

# Recovering Shape from a Single Image of a Mirrored Surface from Curvature Constraints

Marshall F. Tappen  
University of Central Florida  
Orlando, Florida  
mtappen@eecs.ucf.edu

## Abstract

*This paper presents models and algorithms for estimating the shape of a mirrored surface from a single image of that surface, rendered under an unknown, natural illumination. While the unconstrained nature of this problem seems to make shape recovery impossible, the curvature of the surface cause characteristic image patterns to appear. These image patterns can be used to estimate how the surface curves in different directions. We show how these estimates can be used to produce constraints that can be used to estimate the shape of the surface. This approach is demonstrated on simple surfaces rendered under both natural and synthetic illuminations.*

## 1. Introduction

This paper introduces a new framework for estimating the shape of a mirrored surface under unknown illumination. Our goal is to use images of surfaces like those shown in Figure 1 to recover the shape of the surfaces pictured.

If the illumination were known, a solution that inverted the rendering equation, similar in spirit to shape-from-shading [9], could be feasible, but working under an unknown illumination makes an inverse rendering approach impossible.

Instead, we take a constraint-based approach where we rely on the statistical behavior of natural illuminations to derive constraints on the curvature of the surface. These constraints, based on elongation patterns that can be seen in Figure 1, are then used to estimate the shape of the surface.

Solving this problem can be roughly divided into two main steps:

1. Estimating the curvature, or change of orientation, in different directions at every point on the image.
2. Integrating these curvature constraints into a constraints or energy functions that can be optimized to recover the shape of the surface.

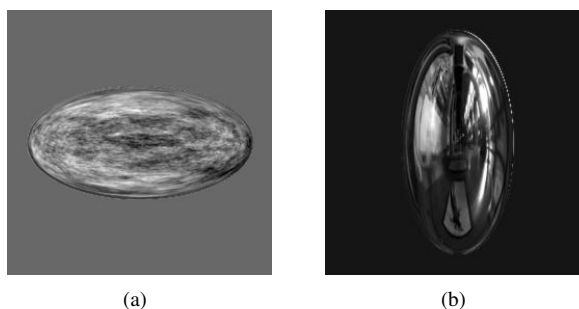


Figure 1. These images of specular ellipsoids, rendered under synthetic and natural illuminations, show how the curvature at different points leads to noticeable image patterns. These can be used to estimate the shape of the surface.

In this paper, we introduce solutions to both of these problems and demonstrate how it is possible to recover the shape of basic, mirrored surfaces under both artificial and natural illuminations.

To our knowledge, this is the first system that recovers shape from a single image rendered under unknown illumination. Thus, we make several simplifications to the problem to make it possible to address the core problem of reconstructing shape, including working with relatively simple surfaces and providing some boundary information, as will be discussed in Sections 8.1 and 8.3. However, we argue that this work is a significant step forward in the ability to recover shape as it points the way for successfully performing shape reconstructions in unknown illumination.

In the remainder of this paper, Sections 2 and 5.3.1 discuss related work on this problem. Sections 3 and 4 explain how curvature affects the surface appearance and can be used to place constraints on the surface shape. The shape optimization is discussed in Sections 5, 6, and 7, while Sections 8 and 9 describe the experiments and results.

## 2. Related Work

Recent work has shown that if the object can be observed while moving, or in a moving environment, the changing reflections form a flow field, known as the specular flow, that can be used to reconstruct the shape of the surface [15, 1, 3]. Earlier work on this reconstruction problem includes [14, 2, 19]. The recent work of [3] is notable in that it provides a linear formulation for recovering shape from the specular flow.

Other work has focused on recovering the surface when the illumination is known [8, 10]. Savarese et al. have shown how it is possible to recover surface orientation from a single image with a known, structured illumination [17].

This approach presented is unique with respect to this previous work in that the surface is estimated from a single image rendered under unknown illumination. This work is influenced by the work of Fleming, Torralba, and Adelson, who explored the connection between image patterns and surface shape [5]. The connections between this work and other previous work will be presented later in Section 5.3.1.

## 3. Curvature and Image Appearance

When the reflectance of the surface and the environment illumination are known, they can be used to derive constraints on the orientation at each point on the surface. The reflectance map can be thought of as defining all of the possible surface orientations at that point. This constraint at every point on the surface, combined with smoothness priors on the surface, makes it possible to recover an estimate of the surface.

Here, we consider mirrored surfaces under an unknown illumination, which makes this strategy impossible. Because the illumination is unknown, there is not a deterministic relationship between image intensity and surface orientation. However, it is still possible to derive constraints on the surface shape.

Previous work has pointed out that when surface orientation changes more quickly in one direction than in the other, elongated patterns are formed in the image [12, 20, 16]. This can be seen in the mirrored ellipsoids shown in Figure 1. In these images, the image intensities along the outer edges of the ellipsoid have an elongated appearance. Using the term curvature to informally mean change in orientation, elongated bands form because the surface curves more quickly in the one direction than in the other direction. This is equivalent to stating that neighboring pixels have roughly the same orientation in one direction, while orientation changes rapidly in the other direction. On the other hand, the intensities vary more isotropically near the center of the ellipsoids, this is because the curvature is relatively similar in all directions.

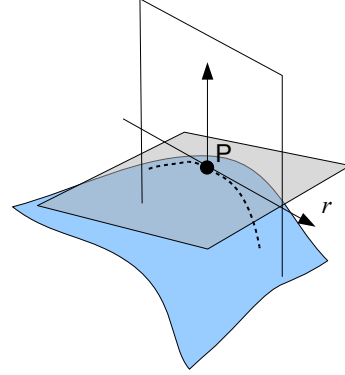


Figure 2. This figure, based on a similar figure in [6], shows how a plane is intersected with a surface to produce plane curves. These plane curves are used to calculate curvature in different directions.

## 4. Curvature Constraints

Given an image of a surface viewed under unknown illumination, these image patterns make it possible to constrain the shape of the surface. These constraints will be defined in a similar fashion to the formal definition of curvature.

For a plane curve, the curvature at a point represents  $\frac{d\theta}{ds}$ , which is the change in the angle of the surface normals,  $\theta$ , with respect to path length  $s$ . This 2D notion of curvature can be generalized to surfaces using normal sections [6]. A normal section is a planar curve created by intersecting the surface with a plane that contains the surface normal. Figure 2 shows an example of how a normal section is created. In this figure, the plane contains the surface normal at point  $P$ . The intersection of this plane with the surface produces a curve in the plane. The *normal curvature* is the curvature of this curve. A family of normal sections can be obtained by rotating the plane around the normal vector. The maximum and minimum curvature in this family of curves are called the principle curvatures.

### 4.1. Curvature Constraints from Images

This definition of curvature is difficult to use as the basis of a constraint on surface shape because it is defined in terms of the path length of the curve. In addition, other curvature related quantities, such as Gaussian curvature, are defined relative to the tangent plane of the surface.

Here, we focus on shape recovery from an orthographically projected image of the surface. Again, change in orientation at a point  $P$  will be measured on plane curves created by intersecting the surface with planes of different orientations. However, instead of rotating those planes around the surface normal, the plane will be rotated around a vector, at point  $P$ , that is oriented parallel to the  $z$ -axis, assuming that the viewing direction is also along the  $z$ -axis.

For a planar curve, this makes it natural to measure the change in orientation with respect to horizontal axis of the

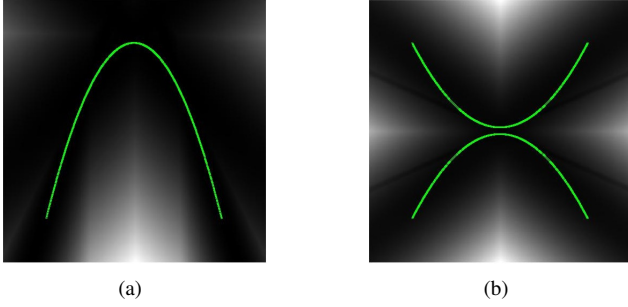


Figure 3. These curves show the combinations of first and second derivative values that satisfy constraints like those in Equations 3 and 4. In both of these plots, the horizontal axis represents the first derivative. Figure (a) represents the constraint when the curvature is known and negative, while Figure (b) is the constraint that can be imposed if only the magnitude of the curvature is known. The background of the plot represents the actual values of the functions in Equations 3 and 4. The energy function, which are the log of Gaussian mixture models, fit the shape of the curves well.

intersecting plane, which will be parametrized with the coordinate variable  $r$ . In derivative notation, instead of measuring  $\frac{d\theta}{ds}$ , the constraint will be based on  $\frac{d\theta}{dr}$ . Although this is different than the standard definition of curvature, we will refer to  $\frac{d\theta}{dr}$  as curvature, for brevity.

If one of these plane curves parametrized as  $f(rx)$ , the orientation at a point can be calculated as

$$\theta = \tan^{-1} \left( \frac{df}{dr} \right). \quad (1)$$

Because we are interested in the change in orientation, Equation 1 is differentiated giving the following relationship between change in orientation and the derivatives at a point:

$$\frac{d\theta}{dr} = \left( \frac{1}{1 + \left( \frac{df}{dr} \right)^2} \right) \left( \frac{d^2f}{dr^2} \right) \quad (2)$$

We expect that the curvature  $\frac{d\theta}{dr}$  will be estimated from the image, so this constraint can be thought of as translating the change in surface orientation of the plane curve into a constraint on the first and second derivative of the curve.

## 4.2. Incorporation into an Energy Function

As will be discussed in Section 5, the shape-estimation problem will be formulated as an energy minimization problem. The constraint in Equation 2 can be expressed as this energy term

$$E_{Curv}(f', f''; C) = \left( C \left( 1 + (f')^2 \right) - f'' \right)^2, \quad (3)$$

where  $f'$  and  $f''$  represent the first and second derivatives, for compactness. The constant  $C$  denotes the curvature estimate from the image.

Figure 3(a) shows the structure of this constraint for a negative curvature. First and second derivative pairs that satisfy Equation 2 are plotted on the graph, with the first derivative along the horizontal axis, while the background shows the value of the energy component in Equation 3.

If it is only possible to estimate the magnitude of the curvature at a point, the number of derivative pairs that satisfy the constraint roughly doubles because the sign of the curvature is ambiguous. Figure 3(b) shows the additional set of points that satisfy the constraint when only the magnitude is known. In this case, the energy term in Equation 3 must be modified to reflect that only the magnitude of the curvature is known. Without knowing the sign of the curvature the energy term in Equation 3 becomes

$$E_{Curv}(f', f''; |C|) = \min \left\{ \begin{aligned} & \left( |C| \left( 1 + (f')^2 \right) - f'' \right)^2, \\ & \left( -|C| \left( 1 + (f')^2 \right) - f'' \right)^2 \end{aligned} \right\}. \quad (4)$$

The background in Figure 3(b) shows the value of this energy term.

Of course, Equations 3 and 4 only define an energy term for a plane curve. Different terms in the final energy function will be used to express the curvature in plane-curves created from intersecting planes with different orientations, just as the principal curvatures are measured by intersecting planes of different orientation.

## 5. Estimating the Shape by Energy Minimization

Shape estimates are produced using an energy function that has the commonly used form

$$E(\mathbf{x}) = E_{Data}(\mathbf{x}) + E_{Smooth}(\mathbf{x}) + E_{Boundary}(\mathbf{x}), \quad (5)$$

In addition to the two terms representing the smoothness and data costs, which are common in most Markov Random Field models, the term  $E_{Boundary}(\mathbf{x})$  denotes the boundary conditions, described in more detail in Section 8.3, used in the shape reconstruction process. In this work, the smoothness cost,  $E_{Smooth}(\mathbf{x})$ , is a basic regularization on the second-order derivatives:

$$E_{Smooth}(\mathbf{x}) = \sum_p (f_{xx})^2 + (f_{xy})^2 + (f_{yy})^2 \quad (6)$$

### 5.1. Fitting the Data Term with Mixture Models

The energy functions shown in Figures 3(a) and 3(b), which are the basis for the data term, have a complicated shape that makes the overall energy-function non-convex. In response to this, the elements of the data term will be approximated with Gaussian mixture models, similar to [11]. This formulation makes it possible to use an upper-bound minimization approach to minimize the energy function in

Equation 5. It also has the benefit of making it possible to learn parameters of the model using the Variational Mode Learning technique from [18].

Formally, the vector  $\mathbf{d} = [f', f'']$  will denote a vector containing the first and second derivatives of a plane curve at point. A Gaussian mixture model on  $\mathbf{d}$ , with  $N_C$  mixture components, will be denoted as  $M(\mathbf{d}; \Theta)$ , where  $\Theta$  will represent all of the parameters of the mixture model, including the mixing coefficients, means, and covariance matrices, for all of the mixture components in the model.

Figures 3(a) and 3(b) show examples of how mixture models can fit these curvature-based energy functions. The values shown in this figures are  $-\log M(\mathbf{d}; \Theta)$ , where  $\Theta$  has been fit to the points plotted in green using the Expectation-Maximization algorithm. Using the negative logarithm makes it possible to treat the mixture model as an energy function. Each pixel in these images represents a different value of  $\mathbf{d}$ . As can be seen in these figures, the mixture approximates the shape of the energy function well.

## 5.2. Implementing the Data Term – Estimating Curvature

As discussed in Section 4.2, the structure of the data term at a point depends on the curvature in different directions. For different values of curvature, the shape of the energy functions in Figure 3 will change. This can be easily accommodated by discretizing the range curvature values and fitting a mixture model for each different curvature, similar to how mixture models were fit for different intensity levels in [11].

The key problem is estimating curvature. Just as the traditional principal curvatures are found by finding the direction of the maximal and minimal curvature, the observed image will be analyzed to recover both the directions<sup>1</sup> along which the curvature constraint will be enforced and the magnitude of the change in orientation along these directions.

### 5.2.1 Estimating Curvature with the Image Structure Tensor

The curvature directions and magnitude are estimated using the image structure tensor. The estimation process begins by computing the vertical and horizontal derivatives of the image, which will be denoted  $I_x$  and  $I_y$ . If  $I^p$  denotes a patch of the image at pixel  $p$ , with  $I_x^p$  and  $I_y^p$  denoting the image derivatives in the patch, then the image structure tensor, or second moment matrix, is the matrix

$$S_p = \begin{bmatrix} w * (I_x^p)^2 & w * (I_x^p)(I_y^p) \\ w * (I_x^p)(I_y^p) & w * (I_y^p)^2 \end{bmatrix}. \quad (7)$$

In this formulation  $w * (I_x^p)^2$  denotes an element-wise squaring of the pixels in  $I_x^p$ , followed by a convolution with

<sup>1</sup> The direction vectors will lie in the  $xy$ -plane as the viewer is looking down the  $z$  axis.

a weighting mask,  $w$ . Assuming that  $w$  is the same size as  $I_x^p$ , the convolution will return one value.

The orientations of the curvature constraints are the eigenvectors,  $\mathbf{e}_1$  and  $\mathbf{e}_2$ , of the second moment matrix. These eigenvectors describe the orientation of maximum variation in the image and the direction of minimum variation. As changes in image intensity signify changes in surface orientation, these two directions be close to the directions of maximum and minimum curvature.

The eigenvalues,  $\lambda_1$  and  $\lambda_2$ , are used to estimate the amount of change in orientation. Again, more change in orientation should correspond to more image variation, and a larger eigenvalue. These eigenvalues will always be positive because Equation 7 is a second moment matrix. For this system, the practical consequence of this is that only the magnitude of curvature can be estimated.

### 5.2.2 Learning the Data Term at Each Pixel

In implementing the system, we eliminated the intermediate step of estimating curvature from the eigenvalues then using that estimate to choose the mixture parameters. Instead, the approach is more directly driven by pattern recognition. Our system uses a training database to learn a set of mixture models, like the models in Figure 3(a), that express the constraint between the first and second derivative for different amounts of curvature. The data term is constructed by using the appropriate mixture model at each point. The training database used to find the mixture models, discussed in Section 8, contains images similar to those in Figure 1 and corresponding ground-truth surfaces.

As a constraint will be placed at every pixel, the appropriate mixture model must be chosen for each pixel. This choice is made using a descriptor based on the eigenvalues of the second moment matrix. The descriptor,  $\mathbf{d}$ , at pixel  $p$ , is computed as

$$\mathbf{d}(p) = \log \left( \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} \right) + 10^{-2} \quad (8)$$

The advantage of this descriptor is that it is invariant to the location orientation of image. The purpose of the logarithm is to make for a better spacing of descriptor values.

After the descriptors have been computed the  $k$ -means algorithm is used to cluster divide them into clusters. For each cluster, two mixture models are fit to the steered first and second derivatives of the surface [7]. One mixture model is fit to the derivatives in the direction of the eigenvector  $\mathbf{e}_1$ , while the other mixture model is fit to the steered derivatives in the direction of  $\mathbf{e}_2$ . The Expectation-Maximization algorithm is used to fit both mixtures. All mixture models are fit to values of steered  $5 \times 5$  derivative of Gaussian filters.

### 5.3. Constructing the Data Term on Novel Images

Given a novel test image, the data term is constructed by computing the eigenvectors and eigenvalues of the second moment matrix at each pixel. If  $\mathbf{d}_p$  is the descriptor at pixel  $p$ , the data term at pixel  $p$  is based on the cluster center that is closest to  $\mathbf{d}_p$ . Formally, the data term at pixel  $p$ , which will be denoted  $E_{Data}^p(\mathbf{x}_p)$  has the form

$$E_{Data}^p(\mathbf{x}_p) = -\log M([x'_{\mathbf{e}_1} x''_{\mathbf{e}_1}]; \Theta_{c(\mathbf{d}_p)}^1) \\ + -\log M([x'_{\mathbf{e}_2} x''_{\mathbf{e}_2}]; \Theta_{c(\mathbf{d}_p)}^2). \quad (9)$$

In this equation,  $c(\mathbf{d}_p)$  denotes the index of cluster center that is closest to  $\mathbf{d}_p$ . The vector  $\mathbf{x}_p$  denote a small image patch centered at pixel  $p$ , while  $x'_{\mathbf{e}_1}$  and  $x''_{\mathbf{e}_1}$  denote the directional first and second derivatives at  $p$ , in the direction  $\mathbf{e}_1$ , computed from  $\mathbf{x}_p$ . Again,  $M(\mathbf{d}, \Theta_{c(\mathbf{d}_p)}^1)$  denotes a Gaussian mixture model with parameters indexed by the cluster index assigned to  $p$ . The superscript denotes whether the parameters correspond to the direction  $\mathbf{e}_1$  or  $\mathbf{e}_2$ .

The complete data term is simply the sum across all pixels  $p$  inside the object:

$$E_{Data}(\mathbf{x}) = \sum_{p=1}^{N_P} E_{Data}^p(\mathbf{x}_p) \quad (10)$$

#### 5.3.1 Related Work

In [5], Fleming et al. show how the local orientation as measured by a population of linear filters is similar to the local orientation of the surface. Similar to the discussion of curvature and second derivatives, Weidenbacher et al. analyze image information to find areas of high curvature and produce a sketch representation of a surface [20]. This sketch does not produce an estimate of the surface, but does capture an accurate visual representation of the surface.

More recently, [16] used the image structure tensor to find an invariant indicating parabolic lines in the image. In [16], multiple images are used to find this invariant. Using multiple images reduces errors from structure in the environment being confused as parabolic lines. This invariant is used in [16] to perform some shape-related tasks, but is not used to directly recover shape. In this work, we use the image structure tensor to identify curvature in a single image, but find that reasonable estimates can still be obtained, even in the presence of structure in the environment.

At a root level, our work and the previous work just described can be thought of as building on the patterns induced by parabolic lines and first studied by Koenderink and van Doorn[12]. Our goal in this work is to transform the identifiable image patterns into actual shape estimates.

## 6. Optimization for Producing Shape Estimates

Once the energy functions for the data term have been defined, the estimated shape  $\mathbf{x}^*$  is computed by minimizing the energy function in Equation 5. As mentioned above,

constructing the data term using mixture models makes it possible to use an upper-bound minimization scheme to compute  $\mathbf{x}^*$ .

The key step in the optimization process is fitting a tight, quadratic upper-bound to the data term, which can be combined with the smoothness term that is already quadratic. Once the bound is found, the resulting quadratic function can be minimized and the the bound re-computed. This leads to a alternating set of steps that can be shown to make the energy function decrease monotonically.

It is well-known that the logarithm of a Gaussian mixture model can be upper-bounded [13], which corresponds to lower-bounding the probability density function. The supplemental material reviews the basic concepts behind the bounds.

## 7. Learning Weights

An advantage of the bound-minimization procedure in Section 6 is that it makes it possible to learn weights for the different components of the data term. We incorporate weights similar to [11] where a weight is assigned to the data term at a pixel based on the cluster index that is used to choose the mixture component at that pixel. With this weighting, the data term becomes

$$E_{Data}(\mathbf{x}) = \sum_{p=1}^{N_P} w_{c(\mathbf{d}_p)} E_{Data}^p(\mathbf{x}_p) \quad (11)$$

where  $c(\mathbf{d}_p)$  denotes the cluster index computed using the descriptor from Section 5.2.1.

We refer the reader to [11] and [18] for details on how the weight values associated with each cluster are optimized. The key aspect of the learning is that the optimization is run for a fixed number of iterations, which makes it possible to treat the optimization as a series of differentiable operations. This, in turn, makes it possible to use gradient descent to optimize the weights.

## 8. Data for Evaluation

The ability of this approach to recover shape estimates is demonstrated using a set of smooth, blobby shapes rendered under different illuminations. This section will discuss the data and assumptions in the experiments.

### 8.1. Surfaces

We use a set of twenty synthetically-generated surfaces. These surfaces are created to be smooth and have an occluding contour that matches a predefined outline. The surfaces are split into a test and training set, with ten surfaces in each set.

Figures 4, 5, and 6 show examples of these surfaces. These surfaces are relatively simple in that they do not contain concavities on the surfaces. Simple surfaces were chosen deliberately because the constraints described here

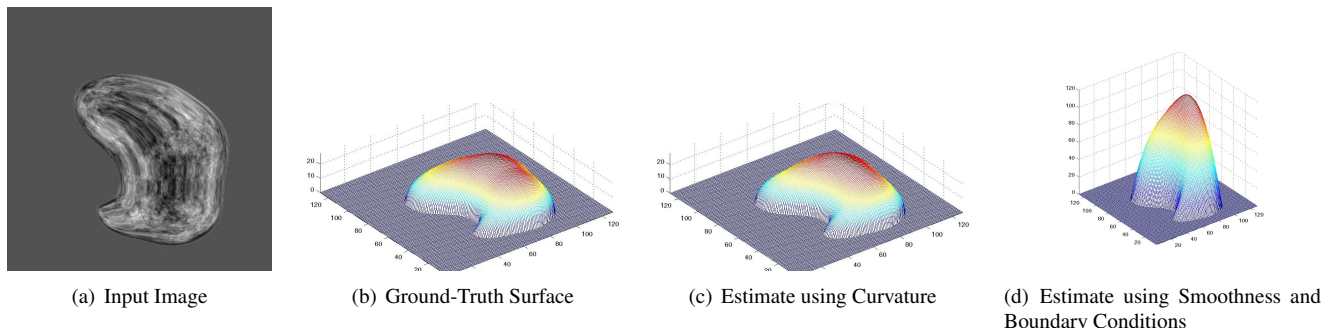


Figure 4. Illustration of results obtained using the image in (a), which has been rendered under a random noise illumination. The estimate produced using the curvature constraints, shown in (c) is quite similar to the ground-truth. On the other hand, the estimate produced using just the boundary conditions and smoothness prior is a poor estimate of the shape.

are ambiguous with respect to the convexity and concavity of the surface, as in shape-from-shading. Properly handling the concave/convex ambiguity will be the focus of future research.

## 8.2. Illumination

This work assumes that the surfaces are perfect mirrors. To produce rendered imagery, we use both synthetic and real environmental illumination. The real illuminations come from Debevec’s Light Probe Gallery [4]. Because these light maps are high-dynamic range, we limit the range by thresholding and a gamma-like operator for the beach illumination.

In addition to the real illuminations, this approach is also evaluated using synthetically-generated noise. The environmental illumination is produced by sampling a cube of  $1/f$  noise. We use this synthetic noise because this illumination is ideal for the curvature estimation. Two of the most significant sources of error for the curvature estimation are the lack of image structure to analyze and structure in the environment, such as edges, being confused for the elongation described in Section 3. Using random noise reduces possible error from either of these problems.

## 8.3. Boundary Conditions

As shown in Equation 5, boundary conditions are used to enable shape reconstruction. In this work, the height values are provided in a three pixel border around the edge of each surface. The boundary conditions are provided because the main purpose of this work is to examine if the interior of the surface can be reconstructed accurately. In future work, we expect to incorporate boundary constraints into the reconstruction system.

## 9. Experimental Results

In the results described in this section, the quality of the surfaces will be assessed both qualitatively and quantitatively. We first present results on images rendered under the

noise illumination, then show results on images rendered under natural illuminations.

### 9.1. Noise Illumination

We first present results on images rendered under random noise illumination.

#### 9.1.1 Comparison with Smoothing Only

The most basic comparison is whether the model described here improves on what could be accomplished just using the combination of the boundary terms and smoothness costs. As shown in Figure 4, the addition of the curvature-driven data terms makes a significant improvement in the quality of the shape reconstruction. Figure 4(d) shows the surface estimated using just the shape constraint. This estimate strongly overshoots the ground-truth surface, shown in Figure 4(b), because the second-order smoothing prior makes the overall height of the surface dependent on the magnitude of the derivatives around the boundary.

On the other hand, the estimate produced using the data term introduced here, shown in Figure 4(c), is quite similar to the ground-truth surface. This qualitative similarity can also be quantitatively verified. The root-mean-square(RMS) error of the estimated surface in Figure 4(c) is 56 times less than the RMS error of the surface computed using smoothing, shown in Figure 4(d).

The significant difference in error can be seen across the entire set of testing images. On average, the estimate produced by our system has 22 times less error than the estimate produced using only smoothing.

#### 9.1.2 Model Complexity

In the data term described in Section 5.1, the image information is used to choose which mixture models constrains the surface derivatives at each point. This causes the mixture models to differ according to the type of curvature that they express. Thus, using more mixture models makes it possible to more finely specify the curvature – at the risk of imposing an incorrect curvature.

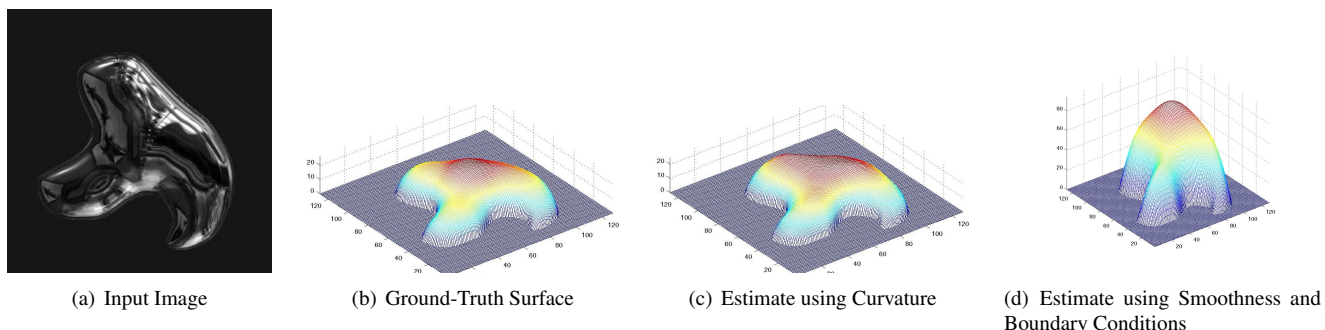


Figure 5. These figures, similar to those in Figure 4, show that the system is also able to produce good estimates from surfaces rendered under a real illumination.

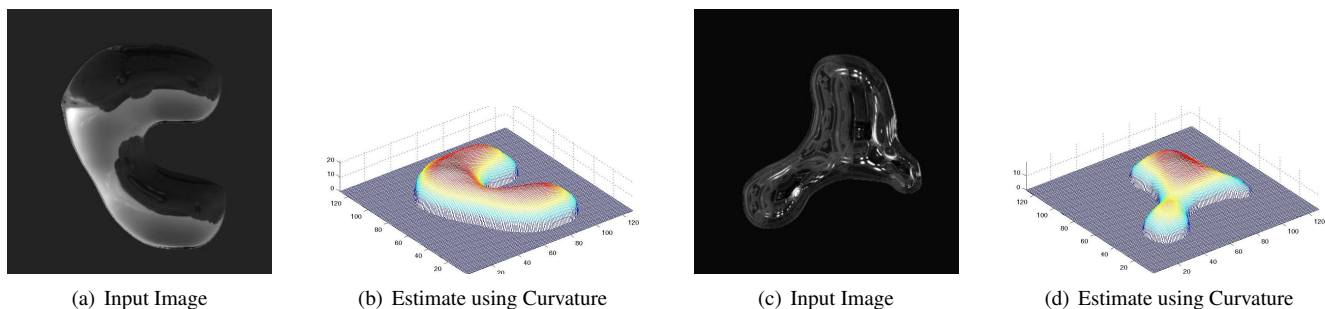


Figure 6. This images show how our approach is successfully able to recover surfaces rendered under the St. Peter's and Beach illuminations.

Using the training set generated under the random-noise illumination, we trained three different data term models, composed of one, ten, and fifty different mixture models in the term<sup>2</sup>. While the data term built from fifty different mixture models performed the best on the training set, we found that the data-term composed of one mixture model performed the best on both the test set and the real illumination experiments described in the next section. This is likely because the simpler model is less likely to strongly constrain the surface based on incorrect curvature estimates.

Using the one-component model makes the estimated shape primarily dependent on the eigenvectors of image structure tensor,  $e_1$  and  $e_2$  in Section 5.2.1, at each point. This is similar to just using the local orientation cues discussed in [5] and these results validate that local orientation is a useful cue for shape estimation cue.

## 9.2. Real Illumination

In addition to the random noise illuminations, this shape estimation system was also evaluated on three different real-world illuminations from [4]: Galileo's Tomb, St. Peter's Basilica, and the beach scene. In general, we found that the shape estimates produced from images rendered under these illuminations were comparable in quality to the estimates recovered from images rendered under the random

noise illumination. Estimate produced under these illuminations are shown in Figures 5 and 6.

As Figure 5 shows, the system is also able perform well on images rendered under the Galileo's Tomb Illumination. Using surfaces rendered under this illumination, we experimented with models trained under two different illuminations: the model trained under the noise illumination described in the previous section and a model trained under the Galileo illumination itself, but using the training surfaces. Surprisingly, we found model trained under noise illumination performed quite well on these images, performing slightly better than the model trained on surfaces rendered under this same illumination.

Figure 7 gives some insight into why the system performs well under this illumination. This figure shows a histogram on the angle between orientation of the direction of maximum second-derivative on the surface and the orientation of  $e_1$  recovered from the image structure tensor for both the random noise and Galileo illumination. As can be seen in this figure, the histograms are quite similar, indicating that the directions of curvature can be estimated well in both illuminations. It should be noted that the angles near 90 degrees may not be significant errors. If the curvature is roughly equal in both directions, such as at an umbilical point on the surface, the angle between the orientation vectors can be large without significantly affecting the model.

<sup>2</sup>Each of these mixture models was itself composed of three Gaussian components.

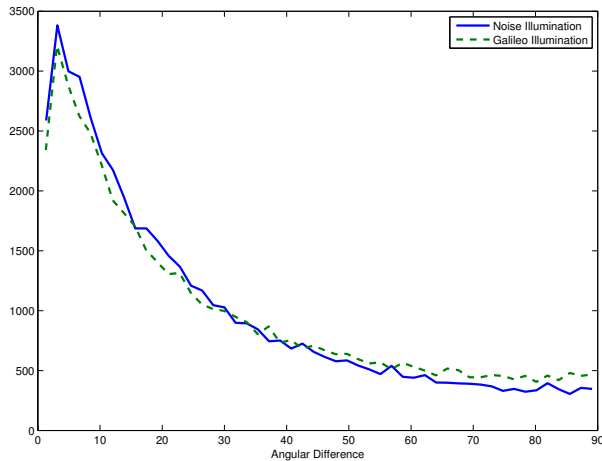


Figure 7. This histogram measures the proportion of different angular errors in the estimation of the orientation of curvature. The estimated orientations found from the image structure tensor are compared with the orientations of the eigenvectors of the Hessian matrix at each point on the surface. The histograms for both noise illumination, plotted with the solid line, and Galileo’s tomb illumination, plotted with a dotted line, are similar, indicating that the system perform roughly as well on natural illuminations as synthetic illuminations.

## 10. Conclusion

This work has shown how curvature constraints on both the direction and magnitude of the curvature of a surface can be estimated from a single image of a specular surface. In addition, we have shown how these constraints can be integrated into a shape-estimation system. This system is able to produce good results, even when operating on surfaces rendered under natural illuminations.

In this work, we have somewhat simplified the problem to make it possible to focus on the core curvature constraints. Nonetheless, we believe these results show a path for robustly estimating shape in unknown illumination. In addition to improving our ability to accurately estimate curvature, our future work will focus on removing the simplifying aspects of the problem.

## Acknowledgements

The author thanks Edward Adelson, Antonio Torralba, and Roland Fleming for helpful discussions and ideas. This work was supported by NSF grants IIS-0905387, IIS-0916868, and a grant under the NGA NURI program.

## References

[1] Y. Adato, Y. Vasilyev, O. Ben-Shahar, and T. Zickler. Toward a theory of shape from specular flow. In *Proceedings of the IEEE International Conference on Computer Vision*, 2007. 2

[2] A. Blake and G. Brelstaff. Geometry from specularities. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 394–403, 1988. 2

[3] G. D. Canas, Y. Vasilyev, Y. Adato, T. Zickler, S. Gortler, and O. Ben-Shahar. A linear formulation of shape from specular flow. In *Proceedings of the IEEE International Conference on Computer Vision*, 2009. 2

[4] P. Debevec. Image-based lighting. *IEEE Computer Graphics and Applications*, 22:26–34, 2002. 6, 7

[5] R. W. Fleming, A. Torralba, and E. H. Adelson. Specular reflections and the perception of shape. *Journal of Vision*, 4(9):798–820, 9 2004. 2, 5, 7

[6] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, us ed edition, August 2002. 2

[7] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Transactions Pattern Analysis Machine Intelligence*, 13:891–906, September 1991. 4

[8] G. Healy and T. O. Binford. Local shape from specularity. *Computer Vision, Graphics, and Image Processing*, (42):62–86, 1988. 2

[9] B. K. P. Horn. *Robot Vision*. MIT Press, Cambridge, Massachusetts, 1986. 1

[10] K. Ikeuchi. Determining surface orientations of specular surfaces by using the photometric stereo method. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 3(6):661–669, November 1981. 2

[11] N. Khan, L. Tran, and M. F. Tappen. Training many-parameter shape-from-shading models using a surface database. In *International Conference on 3-D Digital Imaging and Modeling at ICCV 2009*, 2009. 3, 4, 5

[12] J. Koenderink and A. van Doorn. Photometric invariants related to solid shape. *Optica Acta*, (27):981996, 1980. 2, 5

[13] R. Neal and G. E. Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, pages 355–368. Kluwer Academic Publishers, 1998. 5

[14] M. Oren and S. K. Nayar. A theory of specular surface geometry. *Int. J. Comput. Vision*, 24(2):105–124, 1997. 2

[15] S. Roth and M. J. Black. Specular flow and the recovery of surface structure. In *CVPR ’06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1869–1876, Washington, DC, USA, 2006. IEEE Computer Society. 2

[16] A. C. Sankaranarayanan, A. Veeraraghavan, O. Tuzel, and A. Agrawal. Image invariants for smooth reflective surfaces. In *European Conference on Computer Vision*, Crete, Greece, September 2010. 2, 5

[17] S. Savarese, M. Chen, and P. Perona. Local shape from mirror reflections. *International Journal of Computer Vision*, 64(1):31–67, 2005. 2

[18] M. F. Tappen. Utilizing variational optimization to learn markov random fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR07)*, 2007. 4, 5

[19] S. Waldon and C. R. Dyer. Dynamic Shading, Motion Parallax and Qualitative Shape. In *IEEE Workshop on Qualitative Vision*, pages 61–70, Los Alamitos, CA, 1993. IEEE Computer Society Press. 2

[20] U. Weidenbacher, P. Bayerl, H. Neumann, and R. Fleming. Sketching shiny surfaces: 3d shape extraction and depiction of specular surfaces. *ACM Transactions on Applied Perception*, 3:262–285, July 2006. 2, 5