

# Shape, Illumination, and Reflectance from Shading

*Jonathan Barron*  
*Jitendra Malik*

Electrical Engineering and Computer Sciences  
University of California at Berkeley

Technical Report No. UCB/EECS-2013-117

<http://www.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-117.html>

May 29, 2013



Copyright © 2013, by the author(s).  
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

# Shape, Illumination, and Reflectance from Shading

Jonathan T. Barron and Jitendra Malik,

**Abstract**—A fundamental problem in computer vision is that of inferring the intrinsic, 3D structure of the world from flat, 2D images of that world. Traditional methods for recovering scene properties such as shape, reflectance, or illumination rely on multiple observations of the same scene to overconstrain the problem. Recovering these same properties from a single image seems almost impossible in comparison — there are an infinite number of shapes, paint, and lights that exactly reproduce a single image. However, certain explanations are more likely than others: surfaces tend to be smooth, paint tends to be uniform, and illumination tends to be natural. We therefore pose this problem as one of statistical inference, and define an optimization problem that searches for the *most likely* explanation of a single image. Our technique can be viewed as a superset of several classic computer vision problems (shape-from-shading, intrinsic images, color constancy, illumination estimation, etc) and outperforms all previous solutions to those constituent problems.

## 1 INTRODUCTION

AT the core of computer vision is the problem of taking a single image, and estimating the physical world which produced that image. The physics of image formation makes this “inverse optics” problem terribly challenging and underconstrained: the space of shapes, paint, and light that exactly reproduce an image is vast.

This problem is perhaps best motivated using Adelson and Pentland’s “workshop” metaphor [1]: consider the image in Figure 1(a), which has a clear percept as a twice-bent surface with a stroke of dark paint (Figure 1(b)). But this scene could have been created using any number of physical worlds — it could be realistic painting on a canvas (Figure 1(c)), a complicated arrangement of bent shapes (Figure 1(d)), a sophisticated projection produced by a collection of lights (Figure 1(e)), or anything in between. The job of a perceptual system is analogous to that of a prudent manager in this “workshop”, where we would like to reproduce the scene using as little effort from our three artists as possible, giving us Figure 1(b).

This metaphor motivates the formulation of this problem as one of statistical inference. Though there are infinitely many possible explanations for a single image, some are more likely than others. Our goal is therefore to recover the *most likely explanation* that explains an input image. We will demonstrate that in natural depth maps, reflectance maps, and illumination models, very strong statistical regularities arise that are similar to those found in natural images [2], [3]. We will construct priors similar to those used

in natural image statistics, but applied separately to shape, reflectance, and illumination. Our algorithm is simply an optimization problem in which we recover the most likely shape, reflectance, and illumination under these priors that exactly reproduces a single image. Our priors are powerful enough that these intrinsic scene properties can be recovered from a single image, but are general enough that they work across a variety of objects.

The output of our model relative to ground-truth can be seen in Figure 2. Our model is capable of producing qualitatively correct reconstructions of shape, surface normals, shading, reflectance, and illumination, from a single image. We quantitatively evaluate our model on variants of the MIT intrinsic images dataset [4], on which we quantitatively outperform all

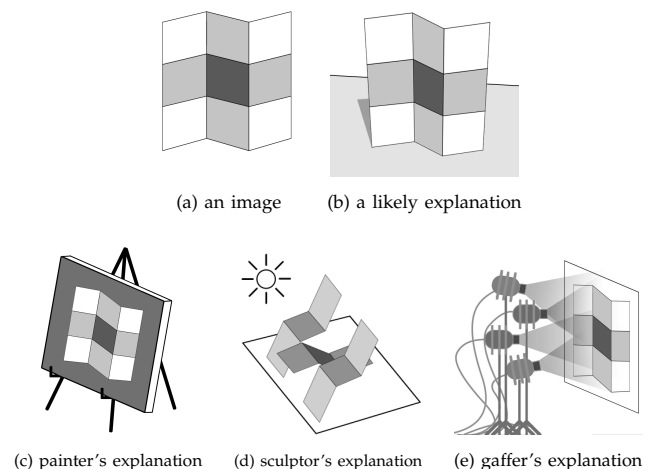


Fig. 1. A visualization of Adelson and Pentland’s “workshop” metaphor [1]. The image in 1(a) clearly corresponds to the interpretation in 1(b), but it could be a painting, a sculpture, or an arrangement of lights.

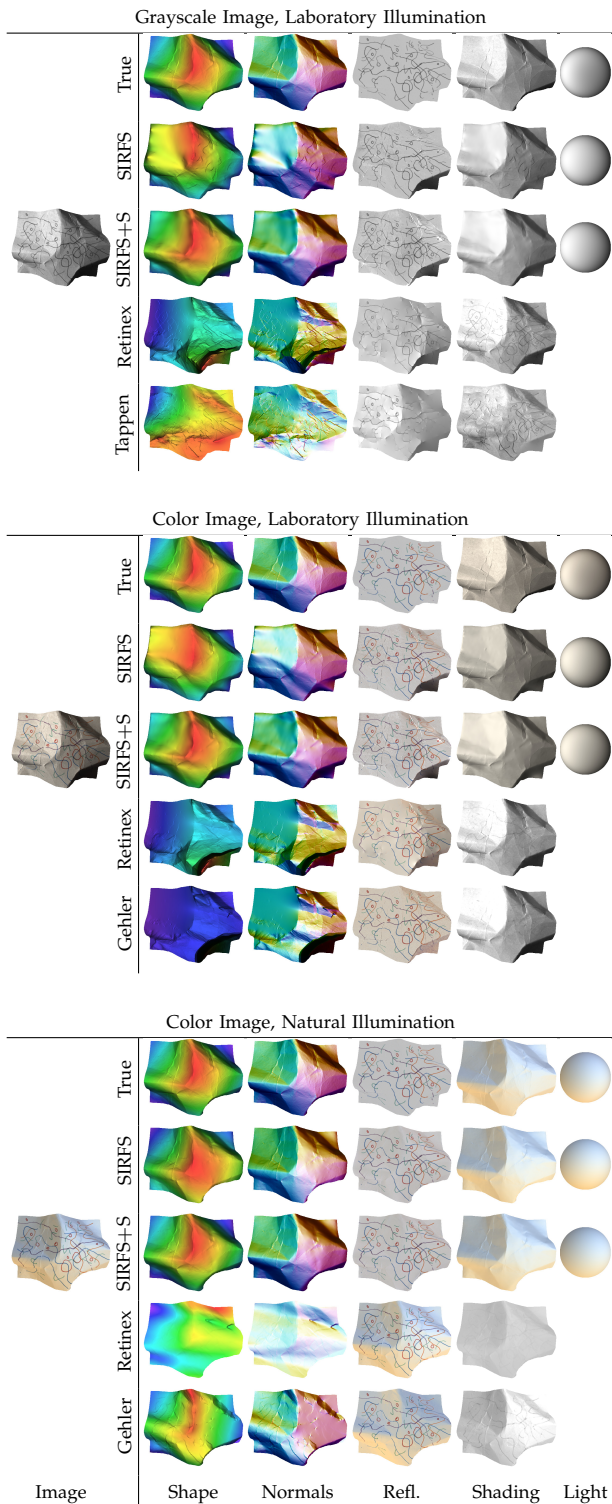


Fig. 2. A single image from our dataset, under three color/illumination conditions. For each condition, we present the ground-truth, the output of SIRFS, the output of SIRFS+S (which uses external shape information), and the two best-performing intrinsic image techniques (for which we do SFS on the recovered shading to recover shape). Best viewed in color.

previously published intrinsic image or shape-from-

shading algorithms. We additionally present qualitative results for many more real-world images, for which we do not have ground-truth explanations.

Earlier versions of this work have been presented in a piecemeal fashion, over the course of many papers [5], [6], [7]. This paper is meant to simplify and unify those previous methods.

This paper will proceed as follows: In Section 2, we will review past work as it relates to our own. In Section 3 we will formulate our problem as one of statistical inference and optimization, with respect to a set of priors over shape, reflectance, and illumination. In Sections 4, 5, and 6 we present and motivate our priors on reflectance, shape, and illumination, respectively. In Section 7 we explain how we solve our proposed optimization problem. In Section 8 we present a series of experiments with our model on variants of the MIT Intrinsic Images dataset [4] and on real-world images, and in Section 9 we conclude.

## 2 PRIOR WORK

The question of how humans solve the underconstrained problem of perceiving shape, reflectance, and illumination from a single image appears to be at least one thousand years old, dating back to the scientist Alhazen, who noted that “Nothing of what is visible, apart from light and color, can be perceived by pure sensation, but only by discernment, inference, and recognition, in addition to sensation.” In the 19th century the problem was studied by such prominent vision scientists as von Helmholtz, Hering and Mach [8], who framed the problem as one of “lightness constancy” — how humans, when viewing a flat surface with patches of varying reflectances subject to spatially varying illumination, are able to form a reasonably veridical percept of the reflectance (“lightness”) in spite of the fact that a darker patch under brighter illumination may well have more light traveling from it to the eye compared to a lighter patch which is less well illuminated.

Land’s Retinex theory of lightness constancy [9] has been particularly influential in computer vision since its introduction in 1971. It provided a computational approach to the problem in the “Mondrian World”, a 2D world of flat patches of piecewise constant reflectance. Retinex theory was later made practical by Horn [10], who was able to obtain a decomposition of an image into its shading and reflectance components using the prior belief that sharp edges tend to be reflectance, and smooth variation tends to be shading.

In 1978, Barrow and Tenebaum defined what they called the problem of “intrinsic images”: recovering properties such as shape, reflectance, and illumination from a single image [11]. In doing so, they described a challenge in computer vision which is still largely unsolved, and which our work directly addresses. Because this problem is so fundamentally



underconstrained and challenging, the computer vision community has largely focused its attention on more constrained and tractable sub-problems. Over time, “intrinsic images” has become synonymous with the problem that Retinex addressed, that of separating an image into shading and reflectance components [4], [10], [9]. This area has seen some recent progress [12], [13], [14], [15], though the performance of Retinex, despite its age, has proven hard to improve upon [4]. The limiting factor in many of these “intrinsic image” algorithms appears to be that they treat “shading” as a kind of image, ignoring the fact that shading is, by construction, the product of some shape and some model of illumination. By addressing a superset of this “intrinsic image” problem and recovering shape and illumination instead of shading, our model produces better results than any intrinsic image technique.

Related to the problem of lightness constancy or “intrinsic images” is the problem of color constancy, which can be thought of as a generalization of lightness constancy from grayscale to color, in which the problem is simplified by assuming that there is just one single model of illumination for an entire image, rather than a spatially-varying “shading” effect. Early techniques for color constancy used gamut mapping techniques [16], finite dimensional models of reflectance and illumination [17], and physically based techniques for exploiting specularities [18]. More recent work uses contemporary probabilistic tools, such as modeling the correlation between colors in a scene [19], or performing inference over priors on reflectance and illumination [20]. All of this work shares the assumptions of “intrinsic image” algorithms that shape (and to a lesser extent, shading) can be ignored or abstracted away.

The second subset of the Barrow and Tenenbaum’s original “intrinsic image” formulation that the computer vision research community has focused on is the “shape-from-shading” (SFS) problem. SFS is traditionally defined as: recovering the shape of an object given a single image of it, assuming illumination and reflectance are known (or assuming reflectance is uniform across the entire image). This problem formulation is very complimentary to the shape-vs-reflectance version of the “intrinsic images” problem, as it focuses on the parts of the problem which “intrinsic images” ignores, and vice-versa.

The shape-from-shading problem was first formulated in the computer vision community by Horn in 1975 [21], though the problem existed in other fields as that of “photoclinometry” [22]. The history of SFS is well surveyed in [23], [24]. Despite being a severe simplification of the complete intrinsic images problem, SFS is still a very ill-posed and underconstrained, and challenging problem. One notable difficulty in SFS is the Bas-relief ambiguity [25], which states (roughly) that the absolute orientation and scaling of a surface

is ambiguous given only shading information. This ambiguity holds true not only for SFS algorithms, but for human vision as well [26]. We address this ambiguity by imposing priors on shape, building on notions of “smoothness” priors in SFS [27], and by allowing for external observations of shape (such as those produced by a stereo system or depth sensor) to be introduced.

Our model can be viewed as a generalization of an “intrinsic image” algorithm or color constancy algorithm in which shading is explicitly parametrized as a function of shape and illumination. Similarly, our model can be viewed as a shape-from-shading algorithm in which reflectance and illumination are unknown, and are recovered. Our model therefore addresses the “complete” intrinsic images problem, as it was first formulated. By addressing the complete problem, rather than two sub-problems in isolation, we outperform all previous algorithms for either sub-problem. This is consistent with our understanding of human perception, as humans use spatial cues when estimating reflectance and shading [8], [28].

Because the intrinsic images problem is so challenging given only a single image, a much more popular area of research in computer vision has been to introduce additional data to better constrain the problem. Instances of this approach are photometric stereo [29], which use additional images with different illumination conditions to estimate shape, and in later work reflectance and illumination [30]. Our algorithm produces the same kinds of output as the most advanced photometric stereo algorithm, while requiring only a single image. “Structure from motion” or binocular stereo [31], [32] uses multiple images to recover shape, but ignores shading, reflectance, and illumination. Inverse global illumination [33] recovers reflectance and illumination given shape and multiple images, while we recover shape and require only a single image.

Recent work has explored using learning to directly infer the spatial layout of a scene from a single image [34], [35]. These techniques ignore illumination and reflectance, and produce only a coarse estimate of shape.

A similar approach to our technique is that of category-specific morphable models [36] which, given a single image of a very specific kind of object (a face, usually), estimates shape, reflectance, and illumination. These techniques use extremely specific models (priors) of the objects being estimated, and therefore do not work for general objects, while our priors are general enough to be applicable on a wide variety of objects: a single model learned on teabags and squirrels can be applied to images of coffee cups and turtles.

The driving force behind our model are our priors on shape, reflectance, and illumination. To construct these priors we build upon past work on natural

image statistics, which has demonstrated that simple statistics govern local patches of natural images [2], [3], [37], and that these statistics can be used for denoising [38], inpainting [39], deblurring [40], etc. But these statistical regularities arise in natural images only because of *the statistical regularities in the underlying worlds that produced those images*. The primary contribution of this work is extended these ideas from natural images to the world that produced that natural image, which is assumed to be composed of natural depth maps and natural reflectance images. There has been some study of the statistics of natural depth maps [41], reflectance images [42] and models of illumination [43], but ours is the first to use these statistical observations for recovering all such intrinsic scene properties simultaneously.

### 3 PROBLEM FORMULATION

We call our problem formulation for recovering intrinsic scene properties from a single image of a (masked) object “shape, illumination, and reflectance from shading”, or “SIRFS”. SIRFS can be thought of as an extension of classic shape-from-shading models [44] in which not only shape, but reflectance and illumination are unknown. Conversely, SIRFS can be framed as an “intrinsic image” technique for recovering shading and reflectance, in which shading is parametrized by a model of shape and illumination. The SIRFS problem formulation is:

$$\begin{aligned} & \underset{R, Z, L}{\text{maximize}} && P(R)P(Z)P(L) \\ & \text{subject to} && I = R + S(Z, L) \end{aligned} \quad (1)$$

Where  $R$  is a log-reflectance image,  $Z$  is a depth-map, and  $L$  is a spherical-harmonic model of illumination [45].  $Z$  and  $R$  are “images” with the same dimensions as  $I$ , and  $L$  is a vector parametrizing the illumination.  $S(Z, L)$  is a “rendering engine” which linearizes  $Z$  into a set of surface normals, and produces a log-shading image from those surface normals and  $L$  (see the supplementary material).  $P(R)$ ,  $P(Z)$ , and  $P(L)$  are priors on reflectance, shape, and illumination, respectively, whose likelihoods we wish to maximize subject to the constraint that the log-image  $I$  is equal to a rendering of our model  $R + S(Z, L)$ . We can simplify this problem formulation by reformulating the maximum-likelihood aspect as minimizing a sum of cost functions (by taking the negative log of  $P(R)P(Z)P(L)$ ) and by absorbing the constraint and removing  $R$  as a free parameter. This gives us the following unconstrained optimization problem:

$$\underset{Z, L}{\text{minimize}} \quad g(I - S(Z, L)) + f(Z) + h(L) \quad (2)$$

where  $g(R)$ ,  $f(Z)$ , and  $h(L)$  (Sections 4, 5, and 6, respectively) are cost functions for reflectance, shape, and illumination respectively, which we will refer to

as our “priors” on these scene properties<sup>1</sup>. Solving this problem (Section 7) corresponds to searching for the least costly (or most likely) explanation  $\{Z, R, L\}$  for image  $I$ .

### 4 PRIORS ON REFLECTANCE

Our prior on reflectance consists of three components: 1) An assumption of piecewise constancy, which we will model by minimizing the local variation of log-reflectance in a heavy-tailed fashion. 2) An assumption of parsimony of reflectance — that the palette of colors with which an entire image was painted tends to be small — which we model by minimizing the global entropy of log-reflectance. 3) An “absolute” prior on reflectance which prefers to paint the scene with some colors (white, gray, green, brown, etc) over others (absolute black, neon pink, etc), thereby addressing color constancy. Formally, our reflectance prior  $g(A)$  is a weighted combination of three costs:

$$g(R) = \lambda_s g_s(R) + \lambda_e g_e(R) + \lambda_a g_a(R) \quad (3)$$

where  $g_s(R)$  is our smoothness prior,  $g_e(R)$  is our parsimony prior, and  $g_a(R)$  is our “absolute” prior. The  $\lambda$  multipliers are learned through cross-validation on the training set.

Our smoothness and parsimony priors are on the differences of log-reflectance, which makes them equivalent to priors on the ratios of reflectance. This makes intuitive sense, as reflectance is defined as a ratio of reflected light to incident light, but is also crucial to the success of our algorithm: Consider the reflectance-map  $\rho$  implied by log-image  $I$  and log-shading  $S(Z, L)$ , such that  $\rho = \exp(I - S(Z, L))$ . If we were to manipulate  $Z$  or  $L$  to increase  $S(Z, L)$  by some constant  $\alpha$  across the entire image, then  $\rho$  would be divided by  $\exp(\alpha)$  across the entire image, which would accordingly decrease the differences between pixels of  $\rho$ . Therefore, if we placed priors on the differences of reflectance it would be possible to trivially satisfy our priors by manipulating shape or illumination to increase the intensity of the shading image. However, in the log-reflectance case  $R = I - S(Z, L)$ , increasing all of  $S$  by  $\alpha$  (increasing the brightness of the shading image) simply decreases all of  $R$  by  $\alpha$ , and does not change the differences between log-reflectance values (it would, however, affect our absolute prior on reflectance). Priors on the differences of log-albedo are therefore invariant to scaling of illumination or

1. Throughout this paper we use the term “prior” loosely. We refer to loss functions or regularizers on  $Z$ ,  $A$ , and  $L$  as “priors” because they often have an interpretation as the negative log-likelihood of some probability density function. We refer to minimizing entropy as a “prior”, which is again an abuse of terminology. Our occluding contour “prior” and our external observation “prior” require first observing the silhouette of an object or some external observation of shape, respectively, and are therefore posteriors, not priors.

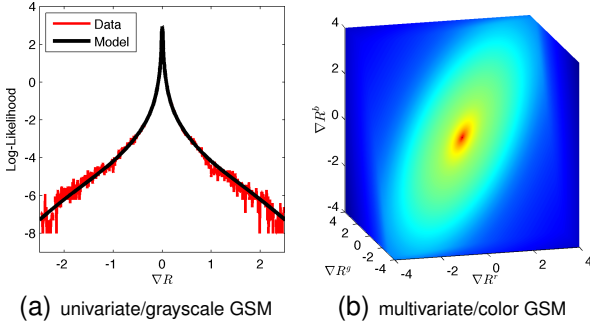


Fig. 3. Our smoothness prior on log-reflectance is a univariate Gaussian scale mixture on the differences between nearby reflectance pixels for grayscale images, or a multivariate GSM for color images. These distribution prefers nearby reflectance pixels to be similar, but its heavy tails allow for rare non-smooth discontinuities. Our multivariate color model captures the correlation between color channels, which means that chromatic variation in log-reflectance lies further out in the tails, making it more likely to be ignored during inference.

shading, which means they behave similarly in well-lit regions as in shadowed regions, and cannot be trivially satisfied.

#### 4.1 Smoothness

The reflectance images of natural objects tend to be piecewise constant — or equivalently, variation in reflectance images tends to be small and sparse. This is the insight that underlies the Retinex algorithm [4], [9], [10], and informs more recent intrinsic images work [13], [14], [15].

Our prior on grayscale reflectance smoothness is a multivariate Gaussian scale mixture (GSM) placed on the differences between each reflectance pixel and its neighbors. We will maximize the likelihood of  $R$  under this model, which corresponds to minimizing the following cost function:

$$g_s(R) = \sum_i \sum_{j \in N(i)} c(R_i - R_j; \alpha_R, \sigma_R) \quad (4)$$

Where  $N(i)$  is the  $5 \times 5$  neighborhood around pixel  $i$ ,  $R_i - R_j$  is a the difference in log-RGB from pixel  $i$  to pixel  $j$ , and  $c(\cdot; \alpha, \sigma)$  is the negative log-likelihood of a discrete univariate Gaussian scale mixture (GSM), parametrized by  $\alpha$  and  $\sigma$ , the mixing coefficients and standard deviations, respectively, of the Gaussians in the mixture:

$$c(x; \alpha, \sigma) = -\log \sum_{j=1}^M \alpha_j \mathcal{N}(x; 0, \sigma_j^2) \quad (5)$$

We set the mean of the GSM is 0, as the most likely reflectance image under our model should be flat. We set  $M = 40$  (the GSM has 40 discrete Gaussians),

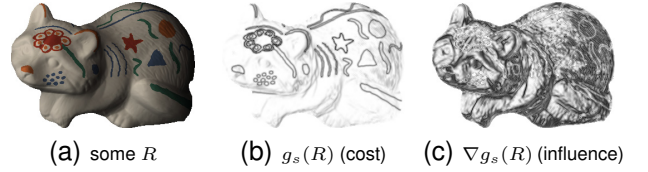


Fig. 4. Here we have a color reflectance image  $R$ , and its cost and influence (derivative of cost) under our multivariate GSM smoothness prior. Strong, colorful edges, such as those caused by reflectance variation, are very costly, while small edges, such as those caused by shading, are less costly. But in terms of *influence* — the gradient of cost with respect to each pixel — we see an inversion: because sharp edges lie in the tails of the GSM, they have little influence, while shading variation has great influence. This means that during inference our model attempts to explain shading (small, achromatic variation) in the image by varying shape, while explaining sharp or chromatic variation by varying reflectance.

and  $\alpha_R$  and  $\sigma_R$  are trained on reflectance images in our training set using expectation-maximization. The log-likelihood of our learned model can be seen in Figure 3(a).

Gaussian scale mixtures have been used previously to model the heavy-tailed distributions found in natural images [38], for the purpose of denoising or inpainting. Effectively, using this family of distributions gives us a log-likelihood which looks like a smooth, heavy-tailed spline which decreases monotonically with distance from 0. Because it is monotonically decreasing, the cost of log-reflectance variation increases with the magnitude of variation, but because the distribution is heavy tailed, the influence of variation (the derivative of log-likelihood) is strongest when variation is small (that is, when variation resembles shading) and weaker when variation is large. This means that our model prefers a reflectance image that is mostly flat but occasionally varies heavily, but abhors a reflectance image which is constantly varying slightly. This behavior is similar to that of the Retinex algorithm, which operates by shifting strong gradients to the reflectance image and weak gradients to the shading image.

To extend our model to color images, we simply extend our smoothness prior to a multivariate Gaussian scale mixture

$$g_s(R) = \sum_i \sum_{j \in N(i)} C(R_i - R_j; \alpha_R, \sigma_R, \Sigma_R) \quad (6)$$

Where  $R_i - R_j$  is now a 3-vector of the log-RGB differences,  $\alpha$  are mixing coefficients,  $\sigma$  are the scalings of the Gaussians in the mixture, and  $\Sigma$  is the covariance matrix of the entire GSM (shared among all Gaussians)

of the mixture).

$$C(\mathbf{x}; \boldsymbol{\alpha}, \boldsymbol{\sigma}, \Sigma) = -\log \sum_{j=1}^M \alpha_j \mathcal{N}(\mathbf{x}; \mathbf{0}, \sigma_j \Sigma) \quad (7)$$

We set  $M = 40$  (the GSM has 40 discrete Gaussians), and we train  $\boldsymbol{\alpha}_R$ ,  $\boldsymbol{\sigma}_R$ , and  $\Sigma_R$  on color reflectance images in our training set (we train a distinct model from the grayscale smoothness model). The log-likelihood of our learned model, and the training data used to learn that model, can be seen in Figure 3(b).

In color images, variation in reflectance tends to manifest itself in both the luminance and chrominance of an image (white transitioning to blue, for example) while shading, assuming the illumination is mostly white, primarily affects the luminance of an image (light blue transitioning to dark blue, for example). Past work has exploited this insight by building specialized models that condition on the chrominance variation of the input image [4], [10], [13], [14], [15]. By placing a multivariate prior over differences in reflectance, we are able to capture the correlation of the different color channels, which implicitly encourages our model to explain chromatic variation using reflectance and achromatic variation using shading without the need for any hand-crafted heuristics. See Figure 4 for a demonstration of this effect. Our model places more-colorful edges further into the tails of the distribution, thereby reducing their influence. Again, this is similar to color variants of the Retinex algorithm [4] which uses the increased chrominance of an edge as a heuristic for it being a reflectance edge. But this approach (which is common among intrinsic image algorithms) of using image chrominance as a substitute for reflectance chrominance means that these techniques fail when faced with non-white illumination, while our model is robust to non-white illumination.

## 4.2 Parsimony

In addition to piece-wise smoothness, the second property we expect from reflectance images is for there to be a small number of reflectances in an image — that the palette with which an image was painted be small. As a hard constraint, this is not true: even in painted objects, there are small variations in reflectance. But as a soft constraint, this assumption holds. In Figure 5 we show the marginal distribution of grayscale log-reflectance for three objects in our dataset. Though the man-made “cup1” object shows the most clear peakedness in its distribution, natural objects like “apple” show significant clustering.

We will therefore construct a prior which encourages parsimony — that our representation of the reflectance of the scene be economical and efficient, or “sparse”. This is effectively an instance of Occam’s

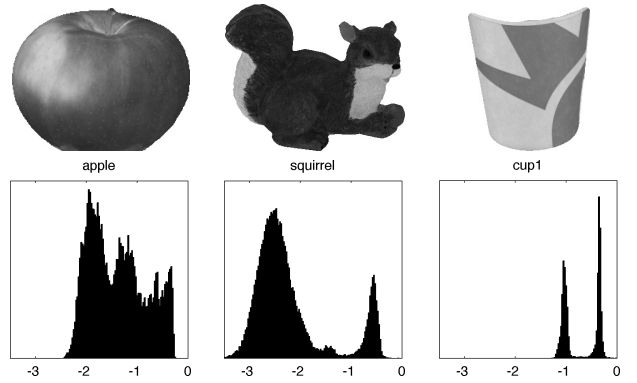


Fig. 5. Three grayscale log-reflectance images from our dataset and their marginal distributions. Log-reflectance in an image tend to be grouped around certain values, or equivalently, these distributions tend to be low-entropy.

razor, that one should favor the simplest possible explanation. We are not the first to explore global parsimony priors on reflectance: different forms of this idea have been used in intrinsic images techniques [15], photometric stereo [46], shadow removal [47], and color representation [48]. We use the quadratic entropy formulation of [49] to minimize the entropy of log-reflectance, thereby encouraging parsimony. Formally, our parsimony prior for reflectance is:

$$g_e(R) = -\log \left( \frac{1}{Z} \sum_{i=1}^N \sum_{j=1}^N \exp \left( -\frac{(R_i - R_j)^2}{4\sigma_R^2} \right) \right) \quad (8)$$

$$Z = N^2 \sqrt{4\pi\sigma^2}$$

This is quadratic entropy (a special case of Rényi entropy) for a set of points  $\mathbf{x}$  assuming a Parzen window (a Gaussian kernel density estimator, with a bandwidth of  $\sigma_R$ ) [49]. Effectively, this is a “soft” and differentiable generalization of Shannon entropy, computed on a set of real values rather than a discrete histogram. By minimizing this quantity, we encourage all pairs of reflectance pixels in the image to be similar to each other. However, minimizing this entropy does not force all pixels to collapse to one value, as the “force” exerted by each pair falls off exponentially with distance — it is robust to outliers. This prior effectively encourages Gaussian “clumps” of reflectance values, where the Gaussian clumps have standard deviations of roughly  $\sigma_R$ .

At first glance, it may seem that this global parsimony prior is redundant with our local smoothness prior: Encouraging piecewise smoothness seems like it should cause entropy to be minimized indirectly. This is often true, but there are common situations in which both of these priors are necessary. For example, if two regions are separated by a discontinuity in the image then optimizing for local smoothness will never cause the reflectance on both sides of the discontinuity to be

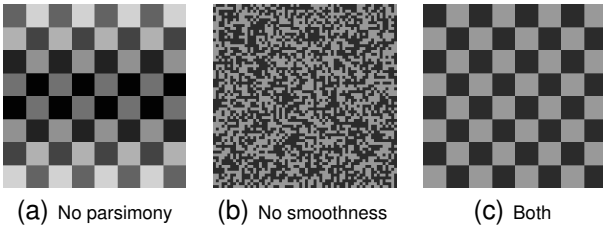


Fig. 6. A demonstration of the importance of both our smoothness and parsimony priors on reflectance. Using only a smoothness prior, as in 6(a), allows for reflectance variation across disconnected regions. Using only the parsimony prior, as in 6(b), encourages reflectance to take on a small number of values, but does not encourage it to form large piecewise-constant regions. Only by using the two priors in conjunction, as in 6(c), does our model correctly favor a normal, paint-like checkerboard configuration.

similar. Conversely, simply minimizing global entropy may force reflectance to take on a small number of values, but need not produce large piecewise-smooth regions. The merit of using both priors in conjunction is demonstrated in Figure 6.

Generalizing our grayscale parsimony prior to color reflectance images requires generalizing our entropy model to higher dimensionalities. A naive extension of this one-dimensional entropy model to three dimensions is not sufficient for our purposes: The RGB channels of natural reflectance images are highly correlated, causing a naive “isotropic” high-dimensional entropy measure to work poorly. To address this, we pre-compute a whitening transformation from log-reflectance images in the training set, and compute an isotropic entropy measure in this whitened space during inference, which gives us an anisotropic entropy measure. Formally, our cost function is quadratic entropy in the space of whitened log-reflectance:

$$g_e(R) = -\log \left( \frac{1}{Z} \sum_{i=1}^N \sum_{j=1}^N \exp \left( -\frac{\|W_R(R_i - R_j)\|_2^2}{4\sigma_R^2} \right) \right) \quad (9)$$

Where  $W_R$  is the whitening transformation learned from reflectance images in our training set, as follows: Let  $X$  be a  $3 \times n$  matrix of the pixels in the reflectance images in our training set. We compute the covariance matrix  $\Sigma = XX^T$  (ignoring centering), take its eigenvalue decomposition  $\Sigma = \Phi\Lambda\Phi^T$ , and from that construct the whitening transformation  $W_R = \Phi\Lambda^{1/2}\Phi^T$ .  $\sigma_R$  is the bandwidth of the Parzen window, which determines the scale of the clusters produced by minimizing this entropy measure, and is tuned through cross-validation (independently of the same variable for the grayscale case). See Figure 7 for a motivation of this model.

Naively computing this quadratic entropy measure requires calculating the difference between all  $N$  log-

reflectance values in the image with all other  $N$  log-reflectance values, making it quadratically expensive in  $N$  to compute naively. In the supplementary material, we describe an accurate linear-time algorithm for approximating this quadratic entropy and its gradient, based on the bilateral grid [50].

### 4.3 Absolute Reflectance

The previously described priors were imposed on *relative* properties of reflectance: the differences between nearby or not-nearby pixels. We must impose an additional prior on *absolute* reflectance: the raw value of each pixel in the reflectance image. Without such a prior (and the prior on illumination presented in Section 6) our model would be equally pleased to explain a gray pixel in the image as gray reflectance under gray illumination as it would nearly-black reflectance under extremely-bright illumination, or blue reflectance under yellow illumination, etc.

This sort of prior is fundamental to color-constancy, as most basic white-balance or auto-contrast/brightness algorithms can be viewed as minimizing a similar sort of cost: the gray-world assumption penalizes reflectance for being non-gray, the white-world assumption penalizes reflectance for being non-white, and gamut-based models penalize reflectance for lying outside of a gamut of previously-seen reflectances. We experimented with variations or combinations of these types of models, but found that what worked best was using a regularized smooth spline to model the log-likelihood of log-reflectance values.

For grayscale images, we use a 1D spline, which we have fit to log-reflectance images in the training set as follows:

$$\underset{\mathbf{f}}{\text{minimize}} \quad \mathbf{f}^T \mathbf{n} + \log \left( \sum_i \exp(-\mathbf{f}_i) \right) + \lambda \sqrt{(\mathbf{f}'')^2 + \epsilon^2} \quad (10)$$

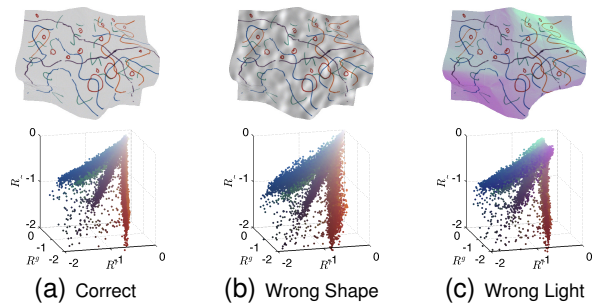


Fig. 7. Some reflectance images and their corresponding log-RGB scatterplots. Mistakes in estimating shape or illumination produce shading-like or illumination-like errors in the inferred reflectance, causing the the log-RGB distribution of the reflectance to be “smeared”, and causing entropy (and therefore cost) to increase.



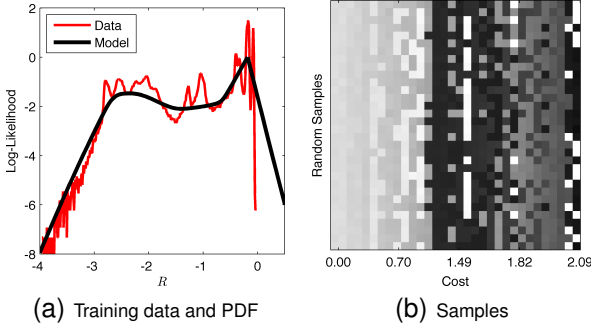


Fig. 8. A visualization of our “absolute” prior on grayscale reflectance. In 8(a) we have the log-likelihood of our density model, and the data on which it was trained. In 8(b) we have samples from our model, where the  $x$  axis is sorted by cost ( $y$  axis is random). Our model prefers reflectances that are close to white or gray, and that lie within the gamut of previously seen colors.

Where  $\mathbf{f}$  is our spline, which determines the non-normalized negative log-likelihood (cost) assigned to every reflectance,  $\mathbf{n}$  is a 1D histogram of log-reflectance in our training data, and  $\mathbf{f}''$  is the second derivative of the spline, which we robustly penalize ( $\epsilon$  is a small value added in to make our regularization differentiable everywhere). Minimizing the sum of the first two terms is equivalent to maximizing the likelihood of the training data (the second term is the log of the partition function for our density estimation), and minimizing the third term causes the spline to be piece-wise smooth. The smoothness multiplier  $\lambda$  is tuned through cross-validation. A visualization of our prior can be found in Figure 8.

During inference, we maximize the likelihood of the grayscale reflectance image  $R$  by minimizing its cost under our learned model:

$$g_a(R) = \sum_i \mathbf{f}(R_i) \quad (11)$$

where  $\mathbf{f}(R_i)$  is the value of  $\mathbf{f}$  at  $R_i$ , the log-reflectance at pixel  $i$ , which we computed using linear interpolation (so that this cost is differentiable).

To generalize this model to color reflectance images, we simply use a 3D spline, trained on whitened log-reflectance pixels in our training set. Formally, to train our model we minimize the following:

$$\text{minimize}_F \langle F, N \rangle + \log \left( \sum_i \exp(-F_i) \right) + \lambda \sqrt{J(F) + \epsilon^2}$$

$$J(F) = F_{xx}^2 + F_{yy}^2 + F_{zz}^2 + 2F_{xy}^2 + 2F_{yz}^2 + 2F_{xz}^2 \quad (12)$$

Where  $F$  is our 3D spline describing cost,  $N$  is a 3D histogram of the whitened log-RGB reflectance in our training data, and  $J(\cdot)$  is a smoothness penalty (the thin-plate spline smoothness energy, made more robust by taking its square root). The smoothness

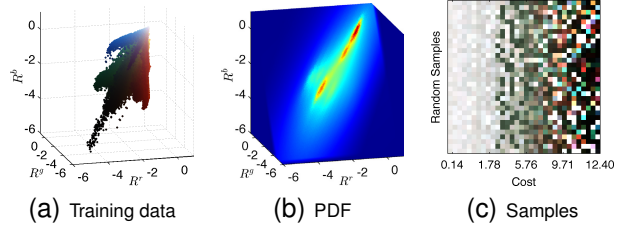


Fig. 9. A visualization of our “absolute” prior on color reflectance. In 9(a) we have the log-RGB reflectance pixels in our training set, and in 9(b) we have a visualization of the 3D spline PDF that we fit to that data. In 9(c) we have samples from our PDF, where the  $x$  axis is sorted by cost ( $y$  axis is random). Our model prefers less saturated, more earthy or subdued colors, and abhors brightly lit neon-like colors or very dark colors. The high-cost reflectances don’t even look like paint, but instead appear glowing and luminescent.

multiplier  $\lambda$  is tuned through cross-validation. As in our parsimony prior, we use whitened log-reflectance to address the correlation between channels, which is necessary as our smoothness term is isotropic. A visualization of our prior can be seen in Figure 9.

During inference, we maximize the likelihood of the color reflectance image  $R$  by minimizing its cost under our learned model:

$$g_a(R) = \sum_i F(W_R R_i) \quad (13)$$

where  $F(W_R R_i)$  is the value of  $F$  at the coordinates specified by the 3-vector  $W_R R_i$ , the whitened reflectance at pixel  $i$  ( $W_R$  is the same as in Section 4.2). To make this function differentiable, we compute  $F(\cdot)$  using trilinear interpolation.

## 5 PRIORS ON SHAPE

Our prior on shape consists of four components: 1) An assumption of smoothness (that shapes tend to bend rarely), which we will model by minimizing the variation of mean curvature. 2) An assumption of isotropy of the orientation of surface normals (that shapes are just as likely to face in one direction as they are another) which reduces to a well-motivated “flatness” prior on shapes. 3) An prior on the orientation of the surface normal near the boundary of masked objects, as shapes tend to face outward at the occluding contour. 4) An optional prior that the shape should resemble some noisy or incomplete external observation, such as an estimate of depth derived from stereo or a depth sensor. Formally, our shape prior  $f(Z)$  is a weighted combination of four costs:

$$f(Z) = \lambda_k f_k(Z) + \lambda_i f_i(Z) + \lambda_c f_c(Z) + \lambda_o f_o(Z, \hat{Z}) \quad (14)$$

where  $f_k(Z)$  is our smoothness prior,  $f_i(Z)$  is our isotropy prior,  $f_c$  is our bounding contour prior, and



$f_o(Z, \hat{Z})$  encourages  $Z$  to be similar to some observation  $\hat{Z}$ , all of which will be explained in detail in the following sections. The  $\lambda$  multipliers are learned through cross-validation on the training set.

Most of our shape priors are imposed on intermediate representations of shape, such as mean curvature or surface normals. This requires that we compute these intermediate representations from a depth map, calculate the cost and the gradient of cost with respect to those intermediate representations, and backpropagate the gradients back onto the shape. In the supplementary material we explain in detail how to efficiently compute these quantities and backpropagate through them.

### 5.1 Smoothness

There has been much work on modeling the statistics of natural shapes [41], [51], with one overarching theme being that regularizing some function of the second derivatives of a surface is effective. However, this past work has severe issues with invariance to out-of-plane rotation and scale. Working within differential geometry, we present a shape prior based on the variation of mean curvature, which allows us to place smoothness priors on  $Z$  that are invariant to rotation and scale.

To review: mean curvature is the divergence of the normal field. Planes and soap films have 0 mean curvature everywhere, spheres and cylinders have constant mean curvature everywhere, and the sphere has the smallest total mean curvature among all convex solids with a given surface area [52]. See Figure 10 for a visualization. Mean curvature is a measure of curvature in *world coordinates*, not image coordinates, so (ignoring occlusion) the marginal distribution of  $H(Z)$  is invariant to out-of-plane rotation of  $Z$  — a shape is just as likely viewed from one angle as from another. In comparison, the Laplacian of  $Z$  and the second partial derivatives of  $Z$  can be made large simply due to foreshortening, which means that priors placed on these quantities [51] would prefer certain shapes simply due to the angle from which those shapes are observed — clearly undesirable.

But priors on raw mean curvature are not scale-invariant. Were we to minimize  $|H(Z)|$ , then the most likely shape under our model would be a plane, while spheres would be unlikely. Were we to minimize  $|H(Z) - \alpha|$  for some constant  $\alpha$ , then the most likely shape under our model would be a sphere of a certain radius, but larger or smaller spheres, or a resized image of the same sphere, would be unlikely. Clearly, such scale sensitivity is an undesirable property for a general-purpose prior on natural shapes. Inspired by previous work on minimum variation surfaces [53], we place priors on the local variation of mean curvature. The most likely shapes under such priors are surfaces of constant mean curvature, which are

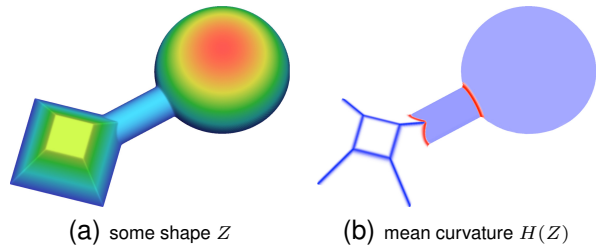


Fig. 10. A visualization of a shape and its mean curvature (blue = positive, red = negative, white = 0). Planes and soap films have 0 mean curvature, spheres and cylinders have constant mean curvature, and mean curvature varies where shapes bend.

well-studied in geometry and include soap bubbles and spheres of any size (including planes). Priors on the variation of mean curvature, like priors on raw mean curvature, are invariant to rotation and viewpoint, as well as concave/convex inversion.

Mean curvature is defined as the average of principle curvatures:  $H = \frac{1}{2}(\kappa_1 + \kappa_2)$ . It can be approximated on a surface using filter convolutions that approximate first and second partial derivatives, as show in [54].

$$H(Z) = \frac{(1 + Z_x^2) Z_{yy} - 2Z_x Z_y Z_{xy} + (1 + Z_y^2) Z_{xx}}{2(1 + Z_x^2 + Z_y^2)^{3/2}} \quad (15)$$

In the supplementary material we detail how to calculate and differentiate  $H(Z)$  efficiently. Our smoothness prior for shapes is a Gaussian scale mixture on the local variation of the mean curvature of  $Z$ :

$$f_k(Z) = \sum_i \sum_{j \in N(i)} c(H(Z)_i - H(Z)_j; \alpha_k, \sigma_k) \quad (16)$$

Notation is similar to Equation 4:  $N(i)$  is the  $5 \times 5$  neighborhood around pixel  $i$ ,  $H(Z)$  is the mean curvature of shape  $Z$ , and  $H(Z)_i - H(Z)_j$  is the difference between the mean curvature at pixel  $i$  and pixel  $j$ .  $c(\cdot; \alpha, \sigma)$  is defined in Equation 5, and is the negative log-likelihood (cost) of a discrete univariate Gaussian scale mixture (GSM), parametrized by  $\alpha$  and  $\sigma$ , the mixing coefficients and standard deviations, respectively, of the Gaussians in the mixture. The mean of the GSM is 0, as the most likely shapes under our model should be smooth. We set  $M = 40$  (the GSM has 40 discrete Gaussians), and  $\alpha_k$  and  $\sigma_k$  are learned from our training set using expectation-maximization. The log-likelihood of our learned model can be seen in Figure 11(a), and the likelihoods it assigns to different shapes can be visualized in Figure 11(b). The learned GSM is very heavy tailed, which encourages shapes to be mostly smooth, and occasionally very non-smooth — or equivalently, our prior encourages shapes to bend rarely.

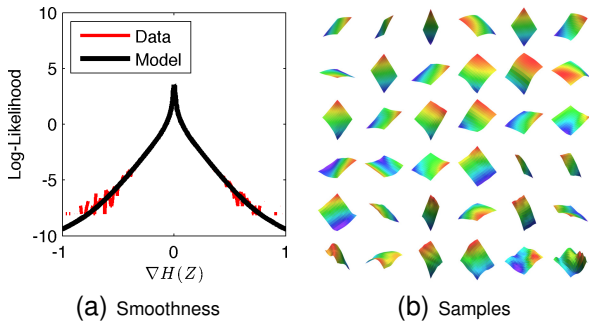


Fig. 11. To encourage shapes to be smooth, we model the variation in mean curvature of shapes using a Gaussian scale mixture, shown in 11(a). In 11(b) we show patches of shapes in our training data, sorted from least costly (upper left) to most costly (lower right). Likely shapes under our model look like soap-bubbles, and unlikely shapes look contorted.

## 5.2 Surface Isotropy

Our second prior on shapes is motivated by the observation that shapes tend to be oriented isotropically in space. That is, it is equally likely for a surface to face in any direction. This assumption is not valid in many settings, such as man-made environments (which tend to be composed of floors, walls, and ceilings) or outdoor scenes (which are dominated by the ground-plane). But this assumption is more true for generic objects floating in space, which tend to resemble spheres (whose surface orientations are truly isotropic) or sphere-like shapes — though there is often a bias on the part of photographers towards imaging the front-faces of objects. Despite its problems, this assumption is still effective and necessary.

Intuitively, one may assume that imposing this isotropy assumption requires no effort: if our prior assumes that all surface orientations are equally likely, doesn't that correspond to a constant cost for all surface orientations? However, this ignores the fact that once we have observed a surface in space, we have introduced a bias: observed surfaces are much more likely to face the observer ( $N^z \approx 1$ ) than to be perpendicular to the observer ( $N^z \approx 0$ ). We must therefore impose an isotropy prior to undo this bias.

We will derive our isotropy prior analytically. Assume surfaces are oriented uniformly, and that the surfaces are observed under orthogonal perspective with a view direction  $(0, 0, -1)$ . It follows that all  $N^z$  (the  $z$ -component of surface normals, relative to the viewer) are distributed uniformly between 0 and 1. Upon observation, these surfaces (which are assumed to have identical surface areas) have been foreshortened, such that the area of each surface in the image is  $N^z$ . Given the uniform distribution over  $N^z$  and this foreshortening effect, the probability distribution over  $N^z$  that we should expect at a given pixel in the image is proportional to  $N^z$ . Therefore, maximizing

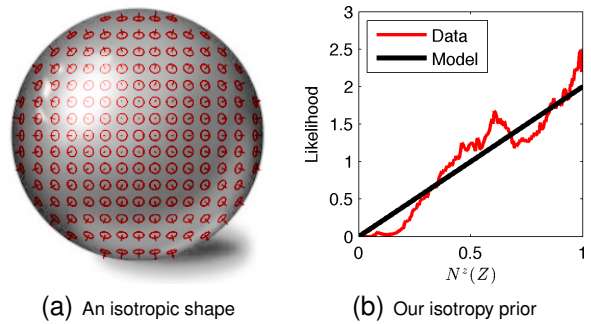


Fig. 12. We assume the surfaces of shapes to be isotropic — equally likely to face in any orientation, like in a sphere. However, observing an isotropic shape imposes a bias, as observed surfaces are more likely to face the observer than to be perpendicular to the observer (as shown by the red gauge figure “thumbtacks” placed on the sphere in 12(a)). We undo this bias by imposing a prior on  $N^z$ , shown in 12(b), which coarsely resembles our training data.

the likelihood of our uniform distribution over orientation in the world is equivalent to minimizing the following in the image:

$$f_i(Z) = - \sum_{x,y} \log(N_{x,y}^z(Z)) \quad (17)$$

Where  $N_{x,y}^z(Z)$  is the  $z$ -component of the surface normal of  $Z$  at position  $(x, y)$  (defined in the supplementary material).

Though this was derived as an isotropy prior, the shape which maximizes the likelihood of this prior is not isotropic, but is instead (because of the nature of MAP estimation) a fronto-parallel plane. This gives us some insight into the behavior of this prior — it serves to as a sort of “flatness” prior. This prior can therefore be thought of as combating the bas-relief ambiguity [25] (roughly, that absolute scale and orientation are ambiguous), by biasing our shape estimation towards the fronto-parallel members of the bas-relief family.

Our prior on  $N^z$  is shown in Figure 12(b) compared to the marginal distribution of  $N^z$  in our training data. Our model fits the data well, but not perfectly. We experimented with learning distributions on  $N^z$  empirically, but found that they worked poorly compared to our analytical prior. We attribute this to the aforementioned photographer’s bias towards fronto-parallel surfaces, and to data sparsity when  $N^z$  is close to 0.

It is worth noting that  $-\log(N^z)$  is proportional to the surface area of  $Z$ . Our prior on slant therefore has a helpful interpretation as a prior on minimal surface area: we wish to minimize the surface area of  $Z$ , where the degree of the penalty for increasing  $Z$ ’s surface area happens to be motivated by an isotropy assumption. This notion of placing priors on surface area has been explored previously [55], but not in the

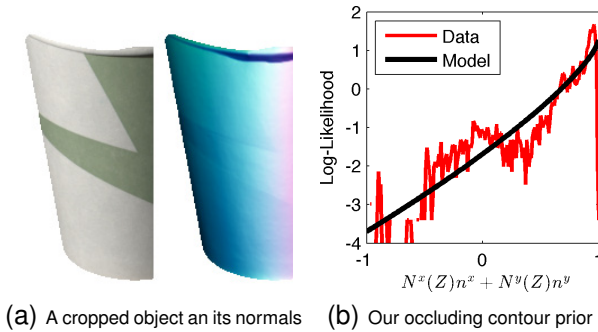


Fig. 13. In 13(a) we have an image and surface normals of a subset of a cup, in our dataset. The side of this cup are “limbs”, points where the surface normal faces outward and is perpendicular to the occluding contour, while the top of the cup are “edges”, sharp discontinuities where the surface is oriented arbitrarily. Our heavy-tailed prior over surface orientation at the occluding contour in 13(b) models the behavior of limbs, but is robust to the outliers caused by edges.

context of isotropy. And of course, this connection relates our model to the study of minimal surfaces in mathematics [52], though this connection is somewhat tenuous as the fronto-parallel planes favored by our model are very different from classical minimal surfaces such as planes and soap films.

### 5.3 The Occluding Contour

The occluding contour of a shape (the contour that surrounds the silhouette of a shape) is a powerful cue for shape interpretation [56] which often dominates shading cues [57], and algorithms have been presented for coarsely estimating shape given contour information [58]. At the occluding contour of an object, the surface is tangent to all rays from the vantage point. Under orthographic projection (which we assume), this means the  $z$ -component of the normal is 0, and the  $x$  and  $y$  components are determined by the contour in the image. In principle, this property is absolutely true, but in practice the occluding contour of a surface tends to be composed of limbs (points where the surface is tangent to rays from the vantage point, like the smooth side of a cylinder) and edges (an abrupt discontinuity of the surface, like the top of a cylinder or the edge of a piece of paper) [59]. See Figure 13(a) for an example of a shape which contains both phenomena. Of course, this taxonomy is somewhat false — all edges are limbs, but some are so small that they appear to be edges, and some are just small enough relative to the image resolution that the “limb” assumption begins to break down.

We present a “soft” version of a limb constraint, one which captures the “limb”-like behavior we expect to see but which can be violated by edges or small limbs. Because our dataset consists of masked objects,

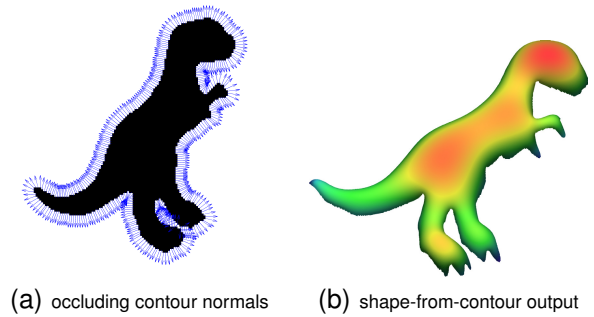


Fig. 14. A subset of our model that includes only our priors on shape is equivalent to a shape-from-contour model. Given only the normals of the silhouette of the object in 14(a), we can produce the coarse estimate of the shape of the object in 14(b).

identifying the occluding contour  $C$  is trivial (see Figure 14(a)). For each point  $i$  on  $C$ , we estimate  $n_i$ , the local normal to the occluding contour in the image plane. Using those we regularize the surface normals in  $Z$  along the boundary by minimizing the following loss:

$$f_c(Z) = \sum_{i \in C} (1 - (N_i^x(Z)n_i^x + N_i^y(Z)n_i^y))^{\gamma_c} \quad (18)$$

Where  $N(Z)$  is the surface normal of  $Z$ , as defined in the supplementary material. We set  $\gamma_c = 0.75$ , which fits the training data best, and which performs best in practice. The inner product of  $n_i$  and  $N_i$  (both of which are unit-norm) is 1 when both vectors are oriented in the same direction, in which case the loss is 0. If the normals do not agree, then some cost is incurred. This cost corresponds to a heavy-tailed distribution (shown in Figure 13(b)) which encourages the surface orientation to match the orientation of the occluding contour at limbs, allows surface normals to violate this assumption at edges.

This occluding-contour prior, when combined with our priors on smooth and isotropic shapes, allows us to easily define an ablation of our entire model that corresponds to a shape-from-contour algorithm: we simply optimize with respect to these shape priors, and ignore our priors on reflectance and illumination, thereby ignoring all but the silhouette of the input image. An example of the output of our shape-from-contour model can be seen in Figure 14(b), and this model is evaluated quantitatively against our complete SIRFS model in Section 8.

### 5.4 Noisy Shape Observation

One of the reasons that using shading cues to recover shape (as we are attempting here) is challenging, is that shading is a fundamentally poor cue for low-frequency (coarse) shape variation. Shading is directly indicative of only the shape of a point relative to its neighbors: fine-scale variations in shape produce

sharp, localized changes in an image, while coarse-scale shape variations produce very small, subtle changes across an entire image. Both algorithms [25] and humans [60] therefore make errors in estimating coarse depth when using only shading. Bas relief sculptures take advantage of this by conveying the impression of a rich, deep 3D scene, using only the shading produced by a physically shallow object.

To deal with this issue, we will construct our prior on shape to allow for an external observation of shape to be incorporated into inference. This observation may be produced by a stereo algorithm, or by some depth sensor such as a laser rangefinder or the Kinect. These depth sensors or stereo algorithms often produce depth maps which are noisy or incomplete, or most often blurry — lacking fine-scale shape detail. Because of the complementary strengths of stereo and shading, combining the two can often yield very accurate results [61], [5].

We will construct a loss function to encourage our recovered depth  $Z$  to resemble the raw sensor depth  $\hat{Z}$ :

$$f_o(Z, \hat{Z}) = \sum_i \left( ((Z * b(\sigma_Z))_i - \hat{Z}_i)^2 + \epsilon^2 \right)^{\frac{\gamma_o}{2}} \quad (19)$$

This is simply a hyperlaplacian distribution with an exponent of  $\gamma_o$  on the difference between  $(Z * b(\sigma_Z))$  and  $\hat{Z}$  at every pixel, with  $\epsilon$  added in to make the loss differentiable everywhere.  $b(\sigma_Z)$  is a 2D Gaussian filter with a standard deviation of  $\sigma_Z$ , and  $*$  is convolution, so  $(Z * b(\sigma_Z))_i$  is the value of a blurry version of our shape estimate  $Z$  at pixel location  $i$ . We tune  $\gamma_o$  on the training set, which sets it to  $\sim 1$ , and we set  $\epsilon = 1/100$ . The robust nature of this cost encourages  $Z$  to resemble  $\hat{Z}$ , while allowing it to occasionally differ drastically. In our experiments we use  $Z * b(30)$  as our  $\hat{Z}$ , which is a reasonably proxy for a stereo algorithm or low-resolution depth-sensor, and we set  $\sigma_Z = 30$  as that value (unsurprisingly) performs best during cross-validation.

## 6 PRIORS OVER ILLUMINATION

Because illumination is unknown, we must regularize it during inference. Our prior on illumination is extremely simple: we fit a multivariate Gaussian to the spherical-harmonic illuminations in our training set. During inference, the cost we impose is the (non-normalized) negative log-likelihood under that model:

$$h(L) = \lambda_L (L - \mu_L)^T \Sigma_L^{-1} (L - \mu_L) \quad (20)$$

where  $\mu_L$  and  $\Sigma_L$  are the parameters of the Gaussian we learned, and  $\lambda_L$  is the multiplier on this prior (learned on the training set).

We use a spherical-harmonic (SH) model of illumination, so  $L$  is a 9 (grayscale) or 27 (color, 9 dimensions per RGB channel) dimensional vector. In contrast to traditional SH illumination, we parametrize

log-shading rather than shading. This choice makes optimization easier as we don't have to deal with "clamping" illumination at 0, and it allows for easier regularization, as the space of log-shading SH illuminations is surprisingly well-modeled by a simple multivariate Gaussian while the space of traditional SH illumination coefficients is not. See Figure 15 for examples of SH illuminations in our different training sets, as well as samples from our model. We see that our samples look similar to the illuminations in the training set, suggesting that our model fits the data well.

## 7 OPTIMIZATION

To estimate shape, illumination, and reflectance, we must solve the optimization problem in Equation 2. This is a challenging optimization problem, and naive gradient-based optimization with respect to  $Z$  and  $L$  fails badly. We therefore present an effective multi-scale optimization technique, which is similar in spirit to multigrid methods [62], but extremely general and simple to implement. We will describe our technique in terms of optimizing  $f(X)$ , where  $f$  is some loss function and  $X$  is some signal.

Let us define  $\mathcal{G}$ , which constructs a Gaussian pyramid from a signal. Because Gaussian pyramid construction is a linear operation, we will treat  $\mathcal{G}$  as a matrix. Instead of minimizing  $f(X)$  directly, we minimize  $f'(Y)$ , where  $X = \mathcal{G}^T Y$ :

$$[\ell, \nabla_Y \ell] = f'(Y) : \quad (21)$$

$X \leftarrow \mathcal{G}^T Y$  // reconstruct signal

$[\ell, \nabla_X \ell] \leftarrow f(X)$  // compute loss & gradient

$\nabla_Y \ell \leftarrow \mathcal{G} \nabla_X \ell$  // backpropagate gradient

We initialize  $Y$  to a vector of all 0's, and then solve for  $\hat{X} = \mathcal{G}^T (\arg \min_Y f'(Y))$  using L-BFGS. Any arbitrary gradient-based optimization technique could be used, but L-BFGS worked best in our experience.

The choice of the filter used in constructing our Gaussian pyramid is crucial. We found that 4-tap binomial filters work well, and that the choice of the magnitude of the filter dramatically affects multi-scale optimization. If the magnitude is small, then the coefficients of the upper levels of the pyramid are so small that they are effectively ignored, and optimization fails (and in the limit, a filter magnitude of 0 reduces our model to single-scale optimization). Conversely, if the magnitude is large, then the coarse scales of the pyramid are optimized and the fine scales are ignored. The filter that we found worked best is:  $\frac{1}{\sqrt{8}}[1, 3, 3, 1]$ , which has twice the magnitude of the filter that would normally be used for Gaussian pyramids. This increased magnitude biases optimization towards adjusting coarse scales before fine scales, without preventing optimization from eventually optimizing fine scales. This filter magnitude does not

appear to be universally optimal — different tasks appear to have different optimal filter magnitudes. Note that this technique is substantially different from standard coarse-to-fine optimization, in that *all* scales are optimized simultaneously. As a result, we find much lower minima than standard coarse-to-fine techniques, which tend to keep coarse scales fixed when optimizing over fine scales. Optimization is also much faster than comparable coarse-to-fine techniques.

To optimizing Equation 2 we initialize  $Z$  and  $L$  to  $\vec{0}$  ( $L = \vec{0}$  is equivalent to an entirely ambient, white illumination) and optimize with respect to a vector that is a concatenation of  $\mathcal{G}^T Z$  and a whitened version of  $L$ . We optimize in the space of whitened illuminations because the Gaussians we learn for illumination mostly describe a low-rank subspace of SH coefficients, and so optimization in the space of unwhitened illumination is ill-conditioned. We precompute a whitening transformation for  $\Sigma_L$  and  $\mu_L$ , and during each evaluation of the loss in gradient descent we unwhiten our whitened illumination, compute the loss and gradient, and backpropagate the gradient onto the whitened illumination. After optimizing Equation 2 we have a recovered depth map  $\hat{Z}$  and illumination  $\hat{L}$ , with which we calculate a reflectance image  $\hat{R} = I - S(\hat{Z}, \hat{L})$ . When illumination is known,  $L$  is fixed. Optimizing to near-convergence (which usually takes a few hundred iterations) for a 1-2 megapixel grayscale image takes 1-5 minutes on a 2011 Macbook Pro, using a straightforward Matlab/C implementation. Optimization takes roughly twice as long if the image is color.

We use this same multiscale optimization scheme with L-BFGS to solve the optimization problems in Equations 10 and 12, though we use different filter magnitudes for the pyramids. Naive single-scale optimization for these problems works poorly.

## 8 EXPERIMENTS

Quantitatively evaluating the accuracy of our model is challenging, as there are no pre-existing datasets with ground-truth shape, surface normals, shading, reflectance, and illumination. Thankfully, the MIT Intrinsic Images dataset [4] provides ground-truth shading and reflectance for 20 objects (one object per image), and includes many additional images of each object under different illumination conditions. Given this, we have created the MIT-Berkeley Intrinsic Images dataset, an augmented version of the MIT Intrinsic Images dataset in which we have used photometric stereo on the additional images of each object to estimate the shape of each object and the spherical harmonic illumination for each image. An example object in our dataset can be seen in Figure 2, and the supplementary material contains additional images and details of our photometric stereo algorithm. In

all of our experiments, we use the following test-set: cup2, deer, frog2, paper2, pear, potato, raccoon, sun, teabag1, turtle. The other 10 objects are used for training.

An additional difficulty in evaluation is the choice of error metrics. Constructing error metrics for specific intrinsic scene properties such as a depth map or a reflectance image is challenging, as naive choices such as mean-squared-error often correspond very poorly with the perceptual salience of an error. Additionally, constructing a single error metric that describes all errors in each intrinsic scene property is difficult. We therefore present six different error metrics that have been designed to capture different kinds of important errors for each intrinsic scene property:  $Z$ -MAE is the shift-invariant absolute error between the estimated shape and the ground-truth shape.  $N$ -MAE is the mean error between our estimated normal field and ground-truth normal field, in radians.  $S$ -MSE and  $R$ -MSE are the scale-invariant mean-squared-error of our recovered shading and reflectance, respectively.  $RS$ -MSE is the error metric introduced in conjunction with the MIT intrinsic images dataset [4], which measures a locally scale-invariant error for both reflectance and shading<sup>2</sup>.  $L$ -MSE is the scale-invariant MSE of a rendering of our recovered illumination on a sphere, relative to a rendering of the ground-truth illumination. To summarize these individual error metrics, we report an “average” error metric, which is the geometric mean of the previous six error metrics. For each error metric and the average metric, we report the geometric mean of error across the test-set images. The use of the geometric mean prevents the average error from being dominated by individual error metrics with large dynamic ranges, or by particularly challenging images. See the supplementary material for a thorough explanation of our choice of error metrics.

Though the MIT dataset has a great deal of variety in terms of the kinds of objects used, the illumination in the dataset is very “laboratory”-like — lights are white, and are placed at only a few locations relative to the object. See Figure 15(a) for examples of these “laboratory” illuminations. In contrast, natural illuminations exhibit much more color and variety: the sun is yellow, outdoor shadows are often tinted blue, man-made illuminants have different colors, and indirect illumination from colored objects may cause very colorful illuminations. To acquire some illumination models that are more representative of the variety seen in the natural world, we took all of the environment maps from the sIBL Archive<sup>3</sup>, expanded that set of environment maps by shifting and mirroring them and varying their contrast and

2. The authors of [4] refer to this error metric to as “LMSE”, but we will call it  $RS$ -MSE to minimize confusion with  $L$ -MSE

3. <http://www.hdrlabs.com/sibl/archive.html>



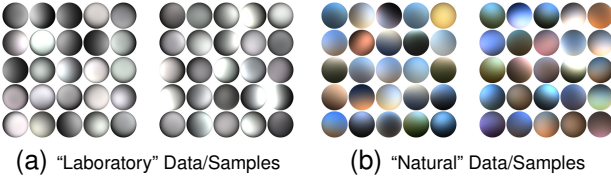


Fig. 15. We use two datasets: the “laboratory”-style illuminations of the MIT intrinsic images dataset [4] which are harsh, mostly-white, and well-approximated by point sources, and a dataset of “natural” illuminations, which are softer and much more colorful. Shown here are some illuminations from the training sets of our two datasets, and samples from a multivariate Gaussian fit to each training set (our illumination prior from Section 6), rendered on Lambertian spheres.

saturation (saturation is only ever decreased, never increased) and produced spherical harmonic illuminations from the resulting environment maps. After removing similar illuminations, the illuminations were split into training and test sets. See Figure 15(b) for examples of these “natural” illuminations. Each object in the MIT dataset was randomly assigned an illumination (such that training illuminations were assigned to training objects, etc), and each object was re-rendered under its new illumination, using that object’s ground-truth shape and reflectance. We will refer to this new pseudo-synthetic dataset of naturally illuminated objects as our “natural” illumination dataset, and we will refer to the original MIT images as the “laboratory” illumination dataset. From our experience applying our model to real-world images, these “natural” illuminations appear to be much more representative of the sort of illumination we see in uncontrolled environments, though the dataset is heavily biased towards more colorful illuminations. We attribute this to a photographer’s bias towards “interesting” environment maps in the SIBL Archive.

Given our dataset, we will evaluate our model on the task of recovering all intrinsic scene properties from a single image of a masked object, under three different conditions: I: the input is a grayscale image and the illumination is “laboratory”-like, II: the input is a color image and the illumination is “laboratory”-like, and III: the input is a color image and the illumination is “natural”. For all tasks, we use the same training/test split, and for each task we tune a different set of hyperparameters on the training set ( $\lambda_s, \lambda_e, \lambda_a, \sigma_R, \lambda_k, \lambda_i, \lambda_c, \lambda_o$ , and  $\lambda_L$ ), and fit a different prior on illumination (as in Section 6). Hyperparameters are tuned using coordinate descent to minimize our “average” error metric for the training set. For each task, we compare SIRFS against several intrinsic images algorithms (meant to decompose an image into shading and reflectance components), upon which we’ve run a shape-from-shading algorithm on the shading image. For the

sake of a generous comparison, the SFS algorithm uses our shape priors, which boosts each baseline’s performance (detailed in the supplementary material). We also compare against a “naive” algorithm, which is a baseline in which  $Z = \vec{0}$  and  $L = \vec{0}$ . Because the intrinsic image baselines do not estimate illumination, we use  $L = \vec{0}$  as their prediction. We were forced to use different baseline techniques for different tasks, as some baselines do not have code available for running on new imagery, and some code that was designed for color images crashes when run on grayscale images.

We also compare against several ablations of our model in which components have been removed:

### I. Grayscale Images, Laboratory Illumination

Algorithm	Z-MAE	N-MAE	S-MSE	R-MSE	RS-MSE	L-MSE	Avg.
(1) Naive Baseline	25.56	0.7223	0.0571	0.0426	0.0353	0.0484	0.2061
(2) Retinex [4], [10] + SFS	67.15	0.8342	0.0311	0.0265	0.0289	0.0484	0.2002
(3) Tappen <i>et al.</i> [14] + SFS	41.96	0.7413	0.0354	0.0252	0.0285	0.0484	0.1835
(4) Shen <i>et al.</i> [13] + SFS	45.57	0.8293	0.0493	0.0427	0.0436	0.0484	0.2348
(A) SIRFS	31.00	0.5343	<b>0.0156</b>	<b>0.0177</b>	<b>0.0209</b>	<b>0.0103</b>	<b>0.0998</b>
(B) SIRFS, no R-smoothness	27.25	0.5361	0.0267	0.0255	0.0290	0.0152	0.1279
(C) SIRFS, no R-parsimony	23.53	0.4862	0.0224	0.0261	0.0228	0.0167	0.1170
(D) SIRFS, no R-absolute	24.02	0.5023	0.0190	0.0201	0.0222	0.0122	0.1037
(E) SIRFS, no Z-smoothness	29.05	0.5783	0.0241	0.0227	0.0337	0.0125	0.1254
(F) SIRFS, no Z-isotropy	98.07	0.7560	0.0200	0.0198	0.0268	0.0104	0.1419
(G) SIRFS, no Z-contour	34.29	0.7676	0.0208	0.0207	0.0232	0.0231	0.1351
(H) SIRFS, no L-gaussian	26.75	0.5929	0.0270	0.0212	0.0327	0.1940	0.1964
(I) SIRFS, no L-multiscale	25.58	0.7233	0.0571	0.0426	0.0353	0.0414	0.2009
(J) SIRFS, no L-whitening	33.93	0.5837	0.0207	0.0208	0.0256	0.0119	0.1171
(K) Shape-from-Contour	<b>18.96</b>	<b>0.4192</b>	0.0571	0.0426	0.0353	0.0484	0.1791
(S) shape observation	4.83	0.1952	-	-	-	-	-
(A+S) SIRFS + shape observation	3.72	0.2414	0.0128	0.0176	0.0210	0.0096	0.0586
(A+L) SIRFS + known illumination	27.32	0.4944	0.0175	0.0179	0.0225	-	-

### II. Color Images, Laboratory Illumination

Algorithm	Z-MAE	N-MAE	S-MSE	R-MSE	RS-MSE	L-MSE	Avg.
(1) Naive Baseline	25.56	0.7223	0.0577	0.0455	0.0354	0.0489	0.2092
(2) Retinex [4], [10] + SFS	85.34	0.8056	0.0204	0.0186	0.0163	0.0489	0.1658
(3) Tappen <i>et al.</i> [14] + SFS	41.96	0.7413	0.0361	0.0379	0.0347	0.0489	0.2040
(4) Shen <i>et al.</i> [13] + SFS	55.95	0.8529	0.0528	0.0458	0.0398	0.0489	0.2466
(5) Gehler <i>et al.</i> [15] + SFS	53.36	0.6844	0.0106	0.0101	0.0131	0.0489	0.1166
(A) SIRFS	19.24	<b>0.3914</b>	<b>0.0064</b>	<b>0.0098</b>	<b>0.0125</b>	0.0096	<b>0.0620</b>
(B) SIRFS, no R-smoothness	19.23	0.4046	0.0125	0.0163	0.0214	0.0092	0.0824
(C) SIRFS, no R-parsimony	19.45	0.4312	0.0096	0.0149	0.0140	0.0091	0.0731
(D) SIRFS, no R-absolute	22.98	0.4288	0.0085	0.0113	0.0135	0.0095	0.0704
(E) SIRFS, no Z-smoothness	19.28	0.4367	0.0114	0.0116	0.0219	<b>0.0088</b>	0.0773
(F) SIRFS, no Z-isotropy	84.08	0.7013	0.0117	0.0128	0.0160	0.0103	0.1063
(G) SIRFS, no Z-contour	32.59	0.7351	0.0103	0.0146	0.0173	0.0444	0.1186
(H) SIRFS, no L-gaussian	20.81	0.4631	0.0199	0.0140	0.0183	0.1272	0.1358
(I) SIRFS, no L-multiscale	25.62	0.7237	0.0574	0.0453	0.0353	0.0401	0.2022
(J) SIRFS, no L-whitening	24.96	0.4766	0.0106	0.0156	0.0188	0.0138	0.0894
(K) Shape-from-Contour	<b>18.96</b>	0.4192	0.0577	0.0455	0.0354	0.0489	0.1818
(S) shape observation	4.83	0.1952	-	-	-	-	-
(A+S) SIRFS + shape observation	3.40	0.2126	0.0070	0.0111	0.0153	0.0063	0.0420
(A+L) SIRFS + known illumination	18.58	0.3761	0.0076	0.0120	0.0146	-	-

### III. Color Images, Natural Illumination

Algorithm	Z-MAE	N-MAE	S-MSE	R-MSE	RS-MSE	L-MSE	Avg.
(1) Naive Baseline	25.56	0.7223	0.0283	0.0266	0.0125	0.0371	0.1364
(2) Retinex [4], [10] + SFS	26.76	0.5851	0.0174	0.0174	0.0083	0.0371	0.1066
(3) Tappen <i>et al.</i> [14] + SFS	53.87	0.7255	0.0255	0.0280	0.0268	0.0371	0.1740
(4) Gehler <i>et al.</i> [15] + SFS	37.66	0.6398	0.0162	0.0150	0.0105	0.0371	0.1149
(A) SIRFS	28.21	0.4057	0.0055	0.0046	<b>0.0036</b>	0.0103	0.0469
(B) SIRFS, no R-smoothness	28.41	0.4192	0.0061	0.0057	0.0062	0.0104	0.0546
(C) SIRFS, no R-parsimony	28.90	0.4184	0.0073	0.0064	0.0041	0.0107	0.0540
(D) SIRFS, no R-absolute	20.63	<b>0.3538</b>	0.0068	0.0058	0.0039	0.0091	<b>0.0466</b>
(E) SIRFS, no Z-smoothness	24.68	0.4441	0.0087	0.0062	0.0095	0.0099	0.0618
(F) SIRFS, no Z-isotropy	50.49	0.4015	<b>0.0046</b>	<b>0.0039</b>	0.0037	<b>0.0086</b>	0.0475
(G) SIRFS, no Z-contour	41.27	0.7036	0.0094	0.0083	0.0062	0.0256	0.0843
(H) SIRFS, no L-gaussian	20.22	0.3937	0.0100	0.0088	0.0075	0.0483	0.0796
(I) SIRFS, no L-multiscale	25.64	0.7205	0.0279	0.0279	0.0124	0.0291	0.1316
(J) SIRFS, no L-whitening	51.74	0.9430	0.0140	0.0106	0.0066	0.0777	0.1246
(K) Shape-from-Contour	<b>19.55</b>	0.4253	0.0283	0.0266	0.0125	0.0371	0.1194
(S) shape observation	4.83	0.1952	-	-	-	-	-
(A+S) SIRFS + shape observation	3.17	0.1471	0.0034	0.0032	0.0030	0.0049	0.0206
(A+L) SIRFS + known illumination	10.28	0.1957	0.0018	0.0014	0.0022	-	-

TABLE 1

We evaluate SIRFS on three different variants of our dataset, and we compare SIRFS to several baseline techniques, several ablations, and two extensions in which additional information is provided.



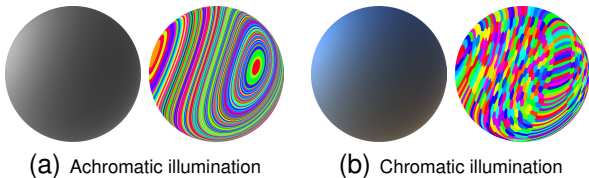


Fig. 16. Chromatic illumination dramatically helps shape estimation. Achromatic isophotes (K-means clusters of log-RGB values) are very elongated, while chromatic isophotes are usually more tightly localized. Therefore, under achromatic lighting a very wide range of surface orientations appear similar, but under chromatic lighting only similar orientations appear similar.

models B-H omit priors by simply setting their  $\lambda$  hyperparameters to 0, and models I and J omit our multiscale optimization over  $Z$  and our whitened optimization over  $L$  respectively. Model K is a shape-from-contour technique, in which only our shape-priors are non-zero and  $L = \vec{0}$ , so the only effective input to the model is the silhouette of the object (for this baseline, the hyperparameters have been completely re-tuned on the training set). We also compare against two extensions of SIRFS: model A+L, in which the ground-truth illumination is known (and fixed during optimization), and model A+S, in which we provide a blurry version of the ground-truth shape (convolved with a Gaussian kernel with  $\sigma = 30$ ) as input and use our prior from Section 5.4 to incorporate that information (in all other models, that prior is ignored by setting its  $\lambda$  multiplier to 0). Model S shows the performance of just the blurry ground-truth shape provided as input to model A+S, for reference. The performance of SIRFS relative to some of these baselines and extensions can be seen in Table 1, in Figure 2 and in the supplementary material.

From Table 1, we see that SIRFS outperforms all baseline techniques. For grayscale images, the improvement is substantial: our error is roughly half that of the best technique. For color images under “laboratory” illumination, our recovered shading and reflectance images are only slightly better than those of the best-performing intrinsic image technique [15], but our recovered shape and surface normals are significantly better, demonstrating the value of a unified technique over a piecewise system that first does intrinsic images, and then does shape from shading. For color images under “natural” illumination, SIRFS outperforms all baseline models by a very large margin, it is the only model that can reason well about color illumination and (implicitly) color shading. From our ablation study, we see that each prior contributes positively to performance, though the improvement we get from each prior is greater in the grayscale case than in the color/natural case. This makes sense, as color images under natural illumination contain much

more information than in grayscale images, and so the “likelihood” dominates our priors during inference. Our ablation study also shows that our multiscale optimization is absolutely critical to performance. Surprisingly, our shape-from-contour baseline performs very well in terms of our shape/normal error metrics. This is probably just a reflection of the fact that all models are bad at absolute shape reconstruction, due to the inherent ambiguity in shape-from-shading, and so the overly-smooth shape predicted by the shape-from-contour model, by virtue of being smooth and featureless, has a low error relative to the more elaborate depth maps produced by other models. Of course, the shape-from-contour model performs poorly on all other error metrics, as we would expect. This analysis of the inherent difficulty of shape estimation is further demonstrated by model A+S, which includes external shape information, and which therefore performs much better in terms of our shape/normal error metrics, but surprisingly performs similarly to model A (basic SIRFS) in terms of all other error metrics. From the performance of model A+L we see that knowing the illumination of the scene a-priori does not help much when the illumination is laboratory-like, but helps a great deal when the illumination is “natural” — which makes sense, as more-varied illumination simply makes the reconstruction task more difficult. One surprising conclusion we can draw is that, though the intrinsic image baselines perform worse in the presence of “natural” illumination, SIRFS actually performs *better* in natural illumination, as it can exploit color illumination to better disambiguate between shading and reflectance (Figure 4), and produce higher-quality shape reconstructions (Figure 16). This finding is consistent with recent work regarding shape-from-shading under natural illumination [63]. However, we should mention that some of the improved performance in the natural illumination task may be due to the fact that the images are pseudo-synthetic (their shading images were produced using our spherical-harmonic rendering) and so they are Lambertian and contain no cast shadows.

In Figure 17, we demonstrate a simple graphics application using the output of our model, for a color image under laboratory illumination. Given just the output of our model from a single image, we can synthesize novel images in which the shape, reflectance, illumination, or orientation of the object has been changed. The output is not perfect — the absolute shape is often very incorrect, as we saw in Table 1, which is due to the inherent ambiguity and difficulty in estimating shape from shading. But such shape errors are usually only visible when rotating the object, and this inherent ambiguity in shape perception often works in our favor when only manipulating reflectance, illumination, or fine-scale shape — low-frequency errors in shape-estimation made by our model are often not noticed by human observers,

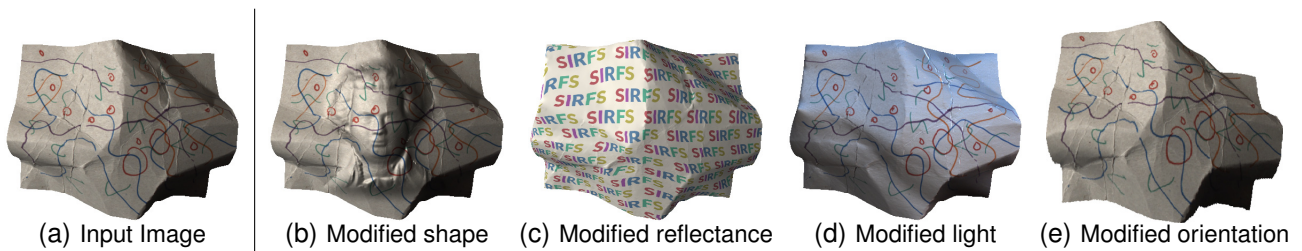


Fig. 17. Our system has obvious graphics applications. Given only a single image, we can estimate an object’s shape, reflectance, or illumination, modify any of those three scene properties (or simply rotate the object), and then re-render the object.

because both the model and the human are bad at using shading to estimate coarse shape.

### 8.1 Real-World Images

Though our model quantitatively performs very well on the MIT-Berkeley Intrinsic Images dataset, this dataset is not very representative of the variety of natural objects in the world — materials are very Lambertian, many reflectances are very synthetic-looking, and illumination is not very varied. We therefore present an additional experiment in which we ran our model on arbitrary masked images of natural objects. We acquired many images (some with an iPhone camera, some with a DSLR, some downloaded from the internet), manually cropped the object in the photo, and used them as input to our model. In Figure 18 we visualize the output of our model: the recovered shape, normals, reflectance, shading, and illumination, a synthesized view of the object from a different angle, and a synthesized rendering of the object using a different (randomly generated) illumination. We did two experiments: one in which we used a grayscale version of the input image and our laboratory illumination model, and one in which we used the color input image and our natural illumination model. We use the same code and hyperparameters for all images in the two constituent tasks, where our hyperparameters are identical to those used in the previous experiments with the MIT-Berkeley Intrinsic Images dataset.

We see that our model is often able to produce extremely compelling shading and reflectance images, and qualitatively correct illumination. Our recovered shape and surface normals are often somewhat wrong, as evidenced by the new synthesized views of each object, but our “reli” objects are often very compelling. The most common mistakes made in shading/reflectance estimation are usually due to our model assuming that the dominant color of the object is due to illumination, not reflectance (such as in the two pictures of faces) which we believe is due to biases in our training data towards white reflectances and colorful illumination.

## 9 CONCLUSION

We have presented SIRFS, a model which takes as input a single (masked) image of an object, and produces as output a reasonable estimate of the shape, surface normals, reflectance, shading, and illumination which produced that image. At the core of SIRFS is a series of priors on shape, reflectance, and illumination: surfaces tend to be isotropic and bend infrequently, reflectance images tend to be piecewise smooth and low-entropy, and illumination tends to be natural. Given these priors and our multiscale optimization technique, we can infer the most-likely explanation of a single image subject to our priors and the constraint that the image be explained. Our unified approach to this problem outperforms all previous solutions to its constituent problems of shape-from-shading and intrinsic image recovery on our challenging dataset, and produces reasonable results on arbitrary masked images of real-world objects in uncontrolled environments. This suggests that the shape-from-shading and intrinsic images problem formulations may be fundamentally limited, and attention should be refocused towards developing models that jointly reason about shape and illumination in addition to shading and reflectance.

But of course, our model has some limitations. Because shading is an inherently poor cue for low-frequency shape estimation [25], [26] our model often makes mistakes in coarse shape estimation. To address this, we have presented a method for incorporating some external observation of shape, such as one from a stereo algorithm or a depth sensor, and we have demonstrated that by incorporating some low-frequency external shape observation (such as what a stereo algorithm or a depth sensor would provide) we can produce high-quality shape estimates. We assume that materials are Lambertian, which is often a reasonable approximation but can cause problems for objects with specularities. Thankfully, because of the modular nature of our algorithm, our simple Lambertian rendering engine can easily be replaced by a more sophisticated model. We assume that images consist of single, masked objects, while real-world natural scenes contain severe occlusion and



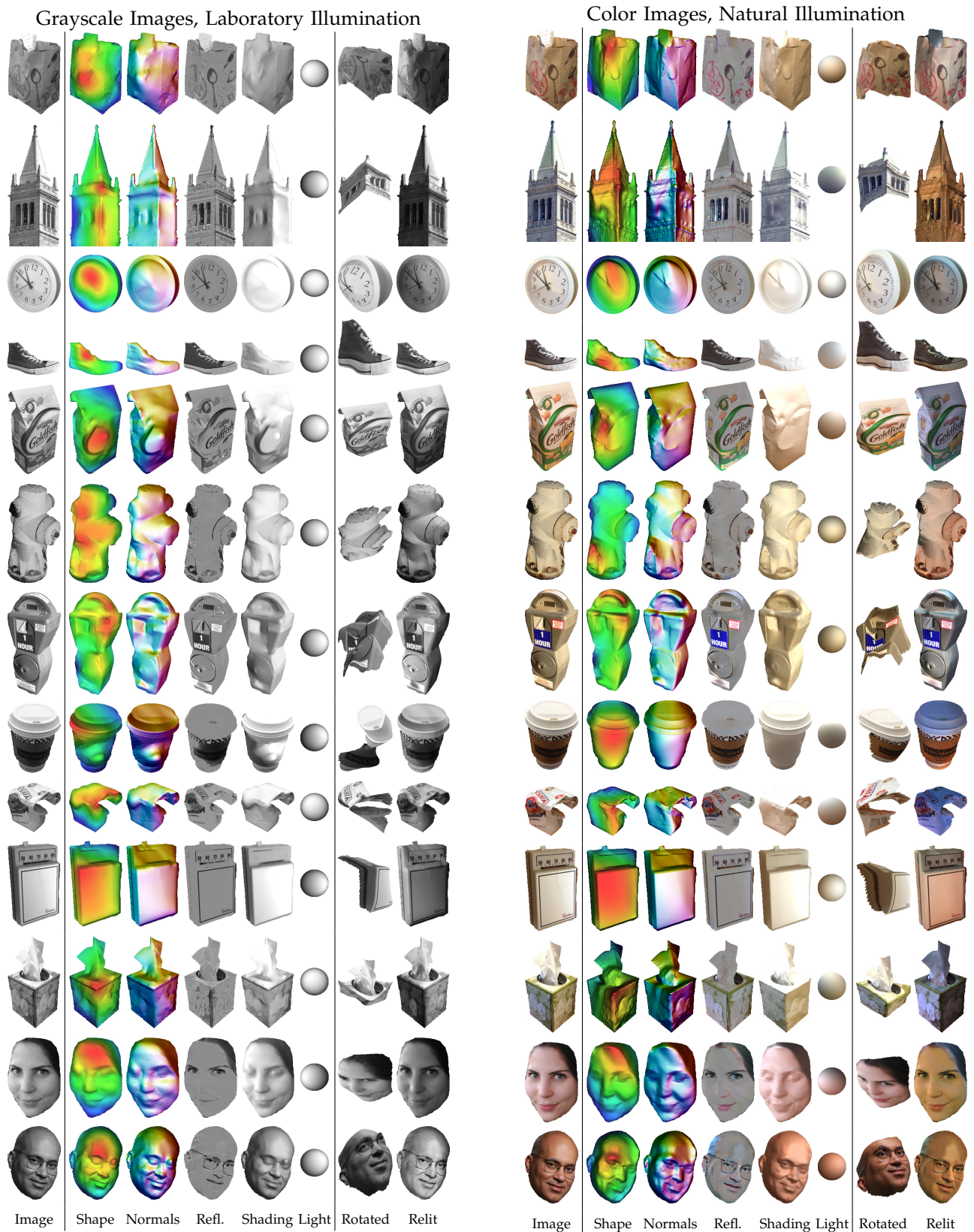


Fig. 18. Our model produces reasonable results on real, manually cropped images of objects. Here are images of arbitrary objects in uncontrolled illumination environments which were downloaded or taken on consumer cameras. For each image, we have the output of our model, and two renderings of our recovered model: one in which we rotate the object, and one in which we relight the object. We run our algorithm on a grayscale version of the image (left), and on the original color image (right). For the color images, we use our “natural” illumination model. The code and parameters used for these images are the same as in all other experiments.

support relationships. We also assume illumination is global, and we ignore illumination issues such as cast shadows, mutual illumination, or other sources of spatially-varying illumination [55], [64]. To address these two issues of occlusion and spatially-varying illumination in natural scenes, we have investigated into the interplay between SIRFS and segmentation techniques, by generalizing SIRFS to a mixture model of shapes and lights which are embedded in a soft segmentation of a scene [65]. Another limitation of our technique is that our priors on shape and reflectance are independent of the category of object present in the scene. We see this as a strength of our model, as it means that our priors are general enough to generalize across object categories, but presumably an extension of our model which uses object recognition techniques to produce class-specific priors should perform better.

## ACKNOWLEDGMENTS

J.B. was supported by NSF GRFP and ONR MURI N00014-10-10933. Thanks to Trevor Darrell, Bruno Olshausen, David Forsyth, Bill Freeman, Ted Adelson, and Estee Schwartz.

## REFERENCES

- [1] E. Adelson and A. Pentland, "The perception of shading and reflectance," *Perception as Bayesian inference*, 1996.
- [2] D. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *JOSA A*, 1987.
- [3] D. Ruderman and W. Bialek, "Statistics of natural images: Scaling in the woods," *Physical Review Letters*, 1994.
- [4] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman, "Ground-truth dataset and baseline evaluations for intrinsic image algorithms," *ICCV*, 2009.
- [5] J. T. Barron and J. Malik, "High-frequency shape and albedo from shading using natural image statistics," *CVPR*, 2011.
- [6] —, "Shape, albedo, and illumination from a single image of an unknown object," *CVPR*, 2012.
- [7] —, "Color constancy, intrinsic images, and shape estimation," *ECCV*, 2012.
- [8] A. Gilchrist, *Seeing in Black and White*. Oxford University Press, 2006.
- [9] E. H. Land and J. J. McCann, "Lightness and retinex theory," *JOSA*, 1971.
- [10] B. K. P. Horn, "Determining lightness from an image," *Computer Graphics and Image Processing*, 1974.
- [11] H. Barrow and J. Tenenbaum, "Recovering intrinsic scene characteristics from images," *Computer Vision Systems*, 1978.
- [12] M. Bell and W. T. Freeman, "Learning local evidence for shading and reflectance," *ICCV*, 2001.
- [13] J. Shen, X. Yang, Y. Jia, and X. Li, "Intrinsic images using optimization," *CVPR*, 2011.
- [14] M. F. Tappen, W. T. Freeman, and E. H. Adelson, "Recovering intrinsic images from a single image," *TPAMI*, 2005.
- [15] P. Gehler, C. Rother, M. Kiefel, L. Zhang, and B. Schoelkopf, "Recovering intrinsic images with a global sparsity prior on reflectance," *NIPS*, 2011.
- [16] D. A. Forsyth, "A novel algorithm for color constancy," *IJCV*, 1990.
- [17] L. T. Maloney and B. A. Wandell, "Color constancy: a method for recovering surface spectral reflectance," *JOSA A*, 1986.
- [18] G. Klinker, S. Shafer, and T. Kanade, "A physical approach to color image understanding," *IJCV*, 1990.
- [19] G. Finlayson, S. Hordley, and P. Hubel, "Color by correlation: a simple, unifying framework for color constancy," *TPAMI*, 2001.
- [20] D. H. Brainard and W. T. Freeman, "Bayesian color constancy," *JOSA A*, 1997.
- [21] B. K. P. Horn, "Obtaining shape from shading information," in *The Psychology of Computer Vision*, 1975.
- [22] T. Rindfleisch, "Photometric method for lunar topography," *Photogrammetric Engineering*, 1966.
- [23] M. J. Brooks and B. K. P. Horn, *Shape from shading*. MIT Press, 1989.
- [24] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, "Shape-from-shading: a survey," *TPAMI*, 2002.
- [25] P. Belhumeur, D. Kriegman, and A. Yuille, "The Bas-Relief Ambiguity," *IJCV*, 1999.
- [26] J. Koenderink, A. van Doorn, C. Christou, and J. Lappin, "Shape constancy in pictorial relief," *Perception*, 1996.
- [27] K. Ikeuchi and B. Horn, "Numerical shape from shading and occluding boundaries," *Artificial Intelligence*, 1981.
- [28] H. Boyaci, K. Doerschner, J. L. Snyder, and L. T. Maloney, "Surface color perception in three-dimensional scenes," *Visual Neuroscience*, 2006.
- [29] R. Woodham, "Photometric method for determining surface orientation from multiple images," *Optical Engineering*, 1980.
- [30] R. Basri, D. Jacobs, and I. Kemelmacher, "Photometric stereo with general, unknown lighting," *IJCV*, 2007.
- [31] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003.
- [32] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment - a modern synthesis," *ICCV*, 1999.
- [33] Y. Yu, P. Debevec, J. Malik, and T. Hawkins, "Inverse global illumination: recovering reflectance models of real scenes from photographs," *SIGGRAPH*, 1999.
- [34] D. Hoiem, A. A. Efros, and M. Hebert, "Recovering surface layout from an image," *IJCV*, 2007.
- [35] A. Saxena, M. Sun, and A. Ng, "Make3d: learning 3d scene structure from a single still image," *TPAMI*, 2008.
- [36] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," *SIGGRAPH*, 1999.
- [37] J. Huang and D. Mumford, "Statistics of natural images and models," *CVPR*, 1999.
- [38] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," *IEEE Trans. Image Process*, 2003.
- [39] S. Roth and M. J. Black, "Fields of experts: A framework for learning image priors," *CVPR*, 2005.
- [40] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. Freeman, "Removing camera shake from a single photograph," *SIGGRAPH*, 2006.
- [41] J. Huang, A. B. Lee, and D. Mumford, "Statistics of range images," *CVPR*, 2000.
- [42] F. Romeiro and T. Zickler, "Blind reflectometry," *ECCV*, 2010.
- [43] R. O. Dror, A. S. Willsky, and E. H. Adelson, "Statistical characterization of real-world illumination," *JOV*, 2004.
- [44] B. K. P. Horn, "Shape from shading: A method for obtaining the shape of a smooth opaque object from one view," MIT, Tech. Rep., 1970.
- [45] R. Ramamoorthi and P. Hanrahan, "An Efficient Representation for Irradiance Environment Maps," *CGIT*, 2001.
- [46] N. Alldrin, S. Mallick, and D. Kriegman, "Resolving the generalized bas-relief ambiguity by entropy minimization," *CVPR*, 2007.
- [47] G. D. Finlayson, M. S. Drew, and C. Lu, "Entropy minimization for shadow removal," *IJCV*, 2009.
- [48] I. Omer and M. Werman, "Color lines: Image specific color representation," *CVPR*, 2004.
- [49] J. C. Principe and D. Xu, "Learning from examples with quadratic mutual information," *Workshop on Neural Networks for Signal Processing*, 1998.
- [50] J. Chen, S. Paris, and F. Durand, "Real-time edge-aware image processing with the bilateral grid," *SIGGRAPH*, 2007.
- [51] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon, "Global stereo reconstruction under second-order smoothness priors," *TPAMI*, 2009.
- [52] D. Hilbert and C. S. Vossen, *Geometry and the Imagination*. Chelsea Publishing Company, 1956.
- [53] H. P. Moreton and C. H. Séquin, "Functional optimization for fair surface design," in *SIGGRAPH*, 1992.
- [54] P. Besl and R. Jain, "Segmentation through variable-order surface fitting," *TPAMI*, 1988.

- [55] D. A. Forsyth, "Variable-source shading analysis," *IJCV*, 2011.
- [56] J. Koenderink, "What does the occluding contour tell us about solid shape?" *Perception*, 1984.
- [57] P. Mamassian, D. Kersten, and D. C. Knill, "Categorical local-shape perception," *Perception*, 1996.
- [58] M. Brady and A. Yuille, "An extremum principle for shape from contour," *TPAMI*, 1983.
- [59] J. Malik, "Interpreting line drawings of curved objects," *IJCV*, vol. 1, 1987.
- [60] J. Koenderink, A. Van Doorn, C. Christou, and J. Lappin, "Perturbation study of shading in pictures," *Perception*, 1996.
- [61] A. Blake, A. Zisserman, and G. Knowles, "Surface descriptions from stereo and shading," *Image and Vision Computing*, 1986.
- [62] D. Terzopoulos, "Image analysis using multigrid relaxation methods," *TPAMI*, 1986.
- [63] M. K. Johnson and E. H. Adelson, "Shape estimation in natural illumination," *CVPR*, 2011.
- [64] D. Forsyth and A. Zisserman, "Reflections on shading," *TPAMI*, 1991.
- [65] J. T. Barron and J. Malik, "Intrinsic scene properties from a single rgb-d image," *CVPR*, 2013.