Per-Image Super-Resolution for Material BTFs

Dennis den Brok, Sebastian Merzbach, Michael Weinmann, and Reinhard Klein

Abstract—Image-based appearance measurements are fundamentally limited in spatial resolution by the acquisition hardware. Due to the ever-increasing resolution of displaying hardware, high-resolution representations of digital material appearance are desirable for authentic renderings. In the present paper, we demonstrate that high-resolution bidirectional texture functions (BTFs) for materials can be obtained from low-resolution measurements using single-image convolutional neural network (CNN) architectures for image super-resolution. In particular, we show that this approach works for high-dynamic-range data and produces consistent BTFs, even though it operates on an image-by-image basis. Moreover, the CNN can be trained on down-sampled measured data, therefore no high-resolution ground-truth data, which would be difficult to obtain, is necessary. We train and test our method’s performance on a large-scale BTF database and evaluate against the current state-of-the-art in BTF super-resolution, finding superior performance.

Index Terms—Super-Resolution, Appearance Capture, Deep Learning

1 INTRODUCTION

Super-resolution, i.e. artificially increasing the sampling rate of a given measuring system, is a problem faced in disciplines as diverse as astronomy and biology. In computer vision and graphics, it typically deals with increasing the image resolution of imaging systems of all kinds, from smartphone cameras to light-field cameras, for static data as well as dynamic. There is an abundance of practical solutions for many of these applications. In contrast, imaging systems of importance to rendering applications, such as acquisition devices for material appearance, have seen a lot less attention. But demand has increased driven by rapid developments in display technology, and while image-space super-resolution for Lambertian reflectance has been studied and understood for a long time, more comprehensive representations like spatially-varying bidirectional reflectance distribution functions (SVBRDFs) and bidirectional texture functions (BTFs) have started to draw interest only recently. This case is a lot more difficult to deal with, as even though typical measurements consist of hundreds or thousands of images with what seems like exploitable per-texel redundancies, the fact that the material’s reflection at each texel can often only be described by complex BRDFs renders traditional multi-view super-resolution approaches inapplicable. Consequently, the current state-of-the-art method for BTF super-resolution by Dong et al. [1] is based on classical single-image super-resolution, albeit applied to eigentextures for a significant performance increase.

In the present paper, we take things a step further and introduce what we believe to be the first super-resolution algorithm for non-Lambertian spatially-varying material appearance based on modern deep learning techniques for single-image super-resolution. Our goal in this work has not been to find the existing method to provide the best possible results for our use cases, which would have involved training and comparing an abundance of network architectures. Instead, we try to answer a number of questions related to the problem at hand:

- Are typical single-image super-resolution networks suitable for the high dynamic range (HDR) radiance maps our setup produces?
- Do we need ground-truth data in the target resolution, which may be difficult to obtain, or is the problem scale-invariant to some extent?
- Does the image-by-image approach lead to visible artefacts in the resulting material representations?

To do so, we construct a simple convolutional neural network (CNN) by extending one of the pioneering works with current best pactices. We train and test our network on a database of down-sampled HDR radiance maps from our material measurements. From the resulting high-resolution test images, we produce high-resolution BTFs for which we provide both a ground-truth comparison and an evaluation against the state-of-the-art. In the latter case, we are able to demonstrate significantly improved reconstruction quality.

2 RELATED WORK

There is a wealth of literature on the general subject of super-resolution. We briefly review some of the publications more relevant to our work and refer to surveys for further reference.

Single-Image Super-Resolution

Approaches for restoring high-resolution representations from a single low-resolution image can be categorized into interpolation-based, reconstruction-based and learning-based methods. Interpolation-based techniques such as linear interpolation, bicubic interpolation or Lanczos resampling [2] offer the possibility of a fast and simple upsampling at the cost of limited accuracy due to remaining blurring artifacts. In contrast, the use of additional prior knowledge allows reconstruction-based approaches (e.g. [3],

* All authors are with the Institute of Computer Science II, University of Bonn, Germany.
E-mail: dnbrok@cs.uni-bonn.de

©2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes,creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
BTF texels are compressed to latent codes by the encoder and the decoder is used to reproduce material appearance under guidance by specified light and view direction.

3 Background

Our goal is an algorithm capable of producing BTFs with high texture resolution from such with low texture resolution. There are a number of natural choices for what the algorithm’s input should be, precisely. In order to be able to justify our particular choice, we first briefly introduce the notion of BTFs and explain how they can be represented digitally. Subsequently, we give an overview of the process from physical material sample to digitally represented BTF, and finally, as their method builds upon some of the techniques explained in this section, we briefly cover the prior art by Dong et al. [1]

3.1 Bidirectional Texture Functions

Consider the rendering equation

$$L_o(x, \omega_i, \omega_o) = \int_{\Omega_i} f_r(x, \omega_i, \omega_o) L_i(x, \omega_i, \omega_o) d\omega_i \quad (1)$$

Most commonly, $f_r$ is assumed to be a spatially-varying BRDF, i.e. it adheres to conservation of energy and Helmholtz reciprocity, and $x$ is defined with respect to the given surface geometry. For BTFs, none of these assumptions are required to hold, which allows them to account for a number of effects not local to a texel, such as parallax and meso-scale inter-reflections. The downside is that they are notoriously difficult to model in a meaningful way, which is why most BTF representations are discrete and image-based. One of the main applications is arguably rendering, particularly in real-time. Due to the way BTFs are typically evaluated at given coordinates $(x, \omega_i, \omega_o)$, it is convenient to assume the set of sampled pairs $(\omega_i, \omega_o)$ to be a Cartesian product $L \times V$ of sets of light and view directions, respectively. In that case, a BTF can be represented as a tensor $B \in \mathbb{R}^{n_l \times n_o \times w \times h}$, where $c$ is the number of color channels, $n_l$, $n_o$ the number of sampled light and view directions, and $w, h$ the size of the bidirectional textures, or more commonly as a matrix $B \in \mathbb{R}^{n_l \times n_o \times c \times w \times h}$. This is the desired final form for our algorithm’s output.

3.2 BTF Compression

In most practical scenarios, $L$ and $V$ need to be sufficiently dense as to avoid interpolation artifacts when the BTF is used in rendering. A typical order of magnitude is $|L \times V| \propto 10^5$, which leads to impractical amounts of data. It is known, however, that BTFs in the matrix representation outlined in Section 3.1 admit a low-rank factorization

$$f(B) \approx U \cdot V, \quad (2)$$

where $U \in \mathbb{R}^{n_l \times n_k \times k}$, $V \in \mathbb{R}^{k \times c \times w \times h}$ are matrices of rank $k \ll n_l \cdot n_o$ [36], which are usually obtained by singular value decomposition (SVD), and $f$ an optional function modifying the effective metric used when computing the factorization, commonly used to deal with the wide dynamic range of BTF data. Common values for $k$ are in the range 100–500.
Dong et al. [1] exploit this factorization to achieve fast and noise-resistant BTF super-resolution by operating per-eigentexture, i.e. per row of the matrix $V$, only. One caveat is that the algorithm relies on the eigentextures to be image-like, i.e. $f = \text{id}$, and therefore likely works best for materials with low-dynamic-range BTF.

3.3 Measuring BTFs

Clearly, it is impractical to sample a material’s BTF at precisely the desired coordinates, as the directions of incoming and outgoing light vary across the material surface. Virtually every available acquisition device therefore produces a set of images of the material as seen and lit from different angles as an intermediate step. In principle, the data acquired this way already constitutes a sampling of the material’s BTF, but for the purpose of storage and rendering efficiency, it is convenient to represent the data as a matrix of the shape outlined in Section 3.1. This involves a number of transformations which alter the image data significantly:

First, in order to obtain proper textures, a homographical transformation is applied image-by-image which warps the region of interest on the material surface from a polygonal shape in image space to a rectangular shape of size $w \times h$. As a result, textures obtained from images taken by cameras seeing the material surface from a grazing angle contain information with much lower frequencies than those from images as seen from a frontal parallel view.

Subsequently, the textures can be arranged in a matrix representation of the described shape, but the columns, which correspond to per- texel BRDFs, do not yet correspond to the desired sampling $L \times V$. The BRDFs are thus re-sampled accordingly, typically by means of multilinear interpolation, which further smoothes the textures.

3.4 Prior Art

The only algorithm for BTF super-resolution to date has been proposed by Dong et al. [1] Similarly to ours it exploits single-image super-resolution, but applied to eigentextures instead of actual images, which provides both a significant speed-up and improved robustness against noise in the data. Contrary to the present paper, they chose a classical approach based on deconvolution with a smoothness prior:

Starting at the image formation model
\[ y = DPx + n, \]
where $D \in \mathbb{R}^{w_{LR} \times h_{LR} \times w_{HR} \times h_{HR}}$ is a downsampling operator, $P \in \mathbb{R}^{w_{HR} \times h_{HR} \times w_{LR} \times h_{LR}}$ a blur matrix accounting for the camera’s point spread function (PSF), $n \in \mathbb{R}^{w_{LR} \times h_{LR}}$ a noise vector, and $x \in \mathbb{R}^{w_{HR} \times h_{HR}}$ and $y \in \mathbb{R}^{w_{LR} \times h_{LR}}$ high- and low-resolution images, respectively, with $w_{HR} > w_{LR}$ and $h_{HR} > h_{LR}$, one arrives at the single-image super-resolution optimization problem

\[ \hat{x} = \arg \min_{x} ||DPx - y||_2^2 + \alpha \phi(x), \]

where $\phi: \mathbb{R}^{w_{HR} \times h_{HR}} \rightarrow \mathbb{R}$ is some regularization function with weight $\alpha$. Dong et al. [1] chose $\phi(x) = ||Lx||_2^2$, where $\text{L}$ is the 2D Laplacian operator. The resulting optimization problem can be solved very efficiently using the regularized normal equations. However, it is known to be sensitive to the presence of noise. Moreover, it has to be applied to each of the BTF’s bidirectional textures individually. Dong et al. [1] thus enhance both efficiency and robustness of their optimization problem by operating on the individual eigentextures $V$ of the factorized low-resolution BTF $B \approx UV$.

Note that $U$ is computed on the low-resolution measured data, which may limit the achievable amount of detail to some extent. Note also that the measured PSF does not directly apply to the bidirectional textures and, by extension, to the eigentextures of the BTF, because they have undergone warping to account for perspective transformation; see Section 3.3. We are not aware of whether Dong et al. took the latter into consideration, but it did not seem to noticeably influence the quality of the results in our experiments.

Arguably, the speed improvement, though significant, is not the algorithm’s primary feature, as the deconvolution is very fast even on an full BTF measurement; nevertheless we shall provide a suggestion how to reach comparable performance with our image-by-image approach.

4 Proposed Method

It should be clear from Section 3 that there are various natural ways to attack the problem of image-by-image super-resolution for BTFs. In the following we shall give a rationale for our particular design choices and a detailed description of the chosen network architecture.

4.1 Design Choices

First of all – why did we choose an image-by-image approach?

Recall that measured material BTFs are of the shape $B \in \mathbb{R}^{n_l \times n_v \times w \times h}$ or similar, with $n_l \cdot n_v \propto 10^5$ (cf. Section 3.1). This brings with it a number of problems. No super-resolution algorithm we are aware of is capable of dealing with anywhere near the necessary amount of $n_l \cdot n_v$ channels. Moreover, the BTF database at our disposal consists of only 24 BTFs with small $w \times h = 128 \times 128$, which is likely insufficient for learning proper parameters even when dividing the available BTFs up into patches. The number of channels can be decreased significantly by compressing the BTFs and operating on the eigentextures $V$, similar to Dong et al. However, the high-resolution eigentextures thus obtained are parameters for a model $U$ of eigen-BRDFs computed on the low-resolution BTF (as the high-resolution BTF is not available), which may limit the effectiveness of the upsampling network. Nevertheless, we conducted experiments with this approach and indeed found it to perform worse than the state-of-the-art, likely due to the still high number of channels (typically 128–256) or the low number of exemplars to learn from.

Secondly, now that we motivated an image-by-image approach, the choice remains between the types of image data that occur in the post-processing process outlined in Section 3.3: eigentextures, bidirectional textures from the re-sampled BTF, warped regions of interest, or simply the HDR radiance maps before any further post-processing.

We see no obvious theoretical a priori reason why any should perform significantly better than the others. By experimentation we found that using HDR radiance maps in
fact works best by quite a margin. We assume that the filters learned from this kind of images are the ones enabling the highest-frequency reconstructions while not being too general: for images corresponding to warped regions of interest, the majority of the images are low-frequency because of the large amount of warping for low-angle views, and bidirectional textures have undergone even further interpolations. Conversely, the range of frequencies is extremely wide in the case of eigntextures, with the eigntexture corresponding to the largest eigenvalue being similar to a mean texture and the ones corresponding to very small eigenvalues being almost indistinguishable from noise. Please note, however, that this is only a hypothesis which we did not verify experimentally.

4.2 Network Architecture

The proposed network is a convolutional neural network (CNN) with skip connections, inspired by the work by Shi et al. [20] We follow their suggestions closely, but modify them in a few places according to current good practices. Their network consists of a $L$ layers, the first $L-1$ of which are consecutive 2D convolutional layers, followed by a single sub-pixel convolution layer:

$$f^{(l)}(I_{LR}) = \phi(W_l \ast I_{LR} + b_l),$$

$$f^{(L)}(I_{LR}) = \mathcal{P}\mathcal{S}(W_L \ast f^{(L-1)}(I_{LR}) + b_L),$$

where $I_{LR}$ is the low-resolution input image, $W_l$ and $b_l$, 1 ≤ $l$ ≤ $L$, are the weights and biases of the corresponding layers, respectively, $\phi$ is the activation function (the same for all except the last layer), and $\mathcal{P}\mathcal{S}$ is a periodic shuffling operator. In this scenario, $b_l \in \mathbb{R}^{n_l}$ is a vector of length $n_l$, and $W_l \in \mathbb{R}^{n_l \times n_l \times k_l \times k_l}$, where $n_l$ is the number of features at layer $l$, $n_1 = 1$, and $k_l$ the corresponding kernel size.

The original network did not perform very well in our scenario in terms of upsampling quality, which is why we introduced some modifications: Following Tong et al. [37] we added a skip connection:

$$f^{(L-1)}(I_{LR}) = \phi(W_{L-1} \ast \left( f^{(1)}(I_{LR}) \right) + b_{L-1}),$$

which allows for combination of low- and high-level features, resulting in enhanced sharpness. See Figure 1 for a schematic of the resulting architecture. Moreover, we added batch normalization to deal with internal covariate shift, allowing for higher learning rates. [38]

We chose this particular architecture for several reasons: it is easy to understand, fast and efficient, in particular due to fact that the upsampling happens at the last layer only, and it does not make any assumptions about the desired output. The latter may seem like a disadvantage; however, in our material BTF scenario, we are interested more in faithful reconstruction in the sense of visual similarity rather than, e.g., an extremely sharp, plausible result which only remotely resembles the material sample. See Figure 4.2 for a reconstruction obtained using a recent, pre-trained state-of-the-art network [39] which may be sharper than what our network produces, but does not look like the original material sample. In particular, if this algorithm were applied on a per-image basis to an entire BTF, it would probably lead to visible artefacts.

4.3 Performance Considerations

Contrary to the work by Dong et al., our network does not seem to work well with eigntextures as input images, as hinted at in Section 4.1. However, once the network is trained, it is still not necessary to upsample all measured images. According to den Brok et al. [40], BRDF reconstruction from sparse measurements as proposed by Nielsen et al. [41] also applies to spatially-varying material appearance via

$$V = (U^T S^T S U + \Gamma^T \Gamma)^{-1} U^T (SB),$$

where $S$ is a subsampling matrix which essentially selects rows, i.e. bidirectional textures, from the full BTF $B$, and $\Gamma$ is a diagonal matrix of weights giving priority to the eigen-BRDFs corresponding to the largest singular values. By leveraging this method we can therefore reduce the amount of images that need to be upsampled significantly, if desired.

Note there are two natural choices for $U$ here, which represent a trade-off between speed and reconstruction quality: we may compute a basis $U_{LR}$ on the low-resolution input BTF, where, contrary to Dong et al., we have the advantage that we can apply non-linear transformations like the logarithm for dynamic-range reduction and thereby obtain more adequate and concise bases (cf. [42]). That way, we only have to upsample approximately the same amount of images as Dong et al., albeit putting an upper limit on the sharpness of the reconstruction because $U_{LR}$ was computed on low-resolution data. The alternative is to first produce a database of high-resolution BTFs using our network, and to use that database to obtain a basis $U_{HR}$ potentially allowing for more high-frequency reconstructions. As $\text{rank}(U_{HR}) > \text{rank}(U_{LR})$, more sample images need to be upsampled prior to sparse reconstruction, which reduces reconstruction speed.

Either way, this approach comes with a penalty on reconstruction quality, which has been studied before. [43] In our experiments, we therefore do not use this performance enhancement and reconstruct full BTFs instead.

5 Experimental Results

5.1 Experimental Setup

We evaluate the proposed algorithm on a database of 24 material samples, 12 of which are leathers and fabrics, respectively (cf. Table 1).

5.1.1 Acquisition Setup

Our acquisition device consists of 11 industrial-grade cameras and 198 LED light sources, with the material sample placed on a turntable which is rotated evenly to 12 different positions to achieve a denser sampling of the hemisphere of viewing directions. The cameras deliver 2048 × 2048 LDR images in the Bayer raw domain, which we combine to HDR images using the classical algorithm by Robertson et al. [44] and subsequently correct radiometrically. In order
The proposed network architecture is an efficient convolutional neural network (CNN) with a skip connection and a sub-pixel convolution layer, extending work by Shi et al. for more detailed results on our data.

Fig. 1

Fig. 2. Left: ground-truth. Right: reconstruction using a state-of-the-art single-image super-resolution network, created by tone-mapping the underlying HDR radiance map. Note that while the result is visually pleasing, it does not resemble the ground-truth very closely. Even if this is considered acceptable, it is unlikely that a BTF constructed this way would behave consistently under varying viewing conditions; i.e., it may exhibit flickering or similar artifacts.

Table 1

TABLE 1

Experiments are performed on a workstation with an Intel Core i7-5820K CPU at 3.30 GHz with 32 GB RAM and an NVIDIA GTX980 GPU with 4 GB of RAM, running Fedora Linux 30 and a scientific computing stack based on the Anaconda Python environment. We use PyTorch with CUDA support for our algorithm’s learning step.
5.2 Implementation Details

5.2.1 Prior Art

We found that the algorithm by Dong et al. crucially depends on the presence of a non-trivial PSF. While no optical system used for BTF acquisition will not meet this criterion, the cameras and lenses used for obtaining our database exhibit extremely little point spread. This is not really a problem, as in practice one can simply take the region of interest slightly out of focus. As we have no such measurements at our disposal, we just applied Gaussian blur with the parameters provided by Dong et al. [1], i.e. a $9 \times 9$ Gaussian blur kernel with $\sigma = 1.5$ to the actual ground-truth, prior to creating the down-sampled simulated high-resolution data. Our algorithm, in contrast, is applied to the original un-blurred data. For dimensionality reduction, we used rank $k = 300$ as proposed by Dong et al., which is a lot greater than what we normally use for compression ($k = 128$); we therefore expect the impact on reconstruction quality to be negligible.

5.2.2 Network Parameters

We aim for an up-sampling factor of $r = 2$. For training, we draw $17 \times 17$ non-overlapping patches from the simulated low-resolution data and the corresponding $34 \times 34$ patches from the simulated high-resolution data. In order to be able to fit the entire training data into RAM, we sub-sampled the BTFs by using approximately 5% of their bidirectional textures, which amounts to 118 images per material, or 2124 images altogether. We consider the sub-sampling justified by the large degree of redundancy in the measurements across different lighting/viewing angles, as experimentally verified by Filip et al. [45].

We implement the proposed network architecture using $L = 5$ layers, the first four of which are the convolutional layers with kernel sizes $k_1 = 5, k_2 = 3, k_3 = 3$ and $k_4 = 3$, respectively. The number of features at the respective layers are $n_1 = 64, n_2 = 32, n_3 = 32$ and $n_4 = r^2 = 4$, closely following Shi et al.

We use the Adam optimizer [46] with its default parameters and an initial learning rate of $10^{-2}$. For the activation functions, we choose $\phi = \text{ReLU}$. Again following Shi et al., we employ the pixel-wise MSE as loss function.

5.3 Ground-Truth Comparison

For the ground-truth comparison, we use the original measured data. The proposed algorithm and the algorithm by Dong et al. are applied to the measurements down-sampled by a factor of 0.5; recall that the mapping for our method was learned on data down-sampled twice.

The learning step converged after less than 100 training epochs, which took only about an hour on the hardware used for training.

Run-time per BTF was approximately 15 minutes for Dong et al., where most of the time was spent on the resampling step, the computation of the SVD, and the actual algorithm.

Our algorithm has a run-time of approximately 32 minutes per BTF, 30 minutes of which are spent on the resampling step, which is a lot slower because it needs to be performed on the up-sampled data in our case, its run-time scaling linearly with the (linear) resolution. Using the approach described in Section 4.3, overall run-time can likely be reduced by as much as 90% depending on the choice of $U$ [40], at the cost of the additional quality penalties imposed by sparse reconstruction. Given that run-time is not excessive as is, we did not investigate how much this amounts to in practice.

For comparison, we provide a number of error measures. Following Dong et al., we provide the mean RMSE

$$m\text{RMSE} = \frac{1}{n_l \cdot n_v} \sum_{i=0}^{n_l \cdot n_v} \left( \frac{1}{\sqrt{\text{w}_{\text{BTF}} \cdot h_{\text{BTF}}}} \| B_{i,:} - \tilde{B}_{i,:} \|_2 \right),$$

the mean PSNR

$$m\text{PSNR} = \frac{1}{n_l \cdot n_v} \sum_{i=0}^{n_l \cdot n_v} \sum_{v=0}^{20 \log_{10}} \frac{\max \{ B_{i,:} \}}{\sqrt{\text{w}_{\text{BTF}} \cdot h_{\text{BTF}}}} \| B_{i,:} - \tilde{B}_{i,:} \|_2,$$

and the mean relative error

$$m\text{RE} = \frac{1}{n_l \cdot n_v} \sum_{i=0}^{n_l \cdot n_v} \left( \frac{\| B_{i,:} - \tilde{B}_{i,:} \|_2}{\| B_{i,:} \|_2} \cdot 100\% \right),$$

the mean being taken over the bidirectional textures. Cf. Table 2 for the numerical errors we obtained. Numerically, our algorithm clearly outperforms the one by Dong et al., surpassing its performance in all of the metrics. Dong et al.’s results are fundamentally limited by both the compressed BTF representation they require and the smoothness prior which counter-acts the super-resolution effect to some extent.

In Table 3, we present bidirectional textures extracted from the ground-truth and the various reconstructions. Again, we obtain significantly improved results when compared to Dong et al., which lacks sharpness and exhibits dampened highlights, probably due to the compression.

Finally, cf. Table 4 for renderings of the ground-truth and the reconstructions in a standard scene for displaying BRDFs using environmental lighting.

5.4 Consistency

The renderings also provide indication for whether the BTFs produced by our algorithm are consistent in the sense that they do not exhibit noticeable artefacts caused by the per-image nature of our approach. Because of the environmental lighting, large parts of the BTF actually contribute to the overall appearance. In a scene like this, artefacts may thus become quite apparent. For the tested materials, this does not appear to be the case for either algorithm.

For further evidence, in Table 5 we provide images of ABRDFs for a single texel, which do not exhibit noticeable artifacts. In the supplemental material to this paper, we provide animated renderings of our results along with difference images, which do not exhibit artifacts like flickering, either.

5.5 Scale Invariance

Given the quality of our reconstructions, which were obtained using an up-sampling mapping learned on practically available low-resolution data, we think it is justified to conclude that, at least for an up-sampling factor of
<table>
<thead>
<tr>
<th></th>
<th>leather #1</th>
<th>leather #2</th>
<th>fabric #1</th>
<th>fabric #2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dong</td>
<td>2.41e-3, 29.7 dB, 19.0 %</td>
<td>3.49e-3, 30.5 dB, 10.4 %</td>
<td>6.16e-3, 23.3 dB, 26.6 %</td>
<td>4.14e-3, 30.2 dB, 21.8 %</td>
</tr>
<tr>
<td>Ours</td>
<td>1.41e-3, 33.1 dB, 12.8 %</td>
<td>2.29e-3, 33.8 dB, 6.6 %</td>
<td>2.16e-3, 32.1 dB, 9.7 %</td>
<td>1.80e-3, 37.1 dB, 10.0 %</td>
</tr>
</tbody>
</table>

Numerical reconstruction errors for the selected test materials. From left to right per table cell: mRMSE, mPSNR, mRE. Our algorithm clearly outperforms the state-of-the-art here, which relies on a lossy compressed BTF representation.

\( r = 2 \), our algorithm is sufficiently scale-invariant as to allow for surprisingly detailed reconstructions. For further evaluation, we conducted an additional experiment where we used a model trained to upsample data which has been downsampled only once instead of twice to the ground-truth resolution. We observed an arguably expected slight degradation in purely numerical terms, e.g. the relative error was reduced further by 3.5% on average when compared to our actual method’s model. However, the resulting images and renderings turned out to be practically indistinguishable from our previous results.

### 6 Conclusion

We presented a BTF super-resolution algorithm based on a simple but efficient convolutional neural network architecture. We demonstrated that, when trained on down-sampled real-world measurements, our network is capable of up-sampling high-dynamic-range BTF measurements, which allows for BTFs of much higher resolution than what was measured, outperforming the state-of-the-art in BTF super-resolution. Along the way, we gave what we believe to be first, positive answers to the questions:

- Do conventional deep single-image super-resolution algorithms apply to high dynamic range data?
- Are networks trained on downsampled BTF measurements scale-invariant enough for super-resolution?
- Can we reconstruct a consistent high-resolution BTF from measurements upsampled on an image-by-image basis?

Our algorithm shares the limitations of learning-based single-image super-resolution methods. The network was chosen such that it favors faithfulness of the reconstruction over sharp, but visibly hallucinated results, which limits the achievable degree of reconstructed detail.

We think it worthwhile to investigate whether deep learning methods can be used even more effectively for our scenario. As it is, our network is not specifically tailored to BTF data. It should be possible to exploit the very specific nature of BTF measurements further, e.g. by means of a multi-view approach, or by using novel, more compact BTF representations which would result in less input channels in the network architecture. The former would be particularly useful to overcome the intrinsic limitations of single-image approaches. However, the lack of large, publicly available BTF databases may prove a major obstacle on any of these paths.

Lastly, many modern super-resolution algorithms including ours are trained and/or evaluated on synthetic data generated by down-sampling in a very straight-forward manner. However, the actual image formation model is a lot more involved. Recent work by Zhang et al. demonstrates a noticeable performance improvement when taking this into consideration. [47]. We believe this to be a promising starting point for any future work on image-space super-resolution.

### Acknowledgments

This work was supported by the DFG project KL 1142/11-1 (DFG Research Unit FOR 2535 Anticipating Human Behavior).

We are grateful to Clara Callenberg for help with some of the figures and Julian Iseringhausen for useful discussions.

### References


TABLE 4
Renderings of the (a) ground-truth and BTFs reconstructed using (b) Dong et al. and (c) proposed under environmental lighting.


TABLE 5

ABRDFs extracted from ground-truth and our reconstructions of the four test materials. Note that the reconstructed reflectance distributions closely resemble the ground-truth, exhibiting no apparent artefacts, even though they have been obtained on a pixel-by-pixel basis.


---

**Dennis den Brok** studied Mathematics at the University of Bonn, Germany. After receiving his diploma in 2011, he became a member of the Computer Graphics group at Bonn University. Since 2013, he is working on his PhD, initially within the framework of the graduate school on Digital Material Appearance sponsored by X-Rite. His research interests include problems related to digital material appearance, particularly modeling, acquisition and super-resolution in several dimensions. The latter sparked his recent interest in problems from computational photography.

**Sebastian Merzbach** studied computer science at the University of Bonn, where he received his diploma in 2013. He is currently working on his PhD in computer graphics. His research areas are appearance acquisition, modelling and rendering, with application of deep learning methods. Other projects include spectral material acquisition and rendering, as well as material synthesis.

**Michael Weinmann** studied Electrical Engineering and Information Technology at the University of Karlsruhe, Germany, where he received his degree Dipl.-Ing. in 2009. After joining the Visual Computing Group at the University of Bonn in 2010, he received his PhD in computer science in 2016. His research interests include machine learning, 3D reconstruction, reflectance reconstruction, semantic scene interpretation and visualization where he published respective work on high-ranked conferences including CVPR, ICCV and ECCV as well as reputable journals such as the ISPRS Journal of Photogrammetry and Remote Sensing, ACM Transactions on Graphics, IEEE Transactions on Visualization and Computer Graphics, and Sensors.

**Reinhard Klein** studied Mathematics and Physics at the University of Tübingen, Germany, from where he received his MS in Mathematics (Dipl.-Math.) in 1989 and his PhD in computer science in 1995. In 1999 he received an appointment as lecturer (“Habilitation”) in computer science also from the University of Tübingen, with a thesis in computer graphics. In 1999 he became Associate Professor at the University of Darmstadt, Germany and head of the research group Animation and Image Communication at the Fraunhofer Institute for Computer Graphics. Since 2000 he is professor at the University of Bonn and director of the Institute of Computer Science II.