STAG: Smart Tools and Applications in Graphics (2024)
A. Caputo, V. Garro, U. Castellani, and A. Giachetti (Editors)

Disk-NeuralRTI: Optimized NeuralRTI Relighting through Knowledge Distillation

Tinsae G. Dulecha[1], Leonardo Righetto [1], Ruggero Pintus [2], Enrico Gobbetti[2], and Andrea Giachetti[1]

[1] University of Verona, Italy
[2] CRS4, Italy

Abstract

Relightable images created from Multi-Light Image Collections (MLICs) are among the most employed models for interactive object exploration in cultural heritage (CH). In recent years, neural representations have been shown to produce higher-quality images at similar storage costs to the more classic analytical models such as Polynomial Texture Maps (PTM) or Hemispherical Harmonics (HSH). However, the Neural RTI models proposed in the literature perform the image relighting with decoder networks with a high number of parameters, making decoding slower than for classical methods. Despite recent efforts targeting model reduction and multi-resolution adaptive rendering, exploring high-resolution images, especially on high-pixel-count displays, still requires significant resources and is only achievable through progressive rendering in typical setups. In this work, we show how, by using knowledge distillation from an original (teacher) Neural RTI network, it is possible to create a more efficient RTI decoder (student network). We evaluated the performance of the network compression approach on existing RTI relighting benchmarks, including both synthetic and real datasets, and on novel acquisitions of high-resolution images. Experimental results show that we can keep the student prediction close to the teacher with up to 80% parameter reduction and almost ten times faster rendering when embedded in an online viewer.

CCS Concepts

• ***RTI*** → *Neural RTI, Disk-NeuralRTI;* • ***Relighting*** → *Neural relighting;* • ***RTI Viewer*** → *Web based visualization;*

1. Introduction

Reflectance Transformation Imaging (RTI) is a widely utilized computational photography technique that enables capturing rich surface representations, including geometric details and local reflective behavior of materials. This method involves the acquisition of Multi-Light Image Collections (MLICs), in the form of sets of images from a stationary perspective with different lighting angles. The resulting measurements are then fitted to compact models replicating the measured behavior.

There are different types of classical RTI techniques [PDC*19]. The most popular and widely utilized ones are Polynomial Texture Mapping(PTM), which is based on the second-order polynomial [MGW01]), and Hemispherical Harmonics (HSH) [GKPB04], exploiting the hemispherical basis defined from the shifted associated Legendre polynomials. However, the low-frequency approximation nature of these methods often fails to suitably represent the subtle illumination effects generated by the intertwining of complex local geometric and appearance characteristics [PCS18].

In recent years, neural network-based RTI regressors [RDL*15a, XSHR18a], such as NeuralRTI [DFP*20, RBP*23, RKG*24], have been shown to produce higher-quality images at similar storage costs to the classic analytical models. However, these models use a decoder with many parameters to create the relighted images, which limits performance, hindering their suitability for real-time interactive object exploration, particularly with high-resolution acquisitions. For this reason, recent efforts have targeted the enhancement of efficiency by manually optimizing the number of layers of the network, streamlining decoding inside custom shaders, and introducing an efficient level-of-detail management system supporting fine-grained adaptive rendering through on-the-fly resampling in latent feature space [RKG*24]. While the resulting viewer facilitates interactive neural relighting of large images, exploring high-resolution images, especially on high-pixel-count displays, still requires significant resources and is only achievable through progressive rendering in typical setups. Earlier works have shown, through experimental tests [DFP*20, RBP*23], that reducing the number and size of the layers, keeping the same training procedure, can obtain only limited performance boosts without quality degradation.

Building on previous work on network compression (Sec. 2), this work introduces a knowledge distillation technique called Disk-NeuralRTI for compressing the NeuralRTI decoder. The method makes it possible to produce high-quality relighted images with a limited percentage of the original decoding parameters, making it possible to perform a smooth interactive relighting even in the case of large images and limited computational power. To our knowl-

edge, this is the first work applying this approach to the RTI relighting domain. The results show that the resulting solution outperforms the manual tuning and is highly effective, making the Neural RTI encoding usable in practical settings.

In the following, after briefly discussing related work (Sec. 2), we describe our knowledge distillation solution (Sec. 3) and its performance and capabilities on synthetic and cultural heritage models (Sec. 4). We conclude by summarizing our findings and discussing future works (Sec. 5 and Sec. 6).

2. Related Work

RTI is widely utilized in the CH domain to analyze surface properties [PDC*19] and has also found applications in other domains, such as manufacturing [NLGL*21] and quality assessment [COS19]. The main goal of RTI is interactive relighting, such as inspecting the rendered surface to simulate a manually controlled variation of the illumination direction. It allows CH researchers to interactively visualize the surface based on a compact encoding of the captured image stack.

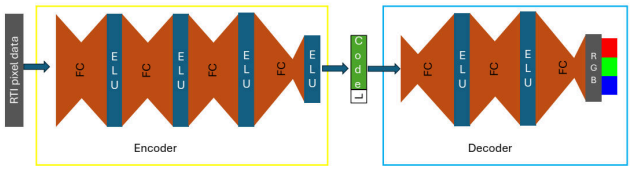
Classical RTI. Most methods approximate the reflectance field by directly mapping lighting parameters to final renderable values [PDC*19,ZD14]. PTM [MGW01,ZD14] and HSH [GKPB04] are the most widely used compact, low-complexity formulations. While these methods require only a few data decodes and arithmetic operations per pixel, they only support relatively low-frequency, smooth behaviors. Radial Basis Function (RBF) interpolation of the original data has been proposed as an alternative to simple parametric functions [GCD*18]. Still, the method requires run-time access to the original massive image stack and is not suitable for interactive relighting. It was later combined with Principal Component Analysis (PCA) compression of the image stack and RBF interpolation in light space to improve efficiency at the cost of a slight reduction in quality [PCS18]. Due to their versatility, compression rate, and fast decoding, PTM, HSH, and/or RBF/PCA are included in publicly available web-based tools for image-based relighting, e.g., *WebRTIViewer* [P*19a], *Pixel+ Viewer* [VPH*20], *Marlie* [JAP*21], *Relight* [P*19b, PCS18], and *OpenLIME* [Ope22]. We provide a plug-in alternative based on neural compression.

Neural-based RTI. In recent years, neural networks have emerged as a viable technique for compression, nonlinear approximation, and interpolation tasks involving large amounts of data. They have also been applied to rendering settings [RDL*15b, XSHR18b, TFT*20]. The NeuralRTI approach [DFP*20] directly targets the MLIC use case and has been proposed as a plug-in replacement for standard RTI representations. It uses a fully connected asymmetric autoencoder to encode the original per-pixel information into a low-dimensional vector and decode it to reconstruct pixel values based on the pixel encoding and a novel light direction. To deal with a large number of input images, the coding complexity has been improved by having the network work on PCA-compressed data rather than in the original images [PB23]. However, due to the estimation of relighted images in Neural RTI being at least two orders of magnitude more complex, which impacts interactivity, practical applications—particularly in the Cultural Heritage domain—still primarily rely on classical methods. To address this issue, Righetto et al. [RBP*23] introduced a modified version of the original Neural RTI. Through a series of experiments, they manually reduced the complexity while maintaining relighting quality. However, this reduction was insufficient to ensure interactive relighting for high-resolution images in case of limited computational power. For this reason, they try to improve interactivity by performing the decoding directly within pixel shaders and by using an adaptive multi-resolution renderer to meet frequency requirements [RKG*24]. Despite these improvements, the relatively high cost of the underlying network still results in unsatisfactory performance when panning over high-resolution images on high-pixel-count displays. To achieve optimal performance, further research is needed to develop more efficient and higher compression techniques for the underlying network.

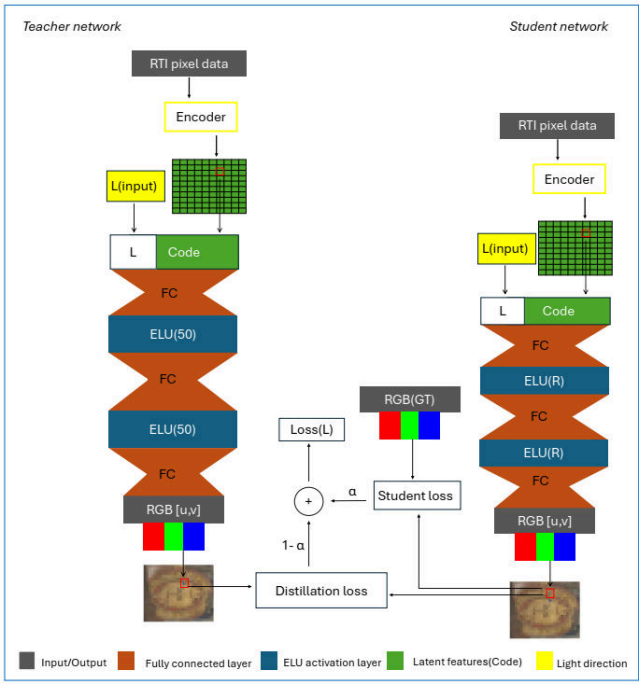
Neural Network Compression. Advancements in neural network compression have led to methods that automatically reduce network sizes and evaluation costs, rather than relying on manual adjustments. These compression techniques are typically categorized into four main types: parameter pruning, low-rank factorization, network quantization, and knowledge distillation. Parameter pruning techniques [HMD15, GYC16, YZZ*17] focus on identifying and removing redundant or non-essential model parameters. While these methods can achieve high compression ratios, they often involve converting fully-connected layers into sparsely-connected ones. This transformation can complicate decoding and potentially slow performance, particularly for small networks like NeuralRTI decoded in GPU shaders. Low-rank factorization techniques [TXZ*15, BL16] utilize matrix and tensor decomposition to identify the critical parameters in convolutional neural networks (CNNs). However, these methods typically yield impressive results mainly for moderately large to very large networks, many orders of magnitude larger than the NeuralRTI decoder. Network quantization techniques [GLYB14, CBD15, HCS*16] reduce the number of bits required to represent each weight, leading to a compressed network. This size reduction also enhances speed by improving cache efficiency but, alone, can only achieve moderate compression if limited to GPU-supported data types. Finally, knowledge distillation [HVD15] focuses on training a smaller (student) model to replicate the behavior of a larger (teacher) model, transferring knowledge from the large network to the smaller one while maintaining prediction performance. In this way, the student model learns to emulate the teacher's predictions. While this technique was initially devised for classification problems, it has also been applied to regression (e.g., [TMI20, SDGA*19]). In this work, we build on a neural-based approach inspired by Neural RTI, and, to the best of our knowledge, we are the first to apply knowledge distillation to RTI neural networks. We aim to reduce the Neural RTI model size to enhance and optimize relighting performance.

3. Disk-NeuralRTI

Our approach is based on the NeuralRTI model, as illustrated in Fig. 1a, which was introduced by Righetto et al. [RBP*23, RKG*24]. This model is the teacher network and includes an encoder and a decoder network. We train the teacher network, T, following the same procedure described by Righetto et al. [RBP*23]. The encoder comprises four layers, each with an Exponential Lin-



(a) *NeuralRTI scheme. The encoder has three hidden layers and the decoder has two hidden layers. Each hidden layer contains 50 units. The encoder receives RTI pixel data and produces a 9-dimensional code. The decoder concatenates the code vector with the light direction and outputs a single RGB value.*



(b) *Disk-NeuralRTI. The encoder has the same architecture for student and teacher networks. The student network decoder contains two layers, each with an R number of units. We tested it with R values of* 10 *and* 20.

Figure 1: *Network architecture for original NeuralRTI (top) and Disk-NeuralRTI (bottom).*

ear Unit (ELU) activation function. It processes per-pixel RTI data, which includes pixel values for various lighting directions, and compresses this data into nine-dimensional latent space features. The decoder network, shown on the right in Fig. 1a, includes two hidden layers, each with 50 units. It takes as input the concatenation of the pixel encoding and a 2D vector representing the light direction. The decoder's output is the predicted RGB pixel value illuminated from the specified light direction. The teacher network is trained end-to-end on all pixels or a subset of the original MLIC. This training involves minimizing the mean squared error between the predicted pixel values and the ground truth values across the specified light directions. Once the training phase is complete, the encoder can be used to generate the final latent features for each pixel, which are stored in a per-pixel latent feature map. The encoder is then discarded. To compute relighted images, only the learned decoder—along with its weights and biases—is used. The decoder applies these to the per-pixel latent features and the interactively set light directions to produce the final output.

The student network S (Fig. 1b, right) is derived by simplifying the original network, specifically modifying only the decoder component, as the encoder network does not impact interactive relighting. Specifically, we decreased the number of units in each hidden layer from the original 50 to significantly smaller values, aiming to reduce the total number of parameters in the decoder while ensuring interactive relighting for large images, even with limited hardware resources.

In the NeuralRTI model, the total number of decoder weights (W) and biases (B) is given by $W = (K+2) \times N + N \times N + N \times 3$ and $B = N + N + 3$. With decoder layers of size $N = 50$ and a latent feature vector of size $K = 9$, the number of weights and related multiplications is $W = 3200$, and the number of biases and related additions is $B = 103$. $K = 9$ latent code values are, instead, stored per pixel, leading to a compression rate similar to standard PTM.

Setting the decoder layer size to $N = 20$ reduces the number of weights and multiplications to $W = 680$ and the number of biases and sums to 43, achieving an 80% reduction in computation and memory fetches. A layer size of $N = 10$, leads, instead to $W = 240$ weights and multiplications, and 23 biases and sum, giving a 92% reduction in computational cost and memory pressure. The theoretical speed-up is thus very significant: between $\approx 5\times$ and $\approx 10\times$ for these configurations. Both cases were tested, and these adjustments effectively maintained smooth interactive relighting during our evaluations while achieving relighting accuracy comparable to the original NeuralRTI (see Sec. 4).

We train the student network on all pixels (or a subset) by minimizing the following loss function:

$$L = \frac{1}{n} \sum_{k=1}^{n} \alpha ||P_s - P_{gt}||_k^2 + (1-\alpha)||P_s - P_t||_k^2 \qquad (1)$$

This function is a weighted combination of two components: the student loss, which measures the difference between the student's predictions (P_s) and the ground truth pixel values (P_{gt}), and the distillation loss, which measures the difference between the teacher's predictions (P_t) and the student's predictions (P_s). The parameter α determines the weight of each loss component. The distillation loss is specifically designed to capture the discrepancy between the student and teacher models. By minimizing this loss during training, we enhance the student model's ability to accurately replicate the teacher's predictions. The basic idea behind the approach is that training a very compact model through distillation should be more effective than training it directly on the original data. Fitting original data with the larger teacher network is easier than fitting it with the smaller student network, thanks to the larger number of parameters. At the same time, the teacher model's outputs are typically smoother/less noisy and may contain richer information than the exact regression target values coming from the original images. During distillation, the teacher model can thus provide hints about the underlying distribution of the data, which can guide the student model to learn more effectively to fit the original data and generalize better. The experiments are implemented using PyTorch on four NVIDIA Ampere A100 GPUs, 64GB each.

4. Results and Evaluation

We evaluated our DisK-NeuralRTI compression using two approaches. First, we assessed the quality of relighted images at different compression levels on the same benchmarks used to showcase NeuralRTI's advantages [DFP*20], namely SynthRTI and RealRTI. SynthRTI includes synthetic multi-light image collections with various shapes and material combinations, while RealRTI contains real captures of diverse surfaces. Additionally, we tested the compressed network's capability to deliver real-time relighting for real-world use cases stemming from the cultural heritage area, where we applied the method to novel high-resolution surface captures. On these new datasets, we demonstrated the increased rendering speed achieved with DisK-NeuralRTI compression using the OpenLIME visualization framework.

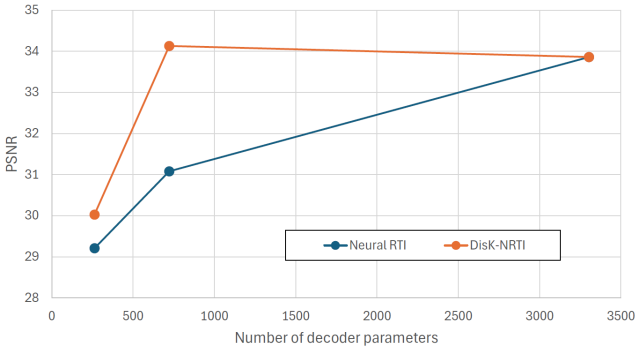


Figure 2: *Line chart showing the relighting quality (PSNR) as a function of the number of decoder parameters for the SynthRTI Single-Material collection. Training with DisK NeuralRTI results in metrics close to or better than the teacher for the 723 parameters version.*

4.1. Evaluation setup

All our results are presented using consistent configurations for training and evaluation.

In all our experiments, we used 90% of the total RTI data pixels for training and reserved 10%, sampled uniformly across pixel locations and light directions, for validation. Training for both the teacher and the student network was carried out using the Adam optimization algorithm [KB14], with a batch size of 64, a learning rate of 0.01, a gradient decay factor of 0.9, and a squared gradient decay factor of 0.99.

For distillation, we determined a single value of the parameter α and used it for all our benchmarks. The selection was made by computing the average relighting quality on a set of five real captures from the ReaRTI dataset, testing the quality of the DisK-Neural RTI compression varying the value of α in the range [0.1-0.9]. We verified that the method is not very sensitive to the selection α in the range [0.1..0.7], with a drop in quality only when alpha exceeds 0.8. For all of our tests, we thus set $\alpha = 0.6$.

The network was implemented using the PyTorch open-source deep-learning library, but was then converted to an OpenGL shader for real-time display, using the methods described in previous work [RBP*23, RKG*24]. After training, the per-pixel latent codes are converted to byte size using offset and scale mapping and stored as an image byte plane. The decoder parameters (weights and biases) and header information are saved in a JSON file. This JSON file is used at runtime by a web viewer, that executes the inference code inside the shader to relight images under new, arbitrary light directions.

4.2. Evaluation on SynthRTI and Real RTI benchmarks

To provide a comparison with other published results, we evaluated the method on public datasets containing real and synthetic samples.

4.2.1. Description of the dataset

SynthRTI [Uni20b] is a collection of 51 multi-light image collections simulated using the Blender Cycles rendering engine. It is divided into two subsets: SingleMaterial, featuring 24 captures of three surfaces with 8 different materials and MultiMaterial, with 27 captures of the same surfaces with 9 material combinations. This dataset allows the evaluation of relighting methods on matte, specular, and metallic materials and their ability to handle a wide range of reflective behaviors. Each collection is divided into two sets of images, corresponding to two different sets of light directions. The first set, called *Dome*, corresponds to a classical multi-ring light dome setup with 49 directional lights arranged in concentric rings in the l_x, l_y plane at 5 different elevation angles $(10, 30, 50, 70, 90$ degrees). The second set, called *Test*, includes 20 light directions at 4 intermediate elevation angles $(20, 40, 60, 80$ degrees). We used the *Dome* set to train our network and the *Test* set to evaluate the quality of the relighted images.

RealRTI [Uni20a] includes 12 multi-light image collections (cropped and resized to allow a fast processing/evaluation) acquired with light domes or handheld RTI protocols [PDC*19] on surfaces with different shape and material complexity. In the original paper [DFP*20], the testing protocol for validating the relighting was based on averaging leave-one-out training and testing results. Instead, we used the same approach used for SynthRTI, removing 5 test images at different elevations for each collection and training the relightable images on the remaining ones. This makes the test faster but similarly effective and challenging.

4.2.2. Religting quality evaluation

Fig. 2 presents the results of our tests on the Synth-RTI single material dataset. The chart compares the relighting accuracy as the decoder layer size is reduced from the original 50 units (3303 decoder parameters) to 20 (723) and 10 (263), while following the original training protocol or using the DisK-NeuralRTI approach. The results show that the proposed method leads to a slower decline in relighting quality as the number of decoder parameters decreases, maintaining performance close to the original with the 723-parameters configuration.

Tab. 1 compares the average PSNR and SSIM values of the comparisons between the ground truth test images and the ones relighted with the original NeuralRTI method [DFP*20], two versions of NeuralRTI with reduced layer size (20 and 10) but same

	NeuralRTI (50)	NeuralRTI (20)	NeuralRTI (10)	DisK-NRTI (20)	DisK-NRTI (10)	PTM	HSH 2.ord	HSH 3ord	PCA/RBF
Canvas	**41,42/0,99**	39,88/0,99	35,82/0,99	40,89/0,99	36,77/0,98	29,03/0.98	35,42/0,99	41,24/0,99	34,2/0,99
Tablet	29,13/0,88	26,45/0,83	25,7/0,81	**30,53/0,90**	26,99/0,84	23,79/0,81	27,63/0,84	29,92/0,87	25,87/0,80
Bas-relief	**31,02**/0,89	26,91/0,83	26,12/0,82	30,97/**0,90**	26,33/0,80	24,47/0.81	27,22/0.85	28,82/0.86	25,55/0,86
Average	33,86/0,92	31,08/0,88	29,21/0,87	**34,13/0,93**	30,03/0,87	25,76/0.87	30,09/0.89	33,33/0.91	28,54/0,88

Table 1: *Average PSNR/SSIM values for the relighting of test images of SynthRTI SingleMaterial collections. Disk-NeuralRTI provides very good results with a per-pixel encoding size of 9 parameters (as PTM and PCA/RBF) and a sufficiently small number of shared decoding parameters per image. With a layer size of 20 elements, it actually provides better metrics than the teacher networks. Bold figures indicate best values. Figures in parentheses indicate the network layers' size.*

training method, and the proposed DisK-NeuralRTI solution with the corresponding layers' sizes.

DisK-NeuralRTI achieves comparable or superior results to the original method while using significantly fewer parameters. The results are also better than all the classical methods, also those requiring a huge per-pixel encoding size.

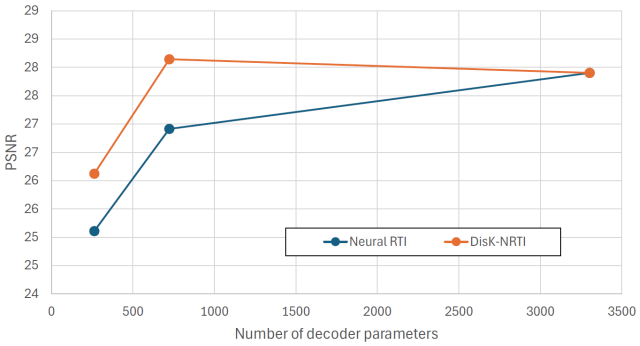


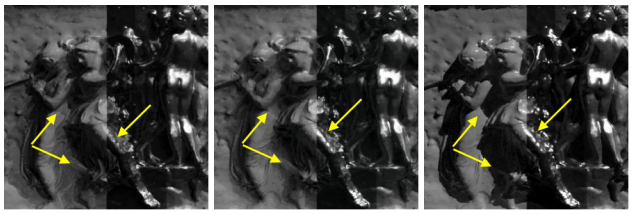
Figure 3: *Line chart showing the relighting quality (PSNR) as a function of the number of decoder parameters for the SynthRTI Multi-Material collection. Training with DisK NeuralRTI results in metrics close to or better than the teacher for the 723 parameters version.*

Similar results are obtained on the SynthRTI multi-material dataset. Fig. 3 shows the PSNR obtained as a function of the number of parameters for the standard and the DisK-NeuralRTI training. Tab. 2 shows the complete comparisons of the average metrics also against classical methods, showing very good results for the compressed Disk-NeuralRTI method up to a layer size of 20. The decrease of the PSNR for smaller layers suggests that 20 can be a nearly optimal solution for the layers' size.

The improvements in the relighting quality for the same layers' size with the new training procedure are also visually evident. Fig. 4 shows an example where it is possible to see that the layers' shrinking without the knowledge distillation approach results in artifacts, especially in shadowed and highlighted details.

For the RealRTI dataset, the average quality decreases with the layers'size behave similarly (Fig. 5). The average metrics, compared with the classical methods (Tab. 3) show here smaller advantages. The compressed neural method does not outperform 3rd order HSH.

It must, however, be considered that the 3rd order HSH encoding requires a large number of per-pixel values (48), compared to



(a) *NeuralRTI (20)* **(b)** *Disk-NRTI (20)* **(c)** *Ground Truth*

Figure 4: *(a) Relight with a test light direction of the SynthRTI multi-material set using the NeuralRTI(20) model. (b) Relight with the same light direction obtained with the Disk-NeuralRTI(20) compressed model. (c) Ground truth image corresponding to the test direction. It is possible to see (see arrows) that the layer size reduction with the original training (a) results in the loss of accuracy of the specular reflections and shadows. The image in (b) presents less artifacts compared with the ground truth (c)*

the 9 elements used in all neural encodings (as well as PTM and PCA/RBF), making it impractical for the transmission and rendering of large images (a 40 Megapixel image would results in a 2GB encoding with 3rd order HSH, while the 9 byte per pixel encoding of Neural RTI would be five times lighter).

A reason for the smaller advantages of NeuralRTI on this benchmark is probably that the sizes of the images are too small (often less than 300×300 pixels). The consequence is that the number of pixels used to train the networks is quite limited compared to a typical acquisition performed in an application domain, which is at least 100 times higher. Some items also present poor relief variations and texture, so the pixels' information is heavily correlated. To verify the performance of the methods in more realistic contexts, we evaluated our method on a novel benchmark composed of high-resolution MLICs from real cultural heritage applications.

4.3. Evaluation on real-world cultural heritage data

To evaluate the advantages of the compressed network in practical settings, we created another relighting quality benchmark with three high-resolution MLIC captures of cultural heritage artifacts. These artifacts are currently investigated in research projects where practitioners need to visualize them interactively with good quality and low latency.

4.3.1. Description of the datasets

The first dataset was captured on a lead sheet found in Cesarea Marittima, Israel, during the excavations of an Italian archaeologi-

	NeuralRTI (50)	NeuralRTI (20)	NeuralRTI (10)	Disk-NRTI (20)	Disk-NRTI (10)	PTM	HSH 2 ord	HSH 3 ord	PCA/RBF
Canvas	**33,33/0,96**	31,94/0,95	28,97/0,93	32,63/0,95	29,95/0,94	25,17/0.93	28,45/0,92	30,03/0,95	27,95/0,95
Tablet	**24,29**/0,77	23,58/0,75	22,38/0,71	24,96/0,79	23,47/0,75	20,56/0,79	22,76/0,82	24,24/**0,84**	20,89/0,76
Bas-relief	26,08/0,83	25,20/0,80	23,97/0,76	**26,83/0,84**	24,94/0,80	22,34/0,76	23,81/0,78	25,10/0,79	21,54/0,81
Average	27,90/0,85	26,91/0,83	25,10/0,80	**28,14/0,86**	26,12/0,83	22,69/0,83	25,01/0,84	26,46/0,86	23,46/0,84

Table 2: *Average PSNR/SSIM values for the relighting of test images of SynthRTI MultiMaterial collections. The 20-elements layer compression achieve better results than the teacher network. Figures in parentheses indicate the network layers' size.*

	NeuralRTI (50)	NeuralRTI (20)	NeuralRTI (10)	Disk-NRTI (20)	Disk-NRTI (10)	PTM	HSH 2 ord	HSH 3 ord	PCA/RBF
Average (12MLICs)	**31,56**/0,88	29,23/0,85	27,52/0,81	30,46/0,88	28,43/0,80	27,95/0,88	30,06/**0,89**	30,77/0,88	28,29/0,87

Table 3: *Average PSNR/SSIM values for the relighting of test images of RealRTI collections. Values differ from [DFP*20] as we changed the testing protocol (see text). Figures in parentheses indicate the network layers' size.*

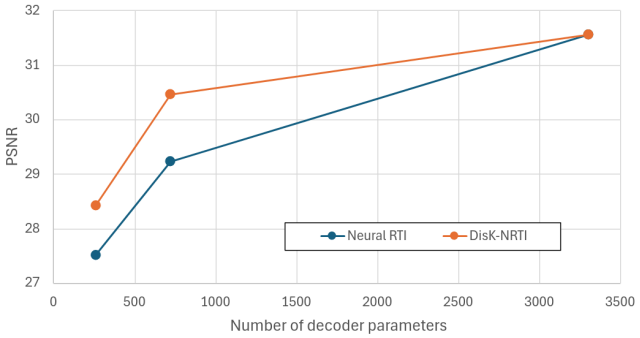


Figure 5: *Line chart showing the average relighting quality (PSNR) as a function of the number of decoder parameters for the RealRTI collection.*

cal mission directed by Antonio Frova in the 1960s, which is now at the Archaeological Museum of Milan. MLIC data were obtained with a light dome (47 LED) and a Nikon D810 DSLR camera.

The other two datasets are two paintings, specifically two panels from the retable of St. Bernardino (1455). This polyptych, originally located in the chapel of St. Bernardino within the St. Francesco church in Cagliari, Italy, is now housed and displayed at the Pinacoteca Nazionale in Cagliari. The first panel, as shown in Fig. 6b, measures 34×25 cm and is painted in oil on a wooden support, depicting the prophet Daniel. The second panel, illustrated in Fig. 6c, is slightly larger (54×36 cm) and features a golden arched frame with an image of Christ in pity, supported by an angel. Both paintings were documented in their pre-restoration condition using a free-form setup. This setup included a 36.3 Megapixel DSLR FX Nikon D810 Camera with a 50 AF Nikkor Lens and a handheld white LED (5500K) that spans the entire visible spectrum. Approximately 60 images were taken for each Multi-Light Image Collection (MLIC).

Using the relightable images created with these datasets after a careful light calibration [PJZ*21], we not only tested the effect of the compression on these images as well (Tab. 4) but also evaluated the frame rate of the interactive relighting performed on these large images with the non-compressed and compressed decoder integrated into the OpenLime online viewer [RKG*24].

4.3.2. Relighting quality evaluation

We split each dataset into a training and test set to test the quality of the relighting on the new high-resolution images. The training set is generated by selecting the images corresponding to the light directions with the smaller angular distance from the directions of the virtual dome of the SynthRTI Train dataset. The test set is generated by selecting images with light directions close to the ones in the SynthRTI test dataset. We used the training set to fit the relighting image model and the test set to evaluate the PSNR and SSIM values against the ground-truth images, exactly as in the SynthRTI benchmark.

Tab. 4 show that not only the quality of the relighting obtained on the high-resolution test images with DisK-NeuralRTI is quite close to the one obtained with the original model but also that the difference in the metrics with respect to the other methods is quite large. The average PSNR obtained with our method is approximately 20% higher than the one obtained with third-order HSH, which is a huge difference. This may indicate that, as the models are trained on the full set of pixels, high-resolution images increase the advantages obtained with the neural model, and these advantages are kept also in the compressed version.

The advantages of the proposed compressed encoding can be seen quite well looking at details of the relighted lamina (Fig. 7). With smaller layer size the original Neural RTI fails to reproduce high-frequency reflectance properties (Fig. 7 a). The proposed technique make the result quite close to the ground truth (Fig. 7 b).

The improvements in the relighting quality with the compressed NeuralRTI method, also when compared with the best Fig. 8. The third-order HSH cannot reproduce the strong highlights and the difference in the reflectance of different materials (a). The same relighting performed with the compressed Disk-NeuralRTI method (b) results in a quite accurate highlight simulation and presents only a slight color shift compared with the ground-truth test image (c).

4.3.3. Interactive relighting performances

As explained by Righetto et al. [RKG*24], NeuralRTI rendering has been integrated into the OpenLIME web-based image viewer. OpenLIME renders an image using a graphic shader, which runs the rendering algorithm in parallel on every pixel. The WebGL 2

(a) *Lamina* **(b)** *Retablo (small)* **(c)** *Retablo (big)*

Figure 6: *Example images of the real-world Cultural Heritage MLICs used for benchmarking. (a): lead sheet found in Cesarea Marittima, Israel. (b), (c): Panels from the retable of St. Bernardino (1455), Cagliari, Italy.*

(a) *NeuralRTI (20)* **(b)** *Disk-NeuralRTI (20)* **(c)** *Ground Truth*

Figure 7: *Detail of the relighting from a test light direction of the Lamina surface, featuring specular regions and varying depth. (a) The Neural RTI with layers size 20 and traditional training fails to reproduce the highlights visible in the ground truth image (c). The use of the student network provides an accurate reproduction of the reflectance with the same layers' size (b).*

technology is exploited, which allows to run the shader on the computer's graphic card, considerably increasing the rendering speed. Once the network is trained, the decoder's parameters are extracted and loaded in the shader as external variables. The latent space is a matrix of dimension $H \times W \times K$, where $H \times W$ is the image resolution, and K is the number of latent space features. The matrix entries are stored as a set of RGB JPEG images, which are loaded in the shader as samplers. The shader reads the content of each sampler pixel by pixel. It applies the decoding operations: scalar product between weights and input vector, sum with the biases, and application of the activation function.

Despite the use of graphic cards, the number of operations required by neural rendering is large, and it could be unfeasible to use it for large images and large viewports on commercially available, medium-cost computers. For this reason, as shown by Righetto et al. [RKG*24], a set of strategies has been adopted to compensate for a large number of operations, which has a big impact, especially on high-resolution images. The image is divided into tiles, and each tile is relighted independently. Thus, only the visible tiles on the screen must be rendered. Moreover, the resolution of the screen and that of the relighted image are decoupled. In practice, when the user changes interactively the light direction or performs a zoom operation, the application tries to preserve rendering speed by decreasing the resolution at which the decoding is performed to match a target value of frame per second (fps), i.e., 20 fps. An upscaled version of the relighted image is finally displayed on the screen. When the user stops moving, full resolution rendering is performed. Fig. 9 (a) shows the effect of this on the web viewer: the snapshot captured during a zooming operation appears blurred due to the low resolution used for decoding. Using the DisK-Neural RTI encoding there is no need for downsampling and the snapshot captured during a similar zooming (b) appears sharp.

To compare Disk-NeuralRTI (20) decoder performance against the original NeuralRTI, we deactivated these optimization strategies. The images have not been divided into tiles. We can thus measure the actual fps reached at full resolution to see how the two networks perform by relighting the whole image and removing the resolution decoupling. To measure fps, we computed a series of relighting operations in sequence, collecting a set of fps values and calculating its average. The evaluation has been done on a commercially available MacBook Pro laptop of 2019. Its specifications are processor 1,4 GHz Intel Core i5 quad-core, graphic card Intel Iris Plus Graphics 645 1536 MB, RAM 8 GB 2133 MHz LPDDR3, operative system macOS Sonoma version 14.3.1 (23D60). The web browser used is Google Chrome (version 128.0.6613.138).

Tab. 5 shows the average fps measured with this setting on the three RTI datasets of the novel high-resolution benchmark: Lamina (4328×2436), Retablo small (3811×2851), Retablo big ($4117 \times$

	NeuralRTI (50)	NeuralRTI (20)	Disk-NRTI (20)	PTM	HSH 2 ord	HSH 3 ord	PCA/RBF
Lamina	**37,29/0,92**	33,00/0,83	36,47/0,94	32,35/0,86	34,36/0,89	34,53/0,88	31,50/0,84
Retablo_small	**31,68/0,84**	26,84/0,72	31,04/0,82	23,06/0,76	23,83/0,77	24,92/0,77	24,34/0,76
Retablo_big	37,57/0,95	33,37/0,92	**38,33/0,95**	27,99/0,92	28,65/0,92	29,54/0,92	29,14/0,92
Average	**35,52/0,90**	31,07/0,82	35,28/0,90	27,8/0,85	28,94/0,86	29,66/0,86	28,32/0,84

Table 4: *Average PSNR/SSIM of the methods on the different high-resolution datasets. The quality of the neural relight is far better with the neural model with respect to classical techniques, and the compression with DisK-NRTI does not affect the quality.*

(a) *HSH 3rd order* **(b)** *Disk-NeuralRTI (20)* **(c)** *Ground truth*

Figure 8: *Relighting of the Retablo big surface with a test light direction not included in the training set. Third order HSH, the best classical method, fails in representing the correct reflectance behavior (a). The compressed Disk-NRTI result (b) is, instead, quite close to the reference image (c).*

(a) *NeuralRTI-50* **(b)** *Disk-NeuralRTI (20)*

Figure 9: *Using the adaptive multiresolution rendering of OpenLime, the system dynamically adapts the rendered images' resolution to guarantee interactivity. (a) Snapshot captured in a zooming interaction with the non-compressed NeuralRTI visualization of the Lamina surface. The image is heavily blurred. (b) Snapshot captured in a similar zooming interaction with the compressed version. Images are always sharp.*

	Lamina (4328×2436)	Retablo small (3811×2851)	Retablo big (4117×3427)
Disk-NRTI (20)	29.68	28.09	22.16
NeuralRTI (50)	1.60	1.42	1.10

Table 5: *Average fps values calculated during relighting of the three high-resolution datasets.*

3427). The models are rendered at full resolution on a screen on a 8K screen (no cropping).

To understand how the fps value changes with the image size in this setting, we performed another test cropping the relightable image parameters' arrays in different ways to vary the total number of pixels to be relighted of one order of magnitude every step, from a 100×100 image (10 thousand pixels) to a 5000×2000 image (10 million pixels). In this way, we can visualize when the performance of the two models worsens. Results are summarized in Fig. 10.

Video recordings captured during the interactive relighting of the three items on the laptop with NeuralRTI(50) and DisKNeuralRTI(20) encoding with and without the adaptive resolution can be watched on the web page https://tgdulecha.github.io/Disk-NeuralRTI/.

5. Discussion

NeuralRTI is currently the RTI relighting method that best reproduces the real reflectance properties of surfaces, especially high-frequency ones. However, the relatively complex custom decoder used by the technique to perform the relighted image rendering may create an annoying latency for high-resolution images on low-end devices. Previous work integrating the method in an online viewer solved the issue by adapting the resolution of the rendered window to the desired frame rate during the interaction [RKG*24], at the cost of an evident detail loss. Using the proposed network compression approach based on Knowledge Distillation, we strongly re-

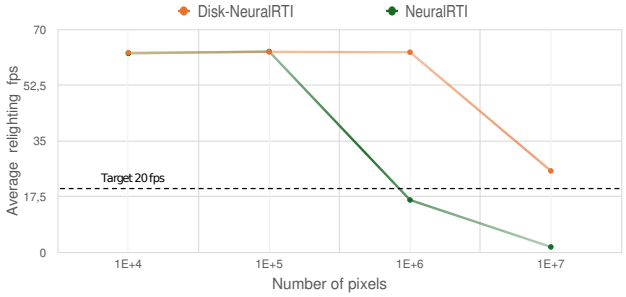


Figure 10: *Average fps calculation for varying number of pixels simultaneously relighted. On the x-axis is the number of pixels in order of magnitude, i.e., from 10 thousand to 10 million. On the y-axis, the average fps during relighting.*

duced the decoding time, making it possible to render large images in real time with interactive performances on standard PCs without lowering the resolution. Our tests show that the Disk-NeuralRTI (20) encoding can guarantee smooth interactive relighting with a higher resolution than the one displayed on 8K UHD screens using low-end hardware. This makes it practical for professional cultural heritage and engineering applications.

Our work is the first attempt to apply network compression approaches to NeuralRTI, and it has given promising results. However, it is certainly possible to improve it. Our experiments used the original NeuralRTI model as a teacher network. This architecture was initially designed with a light decoder architecture to achieve acceptable interactive relighting. By applying knowledge distillation from a deeper teacher architecture, it should be possible to improve the quality of the results further or make the decoding even more efficient.

Moreover, NeuralRTI, similarly to all other learning-based solutions, requires more time to generate a representation than standard fitting-based solutions such as PTM or HSH, as it needs to learn an optimized representation from the presented set of examples. The addition of student network training and the potential adoption of more complex teacher networks further increases the cost of generating the representation for relightable image generation. While this is not critical for the end-user applications, as training is not performed under time-critical constraints and is generally much faster than the cost of acquiring, often on-site, a complex model, there is space for speeding up the solution. In particular, we plan to develop strategies to speed up the networks' training by using a smartly selected subset of the original MLIC pixels for the encoding instead of the complete set.

6. Conclusion

We have shown the potential of using knowledge distillation to create more efficient Neural Reflectance Transformation Imaging (RTI) decoders for interactive object exploration in cultural heritage applications. While neural representations have shown superior image quality at comparable storage costs to traditional models like PTM or HSH, their decoding costs have often hindered their practical usage for real-time high-resolution exploration of large models on high-pixel-count displays. In contrast to previous manual attempts to tune network size, our approach leverages a knowledge distillation framework, where a smaller student network is trained to mimic the output of a larger, more complex teacher network, resulting in a compressed model that retains high-quality relighting capabilities.

The performance of this network compression strategy was evaluated across various RTI relighting benchmarks, including both synthetic and real datasets, as well as newly acquired high-resolution images. Our preliminary experimental results reveal that the student network achieves predictions that remain close to the teacher network's output while significantly reducing the number of parameters—up to 80%, and, most importantly, achieving a substantial improvement in computation time. Combined with previous works on adaptive multiresolution rendering, this reduction in computational overhead makes high-resolution image exploration more feasible without compromising the quality of interactive relighting. This advancement represents a step forward in making neural-based relightable image models more practical for widespread use in cultural heritage applications.

Acknowledgments This study was partially funded by the consortium iNEST funded by the EU Next-GenerationEU PNNR M4C2 Inv1.5 – D.D. 1058 23/06/2022, ECS00000043 and by the project REFLEX (PRIN2022) funded by the EU Next-GenerationEU PNRR M4C2 Inv. 1.1. EG and RP also acknowledge the contribution of Sardinian Regional Authorities under project XDATA (RAS Art9 LR 20/2015). We thank Prof. Attilio Mastrocinque and the Civic Archaeological Museum of Milan for access to the lead sheet and the National Archaeological Museum of Cagliari for access to the retablos. We thank Fabio Bettio (CRS4), Fabio Marton (CRS4), and Federico Ponchio (ISTI-CNR) for their work in developing OpenLime and for the integration of Neural RTI rendering within the framework. We acknowledge ISCRA for awarding this project access to the LEONARDO supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CINECA (Italy).

References

[BL16] BHATTACHARYA S., LANE N. D.: Sparsification and separation of deep learning layers for constrained resource inference on wearables. In *Proc. ACM ENSS* (2016), pp. 176–189. 2

[CBD15] COURBARIAUX M., BENGIO Y., DAVID J.-P.: BinaryConnect: Training deep neural networks with binary weights during propagations. *NeurIPS 28* (2015). 2

[COS19] COULES H., ORROCK P., SEOW C. E.: Reflectance transformation imaging as a tool for engineering failure analysis. *Engineering Failure Analysis 105* (2019), 1006–1017. 2

[DFP*20] DULECHA T. G., FANNI F. A., PONCHIO F., PELLACINI F., GIACHETTI A.: Neural reflectance transformation imaging. *The Visual Computer 36* (2020), 2161–2174. 1, 2, 4, 6

[GCD*18] GIACHETTI A., CIORTAN I. M., DAFFARA C., MARCHIORO G., PINTUS R., GOBBETTI E.: A novel framework for highlight reflectance transformation imaging. *Computer Vision and Image Understanding 168* (2018), 118–131. 2

[GKPB04] GAUTRON P., KRIVÁNEK J., PATTANAIK S. N., BOUATOUCH K.: A novel hemispherical basis for accurate and efficient rendering. *Rendering Techniques 2004* (2004), 321–330. 1, 2

[GLYB14] GONG Y., LIU L., YANG M., BOURDEV L.: Compressing deep convolutional networks using vector quantization. *arXiv preprint arXiv:1412.6115* (2014). 2

[GYC16] GUO Y., YAO A., CHEN Y.: Dynamic network surgery for efficient DNNs. *NeurIPS 29* (2016). 2

[HCS*16] HUBARA I., COURBARIAUX M., SOUDRY D., EL-YANIV R., BENGIO Y.: Binarized neural networks. *NeurIPS 29* (2016). 2

[HMD15] HAN S., MAO H., DALLY W. J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149* (2015). 2

[HVD15] HINTON G., VINYALS O., DEAN J.: Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015). 2

[JAP*21] JASPE VILLANUEVA A., AHSAN M., PINTUS R., GIACHETTI A., GOBBETTI E.: Web-based exploration of annotated multi-layered relightable image models. *ACM JOCCH 14*, 2 (May 2021), 24:1–24:31. 2

[KB14] KINGMA D. P., BA J.: Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014). 4

[MGW01] MALZBENDER T., GELB D., WOLTERS H.: Polynomial texture maps. In *Proc. SIGGRAPH* (2001), pp. 519–528. 1, 2

[NLGL*21] NURIT M., LE GOÏC G., LEWIS D., CASTRO Y., ZENDAGUI A., CHATOUX H., FAVRELIÈRE H., MANIGLIER S., JOCHUM P., MANSOURI A.: HD-RTI: An adaptive multi-light imaging approach for the quality assessment of manufactured surfaces. *Computers in Industry 132* (2021), 103500. 2

[Ope22] OPENLIME TEAM: OpenLime: Open Layered IMage Explorer, 2022. URL: https://github.com/cnr-isti-vclab/openlime and https://github.com/crs4/openlime [Online; accessed 2024-10-19]. 2

[P*19a] PALMA G., ET AL.: WebRTI Viewer, 2019. [Online; accessed 2024-09-20]. URL: http://vcg.isti.cnr.it/rti/webviewer.php. 2

[P*19b] PONCHIO F., ET AL.: Relight, 2019. [Online; accessed 2024-10-19]. URL: http://vcg.isti.cnr.it/relight/. 2

[PB23] PISTELLATO M., BERGAMASCO F.: On-the-go reflectance transformation imaging with ordinary smartphones. In *Proc. ECCV Workshops, Part I* (2023), pp. 251–267. 2

[PCS18] PONCHIO F., CORSINI M., SCOPIGNO R.: A compact representation of relightable images for the web. In *Proc. ACM Web3D* (2018), pp. 1:1–1:10. 1, 2

[PDC*19] PINTUS R., DULECHA T. G., CIORTAN I., GOBBETTI E., GIACHETTI A.: State-of-the-art in multi-light image collections for surface visualization and analysis. *Computer Graphics Forum 38*, 3 (2019), 909–934. 1, 2, 4

[PJZ*21] PINTUS R., JASPE VILLANUEVA A., ZORCOLO A., HADWIGER M., GOBBETTI E.: A practical and efficient model for intensity calibration of Multi-Light Image Collections. *The Visual Computer 37*, 9 (September 2021), 2755–2767. 6

[RBP*23] RIGHETTO L., BETTIO F., PONCHIO F., GIACHETTI A., GOBBETTI E.: Effective interactive visualization of neural relightable images in a web-based multi-layered framework. In *Proc. GCH* (2023), pp. 57–66. 1, 2, 4

[RDL*15a] REN P., DONG Y., LIN S., TONG X., GUO B.: Image based relighting using neural networks. *ACM TOG 34*, 4 (2015), 1–12. 1

[RDL*15b] REN P., DONG Y., LIN S., TONG X., GUO B.: Image based relighting using neural networks. *ACM TOG 34*, 4 (2015), 111:1–111:12. 2

[RKG*24] RIGHETTO L., KHADEMIZADEH M., GIACHETTI A., PONCHIO F., GIGILASHVILI D., BETTIO F., GOBBETTI E.: Efficient and user-friendly visualization of neural relightable images for cultural heritage applications. *ACM JOCCH 17* (2024). To appear. 1, 2, 4, 6, 7, 8

[SDGA*19] SAPUTRA M. R. U., DE GUSMAO P. P., ALMALIOGLU Y., MARKHAM A., TRIGONI N.: Distilling knowledge from a deep pose regressor network. In *Proc. ICCV* (2019), pp. 263–272. 2

[TFT*20] TEWARI A., FRIED O., THIES J., SITZMANN V., LOMBARDI S., SUNKAVALLI K., MARTIN-BRUALLA R., SIMON T., SARAGIH J., NIESSNER M., ET AL.: State of the art on neural rendering. *Computer Graphics Forum 39*, 2 (2020), 701–727. 2

[TMI20] TAKAMOTO M., MORISHITA Y., IMAOKA H.: An efficient method of training small models for regression problems with knowledge distillation. In *Proc. MIPR* (2020), IEEE, pp. 67–72. 2

[TXZ*15] TAI C., XIAO T., ZHANG Y., WANG X., ET AL.: Convolutional neural networks with low-rank regularization. *arXiv preprint arXiv:1511.06067* (2015). 2

[Uni20a] UNIVERSITY OF VERONA: RealRTI, 2020. [Online; accessed-2023-09-06]. URL: https://github.com/Univr-RTI/RealRTI. 4

[Uni20b] UNIVERSITY OF VERONA: SynthRTI, 2020. [Online; accessed-2024-09-06]. URL: https://github.com/Univr-RTI/SynthRTI. 4

[VPH*20] VANWEDDINGEN V., PROESMANS M., HAMEEUW H., VANDERMEULEN B., DER PERRE A. V., VASTENHOUD C., LEMMERS F., WATTEEUW L., GOOL L. V.: Pixel-Plus Viewer, 2020. [Online; accessed-2024-09-20]. URL: https://www.heritage-visualisation.org/pixelplusviewer.html. 2

[XSHR18a] XU Z., SUNKAVALLI K., HADAP S., RAMAMOORTHI R.: Deep image-based relighting from optimal sparse samples. *ACM TOG 37*, 4 (2018), 126. 1

[XSHR18b] XU Z., SUNKAVALLI K., HADAP S., RAMAMOORTHI R.: Deep image-based relighting from optimal sparse samples. *ACM TOG 37*, 4 (2018), 126:1–126:13. 2

[YZZ*17] YAO S., ZHAO Y., ZHANG A., SU L., ABDELZAHER T.: Deepiot: Compressing deep neural network structures for sensing systems with a compressor-critic framework. In *Proceedings of the 15th ACM conference on embedded network sensor systems* (2017), pp. 1–14. 2

[ZD14] ZHANG M., DREW M. S.: Efficient robust image interpolation and surface properties using polynomial texture mapping. *EURASIP Journal on Image and Video Processing 2014*, 1 (2014), 25. 2