Separating Reflection and Transmission Images in the Wild

Patrick Wieschollek[1,2], Orazio Gallo[1], Jinwei Gu[1], and Jan Kautz[1]

[1]NVIDIA, [2]University of Tübingen

Abstract. The reflections caused by common semi-reflectors, such as glass windows, can impact the performance of computer vision algorithms. State-of-the-art methods can remove reflections on synthetic data and in controlled scenarios. However, they are based on strong assumptions and do not generalize well to real-world images. Contrary to a common misconception, real-world images are challenging even when polarization information is used. We present a deep learning approach to separate the reflected and the transmitted components of the recorded irradiance, which *explicitly* uses the polarization properties of light. To train it, we introduce an accurate synthetic data generation pipeline, which simulates realistic reflections, including those generated by curved and non-ideal surfaces, non-static scenes, and high-dynamic-range scenes.

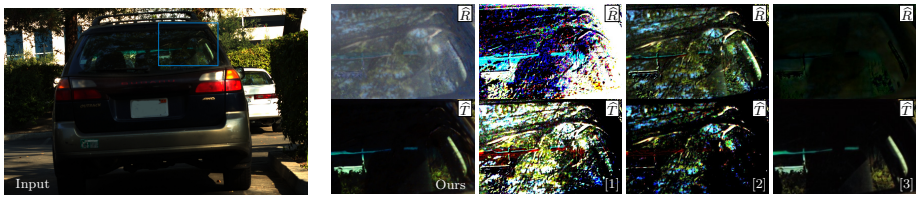


Fig. 1: Glass surfaces are virtually unavoidable in real-world pictures. Our approach to separate the reflection and transmission layers, works even for general, curved surfaces, which break the assumptions of state-of-the-art methods. In this example, only our method can correctly estimate both reflection $\widehat{R}$ (the tree branches) and transmission $\widehat{T}$ (the car's interior).

1 Introduction

Computer vision algorithms generally rely on the assumption that the value of each pixel is a function of the radiance of a single area in the scene. Semi-reflectors, such as typical windows or glass doors, break this assumption by creating a superposition of the radiance of two different objects: the one behind the surface and the one that is reflected. It is virtually impossible to avoid semi-reflectors in man-made environments, as can be seen in Figure 2(a), which shows



Fig. 2: Depending on the ratio between transmitted and reflected radiance, a semi-reflector may produce no reflections ❶, pure reflections ❷, or a mix of the two, which can vary smoothly ❸, or abruptly ❺. The local curvature of the surface can also affect the appearance of the reflection ❹. The last two, ❹ and ❺, are all but uncommon, as shown in (b).

a typical downtown area. Any multi-view stereo or SLAM algorithm would be hard-pressed to produce accurate reconstructions on this type of images.

Several methods exist that attempt to separate the reflection and transmission layers. At a semi-reflective surface, the observed image can be modeled as a linear combination of the reflection and the transmission images: $I_o = \alpha_r I_r + \alpha_t I_t$. The inverse problem is ill-posed as it requires estimating multiple unknowns from a single observation. A solution, therefore, requires additional priors or data. Indeed, previous works rely on assumptions about the appearance of the reflection (*e.g.*, it is blurry), about the shape and orientation of the surface (*e.g.*, it is perfectly flat and exactly perpendicular to the principal axis of the camera), and others. Images taken in the wild, however, regularly break even the most basic of these assumptions, see Figure 2(b), causing the results of state-of-the-art methods [2,3,4] to deteriorate even on seemingly simple cases, as shown in Figure 1, which depicts a fairly typical real-world scene.

One particularly powerful tool is is polarization: images captured through a polarizer oriented at different angles offer additional observations. Perhaps surprisingly, however, our analysis of the state-of-the-art methods indicates that the quality of the results degrades significantly when moving from synthetic to real data, *even when using polarization*. This is due to the simplifying assumptions that are commonly made, but also to an inherent issue that is all too often neglected: a polarizer's ability to attenuate reflections greatly depends on the viewing angle [5]. The attenuation is maximal at an angle called the Brewster angle, θ_B. However, even when part of a semi-reflector is imaged at θ_B, the angle of incidence in other areas is sufficiently different from θ_B to essentially void the effect of the polarizer, as clearly shown in Figure 3. Put differently, because of the limited signal-to-noise ratio, for certain regions in the scene, *the additional observations may not be independent*.

We present a deep-learning method capable of separating the reflection and transmission components of images captured *in the wild*. The success of the method stems from our two main contributions. First, rather than requiring a network to learn the reflected and transmitted images directly from the observations, we leverage the properties of light polarization and use a residual representation, in which the input images are projected onto the canonical polarization angles (Section 3.1 and 3.2). Second, we design an image-based data generator that faithfully reproduces the image formation model (Section 3.3).

We show that our method can successfully separate the reflection and transmission layers even in challenging cases, on which previous works fail. To further validate our findings, we capture the Urban Reflections Dataset, a polarization-based dataset of reflections in urban environments that can be used to test reflection removal algorithms on realistic images. Moreover, to perform a thorough evaluation against state-of-the-art methods whose implementation is not publicly available, we re-implemented several representative methods. As part of our contribution, we release those implementations for others to be able to compare against their own methods [1].

2 Related Work

There is a rich literature of methods dealing with semi-reflective surfaces, which can be organized in three main categories based on the assumptions they make.

Single-image methods can leverage gradient information to solve the problem. Levin and Weiss, for instance, require manual input to separate gradients of the reflection and the transmission [6]. Methods that are fully automated can distinguish the gradients of the reflected and transmitted images by leveraging the defocus blur [7]: reflections can be blurry because the subject behind the semi-reflector is much closer than the reflected image [4], or because the camera is focused at infinity and the reflected objects are close to the surface [8]. Moreover, for the case of double-pane or thick windows, the reflection can appear "doubled" [9], and this can be used to separate it from the transmitted image [10]. While these methods show impressive results, their assumptions are stringent and do not generalize well to real-world cases, causing them to fail on common cases.

Multiple images captured from different viewpoints can also be used to remove reflections. Several methods propose different ways to estimate the relative motion of the reflected and transmitted image, which can be used to separate them [11,12,13,14,15]. It is important to note that these methods assume static scenes—the motion is the apparent motion of the reflected layer relative to the transmitted layer, not scene motion. Other than that, these methods make assumptions that are less stringent than those made by single-image methods. Nonetheless, these algorithms work well when reflected and transmitted scenes are shallow in terms of depth, so that their velocity can be assumed uniform. For the case of spatially and temporally varying mixes, Kaftory and Zeevi propose to use sparse component analysis instead [16].

Multiple images captured under different polarization angles offer a third venue to tackle this problem. Assuming that images taken at different polarization angles offer independent measurements of the same scene, reflection and transmission can be separated using independent component analysis [17,18,19]. An additional prior that can be leveraged is given by double reflections, when the semi-reflective surface generates them [9]. Under ideal conditions, and leveraging polarization information, a solution can also be found in closed form [2,3]. In our experiments, we found that most of the pictures captured in unconstrained settings break even the well-founded assumptions used by these papers, as shown in Figure 2.

3 Method

We address the problem of layer decomposition by leveraging the ability of a semi-reflector to polarize the reflected and transmitted layers differently. Capturing multiple polarization images of the same scene, then, offers partially independent observations of the two layers. To use this information, we take a deep learning approach. Since the ground truth for this problem is virtually impossible to capture, we synthesize it. As for any data-driven approach, the realism of the training data is paramount to the quality of the results. In this section, after reviewing the image formation model, we give an overview of our approach, we discuss the limitations of the assumptions that are commonly made, and how we address them in our data generation pipeline. Finally, we describe the details of our implementation.

3.1 Polarization, Reflections, and Transmissions

Consider two points, P_R and P_T such that $P_R^{'}$, the reflection of P_R, lies on the line of sight of P_T, and assume that both emit unpolarized light, see Figure 3. After being reflected or transmitted, unpolarized light becomes polarized by an amount that depends on θ, the *angle of incidence* (AOI).

At point P_S, the intersection of the line of sight and the surface, the total radiance L is a combination of the reflected radiance L_R, and the transmitted radiance L_T. Assume we place a linear polarizer with polarization angle ϕ in front of the camera. If we integrate over the exposure time, the intensity at *each pixel* x is

$$I_\phi(x) = \alpha(\theta, \phi_\perp, \phi) \cdot \frac{I_R(x)}{2} + (1 - \alpha(\theta, \phi_\perp, \phi)) \cdot \frac{I_T(x)}{2}, \tag{1}$$

where the mixing coefficient $\alpha(\cdot) \in [0, 1]$, the angle of incidence $\theta(x) \in [0, \pi/2]$, the $p-$polarization direction [2] $\phi_\perp(x) \in [-\pi/4, \pi/4]$, and the reflected and transmitted images *at the semi-reflector*, $I_R(x)$ and $I_T(x)$, are all unknown.

At the Brewster angle, θ_B, the reflected light is completely polarized along $\phi_\perp$, *i.e.* in the direction perpendicular to the incidence plane[1], and the trans-

[1] The incidence plane is defined by the direction in which the light is traveling and the semi-reflector's normal.

mitted light along $\phi_{\parallel}$, the direction parallel to the plane of incidence. The angles $\phi_{\perp}$ and $\phi_{\parallel}$ are called the *canonical* polarization angles. In the unique condition in which $\theta(x) = \theta_B$, two images captured with the polarizer at the canonical polarization angles offer independent observations that are sufficient to disambiguate between I_R and I_T. Unless the camera or the semi-reflector are at infinity, however, $\theta(x) = \theta_B$ only holds for few points in the scene, if any, as shown in Figure 3. To complicate things, for curved surfaces, $\theta(x)$ varies non-linearly with x. Finally, even for arbitrarily many acquisitions at different polarization angles, ϕ_j, the problem remains ill-posed as each observation I_{ϕ_j} adds new pixel-wise unknowns $\alpha(\theta, \phi_{\perp}, \phi_j)$.

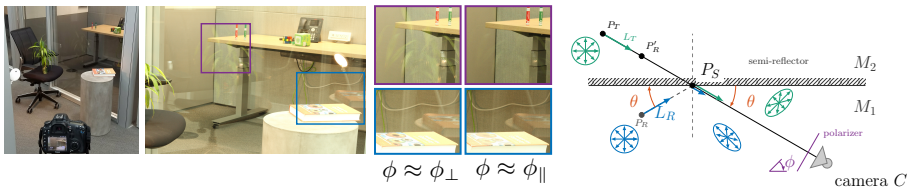


Fig. 3: A polarizer attenuates reflections when they are viewed at the Brewster angle $\theta = \theta_B$. For the scene shown on the left, we manually selected the two polarization directions that maximize and minimize reflections respectively. Indeed, the reflection of the plant is almost completely removed. However, only a few degrees away from the Brewster angle, the polarizerhas little to no effect, as is the case for the reflection of the book on the right.

3.2 Recovering R and T

When viewed through a polarizer oriented along direction ϕ, I_R and I_T, which are the reflected and transmitted images *at the semi-reflector*, produce image I_ϕ *at the sensor*. Due to differences in dynamic range, as well as noise, in some regions the reflection may dominate I_ϕ, or vice versa, see Section 3.3. Without hallucinating content, one can only aim at separating R and T, which we define

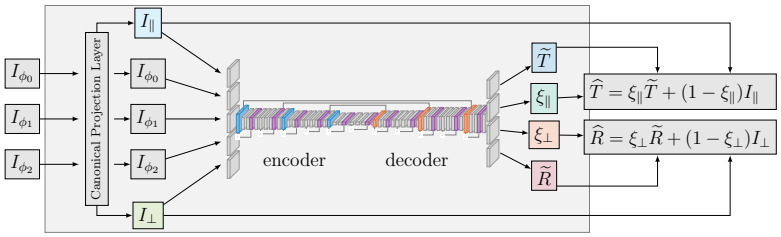


Fig. 4: Our encoder-decoder network architecture with ResNet blocks includes a Canonical Projection Layer, which projects the input images onto the canonical polarization directions, and uses a residual parametrization for $\widehat{T}$ and $\widehat{R}$.

to be the observable reflected and transmitted components. For instance, T may be zero in regions where R dominates, even though I_T may be greater than zero in those regions. To differentiate them from the ground truth, we refer to our estimates as $\widehat{R}$ and $\widehat{T}$.

To recover $\widehat{R}$ and $\widehat{T}$, we use an encoder-decoder architecture, which has been shown to be particularly effective for a number of tasks, such as image-to-image translation [20], denoising [21], or deblurring [22]. Learning to estimate $\widehat{R}$ and $\widehat{T}$ directly from images taken at arbitrary polarization angles does not produce satisfactory results. One main reason is that parts of the image may be pure reflections, thus yielding no information about the transmission, and vice versa.

To address this issue, we turn to the polarization properties of reflected and transmitted images. Recall that R and T are maximally attenuated, though generally not completely removed, at $\phi_\parallel$ and $\phi_\perp$ respectively. The canonical polarization angles depend on the geometry of the scene, and are thus hard to capture directly. However, we note that an image $I_\phi(x)$ can be expressed as [3]:

$$I_\phi(x) = I_\perp(x)\cos^2(\phi - \phi_\perp(x)) + I_\parallel(x)\sin^2(\phi - \phi_\perp(x)). \tag{2}$$

Since Equation 2 has three unknowns, $I_\perp$, $\phi_\perp$, and $I_\parallel$, we can use three different observations of the same scene, $\{I_{\phi_i}(x)\}_{i=\{0,1,2\}}$, to obtain a linear system that allows to compute $I_\perp(x)$ and $I_\parallel(x)$. To further simplify the math we capture images such that $\phi_i = \phi_0 + i \cdot \pi/4$.

For efficiency, we implement the projection onto the canonical views as a network layer in TensorFlow. The canonical views and the actual observations are then stacked in a 15-channel tensor and used as input to our network. Then, instead of training the network to learn to predict $\widehat{R}$ and $\widehat{T}$, we train it to learn the *residual* reflection and transmission layers. More specifically, we train the network to learn an 8-channel output, which comprises the residual images $\widetilde{T}(x)$, $\widetilde{R}(x)$, and the two single-channel weights $\xi_\parallel(x)$ and $\xi_\perp(x)$. Dropping the dependency on pixel x for clarity, we can then compute:

$$\widehat{R} = \xi_\perp \widetilde{R} + (1 - \xi_\perp) I_\perp \qquad \text{and} \qquad \widehat{T} = \xi_\parallel \widetilde{T} + (1 - \xi_\parallel) I_\parallel. \tag{3}$$

While $\xi_\perp$ and $\xi_\parallel$ introduce two additional unknowns per pixel, they significantly simplify the prediction task in regions where the canonical projections are already good predictors of $\widehat{R}$ and $\widehat{T}$. We use an encoder-decoder with skip connections [23] that consists of three down-sampling stages, each with two ResNet blocks [24]. The corresponding decoder mirrors the encoding layers using a transposed convolution with two ResNet blocks. We use an ℓ_2 loss on $\widehat{R}$ and $\widehat{T}$. We also tested ℓ_1 and a combination of ℓ_1 and ℓ_2, which did not yield significant improvements.

The use of the canonical projection layer, as well as the parametrization of residual images is key for the success of our method. We show this in the Supplementary, where we compare the output of our network with the output of the exact same architecture trained to predict $\widehat{R}$ and $\widehat{T}$ directly from the three polarization images $I_{\phi_i}(x)$.

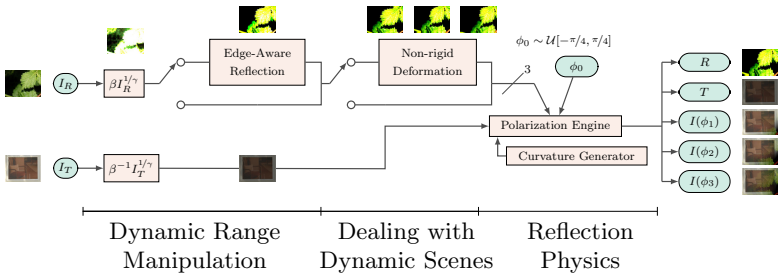


Fig. 5: Our image-based data generation procedure. We apply several steps to images I_R and I_T simulating reflections in most real-world scenarios (Section 3.3).

3.3 Image-Based Data Generation

The ground truth data to estimate $\widehat{R}$ and $\widehat{T}$ is virtually impossible to capture in the wild. Recently, Wan *et al.* released a dataset for single-image reflection removal [25], but it does not offer polarization information. In principle, Equation 1 could be used directly to generate, from any two images, the data we need. The term α in the equation, however, hides several subtleties and nonidealities. For instance, previous polarization-based works use it to synthesize data by assuming uniform AOI, perfectly flat surfaces, comparable power for the reflected and transmitted irradiance, or others. This generally translates to poor results on images captured in the wild: Figures 1 and 2 show common scenes that violate all of these assumptions.

We propose a more accurate synthetic data generation pipeline, see Figure 5. Our pipeline starts from two randomly picked images from the PLACE2 dataset [26], I_R and I_T, which we treat as the image of reflected and transmitted scene *at the surface*. From those, we model the behaviors observed in real-world data, which we describe as we "follow" the path of the photons from the scene to the camera.

Dynamic Range Manipulation at the Surface To simulate realistic reflections, the dynamic range (DR) of the transmitted and reflected images *at the surface* must be significantly different. This is because real-world scenes are generally high-dynamic-range (HDR). Additionally, the light intensity at the surface drops with the distance from the emitting object, further expanding the combined DR. However, our inputs are low-dynamic-range images because a large dataset of HDR images is not available. We propose to artificially manipulate the DR of the inputs so as to match the appearance of the reflections we observe in real-world scenes.

Going back to Figure 3 (right), we note that for regions where $L_T \approx L_R$, a picture taken without a polarizer will capture a smoothly varying superposition of the images of P_R and P_T (Figure 2 ❸). For areas of the surface where $L_R \gg L_T$, however, the total radiance is $L \approx L_R$, and the semi-reflector essentially acts as a mirror (Figure 2 ❷). The opposite situation is also common (Figure 2

❶). To allow for these distinct behaviors, we manipulate the dynamic range of the input images with a random factor $\beta \sim \mathcal{U}[1, K]$:

$$\tilde{I}_R = \beta I_R^{1/\gamma} \qquad \text{and} \qquad \tilde{I}_T = \frac{1}{\beta} I_T^{1/\gamma}, \tag{4}$$

where $1/\gamma$ linearizes the gamma-compressed inputs[2]. We impose that $K > 1$ to compensate for the fact that a typical glass surface transmits a much larger portion of the incident light than it reflects[3].

Images $\tilde{I}_R$ and $\tilde{I}_T$ can reproduce the types of reflections described above, but are limited to those cases for which $L_R - L_T$ changes smoothly with P_S. However, as shown in Figure 2 ❺, the reflection can drop abruptly following the boundaries of an object. This may happen when an object is much closer than the rest of the scene, or when its radiance is larger than the surrounding objects. To properly model this behavior, we treat it as a type of reflection on its own, which we apply to a random subset of the image whose range we have already expanded. Specifically, we set to zero the regions of the reflection or transmission layer, whose intensity is below $T = \text{mean}(\tilde{I}_R + \tilde{I}_T)$, similarly to the method proposed by Fan *et al.* [4].

Dealing with Dynamic Scenes Our approach requires images captured under three different polarization angles. While cameras that can simultaneously capture multiple polarization images exist [27,28,29], they are not widespread. To date, the standard way to capture different polarization images is sequential; this causes complications for non-static scenes. As mentioned in Section 2, if multiple pictures are captured from different locations, the relative motion between the transmitted and reflected layers can help disambiguate them. Here, however, "non-static" refers to the scene itself, such as is the case when a tree branch moves between the shots. Several approaches were proposed that can deal with dynamic scenes in the context of stack-based photography [30]. Rather than requiring some pre-processing to fix artifacts due to small scene changes at inference time, however, we propose to synthesize training data that simulates them, such as local, non-rigid deformations. We first define a regular grid over a patch, and then we perturb each one of the grid's anchors by (dx, dy), both sampled from a Gaussian with variance σ^2_{NR} , which is also drawn randomly for each patch. We then interpolate the position of the rest of the pixels in the patch. For each input patch, we generate three different images, one per polarization angle. We only apply this processing to a subset of the synthesized images—the scene is not always dynamic. Figure 6(a) and (b) show an example of original and distorted patch respectively.

[2] Approximating the camera response function with a gamma function does not affect the accuracy of our results, as we are not trying to produce data that is radiometrically accurate with respect to the original scenes.

[3] At an angle of incidence of $\pi/4$, for instance, a glass surface reflects less than 16% of the incident light.

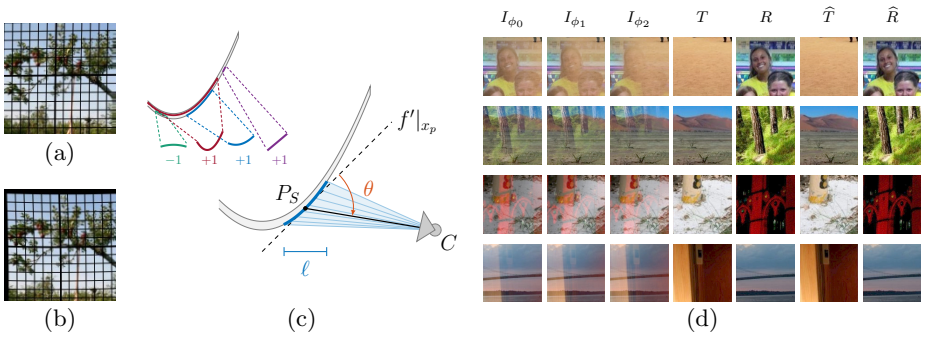


Fig. 6: Examples of our non-rigid motion deformation (a, b) and a curved surface-generator given the camera position, C, a surface-point, P_S, length, ℓ, and the convexity ± 1 (c). Randomly sampled training data (d) with synthesized observations I_{ϕ_0}, I_{ϕ_1}, I_{ϕ_2} from the ground truth data T and R, and estimates $\widehat{T}, \widehat{R}$.

Geometry of the Semi-Reflective Surface The images synthesized up to this point can be thought of as the irradiance of the unpolarized light at the semi-reflector. After bouncing off of, or going through, the surface, light becomes polarized as described in Section 3.1. The effect of a linear polarizer placed in front of the camera and oriented at a given polarization angle, depends on the angle of incidence (AOI) of the *specific* light ray. Some previous works assume this angle to be uniform over the image, which is only true if the camera is at infinity, or if the surface is flat.

We observe that real-world surfaces are hardly ever perfectly flat. Many common glass surfaces are in fact designed to be curved, as is the case of car windows, see Figure 1. Even when the surfaces are meant to be flat, the imperfections of the glass manufacturing process introduce local curvatures, see Figure 2 ❹.

At training time, we could generate unconstrained surface curvatures to account for this observation. However, it would be difficult to sample realistic surfaces. Moreover, the computation of the AOI from the surface curvature may be non-trivial. As a regularizer, we propose to use a parabola. When the patches are synthesized, we just sample four parameters: the camera position C, a point on the surface P_S, a segment length, ℓ, and the convexity as ± 1, Figure 6(c). Since the segment is always mapped to the same output size, this parametrization allows to generate a number of different, realistic curvatures. Additionally, because we use a parabola, we can quickly compute the AOI in closed form, from the sample parameters, see Supplementary.

3.4 Implementation Details

From the output of the pipeline described so far, the simulated AOI, and a random polarization angle ϕ_0, the polarization engine generates three observations

with polarization angles separated by $\pi/4$, see Figure 5. In practice, the polarizer angles ϕ_i will be inaccurate for real data due to the manual adjustments of the polarizer rotation. We account for this by adding noise within $\pm 4°$ to each polarizer angle ϕ_i. Additionally we set $\beta \sim \mathcal{U}[1, 2.8]$. The input for our neural network is $\mathbb{R}^{B\times 128\times 128\times 9}$ when trained on 128×128 patches, where $B = 32$ is the batch size. We trained the model from scratch with a learning rate $5 \cdot 10^{-3}$ using ADAM. See the Supplementary for more details about the architecture. The colors of the network predictions might be slightly desaturated [31,32,4]. We use a parameter-free color-histogram matching against one of the observations to obtain the final results.

4 Experiments

In this section we evaluate our method and data modeling pipeline on both synthetic and real data. For the latter, we introduce the Urban Reflections Dataset (URD), a new dataset of images containing semi-reflectors captured with polarization information. A fair evaluation can only be done against other polarization-based methods, which use multiple images. However, we also compare against single-image methods for completeness.

The Urban Reflections Dataset (URD). For practical relevance, we compile a dataset of 28 high-resolution RAW images (24MP) that are taken in urban environments using two different consumer cameras (Alpha 6000 and Canon EOS 7D, both ASP-C sensors), and which we make publicly available. The Supplementary shows all the pictures in the dataset. This dataset includes examples taken with a wide aperture, and while focusing on the plane of the semi-reflector, thus meeting the assumptions of Fan *et al.* [4].

4.1 Numerical Performance Evaluation

Due to the need for ground-truth, a large-scale numerical evaluation can only be performed on synthetic data. For this task we take two datasets, the VOC2012 [33] and the PLACE2 [26] datasets. A comparison with state-of-the-art methods shows that our method outperforms the second best method by a significant margin in terms of PSNR: ~ 2 dB, see Table 1. For a numerical evaluation on real data, we set up a scene with a glass surface and objects causing reflections. After capturing polarization images of the scene, we removed the glass and captured the ground truth transmission, T_{gt}. Figure 7 shows the transmission images estimated by different methods. Our method achieves the highest PSRN, and the least amount of artifacts.

Table 1: Cross-validation on synthetic data. Best results in bold.

	PASCAL VOC 2012		PLACE2	
Method	RMSE	PSNR	RMSE	PSNR
Farid *et al.* [17]	0.401	7.93	0.380	8.38
Kong *et al.* [3]	0.160	15.88	0.156	16.12
Schechner *et al.* [2]	0.085	21.34	0.086	21.27
Fan *et al.* [4]	0.080	21.89	0.084	21.48
Ours	**0.064**	**23.83**	**0.066**	**23.58**



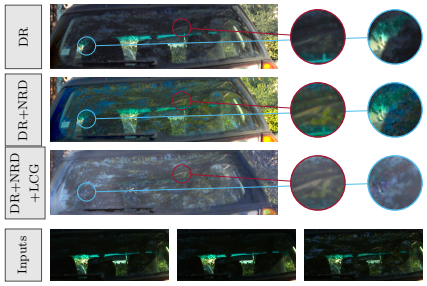
Fig. 7: By removing the semi-reflector, we can capture the ground truth transmission, T_{gt}, optically.

4.2 Effect of Data Modeling

We also thoroughly validate our data-generation pipeline. Using both synthetic and real data, we show that the proposed non-rigid deformation (NRD) procedure and the local curvature generation (LCG) are effective and necessary. To do this, we train our network until convergence on three types of data: data generated only with the proposed dynamic range manipulation, DR for short, data generated with DR+NRD, and data generated with DR+NRD+LCG.

We evaluate these three models on a hold-out synthetic validation set that features all the transformations from Figure 5. The table in Figure 8 shows that the PSNR drops significantly when only part of our pipeline is used to train the network. Unfortunately, a numerical evaluation is only possible when the ground truth is available. However, Figure 8 shows the output of the three models on the real image from Figure 1. The benefits of using the full pipeline are apparent.

A visual inspection of Figure 1 allows to appreciate that, thanks to our ability to deal with curved surfaces and dynamic scenes, we achieve better performance than the state-of-the-art methods.



Model	PSNR
DR	28.17 dB
DR+NRD	30.44 dB
DR+NRD+LCG	31.18 dB

Fig. 8: Our reflection estimation (left) on a real-world curved surface and synthetic data (right Table) using the same network architecture trained on different components of our data pipeline. Only when using the full pipeline (DR+NRD+LCG) the reflection layer is estimated correctly. Note how faint the reflection is in the inputs (bottom row).

4.3 Evaluation on Real-World Examples

We extensively evaluate our method against previous work on the proposed URD. For fairness towards competing methods, which make stronger assumptions or expect different input data, we slightly adapt them, or run them multiple

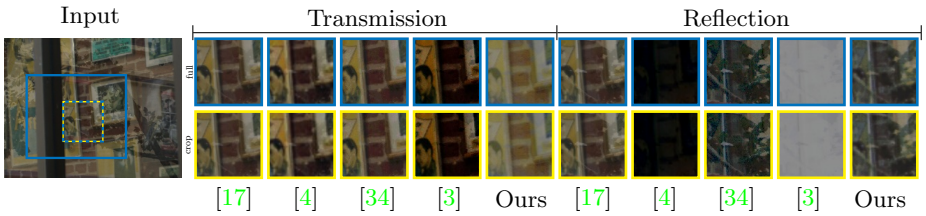


Fig. 9: Applying the different algorithms to the whole image and cropping a region ('full') is equivalent to applying the same algorithms to the cropped region directly ('crop').

times with different parameters retaining only the best result. Due to space constraints, Figure 10 only shows seven of the results. We refer the reader to the Supplementary for the rest of the results and for a detailed explanation about how we adapted previous methods. One important remark is in order. Although the images we use include opaque objects, *i.e.* the semi-reflector does not cover the whole picture, the methods against which we compare are local: applying the different algorithms to the whole image and cropping a region is equivalent to applying the same algorithms to the cropped region directly, Figure 9.

Figure 10, *Curved Window* shows a challenging case in which the AOI is significantly different from θ_B across the whole image, thus limiting the effect of the polarizer in all of the inputs. Moreover, the glass surface is slanted and locally curved, which breaks several of the assumptions of previous works. As a result, other methods completely fail at estimating the reflection layer, the transmission layer, or both. On the contrary, our method separates $\widehat{T}$ and $\widehat{R}$ correctly, with only a slight halo of the reflection in $\widehat{T}$. In particular, notice the contrast of the white painting with the stars, as compared with other methods. While challenging, this scene is far from uncommon.

Figure 10, *Bar* shows another result on which our method performs significantly better than most related works. On this example, the method by Schechner *et al.* [2] produces results comparable to ours. However, recall that, to be fair towards their method, we exhaustively search the parameter space and handpick the best result. Another thing to note is that our method may introduce artifacts in a region for which there is little or no information about the reflected or transmitted layer in any of the inputs, such as the case in the region marked with the red square on our $\widehat{T}$.

We also show an additional comparison showing the superiority of our method (Figure 10, *Paintings*) and a few more challenging cases. We note that in a few examples, our method may fail at removing part of the "transmitted" objects from $\widehat{R}$, as is the case in Figure 10, *Chairs*.

User Study Since we do not have the ground truth for real data, we evaluate our method against previous results by means of a thorough user study. We asked 43 individuals not involved with the project, to rank our results against the state-of-the-art [17,4,7,2,3]. In our study, we evaluate $\widehat{R}$ and $\widehat{T}$ as two separate tasks, because different methods may perform better on one or the other. For each task, the subjects were shown the three input polarization images, and the results of each method on the same screen, in randomized order. They were given the task to rank the results 1–6, which took, on average, 35 minutes per subject. We measure the recall rate in ranking, $R@k$, *i.e.* the fraction of times a method ranks among the top-k results. Table 2 reports the recall-rates. Two conclusions emerge from analyzing the table. First, and perhaps expected, polarization-based methods outperform the other methods. Second, our method ranks higher than related works by a significant margin.

Table 2: Result from the user study. We report the average recall-rate for each method.

	Transmission		Reflection	
Method	$R@1$	$R@2$	$R@1$	$R@2$
Ours	**0.46**	**0.65**	**0.34**	**0.54**
[2]	0.14	0.38	0.23	0.40
[3]	0.11	0.27	0.09	0.20
[4]	0.06	0.17	0.08	0.20
[7]	0.08	0.21	0.10	0.29
[17]	0.06	0.13	0.15	0.37

5 Conclusion

Separating the reflection and transmission layers from images captured *in the wild* is still an open problem, as state-of-the-art methods fail on many real-world images. Rather than learning to estimate the reflection and the transmission directly from the observations, we propose a deep learning solution that leverages the properties of polarized light: it uses a Canonical Projection Layer, and it learns the residuals of the reflection and transmission relative to the canonical images. Another key ingredient to the success of our method is the definition of an image-synthesis pipeline that can accurately reproduce typical nonidealities observed in everyday pictures. We also note that the non-rigid deformation procedure that we propose can be used for other stack-based methods where non-static scenes may be an issue. To evaluate our method, we also propose the Urban Reflection Dataset, which we will make available upon publication. Using this dataset, we extensively compare our method against a number of related works, both visually and by means of a user study, which confirms that our approach is superior to the state-of-the-art methods. Finally, the code for most of the existing methods that separate reflection and transmission is not available: to perform an accurate comparison, we re-implemented representative, state-of-the-art works, and make our implementation of those algorithms available to the community, to enable more comparisons.



Fig. 10: Results on typical real-world scenes. Top pane: comparison with state-of-the-art methods, bottom pane: additional results. More results are given in the Supplementary.

Acknowledgments

We thank the reviewers for their feedback, in particular the reviewer who suggested the experiment in Fig. 7, Hendrik P.A. Lensch for the fruitful discussions, and the people who donated half hour of their lives to take our survey.

References

1. **Project Website**. http://research.nvidia.com/publication/2018-09_Separating-Reflection-and (2018)
2. Schechner, Y.Y., Shamir, J., Kiryati, N.: Polarization and statistical analysis of scenes containing a semireflector. Journal of the Optical Society of America (2000)
3. Kong, N., Tai, Y.W., Shin, J.S.: A physically-based approach to reflection separation: From physical modeling to constrained optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)
4. Fan, Q., Yang, J., Hua, G., Chen, B., Wipf, D.: A generic deep architecture for single image reflection removal and image smoothing. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). (2017)
5. Collett, E.: Field guide to polarization. SPIE press Bellingham (2005)
6. Levin, A., Weiss, Y.: User assisted separation of reflections from a single image using a sparsity prior. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2007)
7. Li, Y., Brown, M.S.: Single image layer separation using relative smoothness. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2014)
8. Arvanitopoulos Darginis, N., Achanta, R., Süsstrunk, S.: Single image reflection suppression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
9. Diamant, Y., Schechner, Y.Y.: Overcoming visual reverberations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2008)
10. Shih, Y., Krishnan, D., Durand, F., Freeman, W.T.: Reflection removal using ghosting cues. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2015)
11. Li, Y., Brown, M.S.: Exploiting reflection change for automatic reflection removal. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). (2013)
12. Xue, T., Rubinstein, M., Liu, C., Freeman, W.T.: A computational approach for obstruction-free photography. ACM Transactions on Graphics (SIGGRAPH) (2015)
13. Szeliski, R., Avidan, S., Anandan, P.: Layer extraction from multiple images containing reflections and transparency. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2000)
14. Guo, X., Cao, X., Ma, Y.: Robust separation of reflection from multiple images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2014)
15. Han, B.J., Sim, J.Y.: Reflection removal using low-rank matrix completion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
16. Kaftory, R., Zeevi, Y.Y.: Blind separation of time/position varying mixtures. IEEE Transactions on Image Processing (TIP) (2013)
17. Farid, H., Adelson, E.H.: Separating reflections and lighting using independent components analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (1999)
18. Barros, A.K., Yamamura, T., Ohnishi, N., et al.: Separating virtual and real objects using independent component analysis. IEICE Transactions on Information and Systems (2001)

19. Bronstein, A.M., Bronstein, M.M., Zibulevsky, M., Zeevi, Y.Y.: Sparse ica for blind separation of transmitted and reflected images. International Journal of Imaging Systems and Technology (2005)
20. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
21. Mao, X., Shen, C., Yang, Y.: Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In: Advances in Neural Information Processing Systems (NIPS). (2016)
22. Wieschollek, P., Schölkopf, M.H.B., Lensch, H.P.A.: Learning blind motion deblurring. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). (2017)
23. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. arXiv preprint arXiv:1505.04597 (2015)
24. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2016)
25. Wan, R., Shi, B., Duan, L.Y., Tan, A.H., Kot, A.C.: Benchmarking single-image reflection removal algorithms. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). (2017)
26. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2017)
27. Fluxdata. http://www.fluxdata.com/products/fd-1665p-imaging-polarimeter (Accessed on July 10, 2018)
28. Ricoh. https://www.ricoh.com/technology/tech/051_polarization.html (Accessed on July 10, 2018)
29. Polarcam. https://www.4dtechnology.com/products/polarimeters/polarcam/ (2018)
30. Gallo, O., Sen, P.: Stack-based algorithms for HDR capture and reconstruction. In: High Dynamic Range Video. Elsevier (2016)
31. Wieschollek, P., Schölkopf, B., Lensch, H.P.A., Hirsch, M.: End-to-end learning for image burst deblurring. In: Proceedings of the Asian Conference on Computer Vision (ACCV). (2016)
32. Kim, J., Kwon Lee, J., Mu Lee, K.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2016)
33. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
34. Schechner, Y.Y., Kiryati, N., Shamir, J.: Separation of transparent layers by polarization analysis. In: Proceeding of the Scandinavian Conference on Image Analysis. (1999)