Reflection Removal using Ghosting Cues

YiChang Shih
MIT CSAIL
yichang@mit.edu

Dilip Krishnan
Google Research *
dilipkay@google.com

Frédo Durand
MIT CSAIL
fredo@mit.edu

William T. Freeman
MIT CSAIL
billf@mit.edu

Abstract

Photographs taken through glass windows often contain both the desired scene and undesired reflections. Separating the reflection and transmission layers is an important but ill-posed problem that has both aesthetic and practical applications. In this work, we introduce the use of ghosting cues that exploit asymmetry between the layers, thereby helping to reduce the ill-posedness of the problem. These cues arise from shifted double reflections of the reflected scene off the glass surface. In double-pane windows, each pane reflects shifted and attenuated versions of objects on the same side of the glass as the camera. For single-pane windows, ghosting cues arise from shifted reflections on the two surfaces of the glass pane. Even though the ghosting is sometimes barely perceptible by humans, we can still exploit the cue for layer separation. In this work, we model the ghosted reflection using a double-impulse convolution kernel, and automatically estimate the spatial separation and relative attenuation of the ghosted reflection components. To separate the layers, we propose an algorithm that uses a Gaussian Mixture Model for regularization. Our method is automatic and requires only a single input image. We demonstrate that our approach removes a large fraction of reflections on both synthetic and real-world inputs.

(a) *Input* **(b)** *Close-up of ghosting*

(c) *Recovered Transmission* **(d)** *Recovered Reflection*

Figure 1: *Our method removes undesired reflections in an image taken through a glass pane: (a) Input image with reflection artifacts; (b) Close-up shows ghosting on the reflection layer; (c) Recovered transmission layer using our algorithm; (d) Recovered reflection layer. Our method exploits the ghosting cues (seen in (b)) to overcome the ill-posedness of the layer separation problem.*

1. Introduction

When taking photographs through windows or glass panes, reflections of the scene on the same side of the glass as the camera often ruin the picture. To minimize reflection artifacts, one may try to change camera position, use polarizers, or put a piece of dark cloth around the camera, but often it is impractical to make any of these adjustments. This raises the need for post-processing to remove reflection artifacts. Separating transmission T and reflection R is an ill-posed problem, since both T and R are natural images with similar characteristics, and traditional imaging models assume that T and R play symmetric roles when forming the input I, *i.e.*, $I = T + R$. Most previous work tackles the ill-posedness through the use of multiple input images [3, 10, 19, 26, 28, 29, 31, 32] or through user inputs which mark regions of the observed image as belonging either to T or R (see [21]).

In this paper, we address the reflection removal problem using "ghosting" effects – multiple reflections on glasses in the captured image. A common example is a double-pane window, which consists of two thin glass panes separated by some distance for insulation [1]. The glass pane at the inner side (closer to the camera) generates the first reflection, and the outer side generates the second, which is a shifted

*Part of this work was done while the author was a postdoctoral associate at MIT CSAIL. Supplemental material, code and data is available at: https://dilipkay.wordpress.com/reflection_ghosting.

and attenuated version of the first reflection. The distance between the two reflections depends on the space between the two panes.

In single-pane windows of typical thickness 3-10mm, ghosting arises from multiple reflections by the near and far surfaces of the glass (see Figure 3 and Section 3). We have calculated that[1], for a modern SLR and standard 50mm lens placed 30cm (1 foot) or less from the glass, ghosting of more than 4 pixels occurs if the camera is at angle of more than 10 degrees and reflected objects are less than 6m away. We include a video in the supplementary material to visualize ghosting with glass panes of varying thickness and camera viewpoints.

To quantify the frequency of ghosting, we analyzed images returned by Google's Image Search. We used the keywords "window reflection photography problems" and "reflections on windows." After removing irrelevant results such as cartoon images and water reflection, we examined 197 randomly sampled images, and observe 96 of them exhibit significant ghosting (49%). Some examples are shown in the supplemental materials.

Ghosting provides a critical cue to separate the reflection and transmission layers, since it breaks the symmetry between the two layers. We model the ghosting as convolution of the reflection layer R with a kernel k. Then the observed image I can be modeled as an additive mixture of the ghosted reflection and transmission layers by R and T respectively:

$$I = T + R \otimes k \tag{1}$$

Following the work in Diamant *et al*. [9], we model the kernel k as a two-pulse kernel, parameterized by the distance and the relative intensity between the primary and secondary reflections. As we show later, higher-order reflections carry minimal energy and can be ignored. Given the input image I, our goal is to recover the kernel k, transmission layer T, and reflection layer R. Most previous work (with the exception of [9]), assumed that $I = T + R$ in their imaging models. Solving this ill-posed problem requires either very effective image priors, or auxiliary data such as multiple images captured with motion or polarizers, or user input. Each of these solutions has drawbacks.

The benefit of using ghosting cues is illustrated with a toy example in Figure 2. We generate a synthetic example with a circle as the transmission layer and a rectangle as the reflection layer. We show the ground truth decomposition, and two other "extreme" decompositions as pure transmission and pure reflection. We then measure the likelihoods of these solutions under the Gaussian Mixture Model (GMM) of Zoran and Weiss [34]. Intuitively, the ground-truth decomposition is sparsest, e.g., in the gradient domain, and therefore the most "natural". This decomposition has the highest likelihood of this decomposition (3.26×10^4) under the GMM model. The two extreme decompositions include ghosting artifacts and are less sparse. They are less natural, and their likelihoods are lower (1.81×10^4 and 1.55×10^4, respectively). In the absence of ghosting, the extreme decompositions are both equally probable under the GMM model.

[1]Based on optical simulation described in the appendix. We include the simulator in the supplemental materials.

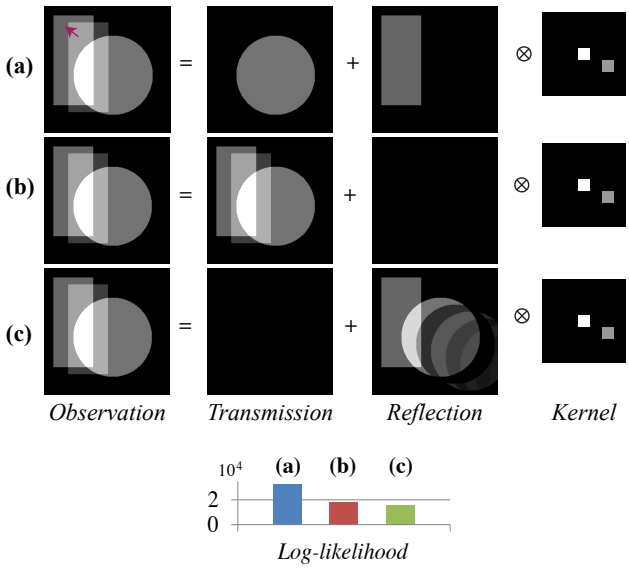


Figure 2: *Ghosting cues for layer separation: (**a**) A synthetic example using a circle and a rectangle as transmission and reflection respectively. The red arrow points to the ghosting. The log-likelihood of this decomposition under a GMM model is* 3.26×10^4*; (b) A feasible decomposition: the image is interpreted as only transmission and has log-likelihood* 1.81×10^4*; (c) Another decomposition: the image is interpreted as pure reflection with log-likelihood* 1.55×10^4*. Both (b) and (c) introduce additional ghosting compared to the true decomposition in (a). This results in lower likelihoods under the GMM model.*

2. Related Work

The layer separation problem, in the absence of ghosting cues, is ill-posed and requires image priors to reduce ambiguities. Unfortunately, modern image priors are not discriminative enough to distinguish the sum of two natural images from the ground truth images. Thus, most previous work on layer separation has relied on other mechanisms to achieve good separation.

A substantial amount of work uses multiple images captured with different polarizers [19, 29] , or different proportions of layer mixing [26], in which information is transferred between images to minimize correlation. In [10], similar ideas are used with two images and Independent Components Analysis for layer decorrelation. A pair of

images taken with and without a flash is used in [3]; separation is achieved by exploiting different artifacts in the flash and non-flash images. Differential focusing is used in [28] with a pair of images taken, each focused on one layer. Other works use video as input, exploiting the decorrelation of motion between the transmission and reflection layers [11, 12, 15, 20, 24, 27, 31, 32].

Some prior work exploits the statistics of natural images [22, 23] to enable separation from a single input image. Specifically, the sparsity of image gradient distributions and sparsity of corner detectors on an image are utilized. However, as acknowledged in [22], real-life scenarios with rich textures are a challenge due to the problems in robustly detecting corners in such images. To reduce ambiguities, follow-up work from the same authors requires the use of sparse user annotations to guide the layer separation process [21]. To overcome the ill-posedness, recent works exploit statistical asymmetries between transmission and reflection, such as considering the case when the reflected scene is achromatic [17], or when the reflection is blurred relative to the transmission [25].

Some researchers in psychology and cognitive science have discovered that local features such as X-junctions are related to transparency perception [2, 18], and can be used to separate reflection and transmission layers in simple scenes [30]. However, as with corner detectors, these cues are difficult to recover robustly from textured images.

The problem of ghosting has been considered in the radio communication literature, and deconvolution is used to remove these artifacts [6]. In image processing, the use of the ghosting effect for reflection removal was introduced in [9], and two polarized images were used to achieve layer separation.

3. Ghosting Formation Model

Our work uses the ghosting model proposed by Diamant and Schechner [9]. They quantified the ghosting in both the transmission layer T and reflection layer R. In this work, we consider a simplified version of their model, involving only the first-order reflection in R; our model is shown in Figure 3.

We denote light rays transmitted through the glass pane as transmission T. Light rays from the reflected objects (on the same side of the glass pane as the camera) first reflect off the glass pane to give a primal reflection layer, which we denote by R_1. Since the glass is semi-reflective, R_1 only contains a certain fraction of the incident light. The remainder transmits through the glass and reaches the opposite side. A certain fraction of this is reflected back towards the camera. This results in a second reflection denoted by R_2. R_2 is a spatially shifted and attenuated image of R_1. The superposition of R_1 and R_2 gives a ghosted reflection layer R, as shown in Figures 3(b) and 3(c).

As in [9], we assume that the spatial shift and relative attenuation between R_1 and R_2 is spatially invariant. Based on Fresnel's equations [14], these assumptions hold when the reflection layer does not have large depth variations, and when the angle between camera and glass normal is not too oblique. In our simplified model, we ignore higher-order internal reflections of both T and R; these are shown as dotted arrows in Figure 3(a). For typical glass with refraction index around 1.5, these higher-order reflections contribute to less than 1% of the reflected or transmitted energy. Finally, we assume that the glass is planar.

Under these assumptions (see [9]), the ghosting kernel k consists of two non-zero values. k is parameterized by a two-dimensional spatial shift $\mathbf{d}_k$ and relative attenuation factor c_k. Given an image X, the result of convolving it with the kernel k gives an output Y, whose value at pixel $\mathbf{i}$ is:

$$Y_{\mathbf{i}} = X_{\mathbf{i}} + c_k X_{\mathbf{i}-\mathbf{d}_k} \tag{2}$$

The spatial shift $\mathbf{d}_k$ is affected by geometric optics and depends on glass thickness, relative positions of the glass, camera, reflected objects, and camera focal length. The attenuation factor c_k is affected by wave optics through Fresnel's equations, and is dependent on the refractive index of the glass, the incidental angle of light, and the polarization of the reflected light rays.

Ghosting effects are more pronounced for thicker glass, because the image offsets are large, and for large angles between the camera viewing angle and the glass surface normal.

4. Layer Separation Algorithm

Our formation model for the observed image I, given the transmission T, reflection R and ghosting kernel k, is:

$$I = T + R \otimes k + n \tag{3}$$

where n is additive i.i.d. Gaussian noise with variance σ^2. We first estimate the ghosting kernel k; details are given in Section 4.2. Given k, the above formation model leads to a data (log-likelihood) term for reconstruction of T and R:

$$L(T, R) = \frac{1}{\sigma^2} \| I - T - R \otimes k \|_2^2 \tag{4}$$

However, minimizing $L(T, R)$ for the unknowns T and R is ill-posed. Additional priors are needed to regularize the inference.

We experimented with a number of priors including sparse functions on the gradients of T and R, based on natural image statistics. The best-performing prior turned out to be a recently proposed patch-based prior based on Gaussian Mixture Models (GMM) [34]. The GMM prior captures covariance structure and pixel dependencies over patches of size 8×8, thereby giving superior reconstructions

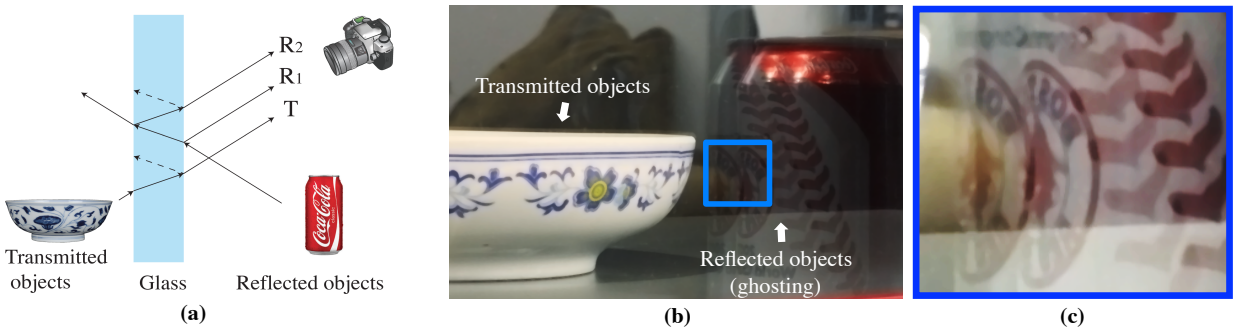


Figure 3: *Ghosting image formation: (a) Light rays from an object on the same side of the glass as the camera (a soft-drink can) are partially reflected by the inner-side of the glass, generating the primal reflection R_1. The remainder is transmitted, and partially reflected by the far side of the glass, generating a secondary reflection R_2. R_2 is a shifted and attenuated version of R_1. The superposition of R_1 and R_2 leads to the observed ghosted image in (b). Higher-order reflections, denoted by dashed arrows, are less than 1% of the energy of T and R_1 and R_2, and can be ignored; (b) The ghosted image captured by the camera; (c) The inset shows the ghosting effect more clearly.*

PSNR: 7.04 dB	13.22 dB	14.01 dB	26.76 dB	-
SSIM: 0.4012	0.5267	0.5499	0.9083	-
(a) Synthetic input	(b) Sparsity-inducing filters	(c) GMM patch prior	(d) Non-negativity	(e) Ground truth

Figure 4: *A synthetically generated example comparing different image regularizers: (a) Synthetically generated input; (b) Recovered transmission layer using sparsity-inducing gradient filters as regularizers; (c) GMM patch prior significantly reduces these artifacts; (d) adding non-negativity constraints in the optimization further improves recovered color; (e) ground truth transmission layer. We show the PSNR and SSIM of the recovered transmissions compared with the ground truth.*

to simple gradient-based filters, which assume independence between filter responses of individual pixels. Following the formulation in [34], our regularizer seeks to minimize the following cost:

$$-\sum_i \log(\text{GMM}(P_i T)) - \sum_i \log(\text{GMM}(P_i R)) \quad (5)$$

where $\text{GMM}(P_i X) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\overline{P_i X}; 0, \Sigma_j)$. The cost sums over all overlapping patches $P_i T$ in T, and $P_i R$ in R; where P_i is the linear operator that extracts the i^{th} patch from T or R. We use the pre-trained zero-mean GMM model from [34] with 200 mixture components, and patch size 8×8. The mixture weights are given by $\{\pi_j\}$, and the covariance matrices by $\{\Sigma_j\}$. In Equation 5, $\mathcal{N}$ is a zero-mean 64-dimensional Gaussian distribution; and $\overline{P_i X}$ is the patch $P_i X$ with mean removed.

Our final cost function combines (4) and (5) for the recovery of T and R, and also includes a non-negativity constraint for each pixel of T and R:

$$\min_{T,R} \; \tfrac{1}{\sigma^2}\|I - T - R \otimes k\|_2^2 - \textstyle\sum_i \log(\text{GMM}(P_i T)) - \sum_i \log(\text{GMM}(P_i R)), \;\; \text{s.t. } 0 \le T, R \le 1 \quad (6)$$

The non-negativity constraints on T and R are very useful in regularizing the low-frequencies [32]. This provides regularization that is complementary to the GMM prior, which is more useful in regularizing higher frequencies. Please note that we abuse notation slightly to avoid clutter; the constraints are per-pixel. For color images, we solve (6) independently on the red, green and blue channels.

As stated above, we compared GMM priors to sparsity inducing priors on filter responses, which can be represented

as cost functions of the form $\sum_i (|f_i \otimes T|^\alpha + |f_i \otimes R|^\alpha)$, where $\alpha \leq 1$, and the filter set $\{f_i\}$ includes gradients, Laplacians, and higher-order gradient filters. Figures 4(b) and 4(c) show that the GMM prior outperforms the sparsity-inducing filters with improved layer separation and decorrelation. In Figure 4(b), using filters with small spatial support leads to longer edges being split between the T and R layers. The GMM prior alleviates this problem by capturing longer range relationships over patches.

4.1. Optimization

The cost in (6) is non-convex due to the use of the GMM prior. We use an optimization scheme based on half-quadratic regularization [13, 34]. We introduce auxiliary variables z_T^i and z_R^i for each patch P_iT and P_iR, respectively. We then optimize the the following cost function:

$$\min_{T,R,z_T,z_R} \frac{1}{\sigma^2}\|I - T - R \otimes k\|_2^2 \tag{7a}$$

$$+ \frac{\beta}{2}\sum_i \left(\|P_iT - z_T^i\|^2 + \|P_iR - z_R^i\|^2\right) \tag{7b}$$

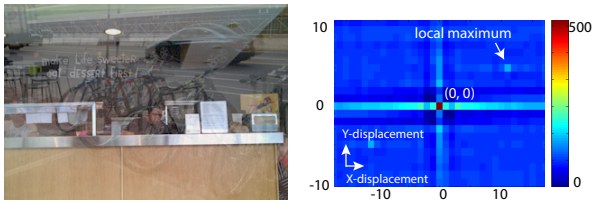
$$- \sum_i \log(\text{GMM}(z_T^i)) - \sum_i \log(\text{GMM}(z_R^i)) \tag{7c}$$

$$\text{s.t. } 0 \leq T, R \leq 1 \tag{7d}$$

To solve the above auxiliary problem, we use increasing values of β. We start from $\beta = 200$, and increase the value by a multiple of 2 after each iteration. We run 25 iterations in all. As β is increased, the values of P_iT and z_T^i are forced to agree; similarly for the values of P_iR and z_R^i. In all our experiments, we set $\sigma = 5 \times 10^{-3}$.

For each fixed value of β, we perform alternating minimization. We first fix $\{z_T^i\}$ and $\{z_R^i\}$ and solve for T and R. This involves the quadratic subproblems (7a) and (7b) and the constraints (7d). We solve this sub-problem using extended L-BFGS [33] to handle box constraints. Since P_i contains only diagonal elements, and k contains only two non-zero entries for each pixel, the pixel domain L-BFGS solver is very efficient. We solve for T and R simultaneously by transforming the term $\|I - T - R \otimes k\|_2^2$ to $\|I - \mathbf{A}X\|_2^2$. Here X vertically concatenates vectors T and R, *i.e.*, $X = [T; R]$, and $\mathbf{A}$ horizontally concatenates the identity matrix $\mathbf{I}$ and convolution matrix $\mathbf{k}$, *i.e.* $\mathbf{A} = [\mathbf{I}|\mathbf{k}]$. We then transform P_iT and P_iR to corresponding patches on X, and use constrained L-BFGS to solve for X.

Next, we fix T and R, and update $\{z_T^i\}$ for each patch i. All the $\{z_T^i\}$ may be updated in parallel, since each patch is independent of other patches. We adopt the approximate optimization suggested by [34]: we first select the component with the largest likelihood in the GMM model, and then perform Weiner filtering using only that component; this is a simple least squares update. An analogous update is used for $\{z_R^i\}$.



(a) Input **(b)** Autocorrelation Map

Figure 5: *Determining ghosting kernel k: (a) Input image with ghosting; (b) 2-D autocorrelation map of the Laplacian of the input image. The local maximum pointed to by the white arrow corresponds to the displacement $\mathbf{d}_k$, which is at an offset of $(13, 5)$ pixels. Using this spatial offset, the attenuation factor c_k is computed. See text for details.*

A good initialization is crucial in achieving better local minima. We initialize the GMM-based model with a sparsity-inducing based model, with a convex L_1 prior penalty:

$$\min_{T,R} \frac{1}{\sigma^2}\|I - T - R \otimes k\|_2^2 + \sum_j \|f_j \otimes T\|_1 + \sum_j \|f_j \otimes R\|_1 \tag{8}$$

The L_1 optimization can be efficiently performed using ADMM [4]. We use a set of sparsity inducing filters $\{f_i\}$ that include gradients and Laplacians. Details of the ADMM optimization are provided in the supplemental material.

4.2. Estimating k

Here we explain the estimation of ghosting convolution kernel k, which is parameterized by a spatial shift vector $\mathbf{d}_k$ and an attenuation factor c_k. We use some ideas from [9] for the estimation. We first estimate $\mathbf{d}_k$ using the 2-D autocorrelation map of $\nabla^2 I$, which is the Laplacian of the input image I. Figure 5 shows an example ghosted image and the autocorrelation map of the Laplacian over a range of spatial shifts. The shifted copies of the reflection layer create a local maximum at $\mathbf{d}_k$ on the autocorrelation map. To detect $\mathbf{d}_k$, we apply a local maximum filter in each 5-by-5 neighborhood. For robust estimation, we discard local maxima in neighborhoods where the first and second maxima are closer than a pre-defined threshold. This removes incorrect maxima that are caused due to locally flat or repetitive structures. We also remove local maxima within 4 pixels of the origin. Finally, of the remaining local maxima, we select the largest one as the ghosting distance $\mathbf{d}_k$. There is an ambiguity on the sign of $\mathbf{d}_k$, and we resolve this by choosing $\mathbf{d}_k$ such that $c_k < 1$, using the property that the second reflection has lower energy than the first.

The estimation of the attenuation factor c_k uses the shift $\mathbf{d}_k$. We first detect a set of interest points from the input using a Harris corner detector. For most corner features within the image, we found that the gradients of a local patch are dominated by the gradients of either R_1, R_2 or

T. Around each corner feature, we extract a 5×5 contrast-normalized patch. For patches that have a strong correlation with a patch at spatial offset $\mathbf{d}_k$, we assume that the edges are due to either reflection layer R_1 or R_2. We then estimate the attenuation between a pair of matching patches p_i, p_j as the ratio $a_{ij} = \sqrt{\frac{\text{var}[p_i]}{\text{var}[p_j]}}$, where $\text{var}[p_i]$ is the variance of the pixels in patch p_i, and we choose the order of (i, j) such that $a_{ij} < 1$. Finally, we sum over all such pairs to give an estimate of c_k:

$$c_k = \frac{1}{Z} \sum_{ij} w_{ij} a_{ij} \tag{9}$$

where $Z = \sum_{ij} w_{ij}$ is the normalization factor, $w_{ij} = e^{-\frac{\|p_i - p_j\|^2}{2\theta^2}}$, $\theta = 0.2$.

While the above method for estimation of k has proven to be robust in our experiments, it can fail on images with strong globally repetitive texture.

5. Results

We evaluate our algorithm on both synthetic and real-world inputs, demonstrating how ghosting cues help in the recovery of transmission and reflection layers. We also demonstrate potential applications of our algorithm, including image classification on the recovered transmissions and automated de-ghosting for product photography.

We start by testing on synthetic data from the MIT-Adobe FiveK dataset [5]. We synthesize 20 inputs from 40 randomly sampled images for T and R, attenuation c_k between 0.5 and 1.0, and ghosting distance $\mathbf{d}_k$ between 4 to 40 pixels. On average, our method achieves an SSIM of 0.84 and PSNR of 23.2 dB for the transmission layer. An example is shown in Figure 4.

Figures 1 and 6 show the results of reflection removal on real-world inputs. All the images are taken between 0.3 to 1 meter away from the window, and the angles between camera and glass range from 10 to 35 degrees. The glass thicknesses are between 4 and 12mm. Note how, in Figure 1, the fruit and the text are recovered in separate layers.

In Figure 7, we compare our method to other reflection removal algorithms that take a single input image. The input shows an image of a building facade and a fire escape, and it contains reflections of the photographer and camera, which are clearly seen in the reflection layer recovered by our algorithm. The second column shows the results of [21], which requires user annotations. Our manual annotations (for the algorithm of [21]) are shown in the inset. Red points label reflection gradients, and blue points label transmission gradients. The third column shows the results of Li and Brown [25]. Their method assumes that the reflection layer is blurry, and therefore cannot handle natural reflection components

Figure 6: *Results on real-world inputs. We show only the recovered transmission layers.*

such as the camera logo. We used implementations from the authors' web sites.

Next, we test image classification performance on recovered transmission layers. An application of this could be in automated driver assistance systems with dashboard cameras for object detection. The input image to the camera could be affected by reflection from the windshield. We use a convolutional neural network (CNN) and the pre-trained ImageNet model over 1000 classes provided in Caffe [16].

The test data is prepared by randomly sampling 20,000 images from the test set in ImageNet [7]. We then draw 10,000 pairs from these images; each pair has an associated transmission and reflection image. We mix each pair to generate 10,000 images; the ghosting kernel k is generated with a random attenuation c_k between 0.5 and 1, and random shift $\mathbf{d}_k$ generated by sampling a shift between -20 to 20 pixels in both the x and y directions. Table 1 compares clas-

	Input	Recovered Transmission	Ground Truth Transmission
Top-1	23.6 %	40.1%	60.5 %
Top-3	36.9 %	58.2%	77.3 %
Top-5	42.4 %	63.9%	82.2 %

Table 1: *Label prediction accuracy on 10,000 synthetically generated inputs. We show the Top-1, Top-3 and Top-5 accuracy on the input, recovered transmission and ground truth transmission layers. The recovered transmission layer has significantly improved accuracy.*

Figure 7: *We compare our result to two other methods on single image reflection removal. The input is a facade and a escape stair, containing reflections of a camera and the photographer. Our method successfully separates the camera reflection. Levin et al. [2007] requires user annotations, which is created by us and shown in the inset.*

sification rates on the input ghosted images, the recovered transmission layer and the ground truth. We see that the recovered transmission layers provide significantly improved classification compared to classifying on the input images. In Figure 8, we show examples of labels predicted by the CNN.

Figure 9 shows an application of our algorithm for automatic de-ghosting. In product photography, the product is often placed on a reflective surface for aesthetic reasons. An example of the resulting image is shown in Figure 9(a). We use our method to decompose the input into the transmission and the reflection layers, and then additively remix them to create the ghosting-free result, shown in Figure 9(b). Our unoptimized MATLAB implementation takes 22 minutes on 24 CPUs to process an input RGB image of size 400×600.

6. Discussion

We have introduced a new algorithm to exploit ghosting cues for layer separation in images taken through glass windows. We show under what conditions this ghost is visible, and present an algorithm which uses patch-based GMM priors to achieve high-quality separation of reflection and transmission images on real-world and synthetic data. Our method is currently limited to spatially-invariant ghosting kernels. Kernels can vary spatially when the reflected scene contains a wide range of depths [8], for very wide angle views, or when the polarization of the reflected light varies spatially.

When the transmission layer contains double features

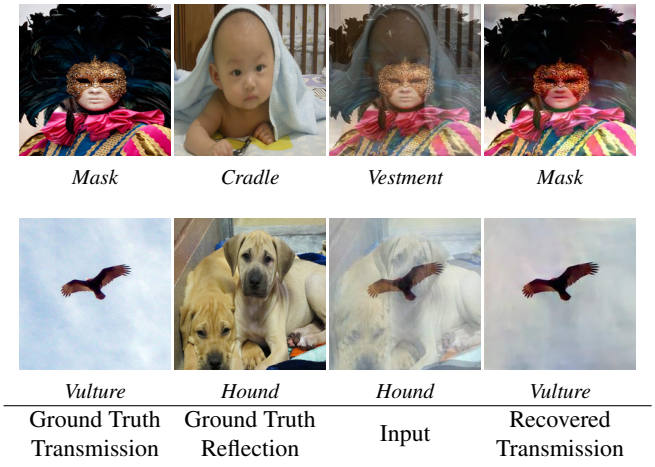
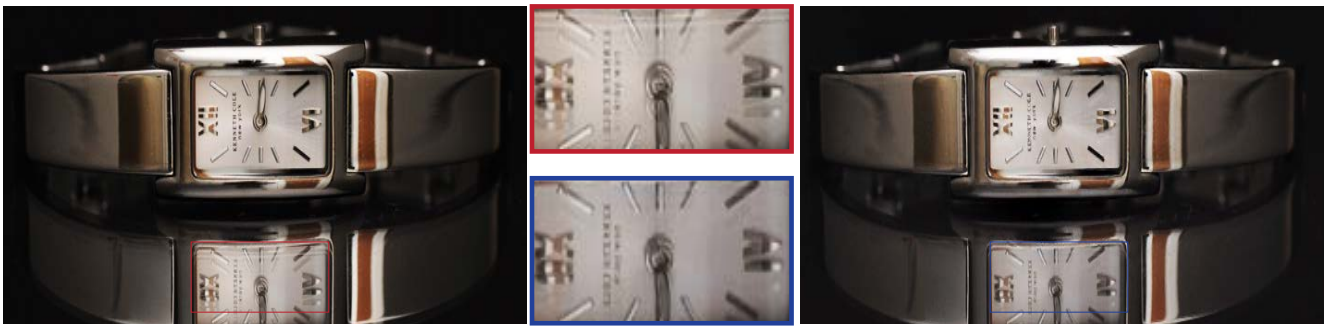


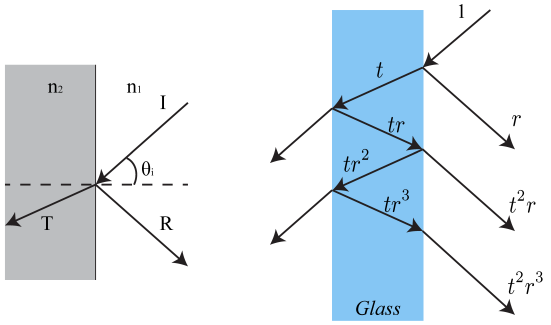
Figure 8: *Examples from ImageNet dataset [7]. From left to right, the columns show the ground truth transmission, ground truth reflection, the synthetically generated input, and our recovered transmission layer. The captions below each image are the labels predicted by a trained CNN [16]. Predicting labels on the image with ghosting gives incorrect results. Separating the layers prior to classification helps predict the correct labels.*

such as edges separated by exactly the ghosting distance and relative strength similar to the same factor c_k, then determining layer ownership is problematic. A corresponding difficulty can occur when the reflection layer contains nega-



(**a**) *Input* (**b**) *Automatically de-ghosted output*

Figure 9: *Automatic de-ghosting: (a) the input shows a watch placed on a mirror surface for product photography. The red inset shows the ghosting artifacts due to the thickness of the mirror, and the blue inset is a zoom into the de-ghosted reflection layer; (b) Our automatic method removes the ghosting artifacts.*



(**a**) *Single interface* (**b**) *Two interfaces*

Figure 10: *Illustration of Fresnel's equations and ghosting effects.*

tively correlated features at the ghosting distance, such as a positive and a negative edge. We have found that the quality of our recovered reflections tends to be lower than that of transmissions, possibly due to the asymmetry in our imaging model. Ghosting cues break the symmetry between reflection and transmission but tend to be less effective for low frequencies. Better low-frequency regularization techniques would be an interesting research direction.

Appendix: Fresnel's Equations and Ghostings

We derive ghosting effects from Fresnel's equations. Consider the setup in Figure 10(a). When a light ray enters from one media to another media, part of the ray is reflected. The fraction of the incident power that is reflected from the interface is given by the reflectance r_s, given by the following:

$$r_s = \left| \frac{n_1 \cos\theta_i - n_2 \sqrt{1 - (\frac{n_1}{n_2} \sin\theta_i)^2}}{n_1 \cos\theta_i + n_2 \sqrt{1 - (\frac{n_1}{n_2} \sin\theta_i)^2}} \right|^2 \quad (10)$$

where θ_i is the incident angle, n_1 and n_2 are the refractive indexes for the two media, which are air and glass in our work. We will use $n_1 = 1$ and $n_2 = 1.5$ in the following, corresponding to typical values for air and glass. The reflectance of the parallel-polarized component, denoted by r_p, is given by:

$$r_p = \left| \frac{n_1 \sqrt{1 - (\frac{n_1}{n_2} \sin\theta_i)^2} - n_2 \cos\theta_i}{n_1 \sqrt{1 - (\frac{n_1}{n_2} \sin\theta_i)^2} + n_2 \cos\theta_i} \right|^2 \quad (11)$$

If the incident light is unpolarized, which means that it contains equal energy of parallel and perpendicularly polarized components, then the effective reflectance r is $\frac{r_s + r_p}{2}$. The transmittance is:

$$t = 1 - r \quad (12)$$

Now we consider the setup in Figure 10(b), which contains two interfaces. Considering the incident light with unit energy, we label the attenuated energy on each reflected light ray. The attenuation factor c in our kernel, which is the ratio between the secondary and the primary reflection, is t^2. Plugging in Equation 10, Equation 11, and Equation 12, using $\theta_i = 45$ degrees, we have $t = 0.95$, and $c = t^2 = 0.91$. The ratio between the third and the primary reflection is $t^2 r^2 \ll c$, since usually $r \ll 1$. In the above setup, this ratio is less than 0.01. Under the assumption of unpolarized light, both t and r are only dependent on the incident angle θ_i, and spatially-invariant across the camera imaging plane.

Acknowledgements

This research was supported by grants from Quanta and Qatar Computing Research Institute.

References

[1] Wikipedia. http://en.wikipedia.org/wiki/Insulated_glazing. 1

[2] E. H. Adelson and P. Anandan. Ordinal characteristics of transparency. AAAI-90 Workshop on Qualitative Vision, 1990. 3

[3] A. Agrawal, R. Raskar, S. K. Nayar, and Y. Li. Removing photography artifacts using gradient projection and flash-exposure sampling. In *ACM Transactions on Graphics (SIGGRAPH)*, volume 24, pages 828–835, 2005. 1, 3

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011. 5

[5] V. Bychkovsky, S. Paris, E. Chan, and F. Durand. Learning photographic global tonal adjustment with a database of input/output image pairs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 97–104, 2011. 6

[6] W. Ciciora, G. Sgrignoli, and W. Thomas. A tutorial on ghost cancelling in television systems. *IEEE Transactions on Consumer Electronics,*, (1):9–44, 1979. 3

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. 6, 7

[8] Y. Diamant and Y. Y. Schechner. Eliminating artifacts when inverting visual reverebrations. *Techical Report, Technion*, 2008. 7

[9] Y. Diamant and Y. Y. Schechner. Overcoming visual reverberations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 2, 3, 5

[10] H. Farid and E. H. Adelson. Separating reflections and lighting using independent components analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, 1999. 1, 2

[11] K. Gai, Z. Shi, and C. Zhang. Blindly separating mixtures of multiple layers with spatial shifts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 3

[12] K. Gai, Z. Shi, and C. Zhang. Blind separation of superimposed moving images using image statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(1):19–32, 2012. 3

[13] D. Geman and C. Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Transactions on Image Processing (TIP)*, 4(7):932–946, 1995. 5

[14] E. Hecht. Optics (second edition). pages 100–102. Addison Wesley, 1987. 3

[15] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *International Journal of Computer Vision (IJCV)*, 12(1):5–16, 1994. 3

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 6, 7

[17] K. Kayabol, E. E. Kuruoglu, and B. Sankur. Image source separation using color channel dependencies. In *Independent Component Analysis and Signal Separation*, pages 499–506. Springer, 2009. 3

[18] J. Koenderink, A. van Doorn, S. Pont, and W. Richards. Gestalt and phenomenal transparency. *Journal of Optical Society of America A*, 25(1):190–202, 2008. 3

[19] N. Kong, Y. Tai, and J. Shin. A physically-based approach to reflection separation: From physical modeling to constrained optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(2):209–221, 2014. 1, 2

[20] J. Kopf, F. Langguth, D. Scharstein, R. Szeliski, and M. Goesele. Image-based rendering in the gradient domain. *ACM Transactions on Graphics (TOG)*, 32(6):199, 2013. 3

[21] A. Levin and Y. Weiss. User assisted separation of reflections from a single image using a sparsity prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 29(9):1647–1654, 2007. 1, 3, 6

[22] A. Levin, A. Zomet, and Y. Weiss. Learning to perceive transparency from the statistics of natural scenes. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1247–1254, 2002. 3

[23] A. Levin, A. Zomet, and Y. Weiss. Separating reflections from a single image using local features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages I–306, 2004. 3

[24] Y. Li and M. S. Brown. Exploiting reflection change for automatic reflection removal. In *IEEE International Conference on Computer Vision (ICCV)*, 2013. 3

[25] Y. Li and M. S. Brown. Single image layer separation using relative smoothness. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 3, 6

[26] B. Sarel and M. Irani. Separating transparent layers through layer information exchange. In *European Conference on Computer Vision (ECCV)*, pages 328–341. Springer, 2004. 1, 2

[27] B. Sarel and M. Irani. Separating transparent layers of repetitive dynamic behaviors. In *IEEE International Conference on Computer Visioin (ICCV)*, volume 1, pages 26–32, 2005. 3

[28] Y. Y. Schechner, N. Kiryati, and J. Shamir. Blind recovery of transparent and semireflected scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 38–43, 2000. 1, 3

[29] Y. Y. Schechner, J. Shamir, and N. Kiryati. Polarization and statistical analysis of scenes containing a semireflector. *The Journal of the Optical Society of America A*, 17(2):276–284, 2000. 1, 2

[30] M. Singh and X. Huang. Computing layered surface representations: an algorithm for detecting and separating transparent overlays. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages II–11, 2003. 3

[31] S. N. Sinha, J. Kopf, M. Goesele, D. Scharstein, and R. Szeliski. Image-based rendering for scenes with reflections. *ACM Transaction on Graphics (SIGGRAPH)*, 31(4):100, 2012. 1, 3

[32] R. Szeliski, S. Avidan, and P. Anandan. Layer extraction from multiple images containing reflections and transparency. In *IEEE Conference Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 246–253, 2000. 1, 3, 4

[33] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4):550–560, 1997. 5

[34] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *IEEE International Conference on Computer Vision*, pages 479–486, 2011. 2, 3, 4, 5