

All-Weather Deep Outdoor Lighting Estimation

Jinsong Zhang^{1*}, Kalyan Sunkavalli[†], Yannick Hold-Geoffroy[†], Sunil Hadap[†],
 Jonathan Eisenman[†], Jean-François Lalonde^{*}
 Université Laval^{*}, Adobe Research[†]

jinsong.zhang.1@ulaval.ca, {sunkaval,holdgeof,hadap,eisenman}@adobe.com, jflalonde@gel.ulaval.ca
<http://www.jflalonde.ca/projects/allweather>

Abstract

We present a neural network that predicts HDR outdoor illumination from a single LDR image. At the heart of our work is a method to accurately learn HDR lighting from LDR panoramas under any weather condition. We achieve this by training another CNN (on a combination of synthetic and real images) to take as input an LDR panorama, and regress the parameters of the Lalonde-Matthews outdoor illumination model [17]. This model is trained such that it a) reconstructs the appearance of the sky, and b) renders the appearance of objects lit by this illumination. We use this network to label a large-scale dataset of LDR panoramas with lighting parameters and use them to train our single image outdoor lighting estimation network. We demonstrate, via extensive experiments, that both our panorama and single image networks outperform the state of the art, and unlike prior work, are able to handle weather conditions ranging from fully sunny to overcast skies.

1. Introduction

Estimating outdoor illumination is critical for a number of tasks such as outdoor scene understanding, image editing, and augmented reality. However, images are formed by illumination interacting with other scene properties like geometry and surface reflectance, thus making the inversion of this process to recover lighting a highly ill-posed problem.

Prior work on this problem has used heuristics to map image features to lighting [16]. Instead of using hand-crafted features, a recent approach proposed learning the mapping from image appearance to outdoor illumination [9] using a deep neural network. In particular, they propose a non-linear optimization scheme to fit the parameters of the Hošek-Wilkie (HW) HDR sky model [10, 11] to SUN360—a large dataset of outdoor, low dynamic range (LDR) panoramas [24]. From this set of panoramas with (now labeled) parameters, they extract limited field of view crops, and train a CNN to regress the HW parameters from a single crop.

¹ Research partly done when Jinsong Zhang was an intern at Adobe Research.

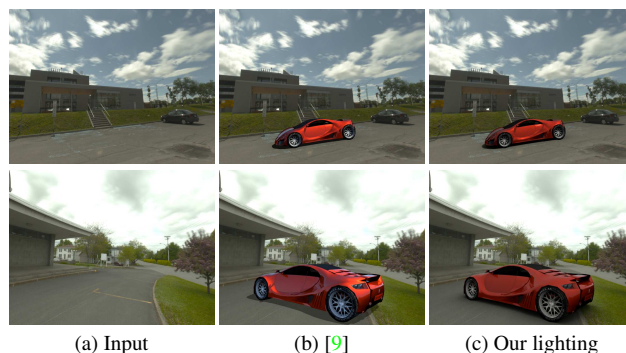


Figure 1. We estimate HDR outdoor lighting from a single image (a) and use it to render a car into the scene. The state-of-the-art method of [9] (b) produces sunny estimates for both clear (top) and overcast scenes (bottom). Our method (c) produces accurate estimates for both conditions leading to more realistic composites.

The accuracy of [9] thus rests on a) the ability of the HW model to represent outdoor illumination, and b) the ability to reliably fit the HW model to sky pixels in an outdoor panorama. Unfortunately, both of these steps have limitations. Indeed, the HW sky model was designed to accurately represent a *subset* of possible weather conditions, specifically, completely clear skies with varying amounts of turbidity (amount of atmospheric aerosol) [13]. Moreover, the optimization scheme to fit the HW model to LDR panoramas can be sensitive to issues like arbitrary camera processing and outliers like patches of clouds. This leads to a clear bias towards sunny skies in their results (see fig. 1(b)).

Other recent approaches to estimating outdoor illumination have eschewed parametric models in favor of completely data-driven models. Zhang et al. [26] learn to hallucinate HDR outdoor environment maps from LDR panoramas by training an encoder-decoder on synthetic data. Calian et al. [2] learn a data-driven sky model by training an auto-encoder on HDR sky panoramas [15, 17], and use this representation in an inverse rendering framework to recover outdoor lighting from a single image of a face. While these learned models can better approximate outdoor lighting conditions, they conflate many of the “intuitive” sky parameters

such as the sun position, sky color, etc. into an opaque representation that can only be decoded by a network.

In this work, we propose a method to robustly estimate a wide variety of outdoor lighting conditions from a single image. Unlike [9], we use the Lalonde-Matthews (LM) sky model [17] that can represent a much wider set of lighting conditions, ranging from completely overcast to fully sunny. While more expressive, the sky and sun illumination components in this model are uncorrelated; as a result, it is not possible to recover HDR sun illumination by fitting an illumination model to LDR sky pixels, as is done by [9]. Thus, our main contribution is a novel method to *learn to label LDR panoramas with HDR lighting parameters*. Specifically, we train a network—*PanoNet*—to take as input an LDR panorama, and regress the parameters of the LM model. We train PanoNet with a combination of synthetic and real data. Moreover, because no sky model will exactly reproduce real sky pixel intensities, we show that merely learning to match *sky appearance* is not sufficient for this task. Instead, we propose a novel render loss that matches the appearance of a *rendered scene under the ground truth and predicted lighting*. PanoNet produces HDR lighting estimates that are significantly better than previous work. We use PanoNet to label the SUN360 dataset with lighting parameters, and similar to [9], train *CropNet*—a network that regresses the lighting labels from a single crop image. Through extensive experiments (and as can be seen in fig. 1), we demonstrate that both PanoNet and CropNet significantly outperform the state-of-the-art, both qualitatively and quantitatively.

2. Related work

A wide array of lighting estimation methods have been presented in the literature. In this section, we focus on outdoor lighting modeling and estimation that is most related to this work.

Outdoor lighting modeling Modeling the solar and sky dome illumination has been extensively studied in atmospheric science, physics, and computer graphics. The Perez All-Weather model [20] was first introduced as an improvement over the previous CIE Standard Clear Sky model, and modeled weather variations using 5 parameters. Preetham et al. [21] simplified this to a model with a single weather parameter—atmospheric turbidity. Hošek and Wilkie introduced an improvement over the Preetham model, which resulted in both a sky dome [10] and solar disc [11] analytical models. See [13] for a comparison of how these sky models approximate clear skies on a rich outdoor dataset provided by the authors. In addition, Lalonde and Matthews [17] proposed an empirical model for HDR skies, which they show better approximate captured skies (in RGB) under a wide variety of illumination conditions. They subsequently employ their sky model for estimating outdoor lighting from

outdoor image collections. In this paper, we exploit that model (which we dub the “LM” model) and show that its parameters can be predicted from a single image by a CNN.

Outdoor lighting estimation Lighting estimation from a single, generic outdoor scene was first proposed by Lalonde et al. [16]; they use a probabilistic combination of multiple image cues—such as cast shadows, shading, and sky appearance variation—to predict lighting. Karsch et al. [12] match the input image to a large dataset of panoramas [24], and transfer the panorama lighting (obtained through a light classifier) to the image. However, the matching metric, that relies on image features, may not yield results that have consistent lighting. Other approaches rely on known geometry [18] and/or strong priors on geometry and surface reflectance [1].

Deep learning for lighting estimation Recent approaches have applied deep learning methods for lighting estimation. Rematas et al. [22] learn to infer a reflectance map (i.e., the convolution of incident illumination with surface reflectance) from a single image of an object. Subsequently, Georgoulis et al. [6] factor reflectance maps into lighting and material properties [5]. Closely related to our work, Hold-Geoffroy et al. [9] model outdoor lighting with the parametric Hošek-Wilkie sky model [10, 11], and learn to estimate its parameters from a single image. As mentioned in the introduction, we take inspiration from this work and significantly improve upon it proposing a novel learning-based approach to robustly annotate LDR panoramas with different weather conditions with the parameters of the LM illumination model. This is closely related to Zhang et al. [26] who learn to map LDR panoramas to HDR environment maps via an encoder-decoder network. Similarly, Calian et al. [2] (as well as the concurrent work of Hold-Geoffroy et al. [8]) employ a deep autoencoder to learn a data-driven illumination model. They use this learned model to estimate lighting from a face image via a multi-step non-linear optimization approach over the space of face albedo and sky parameters, that is time-consuming and prone to local minima. In contrast to the high-dimensional environment map and the learned auto-encoder representations, we use a compact and intuitive sky model—the aforementioned LM model. This allows us to easily annotate a large-scale LDR panorama dataset [24] with lighting parameters and subsequently infer lighting from a single image of a generic outdoor scene in an end-to-end framework. Cheng et al. [3] estimate lighting from the front and back camera of a mobile phone. However, they represent lighting using low-frequency spherical harmonics (SH), which, as demonstrated in [2], does not appropriately model outdoor lighting.

3. Brief review of the LM sky model

In this paper, we make use of the Lalonde-Matthews (LM) sky model [17]. This is a parameteric sun and sky model

which, when fit to HDR panoramas, was determined to better approximate outdoor lighting than other, physically-based models. We now briefly summarize its form and parameters for completeness, but refer the reader to [17] for more details.

The LM hemispherical illumination model can concisely be written as the sum of sun and sky components:

$$f_{\text{LM}}(\mathbf{l}; \mathbf{q}_{\text{LM}}) = f_{\text{sun}}(\mathbf{l}; \mathbf{q}_{\text{sun}}, \mathbf{l}_{\text{sun}}) + f_{\text{sky}}(\mathbf{l}; \mathbf{q}_{\text{sky}}, \mathbf{l}_{\text{sun}}), \quad (1)$$

where $\mathbf{l}_{\text{sun}} = [\theta_{\text{sun}}, \varphi_{\text{sun}}]$ is the sun position in spherical coordinates, and the \mathbf{q}_* are component-specific parameters.

The sky component $f_{\text{sky}}(\mathbf{l})$ in eq. (1) is simply the Preetham sky model [21] $f_{\text{P}}(\cdot)$, multiplied channel-wise with an average sky color $\mathbf{w}_{\text{sky}} \in \mathbb{R}^3$:

$$f_{\text{sky}}(\mathbf{l}; \mathbf{q}_{\text{sky}}, \mathbf{l}_{\text{sun}}) = \mathbf{w}_{\text{sky}} f_{\text{P}}(\theta_{\text{sun}}, \gamma_{\text{sun}}, t), \quad (2)$$

where γ_{sun} is the angle between sky element \mathbf{l} and the sun position \mathbf{l}_{sun} , and t is the sky turbidity. The sun component $f_{\text{sun}}(\mathbf{l})$ in eq. (1) is defined as

$$f_{\text{sun}}(\mathbf{l}; \mathbf{q}_{\text{sun}}, \mathbf{l}_{\text{sun}}) = \mathbf{w}_{\text{sun}} \exp(-\beta \exp(-\kappa/\gamma_{\text{sun}})), \quad (3)$$

where (β, κ) are two parameters controlling the shape of the sun, and $\mathbf{w}_{\text{sun}} \in \mathbb{R}^3$ is the mean sun color. In short, the LM sky model has the following 11 parameters:

$$\mathbf{q}_{\text{LM}} = \{ \mathbf{w}_{\text{sky}}, t, \mathbf{w}_{\text{sun}}, \beta, \kappa, \mathbf{l}_{\text{sun}} \}. \quad (4)$$

4. Estimating HDR parametric lighting from LDR panoramas

One of the main advantages of using the Hošek-Wilkie model is that only its *sky* parameters (sun position, turbidity and exposure) [10] can be fit to an LDR panorama and, because its sun and sky components are linked via the turbidity parameter, an HDR *sun* model can be extrapolated [11]. This is a practical way of estimating the high dynamic range of the sun given a saturated LDR panorama [9]. In this paper, we use the LM model which, because it uses *independent* parameters for the sun and the sky (c.f. sec. 3), is more expressive than the HW model. However, this also means its HDR sun parameters cannot be fit directly to LDR panoramas. In this section, we therefore train a CNN that learns to predict the LM sky parameters from a single LDR panorama.

4.1. Architecture of PanoNet

To regress the parameters of the LM sky model from a single LDR panorama, we take inspiration from the work of [26], who use an autoencoder with skip-links (similar to the well-known U-net architecture [23]) to regress an HDR panorama from LDR. For simplicity, they used the equirectangular format, and assumed the panorama to be rotated such that the sun is in the center. In addition, they also had another path from the latent vector \mathbf{z} to two fully-connected layers that estimate the sun elevation, which was

found to add robustness. Finally, they had a third path from \mathbf{z} to an unsupervised domain adaptation branch that helped the network generalize to real data.

Starting from this base architecture, we add another path from \mathbf{z} , this time to predict the parameters of the LM sky model. More precisely, we ask the network to learn the sun and sky intensities (\mathbf{w}_{sun} and \mathbf{w}_{sky} respectively), the sun shape parameters β and κ , and the sky turbidity t . The new path has a structure of two consecutive FC layers with size of 512 and 256 neurons, the output layer has 9 neurons corresponding to the LM sky parameters. The resulting CNN is illustrated in fig. 2.

4.2. Loss functions

Several loss functions are used to train the PanoNet CNN. First, we use the same loss functions as in [26], namely:

$$\begin{aligned} \mathcal{L}_{\text{pano}} &= \|\mathbf{P}^* - \hat{\mathbf{P}}\|_1 \\ \mathcal{L}_{\theta} &= \|\theta_{\text{sun}}^* - \hat{\theta}_{\text{sun}}\|_2 \\ \mathcal{L}_{\text{render}} &= \|\mathbf{TP}^* - \mathbf{T}\hat{\mathbf{P}}\|_2, \\ \mathcal{L}_{\text{da}} &= -\sum_{i=1} a_i^* \log \hat{a}_i \end{aligned} \quad (5)$$

where symbols $(*)$ and $(\hat{\cdot})$ denote the ground truth and the network output respectively. From top to bottom, the various outputs of the CNN are compared to their ground truth counterparts: on the HDR panorama \mathbf{P} ($\mathcal{L}_{\text{pano}}$), the sun elevation θ_{sun} (\mathcal{L}_{θ}), a rendered image of a synthetic scene—a diffuse sphere on a plane ($\mathcal{L}_{\text{render}}$). The domain adaptation branch is trained with cross-entropy loss (\mathcal{L}_{da}). We make the in-network rendering of the synthetic scene fast by multiplying the reconstructed panorama $\hat{\mathbf{P}}$ with a pre-computed transport matrix \mathbf{T} of the synthetic scene [19].

In addition, we also add loss functions on the estimated LM sun and sky parameters. Unfortunately, we do not have explicit targets for those parameters, so we rely on render losses exclusively:

$$\begin{aligned} \mathcal{L}_{\text{sky}} &= \|\mathbf{TP}_{\text{LDR}}^* - \mathbf{T}f_{\text{sky}}(\hat{\mathbf{q}}_{\text{sky}})\|_2 \\ \mathcal{L}_{\text{sun}} &= \|\mathbf{T}(\mathbf{P}_{\text{HDR}}^* - \mathbf{P}_{\text{LDR}}^*) - \mathbf{T}f_{\text{sun}}(\hat{\mathbf{q}}_{\text{sun}})\|_2. \\ \mathcal{L}_{\text{LM}} &= \|\mathbf{TP}_{\text{HDR}}^* - \mathbf{T}f_{\text{LM}}(\hat{\mathbf{q}}_{\text{LM}})\|_2 \end{aligned} \quad (6)$$

Here, we employ the same transport matrix \mathbf{T} to efficiently render an image at 64×64 resolution, and compute L2 loss on the image. The sky-only loss \mathcal{L}_{sky} relies on a “ground truth” LDR panorama $\mathbf{P}_{\text{LDR}}^*$, which is obtained by clipping the HDR panorama $\mathbf{P}_{\text{HDR}}^*$ at 1, and quantize the result to 8 bits. Among the render losses in eq. 6, we use a smaller weight for the \mathcal{L}_{sky} , which is set to 0.2.

4.3. Datasets

To train the PanoNet CNN, we rely on data from 5 complementary datasets. First, we employ 44,646 panoramas

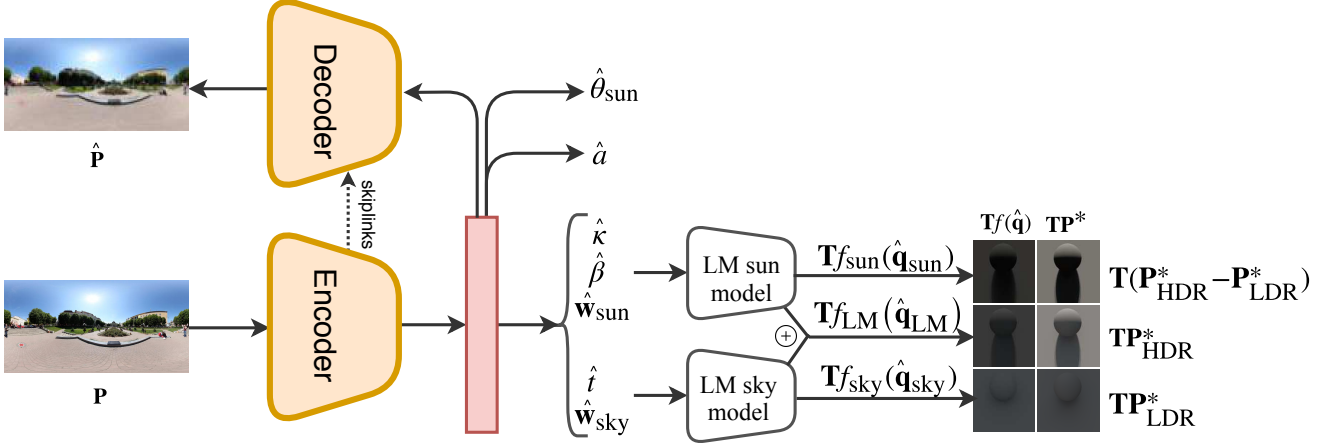


Figure 2. Architecture of PanoNet. We use an autoencoder with skip-links to regress HDR panorama from LDR. The high dynamic range lighting information is likely encoded in the bottleneck layer [26]. From this layer we estimate the LM sky parameters by 2 FC layers. Then we use the estimated parameters to drive the LM sun and sky model and generate HDR environment maps separately. The generated sun and sky environment maps are used to render an object with a pre-computed transport matrix, \mathbf{T} . We compute the loss on the renders w.r.t the supervised data. Additional losses on the sun elevation $\hat{\theta}_{\text{sun}}$, hallucinated HDR sky $\hat{\mathbf{P}}$ and its render are used to force the network to encode as much lighting information in the latent vector (red). We also ask the latent vector to be insensitive to the data domain by adding the unsupervised domain adaptation branch.

from the dataset of synthetic HDR panoramas from [26]. This dataset was created by lighting a virtual 3D city model, obtained from the Unity Store, with 9,732 HDR sky panoramas from the Laval HDR sky database [15]. Second, we use 149 daytime outdoor panoramas from the HDRI Haven database [25]. Third, we use 102 panoramas from a database of HDR outdoor panoramas [8]. Finally, we also train on 19,571 panoramas from the SUN360 dataset [24], and 4,965 from images we downloaded from Google Street View. Since these last two sets of panoramas are LDR, we only use them for the domain adaptation loss \mathcal{L}_{da} . Conversely, that loss is not evaluated when the synthetic panoramas are given as input to the network.

4.4. Training details

For training our PanoNet, we use the ADAM optimizer with a minibatch size of 80 and an initial learning rate of 0.001. Each minibatch contains 36 (45%) synthetic HDR panoramas [26], 4 (5%) captured panoramas [8], 4 (5%) HDRI Haven panoramas [25], 4 (5%) Google Street View images, and 32 (40%) of SUN360 LDR panoramas [24]. Training 500 epochs takes approximately 50 hours on an Nvidia TITAN X GPU. At test time, inference takes approximately 8ms.

4.5. Labeling the SUN360 panorama dataset with the PanoNet network

Once the PanoNet CNN has been trained, we run it on all the outdoor panoramas in the SUN360 database, to obtain their estimated sun and sky LM parameters $\hat{\mathbf{q}}_{\text{sun}}$ and $\hat{\mathbf{q}}_{\text{sky}}$, respectively. As in [9], the sun position \mathbf{l}_{sun} is obtained by

finding the center of mass of the largest saturated region in the sky. We will employ those *estimates*, denoted by $(\hat{\cdot})$, as *targets* for the second network, CropNet, whose goal will be to predict these same labels from a single limited field of view image instead of the panorama. CropNet is the subject of the next section.

5. Learning to estimate HDR parametric lighting from a single LDR image

5.1. Architecture of CropNet

Fig. 3 describes the architecture used for the CropNet CNN. Its task is to take a single LDR image as input, and estimate the LM sun parameters \mathbf{l}_{sun} , $\mathbf{q}_{\text{sun}} = \{\mathbf{w}_{\text{sun}}, \beta, \kappa\}$ and the sky parameters $\mathbf{q}_{\text{sky}} = \{\mathbf{w}_{\text{sky}}, t\}$ respectively, from that image (see sec. 3). It employs a relatively straightforward architecture à la AlexNet [14], composed of 5 convolution layers, followed by two consecutive FC layers. Each convolution layer is followed by a sub-sampling step (stride of 2), batch normalization, and the ELU activation function [4] are used on all convolution layers. The sun position branch outputs a probability distribution over the discretized sun position. As in [9], we use 64 bins for azimuth and 16 bins for elevation.

5.2. Loss functions

As with PanoNet, a variety of loss functions are used to train the CropNet CNN. First, we use a KL-divergence loss for the sun position \mathbf{l}_{sun} . Numerical loss functions are used to compare the estimated other sun/sky parameters $(\mathbf{w}_{\text{sun}}, \beta, \kappa, \mathbf{w}_{\text{sky}}, t)$ with the parameters provided by

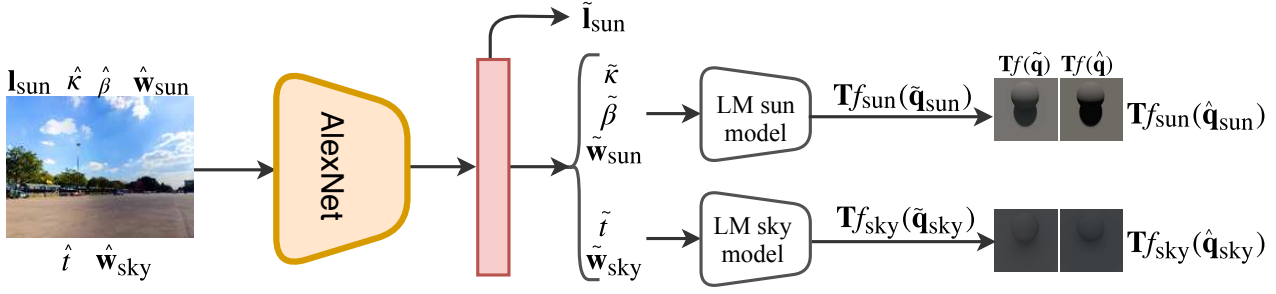


Figure 3. Architecture of CropNet. CropNet estimates LM sky parameters from a single LDR image. All the input images are cropped from the SUN360 panoramas [24], which have been labeled by PanoNet (sec. 4) with LM sky parameters $\hat{\kappa}$, $\hat{\beta}$, $\hat{\mathbf{w}}_{\text{sun}}$, \hat{t} , $\hat{\mathbf{w}}_{\text{sky}}$. The sun position \mathbf{l}_{sun} is detected from the panorama. We enforce a loss on these parameters, as well as on renders with the estimated lighting.

PanoNet (see sec. 4), $(\hat{\mathbf{q}}_{\text{sun}}, \hat{\mathbf{q}}_{\text{sky}})$. Note here that we do not have actual “ground truth” for those parameters—we aim for CropNet to match the predictions of PanoNet, but from a limited field of view image instead of the entire panorama.

We begin by computing the L2 loss on all sun and sky parameters individually:

$$\begin{aligned} \mathcal{L}_{\beta} &= \|\hat{\beta} - \tilde{\beta}\|_2 & \mathcal{L}_{\kappa} &= \|\hat{\kappa} - \tilde{\kappa}\|_2 & \mathcal{L}_t &= \|\hat{t} - \tilde{t}\|_2 \\ \mathcal{L}_{\text{wn}} &= \|\hat{\mathbf{w}}_{\text{sun}} - \tilde{\mathbf{w}}_{\text{sun}}\|_2 & \mathcal{L}_{\text{wy}} &= \|\hat{\mathbf{w}}_{\text{sky}} - \tilde{\mathbf{w}}_{\text{sky}}\|_2 \end{aligned}, \quad (7)$$

where $(\tilde{\cdot})$ denotes CropNet outputs. Prior to computing the loss, each parameter is normalized in the $[0, 1]$ interval according to the minimum and maximum values in the training set, and the weights are all set to 1 except for \mathcal{L}_{κ} , \mathcal{L}_{β} and \mathcal{L}_{wn} which are set to 5, 10, and 10 respectively to balance the loss functions.

As with PanoNet, we also employ render losses to help with the training. Since CropNet does not have ground truth HDR lighting, two render losses are used in contrast to PanoNet. Since the sun position \mathbf{l}_{sun} is treated independently from the other lighting parameters, we exclude the sun position from the render loss. The following render losses are employed:

$$\begin{aligned} \mathcal{L}_{\text{sky}} &= \|\mathbf{T}f_{\text{sky}}(\hat{\mathbf{q}}_{\text{sky}}) - \mathbf{T}f_{\text{sky}}(\tilde{\mathbf{q}}_{\text{sky}})\|_2 \\ \mathcal{L}_{\text{sun}} &= \|\mathbf{T}f_{\text{sun}}(\hat{\mathbf{q}}_{\text{sun}}) - \mathbf{T}f_{\text{sun}}(\tilde{\mathbf{q}}_{\text{sun}})\|_2 \end{aligned} \quad (8)$$

In practice, this corresponds to rendering a simple scene (diffuse sphere on a plane as in sec. 4.2) with the estimated sky(sun)-only parameters, and comparing it with a render of that same scene with the sky(sun)-only parameters predicted from the PanoNet (c.f. sec. 4). The weight for both of these losses is set to 1.

5.3. Training details

To train our CropNet, we use the ADAM optimizer with a minibatch size of 256 and an initial learning rate of 0.001. Each minibatch contains 230 (90%) SUN360 LDR panoramas [24] and 26 (10%) captured panoramas [8]. Training

500 epochs takes approximately 90 hours on an Nvidia TITAN X GPU. At test time, inference takes approximately 25ms.

6. Experimental validation

We evaluate both of our proposed CNNs on a dataset of HDR outdoor panoramas [8] and the SUN360 LDR dataset [24]. First, we show that reliable sky parameters can be estimated from LDR panoramas using our proposed PanoNet through both quantitative and qualitative comparisons with ground truth data. Then, we show how the CropNet network can robustly estimate the same sky parameters from a single LDR image.

6.1. Evaluating the PanoNet CNN

To evaluate the ability of the PanoNet network to predict the LM sky parameters from a single LDR panorama, we employ 103 outdoor HDR panoramas [8]. We first convert each panorama to LDR by applying a random exposure factor to the HDR panorama, clipping its maximum value at 1, and quantizing to 8 bits. To quantify performance, the scale-invariant (si-)RMSE error metric is computed by rendering the diffuse sphere+plane scene (sec. 4.2) with the ground truth panorama (original HDR), the results of the non-linear fitting approach of [9], and our PanoNet. For the scene, we use a simple sphere placed on a ground plane, viewed from the top. We place the sun of both methods to its ground truth position—this ensures that sun position is factored out in the evaluation.

We show some qualitative visual examples in fig. 4; these are accompanied by the quantitative comparison results shown in fig. 5. To better highlight the different lighting conditions in the HDR panorama dataset, we split it into 3 categories based on the softness of shadows in the scene. Shadow softness is estimated by first computing the histogram of horizontal gradients on a 5-pixel row immediately below the sphere. Then, a reference sunny image is manually

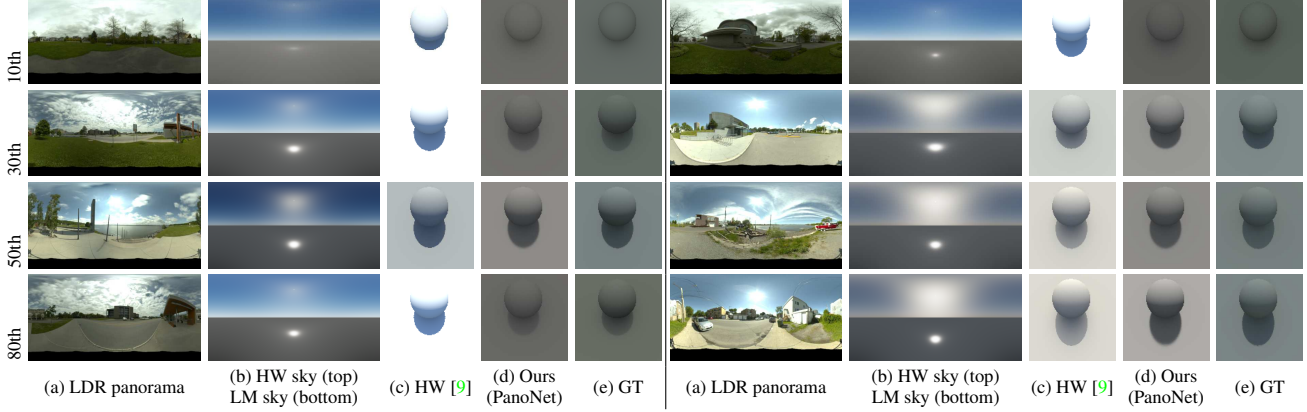


Figure 4. Qualitative comparison between approaches to estimate HDR parametric lighting from a single LDR panorama. We compare our PanoNet method, which learns to predict the parameters of the LM model [17], to the approach of [9] which directly fits the parameters of the HW model [10, 11] using non-linear optimization. For each example, we show (a) the input LDR panorama, (b) the reconstructed skies using both the HW (top) and LM (bottom) models, (c–e) a rendering of a simple scene (viewed from the top) using [9], ours, and the ground truth respectively. The rows are ordered by si-RMSE percentile for our technique (see fig. 5 for the overall error distribution). All the renders are tone-mapped with a $\gamma = 2.2$ for visualization purposes. Our PanoNet can better fit different weather conditions such that the renders are similar to the real lighting both in terms of intensity and shadow softness. Please zoom in for details.

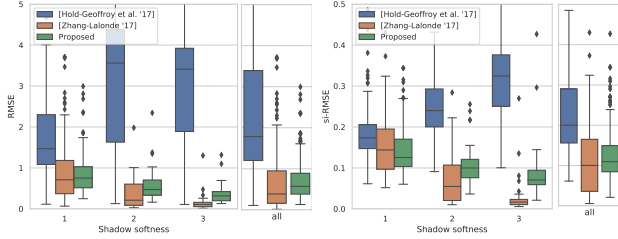


Figure 5. Quantitative comparison of panorama fitting on the real dataset. We show a distribution of (left) RMSE and (right) scale-invariant RMSE [7] as a function of the shadow softness, ranging from very sharp (1) to smooth (3) (see text for the shadow softness definition used), and over the entire dataset (“all”). Compared to the previous state-of-the-art [9], our PanoNet provides better fits to the wide range of illumination conditions. The non-parametric approach of [26] provides a lower bound on the prediction. The lower (upper) edge of each box indicates the 25th (75th) percentiles.

identified, and the KL-divergence between the histograms of each image and that of the reference is computed. Finally, the dataset is sorted by the KL-divergence, and subsequently split into three groups by empirically determining cutting points where shadows appear sharp (corresponding to a softness of 1, or sunny skies) and where shadows are not visible (corresponding to a softness of 3, or overcast skies). The remainder of the dataset is classified as having a softness of 2 (partially sunny skies). As shown in fig. 5, the non-parametric model [26] is unsurprisingly more precise than both parametric models, however it does not produce intuitive sky parameters that can be used to subsequently train CropNet. The approach of [9] yields low errors when

Shadow softness weather		1 clear	2 mixed	3 cloudy	all
RMSE	eq. 6 only	1.07	0.55	0.59	0.82
	eqs 5 and 6	0.92	0.41	0.36	0.73
si-RMSE	eq. 6 only	0.19	0.13	0.08	0.16
	eqs 5 and 6	0.15	0.11	0.09	0.13

Table 1. Ablation study comparing the use of different loss functions (eq. 6 only vs. using both eqs 5 and 6) when training PanoNet.

shadows are sharp (sunny skies), but the error significantly increases when shadows should be softer (partially cloudy to overcast skies). In contrast, our PanoNet performs similarly well across all lighting conditions.

We compare two variants of PanoNet: trained with our proposed combined loss (eqs 5 and 6) and with the parametric loss only (eq. 6). The performance at estimating lighting in different weather conditions is shown in table 1, and indicates that adding the additional losses from [26] helps PanoNet to encode more information in the latent space.

6.2. Evaluating the CropNet CNN

We extract 7 limited field-of-view photos from each panorama in the HDR outdoor panorama dataset [8] using a standard pinhole camera model, and randomly sampling the camera azimuth. Then, we estimate lighting parameters from these crops using the approach of [9] and our CropNet. The sun position angular error from our CropNet and [9] is shown in fig. 7. Note that the sun position is independent from the other radiometric lighting parameters in our approach. In contrast, the radiometric sky properties are

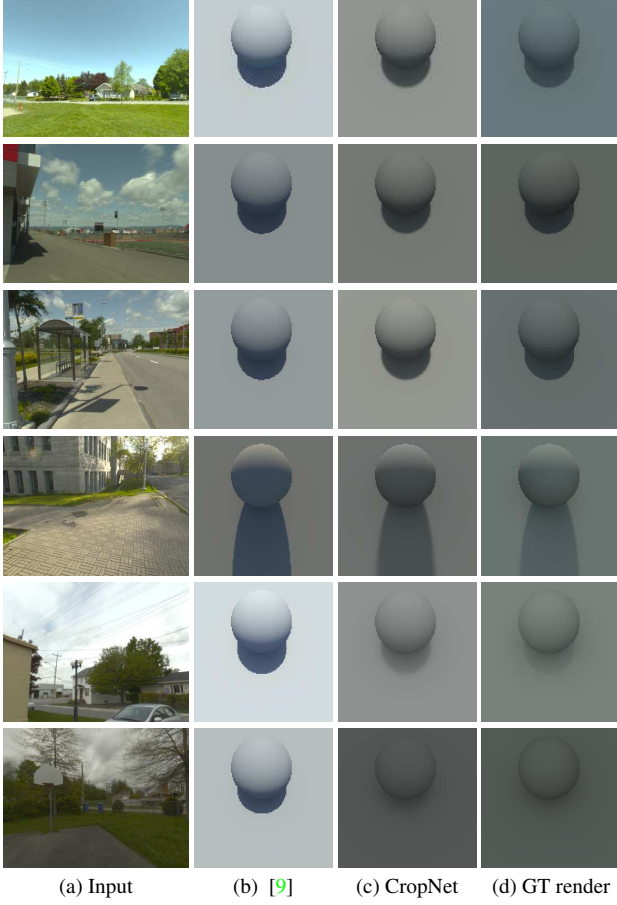


Figure 6. Lighting estimation comparison with ground truth. We crop an image from an HDR panorama database [8], then the cropped image (a) is used to estimate the lighting parameters from [9] and CropNet. The renders from each approach are shown in (b) and (c); (d) shows the ground truth render from the HDR panorama.

constrained by the sun position in [9]. Therefore, in order to fairly compare the radiometric lighting parameters, we employ the network of [9] to estimate the sun position in the subsequent experiments.

Fig. 6 shows qualitative comparison between the prediction from [9] and our CropNet with the ground truth. Again, our method can accurately estimate lighting conditions ranging from clear to overcast. For example, in an overcast day (last row of fig. 6), our approach can successfully estimate the lighting and produce renders with soft shadows. However, we notice that the approach in [9] constantly outputs a clear sky, and usually fails to generate soft shadows.

Those qualitative results are validated quantitatively in fig. 8, which reports both the RMSE and si-RMSE metrics with respect to renders obtained with the ground truth lighting. Again, our approach shows much improved performance across different weather conditions and error metrics. The RMSE plot demonstrates that our CropNet can obtain a

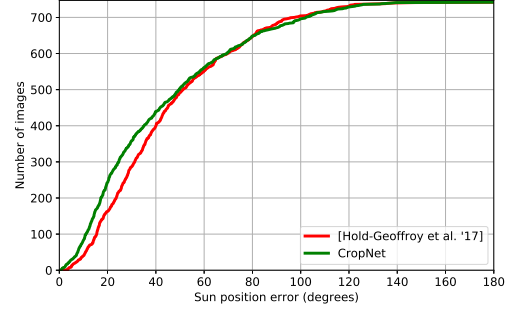


Figure 7. Cumulative sun angular error comparison between our CropNet and [9] on single images extracted from SUN360 dataset [24]. Our method slightly outperforms that of [9].

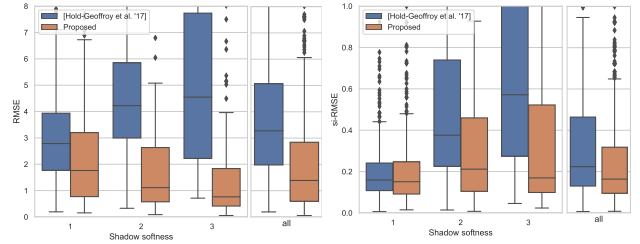


Figure 8. Quantitative comparison between our CropNet and [9] on single images extracted from a dataset of HDR panoramas [8], where the ground truth lighting is available. The render error is shown in the box-plots as function of shadow softness ranging from very sharp (1) to smooth (3) (see text for the exact definition), and over the entire dataset (“all”). The lower (upper) edge of each box indicates the 25th (75th) percentiles. While both techniques perform relatively similarly when shadows are very sharp (in sunny conditions), the error of [9] increases when the sky is not completely clear and shadows start to disappear. In contrast, our method remains much more stable.

much more accurate estimate of exposure in outdoor scenes. The si-RMSE shows that the estimated lighting generates more faithful shadows when compared to [9].

Fig. 9 shows qualitative comparison between our CropNet with [9] for real images. Our approach is able to handle different weather conditions ranging from fully sunny to overcast skies. Fig. 10 compares the lighting between our CropNet and [9] for different viewpoints extracted from the same panorama. Our approach better adapts the *local* crop lighting than [9]. More examples can be found in the supplementary material.

7. Discussion

In this paper, we presented a method for estimating HDR lighting from a single, LDR image. At the heart of our approach lies a CNN that learns to predict the parameters of an analytical sky model from a single LDR panorama such that it 1) more realistically reconstructs the appearance of the sky, and 2) renders the appearance of objects lit by this

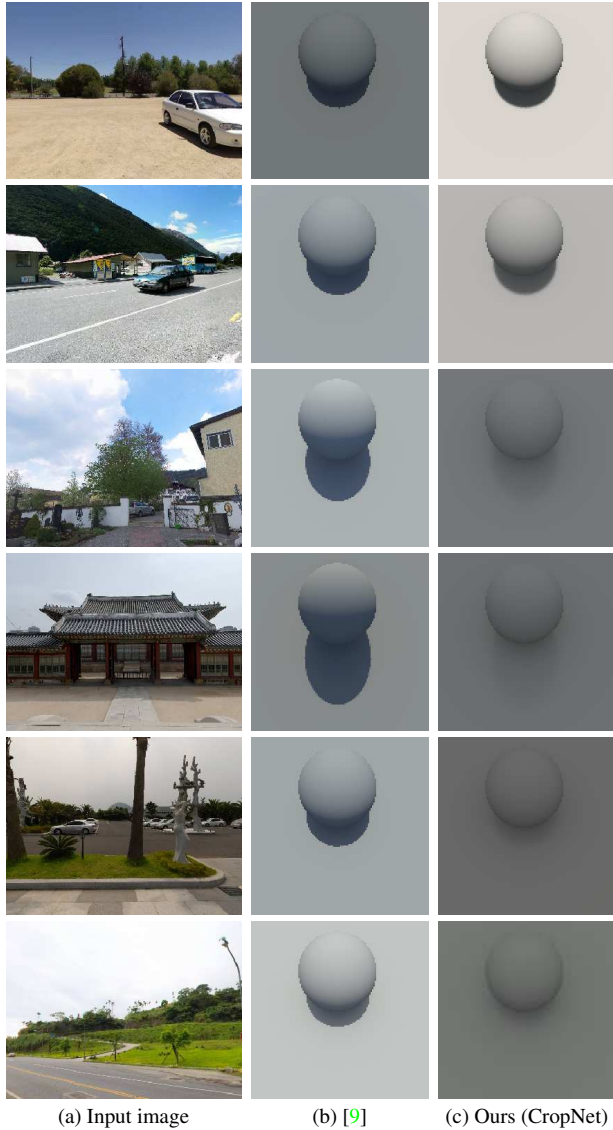


Figure 9. Lighting estimation for real images. We show different lighting estimation results from real images. Our approach provides consistent estimations in a wide range of illumination conditions, ranging from clear (top) to overcast (bottom). Here, the sun azimuth is kept fixed to better compare the renders.

illumination. This CNN is used to label a large dataset of outdoor panoramas, which is in turn used to train a second CNN, this time to estimate the lighting parameters from a single, limited field of view image. Due to its intuitive set of lighting parameters such as the sun shape and color, our approach is particularly amenable to applications where a user might want to modify the estimated lighting parameters, either because they were judged to be not quite right, or to experiment with different lighting effects. In addition, the network outputs can be used to render an environment map, which can readily be used to insert photorealistic objects into photographs (fig. 1).

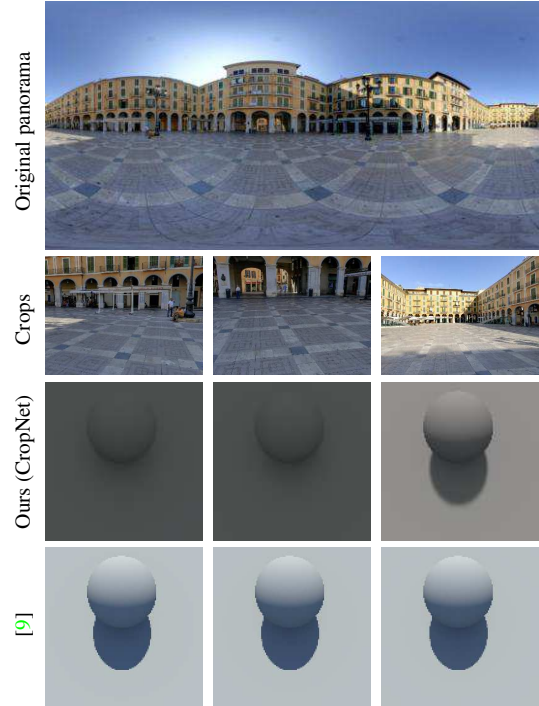


Figure 10. Estimating lighting for different viewpoints extracted from the same panorama. From a single panorama (top row), we extract three different crops from left to right (2nd row), and compare the lighting estimates obtained with our method (3rd row) to that of [9] (bottom row). Despite being extracted from the same panorama, the crops exhibit different lighting conditions. Our method adapts to these changes naturally, and predicts believable illumination conditions in all three cases. Here, the sun azimuth is kept fixed to better compare the renders.

While our approach outperforms the state-of-the-art both qualitatively and quantitatively, it is not without limitations. Its most noticeable limitation is that it typically has difficulty in properly identifying soft shadows, both from panoramas and from crops. We suspect this is because the HDR training data does not contain many examples where this is the case. In addition, since shadows are dimmer, they are harder to see in the images, as such cues to their existence are most subtle. Another limitation is the tendency to estimate gray skies, even when the sky in the image is visible, and blue. Again, we suspect that this is a data issue: the network has difficulty in overcoming the fact that most panoramas have clouds which bring the average sky color closer to gray. These questions pave the way for exciting future work.

Acknowledgements

This work was partially supported by the REPARTI Strategic Network and the NSERC Discovery Grant RGPIN-2014-05314. We gratefully acknowledge the support of Nvidia with the donation of the GPUs used for this research, as well as Adobe for generous gift funding.

References

- [1] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1670–1687, 2015.
- [2] D. A. Calian, J.-F. Lalonde, P. Gotardo, T. Simon, I. Matthews, and K. Mitchell. From Faces to Outdoor Light Probes. *Computer Graphics Forum*, 37(2):51–61, 2018.
- [3] D. Cheng, J. Shi, Y. Chen, X. Deng, and X. Zhang. Learning scene illumination by pairwise photos from rear and front mobile cameras. *Computer Graphics Forum*, 37(7):213–221, 2018.
- [4] 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.
- [5] S. Georgoulis, K. Rematas, T. Ritschel, M. Fritz, T. Tuytelaars, and L. Van Gool. What is around the camera? In *IEEE International Conference on Computer Vision*, 2017.
- [6] S. Georgoulis, K. Rematas, T. Ritschel, E. Gavves, M. Fritz, L. Van Gool, and T. Tuytelaars. Reflectance and natural illumination from single-material specular objects using deep learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(8):1932–1947, 2018.
- [7] 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 (ICCV)*, 2009.
- [8] Y. Hold-Geoffroy, A. Athawale, and J.-F. Lalonde. Deep sky modeling for single image outdoor lighting estimation. In *IEEE International Conference on Computer Vision and Pattern Recognition*, 2019.
- [9] Y. Hold-Geoffroy, K. Sunkavalli, S. Hadap, E. Gambaretto, and J.-F. Lalonde. Deep outdoor illumination estimation. In *IEEE International Conference on Computer Vision and Pattern Recognition*, 2017.
- [10] L. Hošek and A. Wilkie. An analytic model for full spectral sky-dome radiance. *ACM Transactions on Graphics*, 31(4):1–9, 2012.
- [11] 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, 2013.
- [12] 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 Transactions on Graphics*, 33(3):32, 2014.
- [13] 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), 2014.
- [14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*. 2012.
- [15] 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.
- [16] 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.
- [17] J.-F. Lalonde and I. Matthews. Lighting estimation in outdoor image collections. In *International Conference on 3D Vision*, 2014.
- [18] S. Lombardi and K. Nishino. Reflectance and illumination recovery in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1), 2016.
- [19] J. S. Nimeroff, E. Simoncelli, and J. Dorsey. Efficient re-rendering of naturally illuminated environments. In *Photorealistic Rendering Techniques*, pages 373–388, 1995.
- [20] 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.
- [21] 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.
- [22] K. Rematas, T. Ritschel, M. Fritz, E. Gavves, and T. Tuytelaars. Deep reflectance maps. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
- [23] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015.
- [24] 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.
- [25] G. Zaal. HDRI haven. <https://hdrihaven.com>, 2018.
- [26] J. Zhang and J.-F. Lalonde. Learning High Dynamic Range from Outdoor Panoramas. *International Conference on Computer Vision*, 2017.