Deep Outdoor Illumination Estimation

Yannick Hold-Geoffroy[1*], Kalyan Sunkavalli[†], Sunil Hadap[†], Emiliano Gambaretto[†], Jean-François Lalonde[*]
Université Laval[*], Adobe Research[†]
yannick.hold-geoffroy.1@ulaval.ca, {sunkaval,hadap,emiliano}@adobe.com, jflalonde@gel.ulaval.ca
http://www.jflalonde.ca/projects/deepOutdoorLight

Abstract

We present a CNN-based technique to estimate high-dynamic range outdoor illumination from a single low dynamic range image. To train the CNN, we leverage a large dataset of outdoor panoramas. We fit a low-dimensional physically-based outdoor illumination model to the skies in these panoramas giving us a compact set of parameters (including sun position, atmospheric conditions, and camera parameters). We extract limited field-of-view images from the panoramas, and train a CNN with this large set of input image–output lighting parameter pairs. Given a test image, this network can be used to infer illumination parameters that can, in turn, be used to reconstruct an outdoor illumination environment map. We demonstrate that our approach allows the recovery of plausible illumination conditions and enables photorealistic virtual object insertion from a single image. An extensive evaluation on both the panorama dataset and captured HDR environment maps shows that our technique significantly outperforms previous solutions to this problem.

1. Introduction

Illumination plays a critical role in deciding the appearance of a scene, and recovering scene illumination is important for a number of tasks ranging from scene understanding to reconstruction and editing. However, the process of image formation conflates illumination with scene geometry and material properties in complex ways and inverting this process is an extremely ill-posed problem. This is especially true in outdoor scenes, where we have little to no control over the capture process.

Previous approaches to this problem have relied on extracting cues such as shadows and shading [26] and combining them with (reasonably good) estimates of scene geometry to recover illumination. However, both these tasks are challenging and existing attempts often result in poor performance on real-world images. Alternatively, techniques for intrinsic images can estimate low-frequency illumination but rely on hand-tuned priors on geometry and material properties [3, 29] that may not generalize to large-scale scenes. In this work, we seek a single image outdoor illumination inference technique that generalizes to a wide range of scenes and does not make strong assumptions about scene properties.

To this end, our goal is to train a CNN to directly regress a single input low dynamic range image to its corresponding high dynamic range (HDR) outdoor lighting conditions. Given the success of deep networks at related tasks like intrinsic images [42] and reflectance map estimation [34], our hope is that an appropriately designed CNN can learn this relationship. However, training such a CNN requires a very large dataset of outdoor images with their corresponding HDR lighting conditions. Unfortunately, such a dataset currently does not exist, and, because capturing light probes requires significant time and effort, acquiring it is prohibitive.

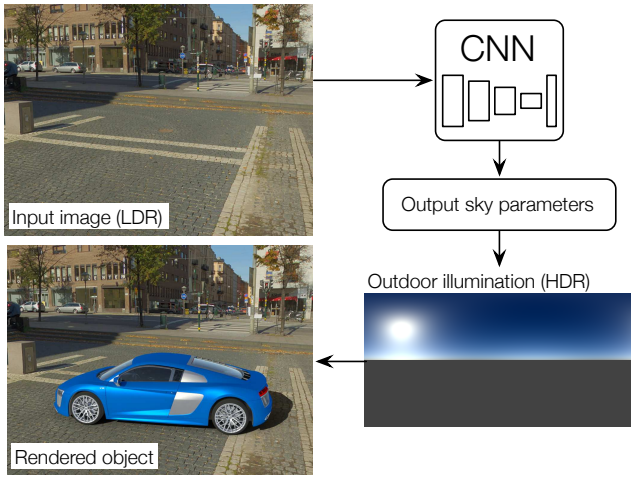


Figure 1. We present an approach for predicting full HDR lighting conditions from a single LDR outdoor image. Our prediction can readily be used to insert a virtual object into the image. Our key idea is to train a CNN using input-output pairs of LDR images and HDR illumination parameters that are automatically extracted from a large database of 360° panoramas.

[1] Research partly done when Y. Hold-Geoffroy was an intern at Adobe Research.

Our insight is to exploit a large dataset of outdoor panoramas [40], and extract photos with limited field of view from them. We can thus use pairs of photos and panoramas to train the neural network. However, this approach is bound to fail since: 1) the panoramas have low dynamic range and therefore do not provide an accurate estimate of outdoor lighting; and 2) even if notable attempts have been made [41], recovering full spherical panoramas from a single photo is both improbable and unnecessary for a number of tasks (e.g., many of the high-frequency details in the panoramas are not required when rendering Lambertian objects into the scene).

Instead, we use a physically-based sky model—the Hošek-Wilkie model [16, 17]—and fit its parameters to the visible sky regions in the input panorama. This has two advantages: first, it allows us to recover physically accurate, high dynamic range information from the panoramas (even in saturated regions). Second, it compresses the panorama to a compact set of physically meaningful and representative parameters that can be efficiently learned by a CNN. At test time, we recover these parameters—including sun position, atmospheric turbidity, and geometric and radiometric camera calibration—from an input image and use them to construct an HDR sky environment map.

To our knowledge, we are the first to address the complete scope of estimating a full HDR lighting representation—which can readily be used for image-based lighting [7]—from a single outdoor image (fig. 1). Previous techniques have typically addressed only aspects of this problem, e.g., Lalonde et al. [26] recover the position of the sun but need to observe sky pixels in order to recover the atmospheric conditions. Similarly, [30] uses a neural network to estimate the sun azimuth to perform localization in roadside environments. Karsch et al. [19] estimate full environment map lighting, but their panorama transfer technique may yield illumination conditions arbitrarily far away from the real ones. In contrast, our technique can recover an accurate, full HDR sky environment map from an arbitrary input image. We show through extensive evaluation that our estimates of the lighting conditions are significantly better than previous techniques and that they can be used "as is" to photorealistically relight and render 3D models into images.

2. Related work

Outdoor illumination models Perez et al. [31] proposed an all-weather sky luminance distribution model. This model was a generalization of the CIE standard sky model and is parameterized by five coefficients that can be varied to generate a wide range of skies. Preetham [32] proposed a simplified version of the Perez model that explains the five coefficients using a single unified atmospheric turbidity parameter. Lalonde and Matthews [27] combined the Preetham sky model with a novel empirical sun model. Hošek and Wilkie proposed a sky luminance model [16] and solar radiance function [17].

Outdoor lighting estimation Lalonde et al. [26] combine multiple cues, including shadows, shading of vertical surfaces, and sky appearance to predict the direction and visibility of the sun. This is combined with an estimation of sky illumination (represented by the Perez model [31]) from sky pixels [28]. Similar to this work, we use a physically-based model for outdoor illumination. However, instead of designing hand-crafted features to estimate illumination, we train a CNN to directly learn the highly complex mapping between image pixels and illumination parameters.

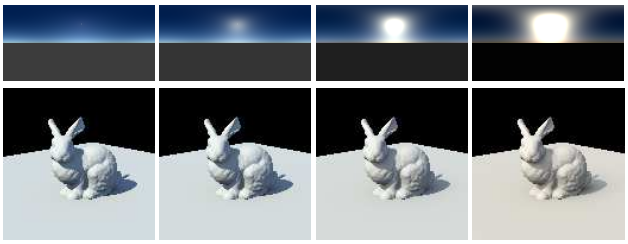
Other techniques for single image illumination estimation rely on known geometry and/or strong priors on scene reflectance, geometry and illumination [3, 4, 29]. These priors typically do not generalize to large-scale outdoor scenes. Karsch et al. [19] retrieve panoramas (from the SUN360 panorama dataset [40]) with features similar to the input image, and refine the retrieved panoramas to compute the illumination. However, the matching metric is based on image content which may not be directly linked with illumination.

Another class of techniques simplify the problem by estimating illumination from image collections. Multi-view image collections have been used to reconstruct geometry, which is used to recover outdoor illumination [14, 27, 35, 8], sun direction [39], or place and time of capture [15]. Appearance changes have also been used to recover colorimetric variations of outdoor sun-sky illumination [37].

Inverse graphics/vision problems in deep learning Following the remarkable success of deep learning-based methods on high-level recognition problems, these approaches are now being increasingly used to solve inverse graphics problems [24]. In the context of understanding scene appearance, previous work has leveraged deep learning to estimate depth and surface normals [9, 2], recognize materials [5], decompose intrinsic images [42], recover reflectance maps [34], and estimate, in a setup similar to physics-based techniques [29], lighting from objects of specular materials [11]. We believe ours is the first attempt at using deep learning for full HDR outdoor lighting estimation from a single image.

3. Overview

We aim to train a CNN to predict illumination conditions from a single outdoor image. We use full spherical, 360° panoramas, as they capture scene appearance while also providing a direct view of the sun and sky, which are the most important sources of light outdoors. Unfortunately, there exists no database containing true high dynamic range outdoor panoramas, and we must resort to using the saturated, low dynamic range panoramas in the



$t = 1$ $t = 3$ $t = 7$ $t = 10$

Figure 2. Impact of sky turbidity t on rendered objects. The top row shows environment maps (in latitude-longitude format), and the bottom row shows corresponding renders of a bunny model on a ground plane for varying values for the turbidity t, ranging from low (left) to high (right). Images have been tonemapped with $\gamma = 2.2$ for display.

SUN360 dataset [40]. To overcome this limitation, and to provide a small set of meaningful parameters to learn to the CNN, we first fit a physically-based sky model to the panoramas (sec. 4). Then, we design and train a CNN that given an input image sampled from the panorama, outputs the fit illumination parameters (sec. 5), and thoroughly evaluate its performance in sec. 6.

Throughout this paper, and following [40], will use the term *photo* to refer to a standard limited-field-of-view image as taken with a normal camera, and the term *panorama* to denote a 360-degree full-view panoramic image.

4. Dataset preparation

In this section, we detail the steps taken to augment the SUN360 dataset [40] with HDR data via the use of the Hošek-Wilkie sky model, and simultaneously extract lighting parameters that can be learned by the network. We first briefly describe the sky model parameterization, followed by the optimization strategy used to recover its parameters from a LDR panorama.

4.1. Sky lighting model

We employ the model proposed by Hošek and Wilkie [16], which has been shown [21] to more accurately represent skylight than the popular Preetham model [32]. The model has also been extended to include a solar radiance function [17], which we also exploit.

In its simplest form, the Hošek-Wilkie (HW) model expresses the spectral radiance L_λ of a lighting direction along the sky hemisphere $\mathbf{l} \in \Omega_{\text{sky}}$ as a function of several parameters:

$$L_\lambda(\mathbf{l}) = f_{\text{HW}}(\mathbf{l}, \lambda, t, \sigma_g, \mathbf{l}_s)\,, \tag{1}$$

where λ is the wavelength, t the atmospheric turbidity (a measure of the amount of aerosols in the air), σ_g the ground albedo, and $\mathbf{l}_s$ the sun position. Here, we fix $\sigma_g = 0.3$ (approximate average albedo of the Earth [12]).

From this spectral model, we obtain RGB values by rendering it at a discrete set wavelengths spanning the 360–700nm spectrum, convert to CIE XYZ via the CIE standard observer color matching functions, and finally convert again from XYZ to CIE RGB [16]. Referring to this conversion process as $f_{\text{RGB}}(\cdot)$, we express the RGB color $C_{\text{RGB}}(\mathbf{l})$ of a sky direction $\mathbf{l}$ as the following expression:

$$C_{\text{RGB}}(\mathbf{l}) = \omega f_{\text{RGB}}(\mathbf{l}, t, \mathbf{l}_s)\,. \tag{2}$$

In this equation, ω is a scale factor applied to all three color channels, which aims at estimating the (arbitrary and varying) exposure for each panorama. To generate a sky environment map from this model, we simply discretize the sky hemisphere Ω_{sky} into several directions (in this paper, we use the latitude-longitude format [33]), and render the RGB values with (2). Pixels which fall within $0.25°$ of the sun position $\mathbf{l}_s$ are rendered with the HW sun model [17] instead (converted to RGB as explained above).

Thus, we are left with three important parameters: the sun position $\mathbf{l}_s$, which indicate where the main directional light source is located in the sky, the exposure ω, and the turbidity t. The turbidity is of paramount importance as it controls the relative sun color (and intensity) with respect to that of the sky. As illustrated in fig. 2, a low turbidity indicates a clear sky with a very bright sun, and a high turbidity represents a sky closer that is closer to overcast situations, where the sun is much dimmer.

4.2. Optimization procedure

We now describe how the sky model parameters are estimated from a panorama in the SUN360 dataset. This procedure is carefully crafted to be robust to the extremely varied set of conditions encountered in the dataset which severely violates the linear relationship between sky radiance and pixel values such as: unknown camera response function and white-balance, manual post-processing by photographers and stitching artifacts.

Given a panorama P in latitude-longitude format and a set of pixels indices $p \in \mathcal{S}$ corresponding to sky pixels in P, we wish to obtain the sun position $\mathbf{l}_s$, exposure ω and sky turbidity t by minimizing the visible sky reconstruction error in a least-squares sense:

$$\begin{aligned} \mathbf{l}_s^*, \omega^*, t^* = &\underset{\mathbf{l}_s, \omega, t}{\arg\min} \sum_{p \in \Omega_s} \left(P(p)^\gamma - \omega f_{\text{RGB}}(\mathbf{l}_p, t, \mathbf{l}_s)\right)^2 \\ &\text{s.t.} \quad t \in [1, 10]\,, \end{aligned} \tag{3}$$

where $f_{\text{RGB}}(\cdot)$ is defined in (2) and $\mathbf{l}_p$ is the light direction corresponding to pixel $p \in \Omega_s$ (according to the latitude-longitude mapping). Here, we model the inverse response function of the camera with a simple gamma curve ($\gamma = 2.2$). Optimizing for gamma was found to be unstable and keeping it fixed yielded much more robust results.

Layer	Stride	Resolution
Input		320×240
conv7-64	2	160×120
conv5-128	2	80×60
conv3-256	2	40×30
conv3-256	1	40×30
conv3-256	2	20×15
conv3-256	1	20×15
conv3-256	2	10×8
FC-2048		
FC-160 LogSoftMax		FC-5 Linear
Output: sun position distribution $\mathbf{s}$		Output: sky and camera parameters $\mathbf{q}$

Figure 3. The proposed CNN architecture. After a series of 7 convolutional layers, a fully-connected layer segues to two heads: one for regressing the sun position, and another one for the sky and camera parameters. The ELU activation function [6] is used on all layers except the outputs.

We solve (3) in a 2-step procedure. First, the sun position $\mathbf{l}_s$ is estimated by finding the largest connected component of the sky above a threshold (98th percentile), and by computing its centroid. The sun position is fixed at this value, as it was determined that optimizing for its position at the next stage too often made the algorithm converge to undesirable local minima.

Second, the turbidity t is initialized to $\{1, 2, 3, ..., 10\}$ and (3) is optimized using the Trust Region Reflective algorithm (a variant of the Levenberg-Marquardt algorithm which supports bounds) for each of these starting points. The parameters resulting in the lowest error are kept as the final result. During the optimization loop, for the current value of t, ω^* is obtained through the closed-form solution

$$\omega^* = \frac{\sum_{p \in \mathcal{S}} P(p) f_{\text{RGB}}(\mathbf{l}_p, t, \mathbf{l}_s)}{\sum_{p \in \mathcal{S}} f_{\text{RGB}}(\mathbf{l}_p, t, \mathbf{l}_s)^2}. \quad (4)$$

Finally, the sky mask $\mathcal{S}$ is obtained with the sky segmentation method of [38], followed by a CRF refinement [23].

4.3. Validation of the optimization procedure

While our fitting procedure minimizes reconstruction errors w.r.t. the panorama pixel intensities, the radiometrically uncalibrated nature of this data means that these fits may not accurately represent the true lighting conditions. We validate the procedure in two ways. First, the sun position estimation algorithm is evaluated on 543 panoramic sky images from the Laval HDR sky database [25, 27], which contains ground truth sun position, and which we tonemapped and converted to JPG to simulate the conditions in SUN360. The median sun position estimation error of this algorithm is 4.59° (25th prct. = 1.96°, 75th prct. = 8.42°). Second, we ask a user to label 1,236 images from the SUN360 dataset, by indicating whether the estimated sky parameters agree with the scene visible in the panorama. To do so, we render a bunny model on a ground plane, and light it with the sky synthesized by the physical model. We then ask the user to indicate whether the bunny is lit similarly to the other elements present in the scene. In all, 65.6% of the images were deemed to be a successful fit, which is testament to the challenging imaging conditions present in the dataset.

5. Learning to predict outdoor lighting

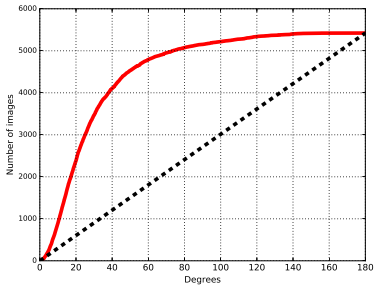
5.1. Dataset organization

To train the CNN, we first apply the optimization procedure from sec. 4.2 to 38,814 high resolution outdoor panoramas in the SUN360 [40] database. We then extract 7 photos from each panorama using a standard pinhole camera model and randomly sampling its parameters: its elevation with respect to the horizon in $[-20°, 20°]$, azimuth in $[-180°, 180°]$, and vertical field of view in $[35°, 68°]$. The resulting photos are bilinearly interpolated from the panorama to a resolution 320×240, and used directly to train the CNN described in the next section. This results in a dataset of 271,698 pairs of photos and their corresponding lighting parameters, which is split into (261,288 / 1,751 / 8,659) subsets for (train / validation / test). These splits were computed on the panoramas to ensure that photos taken from the same panorama do not end up in training and test. Example panoramas and corresponding photos are shown in fig. 6.

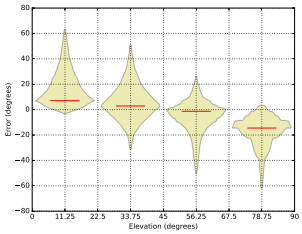
5.2. CNN architecture

We adopt a standard feed-forward convolutional neural network to learn the relationship between the input image I and the lighting parameters. As shown in fig. 3, its architecture is composed of 7 convolutional layers, followed by a fully-connected layer. It then splits into two separate heads: one for estimating the sun position (left in fig. 3), and one for the sky and camera parameters (right in fig. 3).

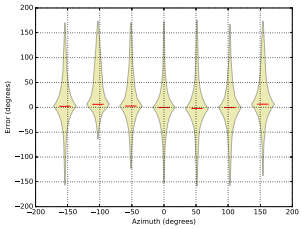
The sun position head outputs a probability distribution over the likely sun positions $\mathbf{s}$ by discretizing the sky hemisphere into 160 bins (5 for elevation, 32 for azimuth), and outputs a value for each of these bins. This was also done in [26]. As opposed to regressing the sun position directly, this has the advantage of indicating other regions believed to be likely sun positions in the prediction, as illustrated in fig. 6 below. The parameters head directly regresses a 4-vector of parameters $\mathbf{q}$: 2 for the sky (ω, t), and 2 for the camera (elevation and field of view). The ELU activation function [6] and batch normalization [18] are used at the output of every layer.



(a) Angular error



(b) Elevation error



(c) Azimuthal error

Figure 4. Quantitative evaluation of sun position estimation on all 8659 images in the SUN360 test set. (a) The cumulative distribution function of the angular error on the sun position. The estimation error as function of the sun elevation (b) and (c) azimuth relative to the camera (0° means the sun is in front of the camera). The last two figures are displayed as "box-percentile plots" [10], where the envelope of each bin represents the percentile and the median is shown as a red bar.

5.3. Training details

We define the loss to be optimized as the sum of two losses, one for each head:

$$\mathcal{L}(\mathbf{s}^*, \mathbf{q}^*, \mathbf{s}, \mathbf{q}) = \mathcal{L}(\mathbf{s}^*, \mathbf{s}) + \beta \mathcal{L}(\mathbf{q}^*, \mathbf{q}) \,, \tag{5}$$

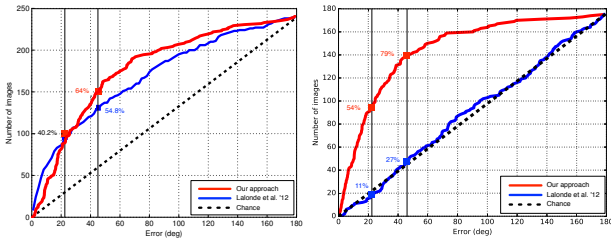
where $\beta = 160$ to compensate for the number of bins in $\mathbf{s}$. The target sun position $\mathbf{s}^*$ is computed for each bin $\mathbf{s}_j$ as

$$\mathbf{s}_j^* = \exp(\kappa \mathbf{l}_s^{*\mathsf{T}} \mathbf{l}_j) \,, \tag{6}$$

and normalized so that $\sum_j \mathbf{s}_j^* = 1$. The equation in (6) represents a von Mises-Fisher distribution [1] centered about the ground truth sun position $\mathbf{l}_s$. Since the network must predict a confident value around the sun position, we set $\kappa = 80$. The target parameters $\mathbf{q}^*$ are simply the ground truth sky and camera parameters.

We use a MSE loss for $\mathcal{L}(\mathbf{q}^*, \mathbf{q})$, and a Kullback-Leibler (KL) divergence loss for the sun position $\mathcal{L}(\mathbf{s}^*, \mathbf{s})$. Using the KL divergence is needed because we wish the network to learn a *distribution* over the sun positions, rather than the most likely position.

The loss in (5) is minimized via stochastic gradient descent using the Adam optimizer [22] with an initial learning rate of $\eta = 0.01$. Training is done on mini-batches of 128 exemplars, and regularized via early stopping. The process typically converges in around 7–8 epochs, because our CNN is not as deep as most modern feed-forward CNN used in vision. Moreover, the high initial learning rate used combined with our large dataset further helps in reducing the number of epochs required for training.



(a) Dataset from [26] (b) Subset of SUN360 test set

Figure 5. Comparison with the method of Lalonde et al. [26] showing the cumulative sun azimuth estimation error on (a) their original dataset, and (b) a 176-image subset from the SUN360 test set. (a) While our method has similar error in an octant (less than $22.5°$), the precision in a quadrant (less than $45°$) significantly improves by approximately 10%. (b) The 176-images SUN360 test subset contains much more challenging images where methods based on the detection of explicit cues (as in [26]) fail. Our deep learning based approach remains robust and achieves high performance on both datasets.

6. Evaluation

We evaluate the performance of the CNN at predicting the HDR sky environment map from a single image in a variety of ways. First, we present how well the network does at estimating the illumination parameters on the SUN360 dataset. We then show how virtual objects relit by the *estimated* environment maps differ from their renders obtained with the ground truth parametric model, still on the SUN360. Finally, we acquired a small set of HDR outdoor panoramas, and compare our relighting results with those obtained with actual HDR environment maps.

6.1. Illumination parameters on SUN360

Sun position We begin by evaluating the performance of the CNN at predicting the sun position from a single input image. Fig. 4 shows the quantitative performance at this task using three plots: the cumulative distribution function of sun angular estimation error, and detailed error histograms for each of the elevation and azimuth independently. We observe that 80% of the test images have error less than 45°. Fig. 4-(b) indicates that the network tends to underestimate the sun elevation in high elevation cases. This may be attributable to a lack of such occurrences in the training dataset—high sun elevations only occur between the tropics, and at specific times of year because of the Earth's tilted rotation axis. Fig. 4-(c) shows that the CNN is not biased towards an azimuth position, and is robust across

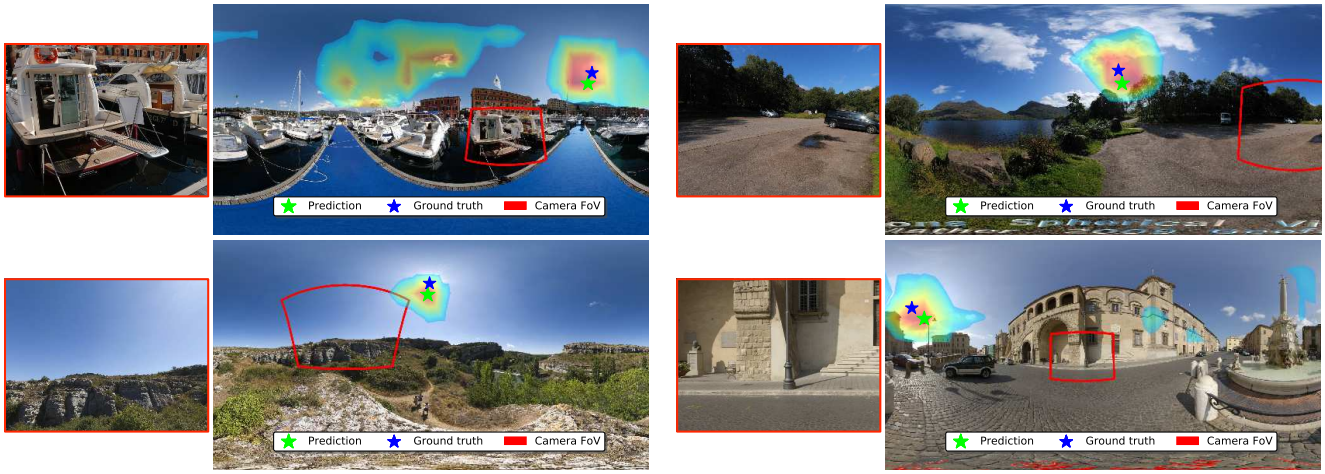


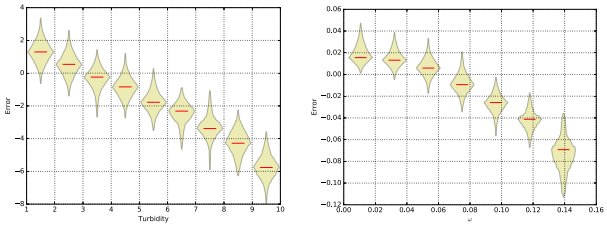
Figure 6. Examples of sun position estimation from a single outdoor image. For each example, the input image is shown on the left, and its corresponding location in the panorama is shown with a red outline. The color overlay displays the probability distribution of the sun position output by the neural network. A green star marks the most likely sun position estimated by the neural network, while a blue star marks the ground truth position.

the entire range. Fig. 6 shows examples of our sun position predictions overlayed over the panoramas that the test images were cropped from. Note that our method is able to accurately predict the sun direction across a wide range of scenes, field of views, and layouts.

We quantitatively compare our approach to that of [26] at the task of sun azimuth estimation from a single image. Results are reported in fig. 5. First, fig. 5-(a) shows a comparison of both approaches on the 239-image dataset of [26]. While our method has similar error in an octant (less than 22.5°), the precision in a quadrant (less than 45°) is significantly improved (by approximately 10%) by our CNN-based approach. Fig. 5-(b) shows the same comparison on a 176-image subset of the SUN360 test set used in this paper. In this case, the approach of Lalonde et al. [26] fails while the CNN reports robust performance, comparable to fig. 5-(a). This is probably due to the fact that the SUN360 test set contains much more challenging images that are often devoid of strong, explicit illumination cues. These cues, which are expressly relied upon by [26], are critical to the success of such methods.

Turbidity and exposure We evaluate the regression performance for the turbidity t and exposure ω lighting parameters on the SUN360 test set, and report the results in fig. 7. Overall, the network tends to favor low turbidity estimates of the sky (as the dataset contains a majority of such examples). In addition, the network successfully estimates low exposure values, but has a tendency to underestimate images with high exposures.

Camera parameters A detailed performance analysis is available in the supplementary material. In a nutshell, the CNN achieves error of less than 7° for the elevation and 11° in field of view for 80% of the test images.



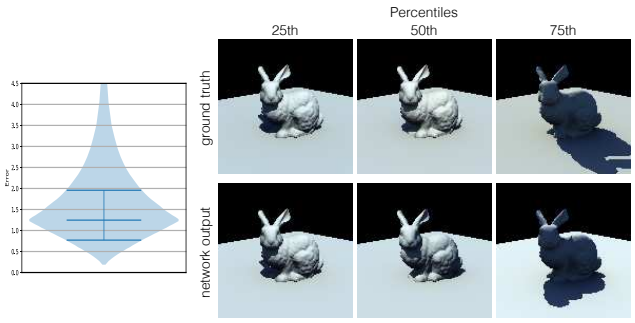
(a) Turbidity t (b) Exposure ω

Figure 7. Quantitative evaluation for turbidity t and exposure ω. The distribution of errors are displayed as "box-percentile" plots (see fig. 4). The CNN tends to favor clear skies (low turbidity), and has higher errors when the exposure is high.

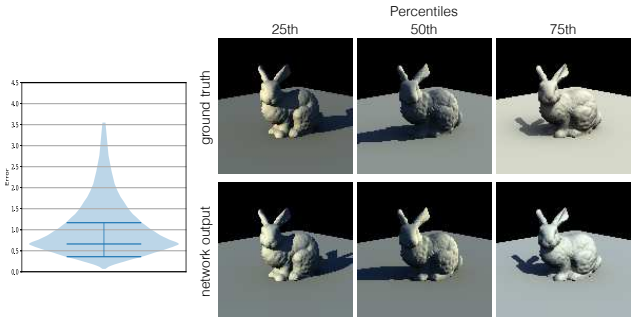
6.2. Relighting on SUN360

Another way of evaluating the performance is by comparing the appearance of a Lambertian 3D model rendered with the estimated lighting, with that of the same model lit by the ground truth. Fig. 8 provides such a comparison, by showing three different error metrics computed on renderings obtained on our test set. The error metrics are the (a) RMSE, (b) scale-invariant RMSE, and (c) per-color scale-invariant RMSE. The scale-invariant versions of RMSE are defined similarly to Grosse et al. [13], except that the scale factor is computed on the entire image (instead of locally as in [13]). The "per-color" variant computes a different scale factor for each color channel to mitigate differences in white balance. The black background in the renders is masked out before computing the metrics.

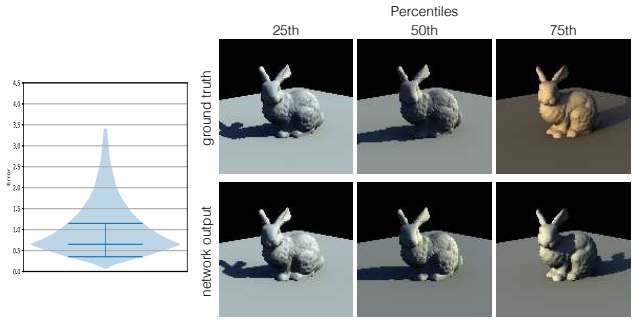
To give a sense of what those numbers mean qualitatively, fig. 8 also provides examples corresponding to each of the (25, 50, 75)th error percentiles. Even examples in



(a) RMSE



(b) Scale-invariant RMSE



(c) Per-color scale-invariant RMSE

Figure 8. Quantitative relighting comparison with the ground truth lighting parameters on the SUN360 dataset. We compute three types of error metrics: (a) RMSE, (b) scale-invariant RMSE [13], and (c) per-color scale-invariant RMSE. The plots on the left shows the distribution of errors with the median, 25th and 75th percentiles identified with blue bars. For each measure, examples corresponding to particular error levels are shown to give a qualitative sense of performance. Renders obtained with the ground truth (estimated) lighting parameters are shown in the top (bottom) row.

the 75th error percentile look good qualitatively. Slight differences in the sun direction and the overall color can be observed, but they still lie within reasonable limits.

Fig. 9 shows examples of virtual objects inserted into images after being rendered with our estimated HDR illumination. As these examples show, our technique is able to infer plausible illumination conditions ranging from sunny to overcast, and high noon to dawn/dusk, resulting in natural-looking composite images. Fig. 10 shows that the camera elevation estimated from the CNN can be used within the rendering pipeline to automatically rotate the virtual camera

Figure 9. Virtual object insertion with automated lighting estimation. From a single image, the CNN predicted a full HDR sky map, which is used to render an object into the image. No additional steps are required. More results on automated object insertion are available in the supplementary materials.

Estimated elevation: -9° Estimated elevation: 3.5°

Figure 10. Virtual object insertion with automated lighting and camera elevation estimation. The two images are taken at the same location with the camera pointing downwards (left) and upwards (right). The elevation of the virtual camera used to render the bunny model is set to the value predicted by the CNN, resulting in a bunny which realistically rests on the ground.

used to render the object. In these results, a simple ground plane is used to model the interactions between the virtual object and its environment, and the object is placed manually at a fixed distance in front of the camera.

6.3. Validation with HDR panoramas

To further validate our approach, we captured a small dataset of 19 unsaturated, outdoor HDR panoramas. To properly expose the extreme dynamic range of outdoor lighting, we follow the approach proposed by Stumpfel et al. [36]. We captured 7 bracketed exposures ranging from 1/8000 to 8 seconds at f/16, using a Canon EOS 5D Mark III camera installed on a tripod, and fitted with a Sigma EXDG 8mm fisheye lens. A 3.0 ND filter was installed behind the lens, necessary to accurately measure the sun intensity. The exposures were stored as 14-bit RAW images at full resolution. The process was repeated at 6 azimuth angles by increments of 60° to cover the entire 360° panorama. The resulting 42 images were fused using the PTGui commer-

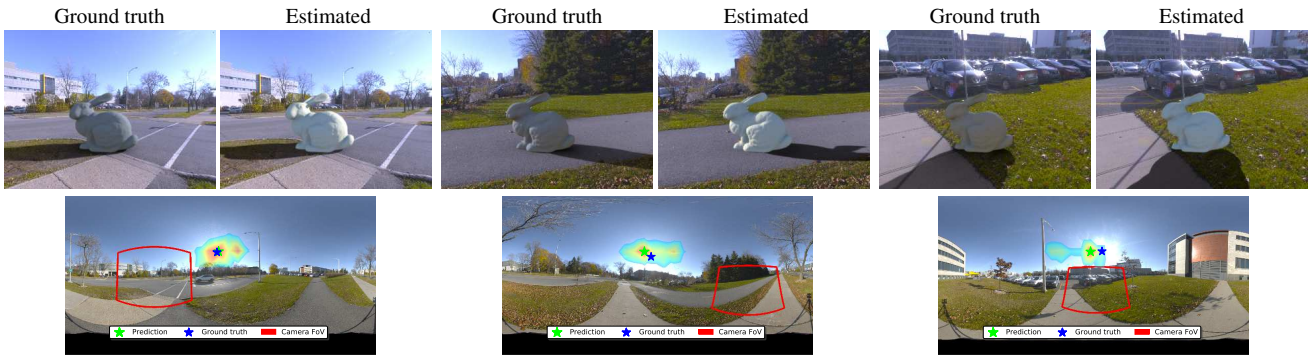


Figure 11. Object relighting comparison with ground truth illumination conditions on captured HDR panoramas. For each example, the top row shows (left) a bunny model relit by the ground truth HDR illumination conditions captured in situ; (right) the same bunny model, relit by the illumination conditions estimated by the CNN solely from the background image, completely automatically. No further adjustment (e.g. overall brightness, saturation, etc.) was performed. The bottom row shows the original environment map, field of view of the camera (in red), and the distribution on sun position estimation (as in fig. 6). Please see additional results on our project page.

cial stitching software. To facilitate the capture process, the camera was mounted on a programmable robotic tripod head, allowing for repeatable and precise capture.

To validate the approach, we extract limited field of view photos from the HDR panoramas and save them as JPEG files. The CNN is then applied to the input photos to predict their illumination conditions. Then, we compare relighting results obtained by rendering a bunny model with: 1) the HDR panorama itself, which represents the ground truth lighting conditions; and 2) the estimated lighting conditions. Example results are shown in fig. 11. While we note that the exposure ω is slightly overestimated (resulting in a render that is brighter than the ground truth), the relit bunny appears quite realistic.

7. Discussion

In this paper, we propose what we believe to be the first end-to-end approach to automatically predict full HDR lighting models from a single outdoor LDR image of a general scene, which can readily be used for image-based lighting. Our key idea is to train a deep CNN on pairs of photos and panoramas in the SUN360 database, which we "augment" with HDR information via a physics-based model of the sky. We show that our method significantly outperforms previous work, and that it can be used to realistically insert virtual objects into photos.

Despite offering state-of-the-art performance, our method still suffers from some limitations. First, the Hošek-Wilkie sky model provides accurate representational accuracy for clear skies, but its accuracy degrades when cloud cover increases as the turbidity t is not enough to model completely overcast situations as accurately as for clear skies. Optimizing its parameters on overcast panoramas often underestimates the turbidity, resulting in a bias toward low turbidity in the CNN. We are currently investigating ways of mitigating this issue by combining the HW model with another sky model, better-suited for overcast skies. Another limitation is that the resulting environment map models the sky hemisphere only. While this does not affect diffuse objects such as the bunny model used in this paper, it would be more problematic for rendering specular materials, as none of the scene texture would be reflected off its surface. It is likely that simple adjustments such as [20] could be helpful in making those renders more realistic.

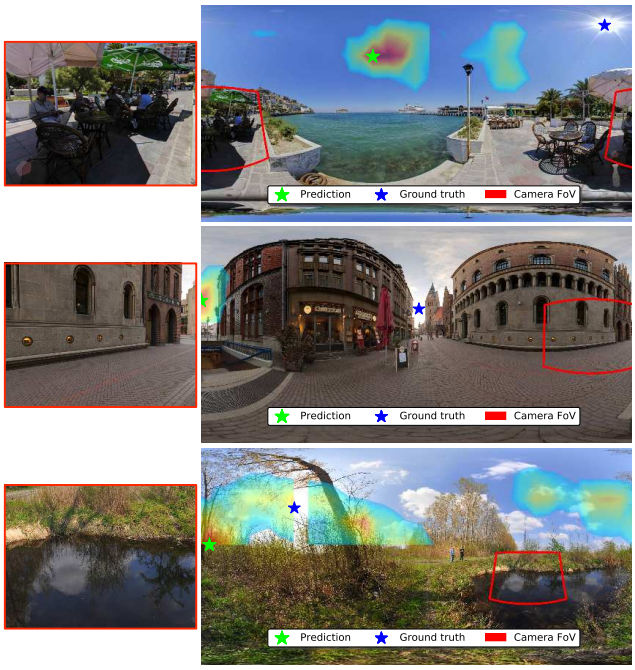


Figure 12. Typical failure cases of sun position estimation from a single outdoor image. See fig. 6 for an explanation of the annotations. Failure cases occur when illumination cues are mixed with complex geometry (top), absent from the image (middle), or in the presence of mirror-like surfaces (bottom).

References

[1] A. Banerjee, I. S. Dhillon, J. Ghosh, and S. Sra. Clustering on the unit hypersphere using von Mises-Fisher distributions. *Journal of Machine Learning Research*, 6:1345–1382, 2005. 5

[2] A. Bansal, B. Russell, and A. Gupta. Marr revisited: 2D-3D model alignment via surface normal prediction. *CVPR*, 2016. 2

[3] J. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1670–1687, 2013. 1, 2

[4] J. T. Barron and J. Malik. Intrinsic scene properties from a single rgb-d image. *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 2

[5] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database. *IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 2

[6] D.-A. Clevert, T. Unterthiner, and S. Hochreiter. Fast and accurate deep network learning by exponential linear units (ELUs). In *International Conference on Learning Representations*, 2016. 4

[7] P. Debevec. Rendering synthetic objects into real scenes : Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proceedings of ACM SIGGRAPH*, 1998. 2

[8] S. Duchêne, C. Riant, G. Chaurasia, J. L. Moreno, P.-Y. Laffont, S. Popov, A. Bousseau, and G. Drettakis. Multiview intrinsic images of outdoors scenes with an application to relighting. *ACM Trans. Graph.*, 34(5):164:1–164:16, Nov. 2015. 2

[9] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. *International Conference on Computer Vision*, 2015. 2

[10] W. W. Esty and J. D. Banfield. The box-percentile plot. *Journal Of Statistical Software*, 8:1–14, 2003. 5

[11] S. Georgoulis, K. Rematas, T. Ritschel, M. Fritz, L. Van Gool, and T. Tuytelaars. Delight-net: Decomposing reflectance maps into specular materials and natural illumination. *arXiv preprint arXiv:1603.08240*, 2016. 2

[12] P. R. Goode, J. Qiu, V. Yurchyshyn, J. Hickey, M.-C. Chu, E. Kolbe, C. T. Brown, and S. E. Koonin. Earthshine observations of the earth's reflectance. *Geophysical Research Letters*, 28(9):1671–1674, 2001. 3

[13] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *IEEE International Conference on Computer Vision*, 2009. 6, 7

[14] T. Haber, C. Fuchs, P. Bekaer, H.-P. Seidel, M. Goesele, and H. Lensch. Relighting objects from image collections. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 2

[15] D. Hauagge, S. Wehrwein, P. Upchurch, K. Bala, and N. Snavely. Reasoning about photo collections using models of outdoor illumination. In *British Machine Vision Conference*, 2014. 2

[16] L. Hošek and A. Wilkie. An analytic model for full spectral sky-dome radiance. *ACM Transactions on Graphics*, 31(4):1–9, 2012. 2, 3

[17] L. Hošek and A. Wilkie. Adding a solar-radiance function to the hosek-wilkie skylight model. *IEEE Computer Graphics and Applications*, 33(3):44–52, may 2013. 2, 3

[18] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Journal of Machine Learning Research*, 37, 2015. 4

[19] K. Karsch, K. Sunkavalli, S. Hadap, N. Carr, H. Jin, R. Fonte, M. Sittig, and D. Forsyth. Automatic scene inference for 3D object compositing. *ACM Trans. Graph.*, 33(3):32:1–32:15, June 2014. 2

[20] E. A. Khan, E. Reinhard, R. W. Fleming, and H. H. Bülthoff. Image-based material editing. *ACM Transactions on Graphics*, 25(3):654, 2006. 8

[21] J. T. Kider, D. Knowlton, J. Newlin, Y. K. Li, and D. P. Greenberg. A framework for the experimental comparison of solar and skydome illumination. *ACM Transactions on Graphics*, 33(6):1–12, nov 2014. 3

[22] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, pages 1–15, 2015. 5

[23] P. Krähenbühl and V. Koltun. Efficient inference in fully connected CRFs with gaussian edge potentials. In *Neural Information Processing Systems*, 2012. 4

[24] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In *NIPS*, pages 2539–2547. 2015. 2

[25] J.-F. Lalonde, L.-P. Asselin, J. Becirovski, Y. Hold-Geoffroy, M. Garon, M.-A. Gardner, and J. Zhang. The Laval HDR sky database. http://www.hdrdb.com, 2016. 4

[26] J. F. Lalonde, A. A. Efros, and S. G. Narasimhan. Estimating the natural illumination conditions from a single outdoor image. *International Journal of Computer Vision*, 98(2):123–145, 2012. 1, 2, 4, 5, 6

[27] J.-F. Lalonde and I. Matthews. Lighting estimation in outdoor image collections. In *International Conference on 3D Vision*, 2014. 2, 4

[28] J.-F. Lalonde, S. G. Narasimhan, and A. A. Efros. What do the sun and the sky tell us about the camera? *International Journal on Computer Vision*, 88(1):24–51, May 2010. 2

[29] S. Lombardi and K. Nishino. Reflectance and illumination recovery in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1):129–141, 2016. 1, 2

[30] W.-C. Ma, S. Wang, M. A. Brubaker, S. Fidler, and R. Urtasun. Find your Way by Observing the Sun and Other Semantic Cues. *IEEE Conference on Robotics and Automation (ICRA)*, 2017. 2

[31] R. Perez, R. Seals, and J. Michalsky. All-weather model for sky luminance distribution - Preliminary configuration and validation. *Solar Energy*, 50(3):235–245, Mar. 1993. 2

[32] A. J. Preetham, P. Shirley, and B. Smits. A practical analytic model for daylight. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques - SIGGRAPH*, 1999. 2, 3

[33] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski. *High Dynamic Range Imaging*. Morgan Kaufman, 2 edition, 2010. 3

[34] K. Rematas, T. Ritschel, M. Fritz, E. Gavves, and T. Tuytelaars. Deep reflectance maps. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1, 2

[35] Q. Shan, R. Adams, B. Curless, Y. Furukawa, and S. M. Seitz. The visual turing test for scene reconstruction. In *3DV*, 2015. 2

[36] J. Stumpfel, A. Jones, A. Wenger, C. Tchou, T. Hawkins, and P. Debevec. Direct HDR capture of the sun and sky. In *Proceedings of ACM AFRIGRAPH*, 2004. 7

[37] K. Sunkavalli, F. Romeiro, W. Matusik, T. Zickler, and H. Pfister. What do color changes reveal about an outdoor scene? In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 2

[38] Y.-H. Tsai, X. Shen, Z. Lin, K. Sunkavalli, and M.-H. Yang. Sky is not the limit: Semantic-aware sky replacement. *ACM Transactions on Graphics (SIGGRAPH 2016)*, 35(4):149:1–149:11, July 2016. 4

[39] S. Wehrwein, K. Bala, and N. Snavely. Shadow detection and sun direction in photo collections. In *International Conference on 3D Vision*, 2015. 2

[40] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba. Recognizing scene viewpoint using panoramic place representation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 2, 3, 4

[41] Y. Zhang, J. Xiao, J. Hays, and P. Tan. Framebreak: Dramatic image extrapolation by guided shift-maps. In *IEEE Conference on Computer Vision and Pattern Recognition Recognition*, pages 1171–1178, 2013. 2

[42] T. Zhou, P. Krähenbühl, and A. A. Efros. Learning data-driven reflectance priors for intrinsic image decomposition. *ICCV*, 2015. 1, 2