{R}^{N_l}$ denotes an independent identically distributed additive white Gaussian noise, $\mathbf{D} \in \mathbb{R}^{N_l \times N_h}$ is the down-sampling matrix, $\mathbf{H} \in \mathbb{R}^{N_h \times N_h}$ is the blurring matrix, which is generated by the blur kernel, and $\mathbf{P}{:=}\mathbf{DH}$. In the BTF system, the blur kernel between the LR and HR images in different viewing and lighting directions is fixed and can be computed by using a collection of HR and LR images [31].

Estimating the HR image from the LR image is an ill-posed problem because, according to Eq. (1), many HR images can produce the same LR image. Incorporating a proper regularization with the target image $\mathbf{x}$, the maximum *a posteriori*

(MAP) estimator of $\mathbf{x}$ for image SR can be obtained by solving the optimization problem

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{P}\mathbf{x}\|_2^2 + \alpha\phi(\mathbf{x}), \tag{2}$$

where the first term is a data fidelity term that enforces the blurred and the down-sampled version of the HR image to be close to the observed LR image, and the second term is the regularization related to the image prior information. The parameter α is a trade-off between the two terms. The regularization term can adopt edge smooth priors [32–34] or image statistical priors [24,35]. In this paper, we simply use the image gradient constraint with ℓ^2 norm [32]. In the following, the image super-resolution based on Eq. (2) will be referred to as the single-image reconstruction (SIR) algorithm.

B. SVD-Based BTF Super-Resolution

Let $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{N_s}] \in \mathbb{R}^{N_l \times N_s}$ be the acquired LR BTF matrix and $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_s}] \in \mathbb{R}^{N_h \times N_s}$ be the corresponding HR BTF matrix to be estimated. Clearly, $\mathbf{X}$ can be recovered in a column by column manner by solving Eq. (2). However, as N_s is very large, the reconstruction process is very time-consuming. In the following, we introduce a fast BTF image SR algorithm based on SVD.

By applying SVD to $\mathbf{Y}$ and adopting only C $(0 < C < N_s)$ principal components, $\mathbf{Y}$ is decomposed into three matrices that, when multiplied, approximate the original matrix

$$\mathbf{Y} \approx \tilde{\mathbf{Y}} = \mathbf{U}_y \mathbf{S}_y \mathbf{V}_y^T = \mathbf{A}_y \mathbf{V}_y^T, \tag{3}$$

where $\mathbf{U}_y \in \mathbb{R}^{N_l \times C}$ is the eigen-texture matrix, $\mathbf{V}_y \in \mathbb{R}^{N_s \times C}$ is the eigen-ABRDF matrix, and $\mathbf{S}_y \in \mathbb{R}^{C \times C}$ is a diagonal matrix of singular values. The weighted eigen-texture matrix $\mathbf{A}_y{:=}\mathbf{U}_y\mathbf{S}_y$ is referred to as intrinsic textures in this work. The parameter C is the number of intrinsic texture components. When $C = N_s$, the matrix $\tilde{\mathbf{Y}}$ will be exactly $\mathbf{Y}$. In practical computation, we always let $C < N_s$ so that the BTF can be compressed or more effectively represented, and image noise can be removed to some extent.

Note that with the decomposition in Eq. (3), the BTF information along the pixel position is separated from that along the viewing/lighting directions. The column vectors of matrix $\mathbf{V}_y$ are actually the basis functions of viewing/lighting directions and are irrelevant to the texture coordinate, or in other words, irrelevant to image resolution. Hence, $\mathbf{V}_y$ can also be the proper basis functions of the HR BTF matrix $\mathbf{X}$. More specifically, the HR BTF matrix can be reconstructed from $\mathbf{V}_y$ as follows:

$$\tilde{\mathbf{X}} = \mathbf{B}_x \mathbf{V}_y^T, \tag{4}$$

where $\mathbf{B}_x$ is computed as

$$\mathbf{B}_x = \mathbf{X}\mathbf{V}_y. \tag{5}$$

When the number of intrinsic texture components, C, is appropriate, matrix $\tilde{\mathbf{X}}$ should approximate $\mathbf{X}$, i.e., $\tilde{\mathbf{X}} \approx \mathbf{X}$. Then, estimating the HR BTF matrix $\mathbf{X}$ is equivalent to estimating $\tilde{\mathbf{X}}$. Moreover, as the basis function matrix $\mathbf{V}_y$ is known, it is also equivalent to estimating $\mathbf{B}_x$. In the following, we will show the relationship between $\mathbf{B}_x$ and $\mathbf{A}_y$.

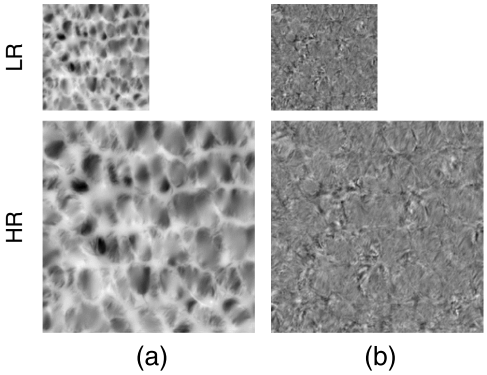

Fig. 4. Images of the LR and HR intrinsic texture components of the material *carpet*06 in the UBO2014 BTF dataset [15]. (a) The image of the first intrinsic texture component. (b) The image of the 300th intrinsic texture component. The images are adjusted for better visualization.

By left multiplying the both sides of Eq. (5) with $\mathbf{P}$, we have $\mathbf{P}\mathbf{B}_x = \mathbf{P}\mathbf{X}\mathbf{V}_y$. For simplicity, at the moment we assume the acquired BTF data is free of noise. Then, according to the imaging model in Eq. (1), we have $\mathbf{P}\mathbf{B}_x = \mathbf{Y}\mathbf{V}_y$. According to SVD, the intrinsic texture matrix $\mathbf{A}_y$ relates to the LR BTFs as $\mathbf{A}_y = \mathbf{Y}\mathbf{V}_y$. Hence, we have

$$\mathbf{A}_y = \mathbf{P}\mathbf{B}_x. \tag{6}$$

When considering image noise, the above relationship becomes $\mathbf{A}_y \approx \mathbf{P}\mathbf{B}_x$. Equation (6) indicates that $\mathbf{A}_y$ is the LR version of $\mathbf{B}_x$ with respect to the blurring down-sampling matrix $\mathbf{P}$.

As illustrated in Fig. 4, by transforming intrinsic textures to images, estimating $\mathbf{B}_x$ from $\mathbf{A}_y$ can be casted to the image SR problem. Letting $\mathbf{a}_{y,i}$, where $i \in \{1, 2, \ldots, C\}$, be the ith column vector of the LR intrinsic texture matrix $\mathbf{A}_y$, the MAP estimator of $\mathbf{b}_{x,i}$ can be obtained by solving the following optimization problem:

$$\hat{\mathbf{b}}_{x,i} = \arg\min_{\mathbf{b}_{x,i}} \|\mathbf{a}_{y,i} - \mathbf{P}\mathbf{b}_{x,i}\|_2^2 + \beta\|\mathbf{L}\mathbf{b}_{x,i}\|_2^2, \tag{7}$$

where $\mathbf{L} \in \mathbb{R}^{N_h \times N_h}$ is a Laplacian matrix and β is a weighted parameter. Accordingly, the HR intrinsic texture matrix is $\hat{\mathbf{B}}_x = [\hat{\mathbf{b}}_{x,1}, \hat{\mathbf{b}}_{x,2}, \ldots, \hat{\mathbf{b}}_{x,C}]$. With the obtained $\hat{\mathbf{B}}_x$, the HR BTF can be reconstructed as

$$\hat{\mathbf{X}} = \hat{\mathbf{B}}_x \mathbf{V}_y^T. \tag{8}$$

In this work, the above algorithm is referred to as the SVD-based SR algorithm. Compared with the SIR algorithm in Eq. (2), the SVD-based SR algorithm efficiently reduces the reconstruction time since the number of intrinsic texture components, C, is far less than the number of images N_s. In addition, the SVD decomposition reduces the noise contained in the raw LR BTF images, and thus the SVD-based SR algorithm can be more robust to image noise.

C. Practical Consideration

In the imaging model in Eq. (1), $\mathbf{H}$ is generally a block circulant matrix with circulant blocks structured blurring matrix formed by the blur kernel when the periodic boundary condition of convolution is used [36]. According to [37], the periodic boundary condition is not satisfied in practical situations, and it can cause ringing artifacts in the boundaries of the reconstructed HR image. As suggested in [37], the occurrence of artifacts can be completely eliminated by using an exact model of the blurring process. Assuming that the blur kernel has size $(2w+1)\times(2w+1)$, Eq. (1) can be rewritten as

$$\mathbf{y} = \mathbf{D}'\mathbf{H}'\mathbf{x}' + \mathbf{n} = \mathbf{P}'\mathbf{x}' + \mathbf{n}, \tag{9}$$

where $\mathbf{x}' \in \mathbb{R}^{(m_h+2w)(n_h+2w)}$, $\mathbf{D}' \in \mathbb{R}^{N_l\times(m_h+2w)(n_h+2w)}$, $\mathbf{H}' \in \mathbb{R}^{(m_h+2w)(n_h+2w)\times(m_h+2w)(n_h+2w)}$, and $\mathbf{P}' \in \mathbb{R}^{N_l\times(m_h+2w)(n_h+2w)}$. Note that $\mathbf{y}$ is still the original LR image as in Eq. (1). With this modification, the unknown pixels around the boundary of the HR image are introduced to produce the true blurred pixels of the image boundary. On the other hand, the wrong blurred pixels produced by them are discarded during the downsampling process. Then, the common image SR image problem is formulated as

$$\hat{\mathbf{x}}' = \arg\min_{\mathbf{x}'}\|\mathbf{y} - \mathbf{P}'\mathbf{x}'\|_2^2 + \alpha\phi(\mathbf{x}'). \tag{10}$$

For the proposed SVD-based SR algorithm, the expanded HR intrinsic texture component can be reconstructed by solving the following optimization problem:

$$\hat{\mathbf{b}}'_{x,i} = \arg\min_{\mathbf{b}'_{x,i}}\|\mathbf{a}_{y,i} - \mathbf{P}'\mathbf{b}'_{x,i}\|_2^2 + \beta\|\mathbf{L}\mathbf{b}'_{x,i}\|_2^2, \tag{11}$$

where $\mathbf{L}$ is the Laplacian operator. The reconstructed HR intrinsic texture component $\hat{\mathbf{b}}_{x,i}$ can be obtained by discarding the boundary zone of width w in the periphery of the intrinsic texture $\hat{\mathbf{b}}'_{x,i}$. Then, the HR BTFs can be reconstructed according to Eq. (8).

4. EXPERIMENTS

In the experiments, we evaluate the proposed SVD-based SR algorithm on both synthetic and real data. The synthetic data are generated based on the Bonn University UBO2014 dataset [15], which was captured using the technique introduced in [14]. The real BTF images were captured using the imaging system presented in our previous work [9]. The effectiveness of the proposed algorithm is compared with four baseline algorithms, i.e., bicubic interpolation [23], anchored neighborhood regression (ANR) [28], locally linear embedding (LLE) [26], and the SIR in Eq. (2). The ANR and LLE algorithms are both learning-based algorithms. In the experiments, the parameter α in SIR and the parameter β in the proposed algorithm are both set as 10^{-3}. The number of intrinsic texture components is set as $C = 300$. With this setting, more than 99% of the information of the raw LR BTF data can be kept after SVD.

As the size of BTF data is huge, it is impractical to use the conventional algorithm [38] to directly compute a truncated SVD for a material considering the high requirement of computation time and memory space. Hence, in this work, we use the technique introduced in [39] to obtain the eigen-ABRDFs $\mathbf{V}_y$, and then compute the intrinsic textures as $\mathbf{A}_y = \mathbf{Y}\mathbf{V}_y$.

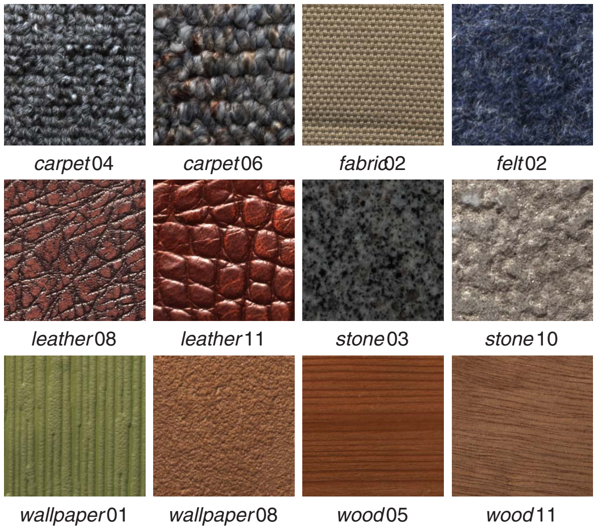

Fig. 5. Example images of the 12 real materials used in the experiment.

We note that the proposed algorithm provides a solution to universal resolution improvement on various textures, under the guidance of a HR reference image. In addition, the algorithm also considers the possible out-of-focus blur that may occur in practical applications. For a fair comparison with the baseline algorithms, the experiments are conducted in the circumstance of 2 × resolution magnification.

A. Synthetic Data Results

The UBO2014 dataset consists of seven semantic categories with 12 different materials in each category. Several examples of these materials are illustrated in Fig. 5. For each material, its full-resolution (or HR) BTFs are measured using the bidirectional sampling of 151 viewing directions and 151 lighting directions. The rectified texture images have a spatial resolution of 400 × 400 pixels.

The LR images are produced from the original HR images by applying a 9 × 9 Gaussian blurring kernel with a standard deviation 1.5 and decimated by a factor of $d_r = 2$. For each material, one HR reference image taken under the front view $\omega_o = (0°, 0°)$ with the lighting direction $\omega_i = (45°, 120°)$ is picked. The picked HR images (one image for each material) and their corresponding LR images are used to estimate the blur kernel required in the proposed and SIR algorithms and are also used as the training database of the ANR and LLE algorithms.

The accuracy of these algorithms is evaluated in terms of mean root mean square error (mRMSE), mean peak signal-to-noise ratio (mPSNR), and mean relative error (mRE) on all the BTFs. The mRMSE of a material is computed as

$$\mathrm{mRMSE} = \frac{1}{N_s}\sum_{k=1}^{N_s}\left(\frac{1}{\sqrt{N_h}}\|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_2\right), \tag{12}$$

where $\mathbf{x}_k$ and $\hat{\mathbf{x}}_k$ are the kth ground truth (GT) and reconstructed HR images (in column vector format), respectively. The mPSNR is defined as

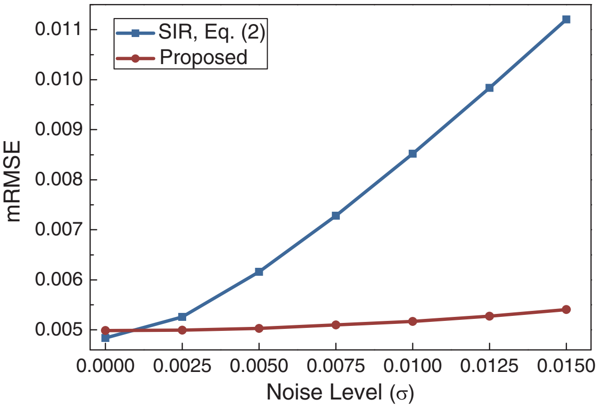

Fig. 6. Average reconstruction error of all materials using the SIR algorithm and the proposed algorithm, under different noise levels (σ).

$$\mathrm{mPSNR} = \frac{1}{N_s}\sum_{k=1}^{N_s} 20\log_{10}\frac{\max(\mathbf{x}_k)}{\frac{1}{\sqrt{N_h}}\|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_2}, \tag{13}$$

where the operator $\max(\mathbf{x}_k)$ returns the maximal entry in $\mathbf{x}_k$. The mRE is computed as

$$\mathrm{mRE} = \frac{1}{N_s}\sum_{k=1}^{N_s}\left(\frac{\|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_2}{\|\mathbf{x}_k\|_2} \times 100\%\right). \tag{14}$$

As discussed above, owing to the employment of SVD, the proposed algorithm is more robust to image noise when compared with the SIR algorithm. Figure 6 shows the mRMSE of all the UBO2014 materials of the two algorithms under various noise levels, with the standard deviation σ changing from 0 to 0.015 in the range [0, 1]. It is observed that when the LR images are free of noise, the reconstruction error of the proposed algorithm is a little higher than that of the SIR algorithm. This slight accuracy degradation is because that the number of intrinsic texture components $C = 300$ is much less than the number of viewing and lighting directions $N_s = 151 \times 151$. In the case of noise corruption, the reconstruction errors of the proposed algorithm are always lower than those of the SIR algorithm. The accuracy improvement is very obvious when the noise level is high.

Table 1 further shows the reconstruction errors of seven semantic categories of the UBO2014 dataset when the noise level is $\sigma = 0.01$. This noise level is adopted to simulate the imaging noise of our real BTF imaging process. It can be seen that the proposed algorithm performs the best, followed by the ANR and LLE algorithms. The bicubic algorithm always gives the highest errors.

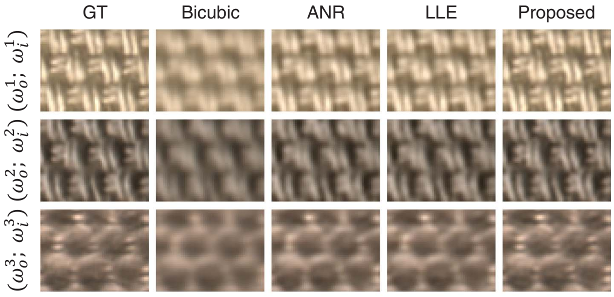
GT
Bicubic
ANR
LLE
Proposed
$(\omega_o^1;\ \omega_i^1)$
$(\omega_o^2;\ \omega_i^2)$
$(\omega_o^3;\ \omega_i^3)$

Fig. 7. GT and reconstructed HR images on material *fabric*02 under different viewing and lighting directions. Row 1: $(\omega_o^1;\omega_i^1) = (0°, 0°; 37.5°, 240°)$. Row 2: $(\omega_o^2;\omega_i^2) = (45°, 120°; 60°, 195°)$. Row 3: $(\omega_o^3;\omega_i^3) = (75°, 240°; 52.5°, 120°)$. The images are adjusted for better visualization.

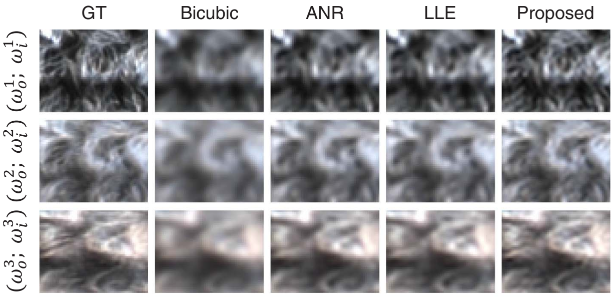
GT
Bicubic
ANR
LLE
Proposed
$(\omega_o^1;\ \omega_i^1)$
$(\omega_o^2;\ \omega_i^2)$
$(\omega_o^3;\ \omega_i^3)$

Fig. 8. GT and reconstructed HR images on material *carpet*04 under different viewing and lighting directions. Row 1: $(\omega_o^1;\omega_i^1) = (0°, 0°; 52.5°, 260°)$. Row 2: $(\omega_o^2;\omega_i^2) = (45°, 120°; 30°, 60°)$. Row 3: $(\omega_o^3;\omega_i^3) = (75°, 240°; 60°, 270°)$. The images are adjusted for better visualization.

To demonstrate the visual quality, the SR results of the materials *fabric02* and *carpet04* under three different combinations of viewing and lighting directions are presented in Figs. 7 and 8, respectively. It is observed that the images produced by the bicubic algorithm are quite blurry. The ANR and LLE algorithms produce clearer edges but fail to recover fine texture details. The reconstructed images of the proposed algorithm contain both sharper edges and finer details and are very close to the ground truths.

Table 1. mRMSE (×10^{-3}), mPSNR, and mRE of Seven Semantic Categories Produced by the Four SR Algorithms

	Bicubic			ANR			LLE			Proposed		
	mRMSE	mPSNR	mRE (%)	mRMSE	mPSNR	mRE (%)	mRMSE	mPSNR	mRE (%)	mRMSE	mPSNR	mRE (%)
Carpet	10.22	32.30	9.85	6.07	35.71	5.97	6.21	35.48	6.12	5.03	37.47	4.88
Fabric	14.40	30.45	10.50	8.11	34.38	6.45	8.52	33.64	7.03	5.74	37.84	4.44
Felt	7.32	32.61	8.24	3.43	35.62	3.92	3.92	34.40	4.63	2.00	40.71	2.19
Leather	20.55	29.70	15.64	12.14	32.61	10.25	12.41	32.14	10.78	9.76	35.57	7.62
Stone	12.78	31.54	8.46	7.66	34.72	6.03	7.98	34.02	6.68	6.15	38.34	4.04
Wallpaper	9.27	29.84	8.85	5.60	34.61	6.11	5.92	33.94	6.97	4.25	37.66	4.32
Wood	6.84	31.60	5.37	4.77	34.06	3.76	5.22	33.45	4.25	3.28	38.04	2.45

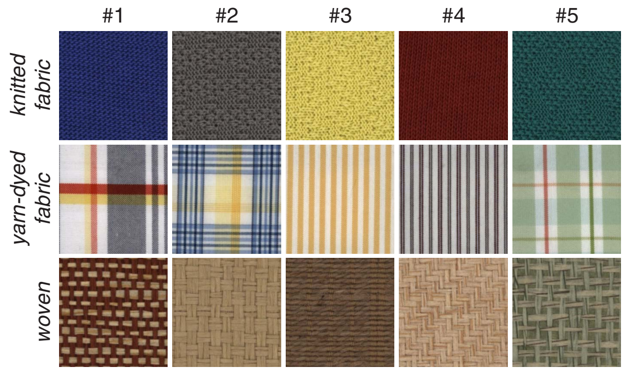

Fig. 9. Example images of the 15 real materials used in the experiment. Row 1: *knitted fabric*. Row 2: *yarn-dyed fabric*. Row 3: *woven*.

B. Real Data Results

Figure 9 shows the example BTF images of 15 materials captured by our BTF imaging system [9]. These materials belong to three categories (*knitted fabric*, *yarn-dyed fabric*, and *woven*) and are of various color patterns. For each material, the full LR BTFs are measured under 96 viewing directions and 77 lighting directions. The resolution of the rectified images is 256 × 256 pixels.

For each material, one HR reference image is taken under the viewing direction $\omega_o = (0°, 0°)$ and the lighting direction $\omega_i = (40°, 120°)$ using a single lens reflex (SLR) camera. The rectified HR images have a spatial resolution of 512 × 512 pixels. Because of the different color characteristics of the imaging devices, the color appearances of the LR and HR images are inconsistent. In the experiment, color consistency is achieved by color calibration via two-order polynomial fitting [40]. As in the synthetic data case, the 15 HR images, as well as the corresponding LR images, are used to estimate the blur kernel for the proposed algorithms and to serve as the training database for the ANR and LLE algorithms.

The SR results of *knitted fabric* #5, *yarn-dyed fabric* #1, and *woven* #4 under the identical viewing and lighting condition to the HR images are shown in Fig. 10. As expected, the proposed algorithm produces the best reconstruction, while the bicubic algorithm produces the worst.

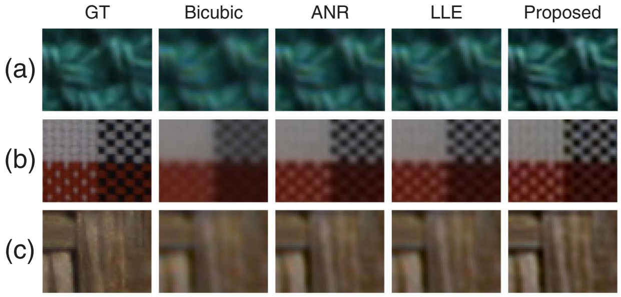

Fig. 10. GT and reconstructed HR images under the viewing direction $\omega_o = (0°, 0°)$ and the lighting direction $\omega_i = (40°, 120°)$. This direction configuration is identical to the HR image acquisition condition. (a) Material *knitted fabric* #5. (b) Material *yarn-dyed fabric* #1. (c) Material *woven* #4.

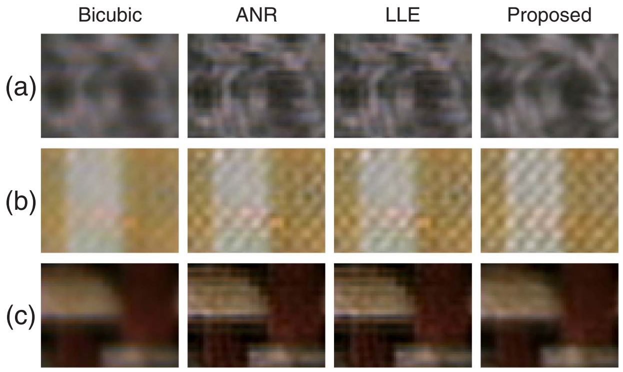

Fig. 11. SR results of the four SR algorithms under the viewing direction $\omega_o = (30°, 0°)$ and the lighting direction $\omega_i = (40°, 240°)$. (a) Material *knitted fabric* #2. (b) Material *yarn-dyed fabric* #3. (c) Material *woven* #1.

To thoroughly evaluate the SR capability, we compare the proposed algorithm with the other algorithms under the viewing and lighting directions different from the HR image acquisition. In this circumstance, the ground truths are not available. Figure 11 shows the SR results of the materials *knitted fabric* #2, *yarn-dyed fabric* #3, and *woven* #1 under the viewing direction $\omega_o = (30°, 0°)$ and the lighting direction $\omega_i = (40°, 240°)$. The images reconstructed by the ANR and LLE algorithms are sharper in appearance than those by the bicubic algorithm. However, high-frequency artifacts are introduced by the training samples. In comparison, in the HR images reconstructed by the proposed algorithm, textures are well recovered, and artifacts are rarely observed.

C. Computational Efficiency

We compare the computational efficiency of the proposed algorithm with those of the bicubic, ANR, LLE, and SIR algorithms. All algorithms are implemented using MATLAB on a personal computer with an Intel(R) Core(TM) i5-4590 CPU at 3.30 GHz and 16 GB RAM. Table 2 shows the average runtime of the five algorithms for one material from the UBO2014 dataset. It is observed that the proposed algorithm runs 3 × faster than ANR, 29 × faster than LLE, and 59 × faster than SIR. The bicubic algorithm runs faster than the proposed algorithm owing to its simple computation, but as it is known, its reconstruction is not satisfactory.

D. Scene Rendering

A conventional application of BTF data is the rendering of three-dimensional models. For the proposed algorithm, the SR results can be used directly as they have already been decomposed via SVD. For the bicubic, ANR, and LLE

Table 2. Computation Time (in Hours) of Different SR Algorithms for a Single Material

Algorithm	Bicubic	ANR	LLE	SIR, Eq. (2)	Proposed
Time	0.76	7.60	48.14	95.00	1.58
Speedup	—	3 ×	29 ×	59 ×	—

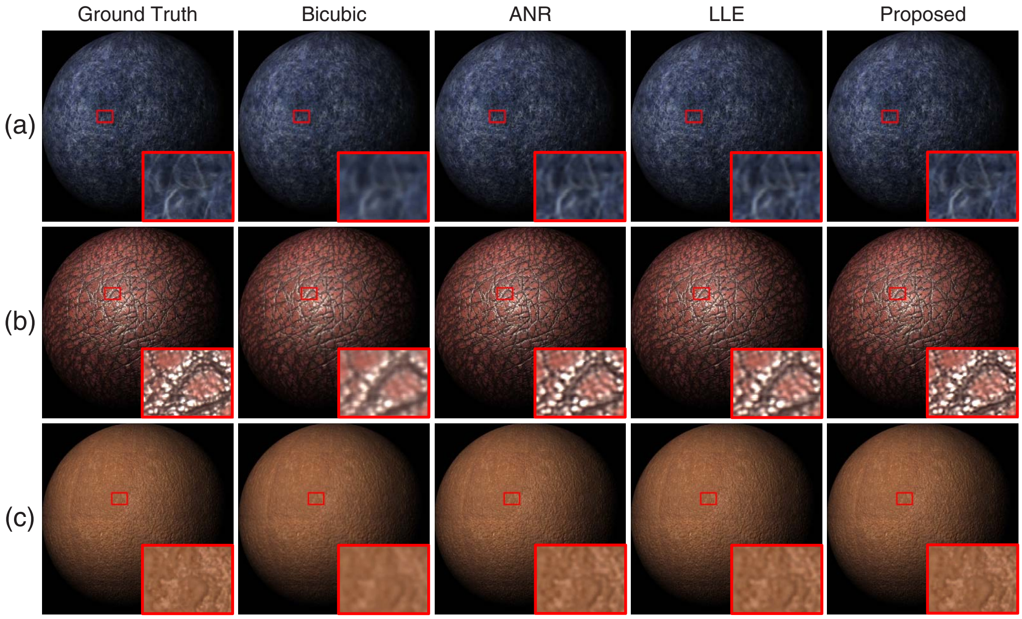

Fig. 12. Three-dimensional balls rendered using the original HR BTFs (first column) and the HR BTFs reconstructed by different SR algorithms (second column to fifth column) under a single point-light illumination. (a) Material *felt*02. (b) Material *leather*08. (c) Material *wallpaper*08. The contents in the red boxes are magnified for better comparison.

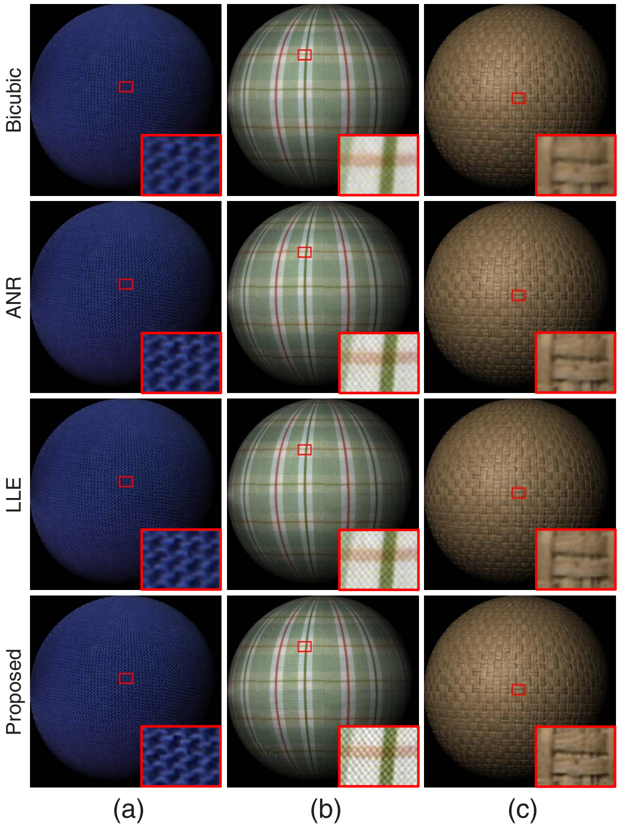

Fig. 13. Three-dimensional balls rendered using the HR BTFs reconstructed by different SR algorithms under a single point-light illumination. (a) Material *knitted fabric* #1. (b) Material *yarn-dyed fabric* #5. (c) Material *woven* #2. The contents in the red boxes are magnified for better comparison.

algorithms, however, the SR results need further compression due to their huge capacity.

Figure 12 illustrates the scenes rendered using the UBO2014 materials under a single point-light illumination. As observed with the proposed SR algorithm, the rendering scenes can keep both large-scale and fine-scale textures, and the appearances are quite close to the ground truths. In comparison, the rendering scenes corresponding to the other SR algorithms do not achieve such appearance fidelity. Figure 13 further shows the rendering results using our materials. As expected, the rendered scenes based on the proposed algorithm are satisfactory.

5. CONCLUSIONS

This paper proposes a fast BTF image SR algorithm using SVD. The size of BTF data can be efficiently reduced by decomposing the raw data into intrinsic textures and eigen-ABRDFs. By exploring the relationship between the LR and HR intrinsic textures, SR is conducted on the intrinsic textures instead of raw images. Thanks to the employment of SVD, the proposed algorithm is computationally efficient and robust to noise corruption. Experimental results indicate that the proposed algorithm outperforms the state-of-the-art algorithms in terms of both accuracy and computational efficiency.

The limitation of the proposed algorithm is that it only uses the simple ℓ^2 norm constraint as the prior regularization. The exploration of texture characteristics and its adoption as prior regularization in the SR optimization framework will be the subject of our future work.

Funding. National Natural Science Foundation of China (NSFC) (61371160); Hong Kong Research Institute of

Textiles and Apparel (ITP/048/13TP); Key Project of Zhejiang Provincial Major Program of Science and Technology.

REFERENCES

1. F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis, *Geometrical Considerations and Nomenclature for Reflectance* (US Department of Commerce, 1977), Vol. **160**.
2. R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," ACM Trans. Graph. **1**, 7–24 (1982).
3. G. J. Ward, "Measuring and modeling anisotropic reflection," ACM SIGGRAPH Comput. Graph. **26**, 265–272 (1992).
4. X. D. He, K. E. Torrance, F. X. Sillion, and D. P. Greenberg, "A comprehensive physical model for light reflection," ACM SIGGRAPH Comput. Graph. **25**, 175–186 (1991).
5. K. J. Dana, B. Van Ginneken, S. K. Nayar, and J. J. Koenderink, "Reflectance and texture of real-world surfaces," ACM Trans. Graph. **18**, 1–34 (1999).
6. G. Müller, J. Meseth, and R. Klein, "Compression and real-time rendering of measured BTFs using local PCA," in *8th International Fall Workshop on Vision, Modeling and Visualisation* (2003), p. 271.
7. Y.-T. Tsai and Z.-C. Shih, "K-clustered tensor approximation: a sparse multilinear model for real-time rendering," ACM Trans. Graph. **31**, 1–17 (2012).
8. M. Sattler, R. Sarlette, and R. Klein, "Efficient and realistic visualization of cloth," in *Rendering Techniques* (2003), pp. 167–178.
9. W. Dong, H.-L. Shen, X. Du, S.-J. Shao, and J. H. Xin, "Spectral bidirectional texture function reconstruction by fusing RGB and spectral images," Appl. Opt. **55**, 10400–10408 (2016).
10. G. Müller, J. Meseth, M. Sattler, R. Sarlette, and R. Klein, "Acquisition, synthesis, and rendering of bidirectional texture functions," Comput. Graph. Forum **24**, 83–109 (2005).
11. J. Y. Han and K. Perlin, "Measuring bidirectional texture reflectance with a kaleidoscope," ACM Trans. Graph. **22**, 741–748 (2003).
12. T.-T. Wong, P.-A. Heng, S.-H. Or, and W.-Y. Ng, "Image-based rendering with controllable illumination," in *Rendering Techniques* (Springer, 1997), pp. 13–22.
13. M. L. Koudelka, S. Magda, P. N. Belhumeur, and D. J. Kriegman, "Acquisition, compression, and synthesis of bidirectional texture functions," in *3rd International Workshop on Texture Analysis and Synthesis (Texture)* (2003), pp. 59–64.
14. G. Müller, G. H. Bendels, and R. Klein, "Rapid synchronous acquisition of geometry and appearance of cultural heritage artefacts," in *Proceedings of the 6th International conference on Virtual Reality, Archaeology and Intelligent Cultural Heritage* (Eurographics Association, 2005), pp. 13–20.
15. M. Weinmann, J. Gall, and R. Klein, "Material classification based on training data synthesized using a BTF database," in *European Conference on Computer Vision* (2014), pp. 156–171.
16. P.-H. Suen and G. Healey, "The analysis and recognition of real-world textures in three dimensions," IEEE Trans. Pattern Anal. Mach. Intell. **22**, 491–503 (2000).
17. H. C. Steinhausen, D. Den Brok, M. B. Hullin, and R. Klein, "Acquiring bidirectional texture functions for large-scale material samples," J. WSCG **22**, 73–82 (2014).
18. D. Den Brok, H. C. Steinhausen, M. B. Hullin, and R. Klein, "Patch-based sparse reconstruction of material BTFs," J. WSCG **22**, 83–90 (2014).
19. S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: a technical overview," IEEE Signal Process. Mag. **20**(3), 21–36 (2003).
20. P. Thévenaz, T. Blu, and M. Unser, "Image interpolation and resampling," in *Handbook of Medical Imaging, Processing and Analysis* (2000), pp. 393–420.
21. X. Li and M. T. Orchard, "New edge-directed interpolation," IEEE Trans. Image Process. **10**, 1521–1527 (2001).
22. X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," IEEE Trans. Image Process. **17**, 887–896 (2008).
23. M. Irani and S. Peleg, "Motion analysis for image enhancement: resolution, occlusion, and transparency," J. Visual Commun. Image Represent. **4**, 324–335 (1993).
24. J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2008), pp. 1–8.
25. X. Yan and J. Shen, "Fast gradient-aware upsampling for cartoon video," in *2010 International Conference on Image Analysis and Signal Processing* (IEEE, 2010), pp. 636–639.
26. H. Chang, D. Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *IEEE Conference on Computer Vision and Pattern Recognition* (2004), pp. 275–282.
27. J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," IEEE Trans. Image Process. **19**, 2861–2873 (2010).
28. R. Timofte, V. De, and L. V. Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *IEEE International Conference on Computer Vision* (2013), pp. 1920–1927.
29. S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," Science **290**, 2323–2326 (2000).
30. M. Jian, K. M. Lam, and J. Dong, "A novel face-hallucination scheme based on singular value decomposition," Pattern Recogn. **46**, 3091–3102 (2013).
31. N. Joshi, R. Szeliski, and D. Kriegman, "PSF estimation using sharp edge prediction," in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2008), pp. 1–8.
32. D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-Laplacian priors," in *Advances in Neural Information Processing Systems* (2009), pp. 1033–1041.
33. S. Dai, M. Han, W. Xu, Y. Wu, and Y. Gong, "Soft edge smoothness prior for alpha channel super resolution," in *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2007), pp. 1–8.
34. M. K. Ng, P. Weiss, and X. Yuan, "Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods," SIAM J. Sci. Comput. **32**, 2710–2736 (2010).
35. R. Fattal, "Image upsampling via imposed edge statistics," ACM Trans. Graph. **26**, 95 (2007).
36. A. K. Jain, *Fundamentals of Digital Image Processing* (Prentice-Hall, 1989).
37. M. Simões, L. B. Almeida, J. Bioucas-Dias, and J. Chanussot, "A framework for fast image deconvolution with incomplete observations," IEEE Trans. Image Process. **25**, 5266–5280 (2016).
38. J. Baglama and L. Reichel, "Augmented implicitly restarted Lanczos bidiagonalization methods," SIAM J. Sci. Comput. **27**, 19–42 (2005).
39. R. Ruiters, M. Rump, and R. Klein, "Parallelized matrix factorization for fast BTF compression," in *Eurographics Symposium on Parallel Graphics and Visualization* (2009), pp. 25–32.
40. G. Hong, M. R. Luo, and P. A. Rhodes, "A study of digital camera colorimetric characterization based on polynomial modeling," Color Res. Appl. **26**, 76–84 (2001).