OutCast: Outdoor Single-image Relighting with Cast Shadows

David Griffiths[1,2] Tobias Ritschel[2] Julien Philip[1]

[1]Adobe Research [2]University College London

Figure 1: *Given only a single RGB image* (left) *in one lighting, our method generates images of that scene in new lighting* (middle / right).

Abstract

We propose a relighting method for outdoor images. Our method mainly focuses on predicting cast shadows in arbitrary novel lighting directions from a single image while also accounting for shading and global effects such the sun light color and clouds. Previous solutions for this problem rely on reconstructing occluder geometry, e. g., using multi-view stereo, which requires many images of the scene. Instead, in this work we make use of a noisy off-the-shelf single-image depth map estimation as a source of geometry. Whilst this can be a good guide for some lighting effects, the resulting depth map quality is insufficient for directly ray-tracing the shadows. Addressing this, we propose a learned image space ray-marching layer that converts the approximate depth map into a deep 3D representation that is fused into occlusion queries using a learned traversal. Our proposed method achieves, for the first time, state-of-the-art relighting results, with only a single image as input. For supplementary material visit our project page at: `https://dgriffiths.uk/outcast`.

CCS Concepts

• ***Rendering*** → *Relighting;*

1. Introduction

Capturing stunning photographs requires a subtle equilibrium between the subject, the composition and the lighting of a scene. While a user can decide what subject to capture and control the properties of the sensor, obtaining the right lighting is much more challenging and often either requires patience and dedication, or is simply out of the user's control. For outdoor pictures, where the sun and sky lighting is dominant, previous methods have proposed to relight an image by removing and re-synthesizing shadows [YDMH99, TSE*04, DRC*15, PGZ*19], this problem is well-defined and known to be notably challenging in computer graphics literature. One of the key hurdles is that occluding geometry casting shadows can be arbitrarily far away from the point receiving the shadow, thus requiring a fine understanding of long-range interactions between objects is necessary. The most dramatic shadow shots e. g., a sunset illuminating architecture, are notoriously difficult in this respect as the shading of a point can depend on arbitrarily far geometry. In addressing this issue, we propose a method that takes a single RGB image as input and enables a user to change its illumination, including dominant cast shadows.

When accurate 3D geometry of a scene is available, cast shadows can be computed precisely using ray-tracing or shadow mapping. Alternatively, given multiple photos from varying view-points of the scene, 3D proxy geometry can be estimated. Such geometry, even if approximate, has been proven sufficient to produce faithful shadows and global illumination effects [PGZ*19, PMGD21]. A

challenge in these approaches is to adapt the light transport computation to become robust to the approximate 3D geometry, e. g., by using a Neural Network (NN) to combine shadows with the actual image (Fig. 2, gray arrows). In this paper we go to the extreme, and show for the first time, how to cast shadows from very approximate and incomplete geometry (a depth map), extracted from a single RGB image alone (Fig. 2, black arrows). To do so we demonstrate how to leverage off-the-shelf NN-based depth maps [EF15, RLH*20, YZW*21] and the limited geometric information they provide to compute plausible cast shadows and shading for an arbitrary sun position.

We achieve this by combining classic screen space shadowing [RGS09] with a learned component to produce both attached and cast shadows as well as more accurate shading. Our learned component is convolutional and able to attend screen space information relevant to casting a shadow, while conventional image-to-image translation [ZPIE17] from depth to shading [NAM*17] is unable to deal with such long-range interactions.

Our main contribution in this work is thus a new hybrid component mixing image space graphics and learnt priors. This component provides a way to compute precise long-range interaction, such as cast shadows, from depth maps alone. This allows plausible relighting in challenging outdoor environments from a single picture.

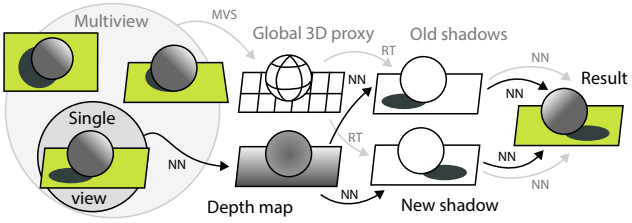

Figure 2: *Our method* (black arrows) *extracts a depth map from a single view to compute cast shadows while previous work* (grey arrows) *relied on a global multi view-generated 3D proxy.*

2. Previous Work

Our work builds on research in both the fields of image lighting estimation/relighting and deep learning-based methods for image-to-image tasks. For a general survey on (deep) lighting estimation and relighting we refer the reader to [EGH21].

2.1. Lighting and Shading Estimation

A key aspect to changing the lighting of an image is understanding its original lighting conditions. For example, the source image shading can inform a relighting algorithm which shadows need to be removed in a target image, or give important cues regarding the light source. Early works on image-based lighting [Deb02] demonstrated the ability to use captured lighting from images covering the hemisphere to render synthetic objects under novel lighting conditions, a technique which was further extended to high dynamic range images [STJ*04]. In contrast, many recent works attempt to directly estimate a parametric lighting representation from a single RGB image. Other work [LEN09] exploits cues extracted from varying portions of the image (e. g., sky, vertical surfaces and ground), as well as shadows, shading and approximate geometry to estimate the sun position, which can in turn be used to generate a synthetic sun dial.

Recent works [HGSH*17, HGAL19, ZSHG*19] leverage advances in deep learning to estimate outdoor sky parameters. CNN-based architectures are shown to be effective in estimating high dynamic range or parametric outdoor illumination from low dynamic range image inputs [LMF*19]. The resulting environment maps can effectively be used to directly render new synthetic objects into the original image, as long as complex interactions between shadows are avoided. The approaches mentioned all assume an outdoor sky model with a single source of light for illumination. Other work [GHGS*19] proposes a method to estimate lighting of indoor scenes for a single image with multiple light sources of varying properties. The authors achieve this by defining a parametric lighting representation describing area lights distributed in 3D for each pixel making the method computationally cumbersome, preventing the computation of hard shadows due to the low resolution of the representation. Similar work [LSR*20] proposed an indoor network which can estimate scene shape, spatially varying lighting (driven by a spherical Gaussian lighting representation) and non-Lambertian surface reflectance.

These major steps forward in lighting estimation, unfortunately, only formulate part of the solution to the relighting problem. For instance, Li et al., [LSR*20] estimate intrinsic parameters such as normals and albedo, allow object insertion and some level of scene editing but not relighting. Even with this rich information, altering the entire image illumination is not straightforward, as one also needs to define a coherent lighting, shading and shadows for the novel illumination.

Another essential component of the relighting problem is the automatic removal of source shadows. Even with access to accurate shadow masks, this is a challenging task. Such algorithms must adjust the shadow pixel intensities, whilst also inferring semantic understanding of the scene to handle fine shading details. DeshadowNet [QTH*17b] is a multi-branch CNN which learns both local and global features of the input image. Wang et al., [WLY18] propose a novel approach where separate shadow detection and removal networks can mutually benefit from each other by introducing adversarial losses. In this work we jointly perform shadow removal, re-casting and relighting in a single unified network. Whilst this demands considerably more from the network, we demonstrate empirically these tasks can be learned together.

Intrinsic decomopsition of images can also be a useful technique for extracting shadows from RGB images [BKPB17]. Such methods can identify (and subsequently remove [QTH*17a]) source shadows and shading in the scenes and can now run at an interactive frame rate [MSZ*21]. Many early works adopted this approach [ISR12] to enable some form of lighting editing. Unfortunately, it is not obvious how these methods can be extended to produce the inverse, shadow generation, especially with long range interaction.

2.2. Relighting

Prior work on image relighting typically relies on scene geometry, light and reflectance models to accurately relight using inverse rendering [YDMH99, LFD*99, MG97]. Having access to the full scene representation (geometry, materials, lighting) allows traditional rendering and shading methods to be used and gives promising results, however, requires a highly complex capture process. The capture process can be simplified through techniques such as semiautomatic vision-based geometry reconstruction [LDR00], or by computing scene parameters through viewing the same scene under varying lighting conditions [ED04, LFD*99], however, a high level of technical competency is still required. Furthermore, lighting and material information requirements can be relaxed using Inverse Path Tracing [ALKN19], however, a full and accurate scene geometry is still essential for high quality rendering. If very high quality results are required (e. g., for film production), sophisticated capture sets ups such as the Light Stage [DHT*00, WGT*05, MHP*19, GLD*19, SXZ*20] can be employed. Whilst such methods give impressive results, they require expensive rigs and trained professionals to operate them.

Advancements in deep learning have resulted in framing full scene representation as an optimization problem [SMB*20, MST*20]. An implicit representation of the scene is jointly optimized for geometry but can also be optimized for surface characteristics such as albedo and normals, enabling rerendering under new lighting conditions using traditional rendering pipelines [BXS*20a, ZSD*21, BBJ*21, BXS*20b, SDZ*21]. More recent work such as PixelNeRF [YYTK21] show that with appropriate conditioning reasonable results can be obtained from single-image inputs. Unlike our method, these works are restricted to objects, rather than entire scenes and require multiple views as input or require strong preconditioning and assumptions of the scene.

For scene scale relighting, deep learning has enabled significant advances in the performance of multi-view relighting systems [PGZ*19, PMGD21]. Typical approaches employ a NN to map from one of the input images and a set of approximate guide maps (depth, normals, shadow images) to a novel illumination condition. 2D image NNs are used to jointly remove the old shadow and shading and change it to the new lighting and shadow without attempting to recover an explicit shadow-free image [SSL12] or an albedo map [LM71], requiring only geometry to guide the relighting. However, the quality of the results still comes at the cost of the capture process. To allow for a proxy geometry, photogrammetric pipelines can be utilized, requiring tens or hundreds of images of the scene from varying view-points. In many practical applications, multi-image data acquisition is not possible. Whilst impressive results have been achieved using only image sets with as little as five images [XSHR18, RDL*15], these approaches are typically constrained to objects or simple scenes.

It is specifically this constraint we aim to relax in our work. We build on the work of Philip et al,. [PGZ*19], however, only require a single-image input. Despite this, we achieve a visually comparable performance on challenging real-world test cases (Fig. 12). Key to our method is the use of an off-the-shelf depth estimator [EF15, RLH*20, YZW*21] as a source of approximate geometry and a novel 3D module to become robust to such inputs.

Single-image relighting is not a new problem. Many studies focus on relighting limited subjects such as human faces [PTMD07, WZL*09, SYH*17, ZHSJ19, SBT*19, NLML20] and bodies [KE18, LSY*21]. On these restricted classes deep priors are easier to build, allowing for very impressive results such as recently demonstrated in Total Relighting [POEL*21]. Regarding more general scenes, Wu et al., [WS17] obtain realistic relighting results from single images, but at the cost of significant user interactions to annotate the scene and estimate the geometry. Ture et al., [TCE*21] focus mostly on sky relighting. Similar to our method, monocular depth estimation is used, however, their approach can only handle shadows cast by clouds. Single-view relighting methods [YME*20, LGZ*20] have also recently built on image-to-image translation methods [ZPIE17, WLZ*18]. One of the key missing components of these methods is their ability to generate accurate and convincing cast shadows in the target image which is our main contribution. Liu et al., [LGZ*20] show some level of shadow casting, but these are overly smooth and soft and the method is restricted to cities.

Other works deal with shadows by assuming video / timelapse input [SMPR07], terrain data [KNC*08] or object templates [KSES14, KHFH11]. The outdoor scenes and the capture modality we target do not match such requirements. Attempts at predicting cast shadows are also made using traditional 2D CNNs [CVJR19, ZLW19, LLZ*20, SZB21], allowing to cast approximate shadows of individual objects with limited quality especially when fine, long-range, interactions are needed, such as with hard cast shadows. Still, we study 2D CNNs as a baseline for our method, demonstrating they are insufficient to solve our task. This is because a typical U-net [RFB15], or even more advanced architectures [WLZ*18], while being able to aggregate information at multiple scales, have no inductive bias to attend the information for long-range shadow interactions, which is in the direction of the light.

2.3. Epipolar Geometry

The shortcoming of 2D Convolutional Neural Networks (CNNs) to explicitly collect relevant features across an image has been identified in prior work. Most notably epipolar Transformers [HYFY20] create feature volumes by sampling along the Epipolar line of 2 images of the same scene with a known transformation. Explicitly sampling along a known ray boosted performance for 3D human joint localization. Similar conclusions have been drawn for depth regression [PDB18], data-adaptive interest points [YMB*19] and keypoint detection [JYP18]. Shin et al., [SRSF19] further adapt epipolar transformers for the task of 3D scene reconstruction for single-view RGB images. Our 3D shadow network (Fig. 9) takes inspiration from such networks. However, instead of our sampling direction being determined by epipolar geometry, the direction is determined by the position of the light source. Furthermore, the above methods sample in feature space, whereas we directly sample the input RGBZ image.

3. Background

Our work is based on a method proposed by Philip et al., [PGZ*19], which relights an RGB image C_o, captured in an (unknown) original

light condition characterized by the sun direction ω_o, to a novel light direction ω_n. A particular strength of their method is the ability to render accurate cast shadows. This is achieved through the use of *shadow images* which are represented as gray-scale images S_o and S_n that hold the shadow information in the old and new light direction, respectively. These intermediate maps are not used in a classic inverse rendering setting, but instead serve as guides to an image-to-image translation network to perform the final relighting. However, constructing these shadow images relies on global 3D geometry, acquired with multi-view reconstruction [Ull79], necessitating tens to hundreds of images.

In this work, we keep the overall structure of the system propsed by Philip et al., but relax the requirement of multi-view images as input. Instead, we produce the shadow images from the original color image C_o alone. To enable this, we assume access to an approximate depth estimation process $\mathcal{Z}(C_o)$ with only scale-invariance, e. g., an off-the-shelf NN [YZW*21]. Alternatively, depth estimation could come from an active depth sensor [Zha12, BCD*21], which are becoming increasingly popular on smartphones.

Casting shadow from such approximate geometry, is more challenging than casting shadows from exact geometry via shadow mapping or ray-tracing. This is largely due to high levels of occlusions, resulting in incomplete geometry. A depth map only provides the geometry of the visible surface when computed from a single viewpoint of the scene. A depth map does not provide information regarding what is behind an object thus direct shadow casting across its back will incorrectly report in no shadows (Fig. 6) or too much shadows (Fig. 4, second column). Our main contribution is a learnable module that takes as input samples obtained from a depth-based shadow casting approach, and outputs a shadow mask, robust to approximate and incomplete geometry.

Formally, the system proposed by Philip et al., [PGZ*19] is a *relighting operator* $\bar{\mathcal{L}}(C_o, S_o, S_n, \omega_o, \omega_n)$. Where C_o is the image which we want to relight, S_o and S_n are the respective old and new shadow images computed thanks to the mulit-image derived 3D proxy geometry, and where ω_o and ω_n represent the old and new sun directions respectively. In this work we focus on removing the requirement for the multi-image derived 3D geometry proxy that allows the computation of old and new shadow images S_o and S_n as system inputs, resulting in a new operator $\mathcal{L}(C_o, \omega_o, \omega_n)$, relying only on the color image C_o and the input light direction ω_o. Details on how we obtain ω_o can be found in Sec. 4.2.

4. Image Relighting using Approximate Depth Maps

Our new relighting approach ($\mathcal{L}$) shares the high level architecture of the one proposed by Philip et al., [PGZ*19] ($\bar{\mathcal{L}}$) using shadow images directly computed from the color image C_o alone as input:

$$C_n = \mathcal{L}(C_o, \omega_o, \omega_n) \simeq \bar{\mathcal{L}}(C_o, \mathcal{S}(C_o, \omega_o), \mathcal{D}(C_o, \omega_n), \omega_o, \omega_n)),$$

where $\mathcal{S}$ produces the new shadow image (Sec. 4.1) and $\mathcal{D}$ produces the old shadow image (Sec. 4.2) that are combined in the final relighting step (Sec. 4.3). Fig. 3 shows an overview of our approach. Fig. 10 shows an example of the intermediate buffer maps on test data.

Depth and normal estimation We make use of a depth estimation method based on MiDaS [RLH*20], denoted as $Z = \mathcal{Z}(C)$. As the initial MiDaS implementation is trained to produce disparity maps that are not only scale-invariant, but also shift-invariant, an unknown non-linear distortion would be applied to depths values if we were to use it directly, leading to poor normals and distorted geometry [YZW*21]. We therefore use a version of MiDaS [RLH*20] trained without using the shift-invariant losses so it recovers the scene with an unknown scale but no additional shift. This is achieved by training on data that is either synthetic, LiDAR sensed, or from stereo camera pairs with known calibration, meaning there is no shift ambiguity in the data. We do not refine the weights of $\mathcal{Z}$ during training.

We compute an approximate normal image N, by first converting the depth map to a 3D position map containing for each pixel its x, y, z coordinates according to the camera frame of reference (operator ρ) and then taking the normalized cross product between the horizontal and vertical components, u and v, of the gradient of the position map. This gradient is computed using a Sobel filter.

$$N = \frac{\partial \rho(Z)}{\partial u} \times \frac{\partial \rho(Z)}{\partial v}$$

This process does not need to be differentiable, it only allows to use an approximate depth Z or normals N whenever working with a color image C. All operations are performed in camera space.

Loss Our relighting $\mathcal{L}_\theta$ is a *tunable mapping*, a function with learnable parameters θ which are trained end-to-end, minimizing a cost function of four terms:

$$f = \lambda_\mathcal{S} f_\mathcal{S} + \lambda_\mathcal{D} f_\mathcal{D} + \lambda_\mathcal{L} f_\mathcal{L} + \lambda_\mathcal{A} f_\mathcal{A}. \quad (1)$$

where λ scales each loss, respectively. We will explain the new-shadow, old-shadow, relighting and adversarial loss terms $f_\mathcal{S}$, $f_\mathcal{D}$, $f_\mathcal{L}$ and $f_\mathcal{A}$ in the following respective Sections 4.1, 4.2 and 4.3.

4.1. Producing a new shadow

Problem analysis The main challenge we address in this paper is how to cast shadows using an imperfect depth map. A depth map differs from the global proxy 3D geometry in two important aspects: it is more inaccurate and it is incomplete. Therefore, directly applying shadow casting to obtain detached shadow-images will produce unsatisfactory results.

It is *inaccurate*, because a NN-based depth estimator is never able to perfectly match the true depth. It lacks details and often suffers from distortion. For example, an even ground plane or a side wall of a building facing away from the light typically does not come out as a proper plane, rather it would appear bumpy and curved. These effects are particularly strong for texture objects in a phenomenon known as "texture copying". Such bumps, when used to compute attached shadows will lead to several false positives: normals computed from bumpy depth will make surfaces sporadically face away from the light, i. e., darker. When used during the computation of detached shadows, they cast small shadows known as "shadow acne" in the shadow mapping literature [DYK*14], this phenomenon is illustrated in the first column of Fig. 4. Severity of these difficulties varies from method to method and from sensor to sensor but are clearly present in the state-of-the-art depth estimator we use in our experiments as well as in low-cost active depth sensors (e. g., LiDAR sensors).

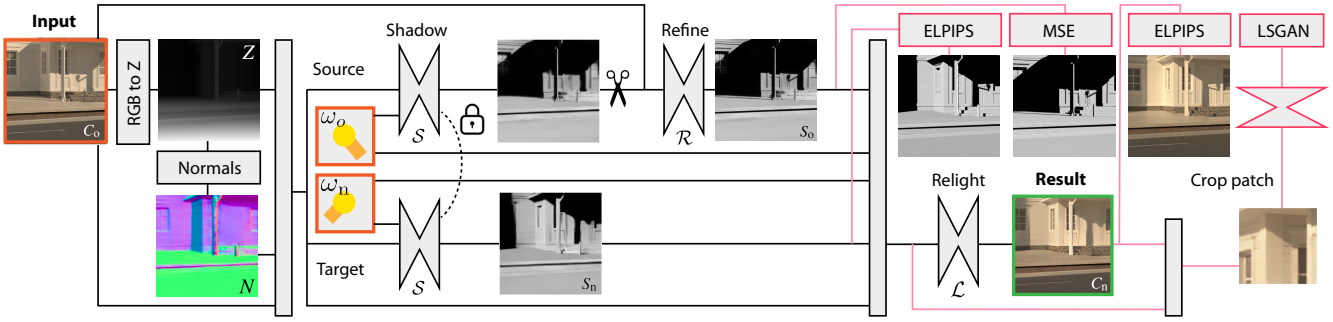

Figure 3: *The main network architecture of our approach. Input is outlined in orange, output in green. Pink lines indicate the cost functions we optimize for. Blocks with learnable parameters are double-wedges, and a dotted line indicates siamese training / weight sharing. The scissor symbol is where the gradients are detached from the network graph, to prevent further back propagation of the gradient.*

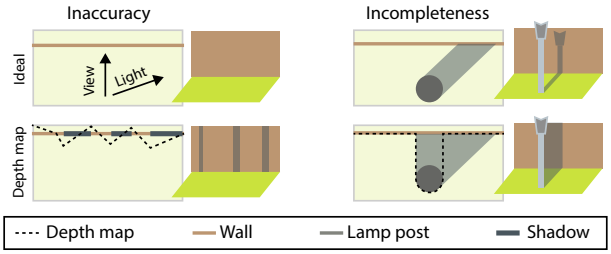

Figure 4: *Illustration of the inaccuracy and the incompleteness problems when casting shadows from a depth map. The top row shows a scene in ideal conditions with complete global geometry. The bottom row shows shadows resulting from a high threshold direct shadow casting with approximate depth map geometry.*

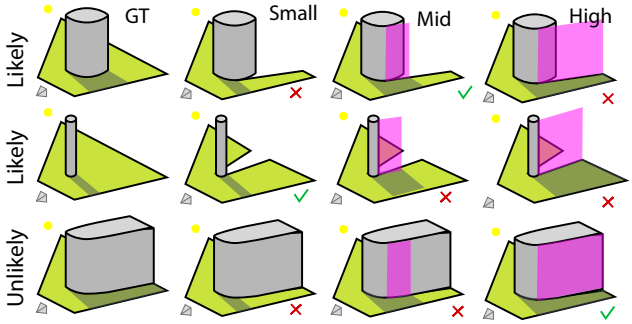

Figure 5: *Different geometries (row) and the respective cast shadows under different direct shadow casting thresholds (columns). The camera is to the left and the light in the back. The top two geometries are common cylinders, but there is no single threshold (pink) that would result in the correct value for both cases. Our method learns to make the decision adapted to the color and depth context. The last row shows that the solution is ambiguous, as the depth map observed from the camera is identical to the wide cylinder. Our network would likely learn to cast a shadow similar to a cylinder.*

Secondly, even if an oracle NN would reconstruct depth values perfectly, these will remain *incomplete*, because a 2D depth map can only store a single depth for every pixel. If we see a frontal RGB image of a box, we cannot know how far it extends behind every pixel. This information, however, is critical to cast a proper shadow. A thin box will cast a thin shadow, a thick box will cast a thick shadow. Hence, ideally we would not only need to know the depth of the surface, but also have a notion of thickness. This is illustrated in Fig. 4 (Col. 2), where we visualize a shadow obtained by naively intersecting a surface created by a depth map if we assume all geometry extends indefinitely away from the camera. We show the shadow obtained by naively intersecting the surface created by the depth if we meshed it, that is, whenever a ray pass behind an observed surface it is considered in shadow. Other possibilities to compute shadows from depth maps using thresholding are shown in Fig. 5.

While the task is ill-posed (Fig. 5 last row), NNs have demonstrated a great ability at building deep priors to infer 3D geometry from only a single view-point [HMR19, NPLT*19], given a large enough set of examples. We often use such priors ourselves, for instance, a vertically elongated object like a lamp post might be an "accidental" view of a very long wall from the side (Fig. 5 last row), but in expectation over a dataset, it is just what it is, a lamp post, casting a lamp post shadow. Similarly, when observing the front of a building, we assume a thick filled volume behind it; we could be observing a movie set with fake buildings, but this is very unlikely.

Possible solutions The most straightforward way to take into account the long-range interactions caused by occluding geometry given a depth map would thus be to ray-march the depth buffer to compute shadows [RGS09]. More precisely, to shade a point, all pixels that fall on the ray toward the light source are considered. If the depth buffer is sufficiently close (e. g., under a specified threshold) to the corresponding depth of the 3D ray for any of these pixels, the point is considered in shadow. Using a small threshold works well as long as the light direction is relatively orthogonal to the camera plane as most of the useful information is contained in the depth map. However, these methods cannot properly deal with unobserved thickness when the light is coming from the side or the top of the image as, there, the threshold plays an important role (Fig. 5). A real example illustrates this issue in Fig. 6. The top of the aqueduct not being observed leads to underestimation of shadow regions. We later refer to this approach as DIRECT.

On the other hand, if we were to accept shadow prediction to be a task worth learning, the most straightforward way would be to employ an image-to-image translation network from the depth and color image to the shadow image [NAM*17]. We will indeed study this ablation (2D), but it has limited quality due to two main reasons. Firstly, convolutions in an encoder-decoder take into account the 2D context at different scales, but for a shadowing task, the context that matters from a query point, in a certain direction, is 1D, along a ray. Meaning that the relevant information can be arbitrarily far from the shadowed point. Secondly, it does not make use of the physics of shadows. They are a combination of attached shadows (also known as self shadows), which depends on local surface orientation and cast shadows, that are produced by the presence of an occluder that can be arbitrarily distant.

Accounting for the duality of shadows (self and detached shadows), the partial effectiveness of ray-marching and recent advances in deep learning inspired our solution.

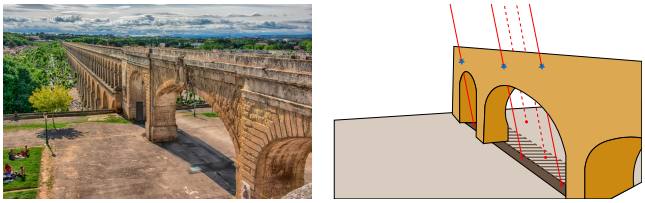

Figure 6: *Illustration of the problem caused by casting shadows using ray-marching directly in the depth map. Left: an input image. Right: a schematic representation of surface-based shadow casting. While the plain brown zone is correctly classified as shadow, casting rays from the hatched one toward the sun does not produce intersections with the observed surface resulting in the incorrect classification of this zone as non-shadow.*

Our solution Our solution computes both the attached and the detached shadow in a single network. Instead of predicting binary shadow images, we use the product of the binary mask with the cosine term (that we later refer to) as the new shadow image S_{n}.

As shown in Fig. 7, this has three main advantages. Firstly, it prevents creating arbitrary high frequency cut-offs on smooth surfaces as the cosine term smoothly goes to zero before the mask does. Secondly, it contains part of the surface shading, giving cues about direct light intensity change. Finally, as we have two different lighting conditions, we also have information regarding the difference of intensity between shadow and non-shadow regions. When the light is at a grazing angle (bottom row Fig. 7), our representation encodes that the difference between shadow and non-shadow region is smaller than when the light is at a higher angle (top row Fig. 7).

To predict a shadow image we first observe that we can provide to the network an approximation of the cosine term, thus helping with attached shadows, by computing the clamped dot product of the light direction and the approximate normal image $\max(0, \langle N, \omega_{\mathrm{n}} \rangle)$. This guide is therefore as inaccurate as the normal map and is a prior that will need to be refined by the network. To do so we concatenate

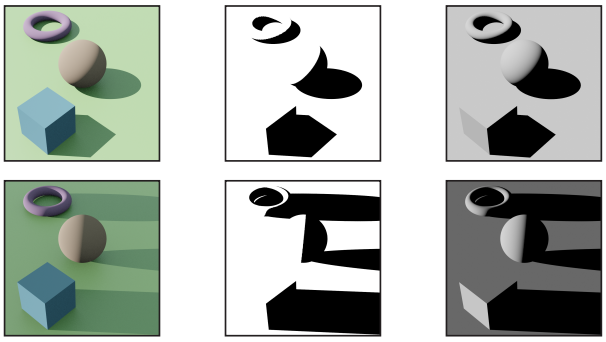

Figure 7: *From left to right: A rendering of an example scene, binary shadow images and our proposed representation: cosine terms times shadow images. We show two illumination directions, a high lighting (top) and lighting at a grazing angle (bottom).*

this term with the original RGB image. We later refer to this four channels image as the "2D features":

$$2D_{\mathrm{feat}} = \mathtt{cat}[C_{\mathrm{o}}, \max(0, \langle N, \omega_{\mathrm{n}} \rangle)] \tag{2}$$

As discussed, shadows *cast* from arbitrarily distant occluders are more difficult to handle. The key insight of this paper is that the information to be considered for deciding if a point is in cast shadow relies on all the image positions, and their respective local neighbors, that fall onto a ray from that point in the direction to the light, i. e., the same pixels used for casting shadows with ray-marching.

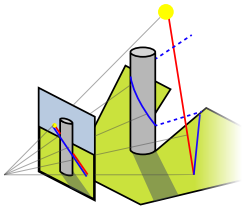

Figure 8: *A cylinder casting a shadow and its projection. The red line shows the depth of the ray while the blue line shows the depth sample in 2D ray-marching.*

Fig. 8 shows a cylinder and two associated depths along a light ray, the one from the observed surface (blue) and the one from the 3D ray (red). To answer if a point is in shadow, we have to consider all points on a ray from that point to the light. The most relevant information available to evaluate this is in the depth and color image along the geometry of that ray. An ideal method would classify points as occluded or not occluded, but using depth information directly, results in the errors previously mentioned seen in Fig. 6 if ray-marching against the implied surface. Instead, we also rely on color, as well as on nearby depth values to account for the full spatial arrangement. Below we detail this process.

For every image pixel, we ray-march $z = 256$ steps in the 2D direction, defined by the projection of the 3D light direction. We

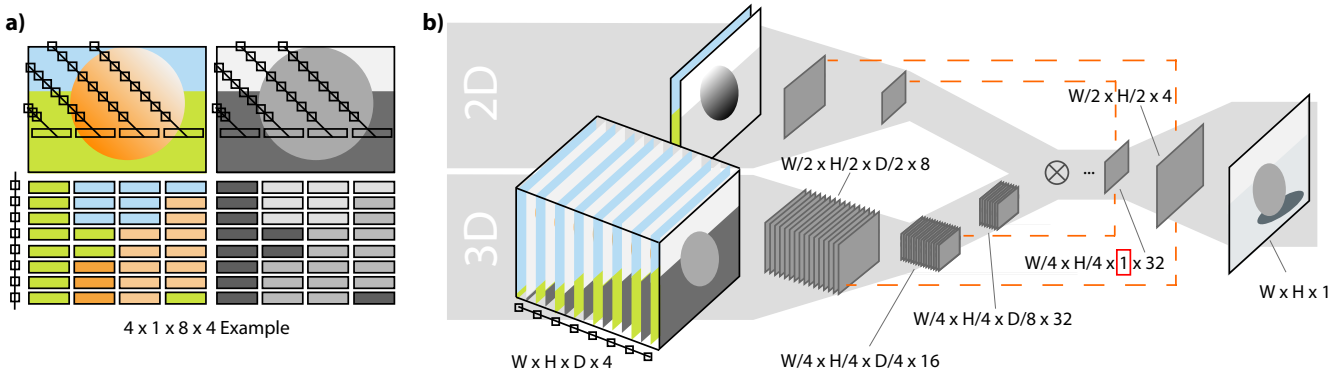

Figure 9: *Our ray-marching procedure. Left (a), we show 4 pixels marched in the RGB and depth map into the direction of the light with 8 steps. For that row, this results in a 3D image of 8 layers, one for each ray-marching step. This epipolar volume is fed into a 3D encoder branch that reduces all dimensions by half in three steps while doubling feature count (b). We opt for this approach to allow the network to take into account immediate local neighbors near the ray but restrict the network from computing spatially global features across the image plane. This is followed by steps that keep spatial resolution in the volume height and width, but reduce the depth dimension to 1. This is decoded into a 2D image. An additional 2D branch is provided the approximate Lambertian term, that is 2D-encoded. On the decoding step the 2D and 3D branches share features Orange lines indicate skip connections.*

sample the input pixel depth and color values at that position and store them into separate channels. Color values are stored as they have been shown to aid learning a per-pixel object thickness value which is important for shadow estimation [NCL19]. Instead of storing depth directly, we store the ratio between the depth in the depth map (Fig. 8, blue line) and the depth of the point along the ray (Fig. 8 red line). This ratio is attractive as it is the quantity that a normal ray-marching would use to decide occlusion. When this ratio is very close to 1, it means that the ray intersects the surface, when it is greater than one it means the ray is in front of the observed surface and behind it when the ratio is smaller than one, which is the case for the points on the cylinder in Fig. 8. Second, this ratio is scale-invariant. Scale-invariance is very important in our scenario as depth-estimators learn disparity, and not absolute depth values, so are themselves also scale independent.

We stack all z features, resulting from the z steps of the ray-marching, into a volume of size $x \times y \times z \times 4$, the four channels being RGB and depth ratios. We refer to this volume as the 3D features $3D_{feat}$. Similar ideas have been used to find correspondences between pairs of images, where for each point in one image, the epipolar line in the other image is marched [HYFY20]. Instead, here we march the epipolar image line of a point with respect to the light.

Finally, a new 3D-2D encoder-decoder with two encoder heads maps the $x \times y \times z \times 4$ $3D_{\text{feat}}$ volume to a bottleneck size of $x' \times y' \times 1 \times n$ using 3D convolution in the 3D head. Intuitively, applying 3D convolution to the proposed volume allows to traverse along the ray direction while also accounting for local context.

In parallel, the 2D features $2D_{\text{feat}}$ are also encoded to a $x' \times y' \times n$ image using standard 2D convolutions.

Finally, a decoder maps the max of the encoded 2D and 3D features back to an image of size $x \times y$ with a single channel using 2D convolutions. Our network architecture is illustrated in Fig. 9 and outlined in detail in the supplementary material. For the skip connections from the 3D branch to the 2D decoder we use a linear upsampling layer to match the volume sizes. This network is asked to match the ground truth new shadow image:

$$f_{\mathcal{S}}(C_{\text{o}}, \omega_{\text{n}}) = \Delta(\mathcal{S}^{\theta}(C_{\text{o}}, \omega_{\text{n}}), S_{\text{n}}) \tag{3}$$

as the loss function Δ we use E-LPIPS [KHL19], it led to more plausible outputs as it is much more resilient to small errors in shadow boundaries than Mean-square Error (MSE) which led to smooth shadow when we tested it.

4.2. Extracting the original shadow

In order to get the source image shadow in the source light direction ω_{o}, we could simply execute $\mathcal{S}$ for that direction i. e., $\mathcal{D} = \mathcal{S}$. Unlike the target direction ω_{n} that is user defined, the original light direction ω_{o} is unknown. To obtain it at test time, we rely on a simple user interface which shows the output of $\mathcal{S}$. The user is asked to roughly align the predicted shadows with the ones in the input image. We show a demonstration of this process in the supplemental video. The interface could also be initialized with outdoor light estimation methods such as [HGAL19, LMF*19]. Furthermore, during training we artificially add noise to the ground truth light direction to make the model more robust to inaccuracies from the user. Whilst running the input and guide images through the $\mathcal{S}$ gives us a shadow image, it misses the opportunity to refine the shadow with the assumption that the very shadow we look for are present in the color image C_{o}, albeit entangled with albedo. In light of this, to make use of both the shadow in the image, and the shadow predicted by the network, a separate network $\mathcal{R}$ is trained, with the sole purpose of refining the output of $\mathcal{S}$, while also accessing the color image, such that $\mathcal{D}(C_{\text{o}}, \omega_{\text{o}}) = \mathcal{R}(\mathcal{S}(C_{\text{o}}, \omega_{\text{o}}), C_{\text{o}}, \omega_{\text{o}})$. $\mathcal{R}$ is supervised, with RGB MSE as here the shadow boundaries are available in the original image and this image needs to be accurate for better shadow removal rather than just plausible:

$$f_{\mathcal{D}}(C_{\text{o}}, \omega_{\text{o}}) = \Delta(\mathcal{D}^{\theta}(C_{\text{o}}, \omega_{\text{o}}), S_{\text{o}}). \tag{4}$$

4.3. Relighting

With both shadow images at our disposal, we can now perform a relighting closely inspired by Philip et al., [PGZ*19] which is a rich image(s)-to-image translation.

Input to the relighting network $\mathcal{L}$ are the two shadow images S_n S_o, the old color image C_o, the computed normals N, as well as the old and new light direction ω_o and ω_n along a direction map Ψ containing the x, y, z direction from the camera center towards each pixel center. The depth is not an input to the relighting network. Output is the final color image C_n. Fig. 3 shows the complete network architecture.

The loss function penalizes the color image in the new light condition $\mathcal{L}(C_\mathrm{o}, \omega_\mathrm{o}\omega_\mathrm{n})$ to match a known reference color image C_n in the new light conditions, so

$$f_\mathcal{L}(C_\mathrm{o}, \omega_\mathrm{o}, \omega_\mathrm{n}) = \Delta(\mathcal{L}^\theta(C_\mathrm{o}, \omega_\mathrm{o}, \omega_\mathrm{n}), C_\mathrm{n}) \tag{5}$$

where Δ is E-LPIPS [KHL19].

4.4. Generating realistic images and shadows

Additionally, we employ a Patch-based Least-squares GAN (LSGAN) [MLX*17] with a 64 square pixel receptive field to enforce the results to align with the distribution of natural images. Instead of conditioning it only on the output image, we also condition it on the original RGB image C_o and the new shadow image S_n. This adversarial loss has three main effects. Firstly, it helps with the overall visual quality and sharpness of the images. Secondly, it improves shadow removal. Our intuition is that by seeing the input and output image the discriminator should be able to detect bad shadow removal. Lastly, having the shadow image as part of the conditioning helps with shadow details and coherency between predicted shadows and the output image.

4.5. Training data

Creating a real-world training dataset for our method with the full distribution of lighting conditions our model accounts for would be exceptionally challenging. Instead, we train our method entirely on synthetic scenes. This enables us to simulate the full spectrum of lighting conditions in a large variety of settings. We utilize 40 Evermotion Archexterior [eve21] scenes for geometry, material and texture of the scenes.

For each scene a camera path is manually created, from which we sample 256 view-points. For each view-point we select an object for the camera to look at, a random focal length and render the scene under 16 lighting conditions at 1024×768 resolution. In total we render 164k images from the 40 scenes at 128 samples per pixel using the Cycles [Com18] path tracer. For lighting, all images are rendered using the Nishita sky model [NSTN93] which implements atmospheric scattering to which we added volumetric clouds to handle their relighting in real images. Using this realistic sky model, clouds and the direction map Ψ allows the relighting network to implicitly detect and relight the sky. Ψ is particularly helpful in removing and synthesizing the sun when it is directly visible in the image as the network can match the old or new sun direction ω_o and ω_n with the direction towards which each pixel points, which is given by Ψ. An illustration of such synthesis and cloud handling is visible in Fig. 1, "Relighting A".

On top of the path-traced RGB image, each individual sample contains a ground truth depth map, normal image and its corresponding shadow image. This shadow image is rendered by computing single bounce direct illumination of the scene, replacing all the materials with white Lambertian BRDF, as shown in Fig. 7, third column. Training examples are are also shown in the supplemental materials.

Out of screen shadow casters Our ray-marching based model is not able to handle shadows cast by objects that are not visible in the image. Thus, we cull all the geometry outside of the screen before rendering each training image. This means none of our training examples exhibits shadows cast by out of screen objects. While at test time the original image may contain such shadows, we found that the network has learnt to remove them correctly. As for the new shadows, we empirically find that synthesizing shadows cast only by the visible content provides sufficient realism in most cases.

4.6. Training strategy and details

Training procedure At train time we sample a viewpoint from the dataset of rendered scenes and a random pair of lighting conditions to define the input C_o and target image C_n. These images are stored in linear space and similar random exposure, saturation and gamma tone-mapping augmentations are applied to both C_o and C_n.

Learning to trust the depth As previously mentioned, predicted depth maps are often distorted. This means that shadows cast by predicted depth maps are often miss-aligned with the ground truth ones, though they may look realistic. When training our cast shadow network only with predicted depth maps, this phenomenon led to very noisy gradients and poor convergence quality. Trying to correct this distortion would be equivalent to trying to beat the best monocular depth estimator. Instead, we teach the cast shadow network to trust the depth maps by training it with a combination of ground truth and NN estimated depth maps. Each training step flips a 80 %-to-20 %-biased coin to determine if it learns new cast shadows from ground truth or estimated depth respectively. Both the old shadows and normals are always computed on the estimated depth maps. In doing this, our cast shadow network learns geometrically founded features (e. g., the relationship between surface normals, light direction and attached shadows), however, is also robust to noisy normals and inaccurate cast shadows computed for the old light direction ω_o and allows to transfer to real data more easily.

Optimization details For the optimization we use the Adam Optimizer [KB15] with a learning rate of $1e-4$ both for the generator and the discriminator. We alternate five steps of generator for one step of discriminator. The loss is weighted so each subnetwork has approximately equivalent losses at the start of training, $\lambda_\mathcal{S} = 10$,$\lambda_\mathcal{D} = 2$,$\lambda_\mathcal{L} = 10$, the adversarial loss is set lower with $\lambda_\mathcal{A} = 0.1$. We train on 384x384px images and a batch size of 4 patches.

An important detail regarding the optimization procedure is visible in Fig. 3. The small scissors denote that the gradients do not flow backward from the refinement network $\mathcal{D}$ to the cast shadow

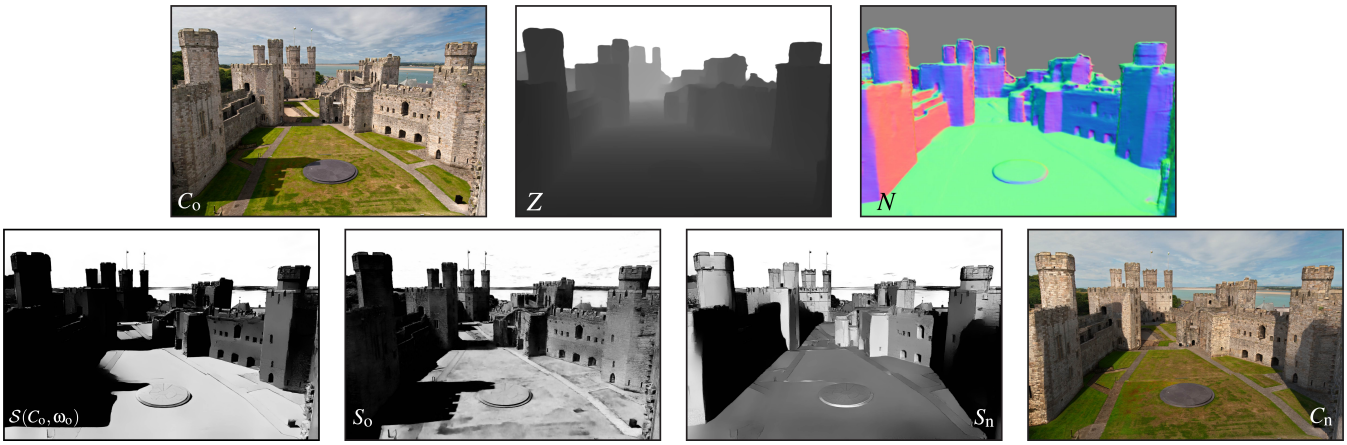

Figure 10: *Visualization of different intermediate buffers for a real image relighting example. Top row, from left to right: Input image C_o, Estimated Depth Z, Normals N. Bottom row, from left to right: Estimated Source Shadows $\mathcal{S}(C_o, \omega_o)$, Refined Source Shadows $\mathcal{D}(C_o, \omega_o)$, Estimated Target Shadows $\mathcal{S}(C_o, \omega_n)$, Output C_n.*

network $\mathcal{S}$. If we were not to do this, half of the gradients of $\mathcal{S}$ are from predicting the original shadows and $\mathcal{S}$ learns to copy the shadows from the original image C_o.

Network Architectures The 3D branch of the cast shadow network $\mathcal{S}$ is composed of 3D convolutions with kernel sizes (4,4,4) allowing to effectively divide by 8 the volume size after each convolution while we multiply the number of features by 2. After 3 down-samplings, additional convolutions with kernel size (4,3,3) reduce the depth dimension to 1.

All other encoder and decoders are CNN inspired by the architecture from Pix2Pix HD [WLZ*18]. They are U-Net-like [RFB15], but composed of residual blocks and residual skip connections.

A detailed blueprint of the system with network architectures is available in the supplemental materials.

5. Experiments and Results

We test our approach on both a set of reserved synthetic scenes, allowing for quantitative assessment on ground truth data (Sec. 5.2), as well as qualitatively on real-world images (Sec. 5.1). Our experiments are formed on a set of ablations of our method, justifying the benefit of our proposed learned 3D ray-marching module.

5.1. Qualitative

We tested our method on a range of photos, typically of outdoor architecture shown in Fig. 11.

Fig. 12 shows examples of results for all methods we compare to, giving qualitative insight into the performance of our approach. We perform favorably to SELF RELIGHTING in all scenarios. Most notably is our ability to predict detached (cast) shadows and our ability to remove source shadows. When comparing to Philip et al.,[PGZ*19] we find our method to produce stronger shadows in the target image.

In Fig. 13, we apply our method to paintings. Despite the shadows not always been physically accurate, our network still creates plausible outputs. The ability for our network to perform well in such a large domain of inputs highlights its generalization abilities.

5.2. Quantitative

In this section we perform an ablation study on our method. All test samples are drawn from scenes that were not available at training time. This eliminates the chance the network will have seen similar view-points of the test images at train time. We describe our ablation methods below.

PIX2PIXHD-LIKE An image-to-image translation network. The source image C_o is directly mapped to the target image C_n using a CNN U-Net architecture [IZZE17]. In practice, for fairness, we use our relighting network $\mathcal{L}$ with all its inputs except for the computed old and new shadow images. The relighting approach is defined as $C_n = \tilde{\mathcal{L}}(C_o, \omega_o, \omega_n)$.

2D The above method ignores any shadow guides for training. Here, we replace our 3D shadow estimation network $\mathcal{S}$ with a 2D CNN similar to the refinement network $\mathcal{D}$, its input are the input image C_o, light direction ω_o or ω_n, the corresponding cosine term and the estimated depth map.

DIRECT We replace our 3D shadow network with a non-learnable direct screen-space ray marching algorithm [RGS09]. Shadow images are computed directly from the estimated depth map and input to the final relighting network. The samples collected for this method are the same samples passed into our learnable 3D shadow network.

OUR Our full method, as described in Sec. 4.

All methods make use of depth extracted by $\mathcal{Z}$, as described in Sec. 4 and no method has access to the ground truth depth or normals, only to RGB and source light direction ω_o.

Metrics As test data is rendered, we know the correct new image C_n' and hence can compute the image error, here using Structural

Figure 11: *Results of our proposed approach for three novel illuminations (three right columns), on a variety of challenging real-word scenes (first column). For more results, including full time lapse videos, please see the supplemental video.*

Figure 12: *Comparison of our methods against Self-supervised Outdoor Relighting [YME*20] and [PGZ*19].* OURS *performs favorably over* SELF RELIGHTING *in all scenarios. We perform on-par to Philip et al., [PGZ*19], however, do not require multi-view inputs. Self-relighting is unable to remove the old shadow, as seen particularly in the top scene.*

Table 1: *Relighting error for different ablations across the test dataset according to different metrics. For all methods with shadow prediction, we also report the MSE. For all metrics, less is better.*

Method	Relight			Shadow
	DSSIM↓	LPIPS↓	MSE↓	MSE↓
PIX2PIX	.165	.0170	.0427	—
2D	.169	.0182	.0515	.187
DIRECT	.157	.0169	.0409	.358
OUR	**.154**	**.0160**	**.0399**	**.175**

Dissimilarity Index (DSSIM), Mean Squared Error (MSE) and Perceptual Similarity (LPIPS). To further evaluate the networks ability to predict shadows, we also report the MSE of the predicted and ground truth shadow images. Note, we cannot do this for PIX2PIX as no intermediate shadows are produced.

Results Results are summarized in Tbl. 1. We see that our approach performs best according to all metrics, both in the image error, as well as for the intermediate shadow. Additional visual evidence is shown in Fig. 14 and discussed in the caption.

6. Limitations

Our method is subject to several assumptions. Firstly, we require the input image light direction. While we use a manually derived input for the results shown in this paper, the proposed system would benefit from more reliable methods to estimate this.

In all scenarios, we assume a dominant single light source, typically, the sun. Extension to a mixture of light sources is straightforward (as light sums linearly) but a more refined solution, e. g., a latent model of illumination, will certainly outperform this.

Currently, we model classic opaque shadows in our ray-marching. Hence, all other shading effects, like reflections, colored shadows, indirect light or caustics are not modeled explicitly, but instead left to the final shading network to approximate. This works, as long as such effects do not become visually dominant. Relighting a modern office interior (mirror reflections), a church interior (colored shadows), a glass vase (caustics) or strong indirect lighting would require both adequate training data, but also likely require adapted guide signals, like we provided for shadows. Whilst our method does not allow for control of shadow softness, this would be an easy extension, as shadow softness can be controlled in the training data.

As discussed in Sec. 4.5, our method assumes there are no shadows cast from out of screen geometry. Although it is conceivable

Figure 13: *Application to artwork. The first column shows the original, the two left columns our result. Both inputs show strong, while not yet entirely physically-correct shadow. The top one is "Odysseus returns Chryseis to Her Father" (ca. 1644) by Claude Lorraine (1604–1682). The bottom one is "The square of Saint Mark's, Venice" (ca. 1723) by Giovanni Antonio Canal (1697–1768) known as Canaletto.*

that such geometries can be reconstructed from their shadows alone, we do not address such scenarios in this work.

Finally, whilst all the results presented in this work are generated using a single model state (including Fig. 13), our network components are learnt through a data-driven approach. Therefore, for some test images where the source image is far from the training distribution our results degenerate. This is, however, more a limitation of our training data than the system architecture.

7. Conclusion

We have shown how the classic idea of ray marching, combined with a learned component, allows to cast shadows in RGB images resulting in faithful outdoor relighting of single-view images. This is made possible by using geometry provided from an off-the-shelf monocular object detector. Our method compares favorably to previous work as well as to strong baselines of ablations. When looking at the evolution of screen space shading however, it appears quite conceivable that the idea of a (differently) ray-marched guide is applicable to all of these in future work. Beyond relighting, handling physically based long-range interactions from a single image, our key technical innovation, might have applications in other graphics tasks such as image material editing and even lead to novel forms of (self) supervision using physical long-range constraints in both vision and graphics.

References

[ALKN19] AZINOVIC D., LI T.-M., KAPLANYAN A., NIESSNER M.: Inverse path tracing for joint material and lighting estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 2447–2456.

[BBJ*21] BOSS M., BRAUN R., JAMPANI V., BARRON J. T., LIU C., LENSCH H.: NerRD: Neural reflectance decomposition from image collections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 12684–12694.

[BCD*21] BARUCH G., CHEN Z., DEHGHAN A., DIMRY T., FEIGIN Y., FU P., GEBAUER T., JOFFE B., KURZ D., SCHWARTZ A., SHULMAN E.: Arkitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data.

[BKPB17] BONNEEL N., KOVACS B., PARIS S., BALA K.: Intrinsic decompositions for image editing. In *Computer Graphics Forum* (2017), vol. 36, Wiley Online Library, pp. 593–609.

[BXS*20a] BI S., XU Z., SRINIVASAN P. P., MILDENHALL B., SUNKAVALLI K., HASAN M., HOLD-GEOFFROY Y., KRIEGMAN D. J., RAMAMOORTHI R.: Neural reflectance fields for appearance acquisition. *arxiv:2008.03824* (2020).

[BXS*20b] BI S., XU Z., SUNKAVALLI K., HASAN M., HOLD-GEOFFROY Y., KRIEGMAN D. J., RAMAMOORTHI R.: Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. *arXiv:2007.09892* (2020).

[Com18] COMMUNITY B. O.: *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018.

[CVJR19] CARLSON A., VASUDEVAN R., JOHNSON-ROBERSON M.: Shadow transfer: Single image relighting for urban road scenes. *arXiv:1909.10363* (2019).

[Deb02] DEBEVEC P.: Image-based lighting. *IEEE Computer Graphics and Applications 22*, 2 (2002), 26–34.

Figure 14: *Qualitative results of our ablation study. In general* PIX2PIXHD-LIKE *achieves reasonable global relighting, however, performs poorly at both shadow removal and placement.* 2D *is not capable of long-range detach shadows as standard 2D convolutions are unsuitable for such a task.* DIRECT *is the most similar to* OUR *method, however, suffers from underestimated shadows.*

[DHT*00] DEBEVEC P., HAWKINS T., TCHOU C., DUIKER H.-P., SAROKIN W., SAGAR M.: Acquiring the reflectance field of a human face. In *Proc. SIGGRAPH* (2000), pp. 145–156.

[DRC*15] DUCHÊNE S., RIANT C., CHAURASIA G., LOPEZ-MORENO J., LAFFONT P.-Y., POPOV S., BOUSSEAU A., DRETTAKIS G.: Multi-view intrinsic images of outdoors scenes with an application to relighting. *ACM Trans Graph (Proc SIGGRAPH Asia) 34*, 5 (2015).

[DYK*14] DOU H., YAN Y., KERZNER E., DAI Z., WYMAN C.: Adaptive depth bias for shadow maps. In *Proc I3D* (2014), pp. 97–102.

[ED04] EISEMANN E., DURAND F.: Flash photography enhancement via intrinsic relighting. In *ACM Trans Graph* (2004), vol. 23, ACM, pp. 673–678.

[EF15] EIGEN D., FERGUS R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV* (2015), pp. 2650–2658.

[EGH21] EINABADI F., GUILLEMAUT J.-Y., HILTON A.: Deep neural models for illumination estimation and relighting: A survey. In *Comp Graph Forum* (2021).

[eve21] Evermotion archexteriors, 2021.

[GHGS*19] GARDNER M.-A., HOLD-GEOFFROY Y., SUNKAVALLI K., GAGNE C., LALONDE J.-F.: Deep parametric indoor lighting estimation. In *ICCV* (2019).

[GLD*19] GUO K., LINCOLN P., DAVIDSON P., BUSCH J., YU X., WHALEN M., HARVEY G., ORTS-ESCOLANO S., PANDEY R., DOURGARIAN J., TANG D., TKACH A., KOWDLE A., COOPER E., DOU M., FANELLO S., FYFFE G., RHEMANN C., TAYLOR J., DEBEVEC P., IZADI S.: The relightables: Volumetric performance capture of humans with realistic relighting. *ACM Trans. Graph. 38*, 6 (2019).

[HGAL19] HOLD-GEOFFROY Y., ATHAWALE A., LALONDE J.-F.: Deep sky modeling for single image outdoor lighting estimation. In *CVPR* (2019).

[HGSH*17] HOLD-GEOFFROY Y., SUNKAVALLI K., HADAP S., GAM-

BARETTO E., LALONDE J.-F.: Deep outdoor illumination estimation. In *CVPR* (2017).

[HMR19] HENZLER P., MITRA N. J., RITSCHEL T.: Escaping Plato's cave: 3D shape from adversarial rendering. In *CVPR* (2019), pp. 9984–9993.

[HYFY20] HE Y., YAN R., FRAGKIADAKI K., YU S.-I.: Epipolar transformers. In *CVPR* (2020), pp. 7779–7788.

[ISR12] ISAZA C., SALAS J., RADUCANU B.: Evaluation of intrinsic image algorithms to detect the shadows cast by static objects outdoors. *Sensors 12*, 10 (2012), 13333–13348.

[IZZE17] ISOLA P., ZHU J.-Y., ZHOU T., EFROS A. A.: Image-to-image translation with conditional adversarial networks. In *CVPR* (2017).

[JYP18] JAFARIAN Y., YAO Y., PARK H. S.: Monet: Multiview semi-supervised keypoint via epipolar divergence. *Unknown Journal* (2018).

[KB15] KINGMA D., BA J.: Adam: A method for stochastic optimization. *ICLR* (2015).

[KE18] KANAMORI Y., ENDO Y.: Relighting humans: occlusion-aware inverse rendering for fullbody human images. *ACM Trans Graph 37*, 270 (2018), 1–270.

[KHFH11] KARSCH K., HEDAU V., FORSYTH D., HOIEM D.: Rendering synthetic objects into legacy photographs. *ACM Trans Graph 30*, 6 (2011), 157:1–157:12.

[KHL19] KETTUNEN M., HÄRKÖNEN E., LEHTINEN J.: E-LPIPS: robust perceptual image similarity via random transformation ensembles. *arXiv:1906.03973* (2019).

[KNC*08] KOPF J., NEUBERT B., CHEN B., COHEN M. F., COHEN-OR D., DEUSSEN O., UYTTENDAELE M., LISCHINSKI D.: Deep Photo: Model-based photograph enhancement and viewing. *ACM Trans. Graph. (Proc. SIGGRAPH Asia 2008) 27*, 5 (2008).

[KSES14] KHOLGADE N., SIMON T., EFROS A., SHEIKH Y.: 3D object manipulation in a single photograph using stock 3D models. *ACM Trans Graph 33*, 4 (2014), 127.

[LDR00] LOSCOS C., DRETTAKIS G., ROBERT L.: Interactive virtual relighting of real scenes. *IEEE Trans Vis and Comp Graph 6*, 4 (2000), 289–305.

[LEN09] LALONDE J.-F., EFROS A. A., NARASIMHAN S. G.: Estimating natural illumination from a single outdoor image. In *ICCV* (2009), IEEE, pp. 183–190.

[LFD*99] LOSCOS C., FRASSON M.-C., DRETTAKIS G., WALTER B., GRANIER X., POULIN P.: Interactive virtual relighting and remodeling of real scenes. In *Rendering Techniques 99*. Springer, 1999, pp. 329–340.

[LGZ*20] LIU A., GINOSAR S., ZHOU T., EFROS A. A., SNAVELY N.: Learning to factorize and relight a city. In *ECCV* (2020).

[LLZ*20] LIU D., LONG C., ZHANG H., YU H., DONG X., XIAO C.: ARShadowGAN: Shadow generative adversarial network for augmented reality in single light scenes. In *CVPR* (2020), pp. 8139–8148.

[LM71] LAND E. H., MCCANN J. J.: Lightness and retinex theory. *J OSA 61*, 1 (1971), 1–11.

[LMF*19] LEGENDRE C., MA W.-C., FYFFE G., FLYNN J., CHARBONNEL L., BUSCH J., DEBEVEC P.: Deeplight: Learning illumination for unconstrained mobile mixed reality. In *CVPR* (2019), pp. 5918–5928.

[LSR*20] LI Z., SHAFIEI M., RAMAMOORTHI R., SUNKAVALLI K., CHANDRAKER M.: Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In *CVPR* (2020), pp. 2475–2484.

[LSY*21] LAGUNAS M., SUN X., YANG J., VILLEGAS R., ZHANG J., SHU Z., MASIA B., GUTIERREZ D.: Single-image full-body human relighting. In *Eurographics Symposium on Rendering (EGSR)* (2021), The Eurographics Association.

[MG97] MARSCHNER S. R., GREENBERG D. P.: Inverse lighting for photography. In *Color and Imaging Conference* (1997), vol. 1997, Society for Imaging Science and Technology, pp. 262–265.

[MHP*19] MEKA A., HAENE C., PANDEY R., ZOLLHÖFER M., FANELLO S., FYFFE G., KOWDLE A., YU X., BUSCH J., DOURGARIAN J., ET AL.: Deep reflectance fields: high-quality facial reflectance field inference from color gradient illumination. *ACM Transactions on Graphics (TOG) 38*, 4 (2019), 1–12.

[MLX*17] MAO X., LI Q., XIE H., LAU R. Y., WANG Z., PAUL SMOLLEY S.: Least squares generative adversarial networks. In *ICCV* (2017), pp. 2794–2802.

[MST*20] MILDENHALL B., SRINIVASAN P. P., TANCIK M., BARRON J. T., RAMAMOORTHI R., NG R.: NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV* (2020).

[MSZ*21] MEKA A., SHAFIEI M., ZOLLHÖFER M., RICHARDT C., THEOBALT C.: Real-time global illumination decomposition of videos. *ACM Transactions on Graphics (TOG) 40*, 3 (2021), 1–16.

[NAM*17] NALBACH O., ARABADZHIYSKA E., MEHTA D., SEIDEL H., RITSCHEL T.: Deep shading: Convolutional neural networks for screen space shading. *Comp Graph Forum (Proc. EGSR) 36*, 4 (2017).

[NCL19] NICASTRO A., CLARK R., LEUTENEGGER S.: X-section: Cross-section prediction for enhanced rgb-d fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 1517–1526.

[NLML20] NESTMEYER T., LALONDE J.-F., MATTHEWS I., LEHRMANN A. M.: Learning physics-guided face relighting under directional light. In *CVPR* (2020), pp. 5123–5132.

[NPLT*19] NGUYEN-PHUOC T., LI C., THEIS L., RICHARDT C., YANG Y.-L.: HoloGAN: Unsupervised learning of 3D representations from natural images. In *ICCV* (2019), pp. 7588–7597.

[NSTN93] NISHITA T., SIRAI T., TADAMURA K., NAKAMAE E.: Display of the Earth taking into account atmospheric scattering. In *SIGGRAPH* (1993), pp. 175–182.

[PDB18] PRASAD V., DAS D., BHOWMICK B.: Epipolar geometry based learning of multi-view depth and ego-motion from monocular sequences. In *Proceedings of the 11th Indian Conference on Computer Vision, Graphics and Image Processing* (2018), pp. 1–9.

[PGZ*19] PHILIP J., GHARBI M., ZHOU T., EFROS A. A., DRETTAKIS G.: Multi-view relighting using a geometry-aware network. *ACM Trans Graph (Proc. SIGGRAPH) 38*, 4 (2019), 1–14.

[PMGD21] PHILIP J., MORGENTHALER S., GHARBI M., DRETTAKIS G.: Free-viewpoint indoor neural relighting from multi-view stereo. *ACM Trans Graph* (2021).

[POEL*21] PANDEY R., ORTS-ESCOLANO S., LEGENDRE C., HAENE C., BOUAZIZ S., RHEMANN C., DEBEVEC P., FANELLO S.: Total relighting: Learning to relight portraits for background replacement. *ACM Trans Graph (Proc. SIGGRAPH) 40*, 4 (2021).

[PTMD07] PEERS P., TAMURA N., MATUSIK W., DEBEVEC P.: Post-production facial performance relighting using reflectance transfer. In *ACM Trans Graph* (2007), vol. 26, ACM, p. 52.

[QTH*17a] QU L., TIAN J., HE S., TANG Y., LAU R. W.: Deshadownet: A multi-context embedding deep network for shadow removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 4067–4075.

[QTH*17b] QU L., TIAN J., HE S., TANG Y., LAU R. W. H.: DeshadowNet: A multi-context embedding deep network for shadow removal. In *CVPR* (2017).

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

[RFB15] RONNEBERGER O., FISCHER P., BROX T.: U-net: Convolutional networks for biomedical image segmentation. In *MICCAI* (2015), Springer, pp. 234–241.

[RGS09] RITSCHEL T., GROSCH T., SEIDEL H.-P.: Approximating dynamic global illumination in image space. In *Proc. ACM i3D* (2009), pp. 75–82.

[RLH*20] RANFTL R., LASINGER K., HAFNER D., SCHINDLER K., KOLTUN V.: Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Trans. Pattern Analysis and Machine Intelligence* (2020).

[SBT*19] SUN T., BARRON J. T., TSAI Y.-T., XU Z., YU X., FYFFE G., RHEMANN C., BUSCH J., DEBEVEC P. E., RAMAMOORTHI R.: Single image portrait relighting. *ACM Trans. Graph. 38*, 4 (2019), 79–1.

[SDZ*21] SRINIVASAN P. P., DENG B., ZHANG X., TANCIK M., MILDENHALL B., BARRON J. T.: Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *CVPR* (2021).

[SMB*20] SITZMANN V., MARTEL J. N., BERGMAN A. W., LINDELL D. B., WETZSTEIN G.: Implicit neural representations with periodic activation functions. In *NeurIPS* (2020).

[SMPR07] SUNKAVALLI K., MATUSIK W., PFISTER H., RUSINKIEWICZ S.: Factored time-lapse video. In *ACM Trans Graph* (2007), vol. 26, ACM, p. 101.

[SRSF19] SHIN D., REN Z., SUDDERTH E. B., FOWLKES C. C.: 3d scene reconstruction with multi-layer depth and epipolar transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 2172–2182.

[SSL12] SANIN A., SANDERSON C., LOVELL B. C.: Shadow detection: A survey and comparative evaluation of recent methods. *Pattern recognition 45*, 4 (2012), 1684–1695.

[STJ*04] STUMPFEL J., TCHOU C., JONES A., HAWKINS T., WENGER A., DEBEVEC P.: Direct HDR capture of the sun and sky. In *Afrigraph* (2004), ACM, pp. 145–149.

[SXZ*20] SUN T., XU Z., ZHANG X., FANELLO S., RHEMANN C., DEBEVEC P., TSAI Y.-T., BARRON J. T., RAMAMOORTHI R.: Light stage super-resolution: Continuous high-frequency relighting. *ACM Transactions on Graphics (TOG) 39*, 6 (2020), 1–12.

[SYH*17] SHU Z., YUMER E., HADAP S., SUNKAVALLI K., SHECHTMAN E., SAMARAS D.: Neural face editing with intrinsic image disentangling. In *CVPR* (2017), pp. 5444–5453.

[SZB21] SHENG Y., ZHANG J., BENES B.: SSN: Soft shadow network for image compositing. In *CVPR* (2021), pp. 4380–4390.

[TCE*21] TURE M., CIKLABAKKAL M. E., ERDEM A., ERDEM E., SATILMIS P., AKYUZ A. O.: From noon to sunset: Interactive rendering, relighting, and recoloring of landscape photographs by modifying solar position. *Comp Graph Forum 40*, 6 (2021), 500–515.

[TSE*04] TCHOU C., STUMPFEL J., EINARSSON P., FAJARDO M., DEBEVEC P.: Unlighting the Parthenon. In *ACM Siggraph 2004 Sketches* (2004), ACM, p. 80.

[Ull79] ULLMAN S.: The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences 203*, 1153 (1979), 405–426.

[WGT*05] WENGER A., GARDNER A., TCHOU C., UNGER J., HAWKINS T., DEBEVEC P.: Performance relighting and reflectance transformation with time-multiplexed illumination. In *ACM Trans Graph* (2005), vol. 24, ACM, pp. 756–764.

[WLY18] WANG J., LI X., YANG J.: Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In *CVPR* (2018), pp. 1788–1797.

[WLZ*18] WANG T.-C., LIU M.-Y., ZHU J.-Y., TAO A., KAUTZ J., CATANZARO B.: High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR* (2018).

[WS17] WU J.-H., SAITO S.: Interactive relighting in single low-dynamic range images. *ACM Trans. Graph. 36*, 2 (2017).

[WZL*09] WANG Y., ZHANG L., LIU Z., HUA G., WEN Z., ZHANG Z., SAMARAS D.: Face relighting from a single image under arbitrary unknown lighting conditions. *IEEE PAMI 31*, 11 (2009), 1968–1984.

[XSHR18] XU Z., SUNKAVALLI K., HADAP S., RAMAMOORTHI R.: Deep image-based relighting from optimal sparse samples. *ACM Trans Graphics 37*, 4 (2018), 126.

[YDMH99] YU Y., DEBEVEC P., MALIK J., HAWKINS T.: Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Proc. SIGGRAPH* (1999), pp. 215–224.

[YMB*19] YANG G., MALISIEWICZ T., BELONGIE S. J., FARHAN E., HA S., LIN Y., HUANG X., YAN H., XU W.: Learning data-adaptive interest points through epipolar adaptation. In *CVPR Workshops* (2019), pp. 1–7.

[YME*20] YU Y., MEKA A., ELGHARIB M., SEIDEL H.-P., THEOBALT C., SMITH W. A. P.: Self-supervised outdoor scene relighting. In *ECCV* (2020), pp. 84–101.

[YYTK21] YU A., YE V., TANCIK M., KANAZAWA A.: pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 4578–4587.

[YZW*21] YIN W., ZHANG J., WANG O., NIKLAUS S., MAI L., CHEN S., SHEN C.: Learning to recover 3d scene shape from a single image. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (CVPR)* (2021).

[Zha12] ZHANG Z.: Microsoft Kinect sensor and its effect. *IEEE multimedia 19*, 2 (2012), 4–10.

[ZHSJ19] ZHOU H., HADAP S., SUNKAVALLI K., JACOBS D.: Deep single-image portrait relighting. In *ICCV* (2019), pp. 7193–7201.

[ZLW19] ZHANG S., LIANG R., WANG M.: ShadowGAN: Shadow synthesis for virtual objects with conditional adversarial networks. *Computational Visual Media 5*, 1 (2019), 8.

[ZPIE17] ZHU J.-Y., PARK T., ISOLA P., EFROS A. A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV* (2017), pp. 2223–2232.

[ZSD*21] ZHANG X., SRINIVASAN P. P., DENG B., DEBEVEC P., FREEMAN W. T., BARRON J. T.: NeRFactor: Neural factorization of shape and reflectance under an unknown illumination. *arXiv:2106.01970* (2021).

[ZSHG*19] ZHANG J., SUNKAVALLI K., HOLD-GEOFFROY Y., HADAP S., , EISENMANN J., LALONDE J.-F.: All-weather deep outdoor lighting estimation. In *CVPR* (2019).

Table 2: *Relighting error for different loss related ablations across the test dataset according to different metrics. We also report the MSE for shadow prediction. For all metrics, less is better.*

Method	Relight			Shadow
	DSSIM↓	LPIPS↓	MSE↓	MSE↓
MSE loss for target shadows (no LPIPS)	.171	**.0159**	.0437	**.150**
No PatchGAN	.162	.0190	.0405	.169
Our	**.154**	.0160	**.0399**	.175

8. Supplementary Material

8.1. Detailed Network Architecture

In Fig. 15 we present a detailed architecture blueprint of our network with all inputs, modules, networks, and outputs.

8.2. Training Data Examples

In Fig. 16, we present random training examples from our dataset. For each viewpoint we provide two lighting conditions and their respective ground truth shadows.

8.3. Real world ground truth evaluation

We evaluate our method on a real world lighting scenario. To enable this we utilize separate images taken roughly from the same viewpoint with different lighting conditions. As shown in Fig. 17 we observe that whilst the network output looks plausible, the shadows are misaligned from the target ground truth. We believe this is due to distortions in the predicted depth estimation for this scene.

8.4. Further Ablations

In addition to the ablations presented in the main paper (Sec. 5.1 and Sec. 5.2), we also undertake further ablations specifically evaluating the use of specific loss function components. Tbl. 2 provides the quantitative evaluation for these ablation. As expected, the MSE loss is smaller for shadows when used as the training metric. Overall the pipeline appears to perform marginally better when the PatchGAN loss term is used which seems surprising. While small variations might be due to the randomness of the training process, it is possible that the PatchGAN loss helps escape local minima leading to better convergence. Moreover, as shown in Fig. 18, this loss helps in producing complex localized effects such as high frequency reflections on tree leaves (Fig. 18 first row) or reflection on the water and better looking clouds (Fig. 18 second row). Training with E-LPIPS [KHL19] for shadows does not provide a strong advantage numerically but has a strong impact on the sharpness on elongated shadows as can be seen in Fig. 19. Without the E-LPIPS loss and with a more traditional MSE loss, the shadow network tends to produce overly smooth shadows as a slight misalignment of boundaries is strongly penalized.

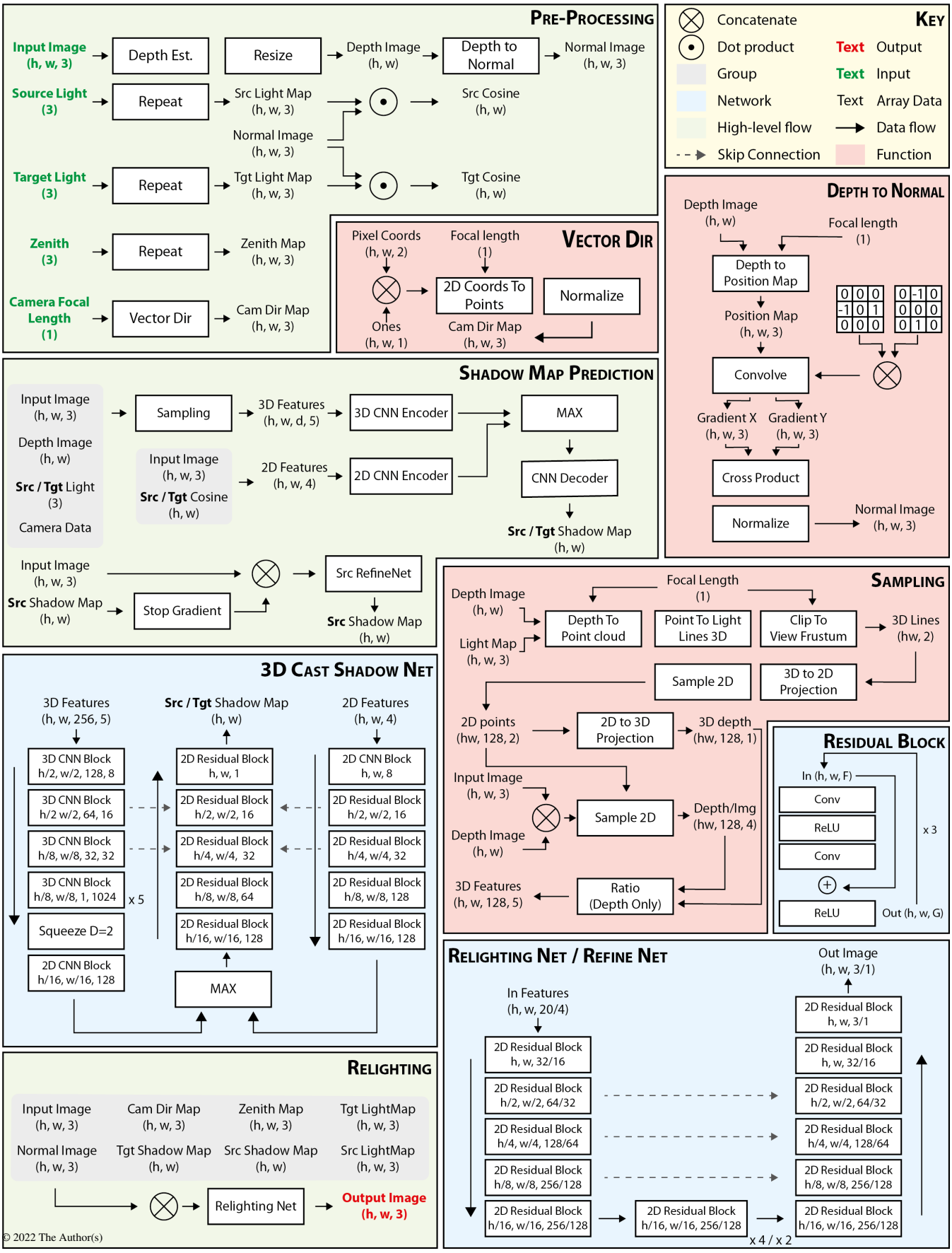

Figure 15: *Blueprint of our full relighting system. Where arrows are not included assume flow remains in the current direction. All deep learning-based functions were implemented in the PyTorch framework (v1.9).*

Figure 16: *Examples of training pairs. From left to right: Input, Ground Truth Source Shadows, Target, Ground Truth Target Shadows.*

Figure 17: *Comparison to real images with different lighting conditions. From left to right: Input, Our relighting, Ground Truth.*

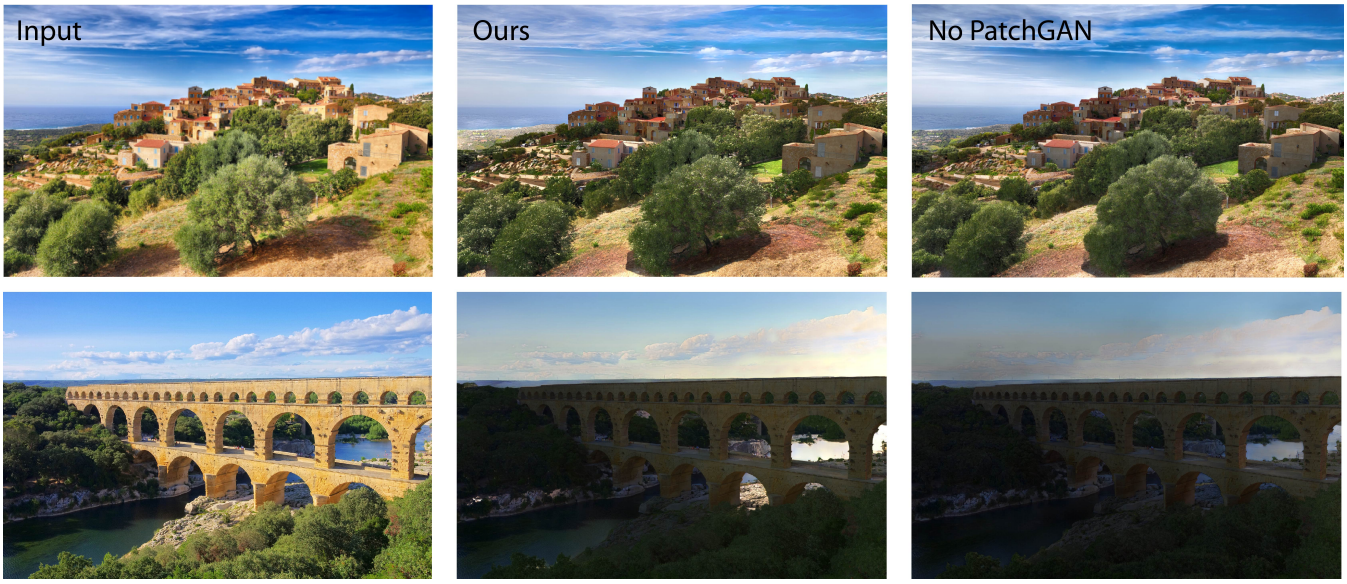

Figure 18: *Ablation results when removing the PatchGAN loss. From left to right: Input, Our relighting, Ablation result.*

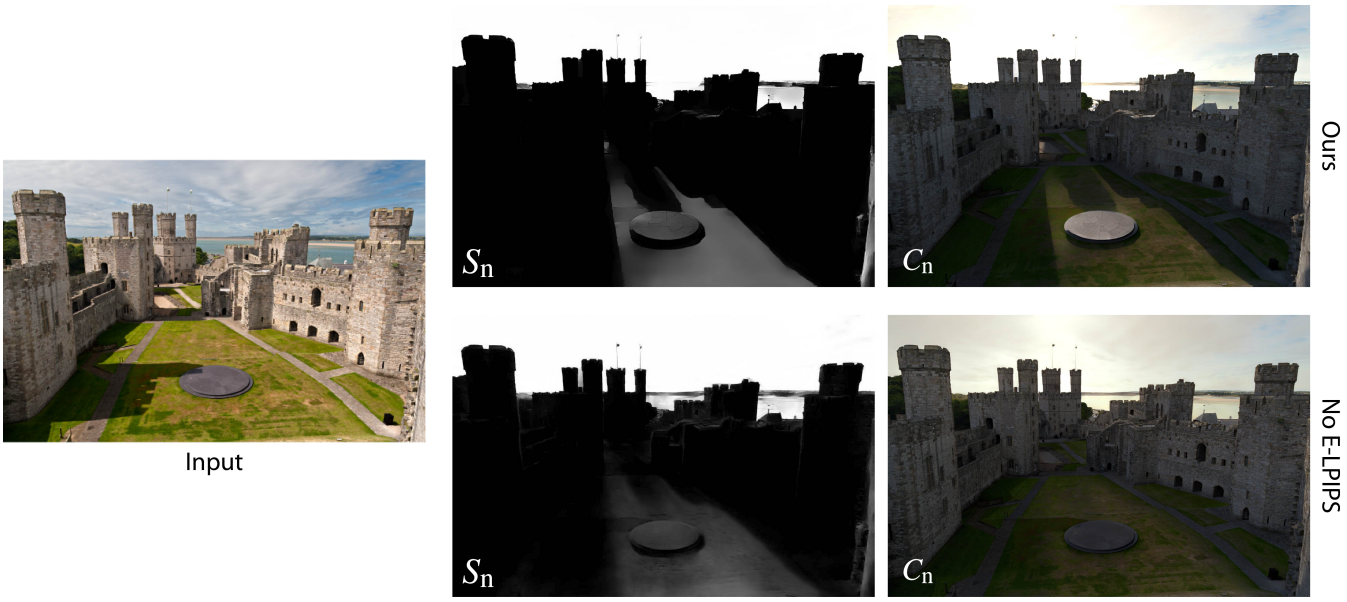

Figure 19: *Ablation results where the E-LPIPS loss is replaced by an MSE loss. Input to the network is shown on the left. (**top**) Ours for target shadow (left) and output (right). (**bottom**) Ablation for target shadow (left) and output (right).*