Variational Shape and Reflectance Estimation under Changing Light and Viewpoints

Neil Birkbeck[1], Dana Cobzas[1], Martin Jagersand[1], and Peter Sturm[2]

[1] Computer Science, University of Alberta, Canada
{birkbeck,dana,jag}@cs.ualberta.ca
[2] INRIA Rhone-Alpes,
France
{peter.sturm}@inrialpes.fr

Paper ID 864

Abstract. Fitting parameterized 3D shape and general reflectance models to 2D image data is challenging due to the high dimensionality of the problem. The proposed method combines the capabilities of classical and photometric stereo, allowing for accurate reconstruction of both textured and non-textured surfaces. In particular, we present a variational method implemented as a PDE-driven surface evolution interleaved with reflectance estimation. The surface is represented on an adaptive mesh allowing topological change. To provide the input data, we have designed a capture setup that simultaneously acquires both viewpoint and light variation while minimizing self-shadowing. Our capture method is feasible for real-world application as it requires a moderate amount of input data and processing time. In experiments, models of people and everyday objects were captured from a few dozen images taken with a consumer digital camera. The capture process recovers a photo-consistent model of spatially varying Lambertian and specular reflectance and a highly accurate geometry.

1 Introduction

The automatic computation of 3D geometric and appearance models from images is one of the most challenging and fundamental problems in computer vision. While a more traditional point-based method provides accurate results for camera geometry, a surface representation is required for modeling and visualization applications. Most surface-based approaches reconstruct the model based on stereo correlation data [1–3]. That works well for textured Lambertian surfaces but fails in the presence of specular highlights or uniform texture. Additionally, stereo-based techniques reconstruct only the shape and not the surface reflectance properties even though some approaches can handle specular objects using robust scores [4, 5].

We are proposing a surface reconstruction method that uses texture and shading information to successfully reconstruct both textured and non-textured objects with general reflectance properties. The similarity cost functional uses a parametric reflectance model that is estimated together with the shape. There exist other approaches that combine stereo for textured regions with shape from shading cues for texture-less regions [6, 7], but, in those works, the two scores are separate terms in the cost function and the combination is achieved either using weights [6] or by manually assigning regions [7]. Additionally, they only exploit diffuse objects whereas our method can also handle specular objects. The uniform albedo assumption is necessary only when there is no light variation on the object. Instead, in our work, we assume that the object is moving relative to the light source, which produces illumination variation on the object; therefore, we allow for a true spatially varying albedo like in the photometric stereo techniques. Zhang et al. [8] and Weber et al. [9] also use light variation for reconstructing spatially varying albedo. But, in contrast to our approach, they do not consider the challenge of dealing with specular surfaces.

More and more researchers have developed methods that recover specular surfaces. Most of the approaches either filter or remove specularities and use only diffuse observations in the reconstruction [5]. Another option is to design similarity scores that account for specular highlights either using constraints on the light and camera positioning [10] (Helmholtz reciprocity) or on surface material [11] (uniform material under constant light). A more general approach is to explicitly model surface reflectance either with a parametric model [12] or a non-parametric model (BRDF map). Obtaining a BRDF map requires carefully calibrated lights and many samples [13]. A parametric model does not necessarily generalize constraints on the observations at a point but it requires less data and can be more easily fitted compared to a non-parametric model. An interesting but rather unpractical approach is presented by Treuille et al. [14] where a known object having similar material with the unknown scene is used to lookup surface orientations with similar observed colors. For our system we made the choice of using a parametric model for reflectance as we are interested in reconstructing both shape and reflectance parameters. We allow varying materials and provide an efficient way of clustering points with similar properties.

Important aspects of shape reconstruction are the surface representation and the optimization method. Different representations have been proposed in the literature. They can be divided in two main classes - image-based (depth/disparity) and object-based (voxel grid, mesh, level set). Image-based representations are suitable for single view or binocular stereo techniques but they are less appealing for multi-view approaches as they become biased by the choice of reference camera. Object-based representations are more suitable for multi-view reconstruction. Mesh and level set techniques have the advantage over voxel representations that they give readily computable normals (essential in recovering shading). Additionally, the regularization terms can be easily integrated into a mesh or level set. An implicit level set representation leads to an elegant algorithm [2], but despite various efficient numerical solutions proposed for the level set methods [15], they are still slow compared to mesh based approaches that can take advantage of graphics hardware acceleration. We therefore decided to implement our method using an adaptive deformable mesh that allows for topological changes. The mesh is evolved in time based on a variational algorithm. Fua and Leclerc [6] and Duan et al. [16] have presented related variational mesh-based approaches but not as general as they only reconstruct diffuse objects.

Due to the high dimensionality, reconstruction can be difficult, slow and require lots of image data. To ameliorate these problems, we propose a multi-resolution algorithm that alternates between shape and reflectance estimation. We also propose a practical capture setup for acquiring images having enough light variation to allow robust fitting of a parametric reflectance model as well as an easy way to automatically obtain calibration for camera and light position. Although in theory a general reflectance model can be estimated at every step, we practically noticed that similar results are obtained more efficiently if the shape reconstruction is performed on filtered diffuse pixels assuming Lambertian reflectance. A Phong parametric model is then calculated using the final shape. Experiments show that the proposed method is able to reconstruct accurate and photo-realistic models that can be rendered in novel illumination conditions. To summarize, the main contributions of the paper are:

- We designed a photo-consistency functional suitable for surfaces with non-uniform general reflectance based on a parametric reflectance model;
- We present a variational method implemented as a PDE-driven mesh evolution interleaved with reflectance estimation. Our particular mesh implementation is robust to self-intersections while allowing topological changes;
- We designed a practical setup that provides the necessary light variation, camera and light calibration and requires only commonly available hardware: a light source, a camera, and a glossy white sphere.

2 Shape Refinement

We present the shape refinement problem beginning with a general continuous formulation that is then discretized on the mesh triangles. Next, we describe a numeric solution to the resultant optimization problem for an object with Lambertian or specular reflectance.

2.1 Problem Definition

The proposed capture setup consists of a single camera viewing an object placed on a turntable illuminated by a desk lamp. We take two sets of images of a full rotation, each with a different light position. Considering the proposed capture setup, the shape recovery problem takes the following as input:

- a set of n images $\mathcal{I} = \{I_i | i = 1 \cdots n\}$;
- the associated projection matrices P_i;
- the illumination information $L_i = (\mathbf{l}_i, l_i)$, assuming a single distant light source with direction $\mathbf{l}_i$ and color l_i the light color;
- an initial shape S_0;

and computes a refined shape, S, and the corresponding reflectance parameters that best agree with the input images. A practical method for automatically calibrating the camera and the light is presented in Section 4.

Given the projection matrix $P_i = K[R_i, \mathbf{t}_i]$, the image coordinates $\mathbf{p}_i = (u_i, v_i, 1)^T$ for a 3D point $\mathbf{x}$ are expressed as $\mathbf{p}_i = \Pi(P_i\mathbf{x})$. Π represents the non-linear operator that transforms homogeneous coordinates into Cartesian ones (division with the homogeneous component).

We assume that surface reflectance is a parametric function implied by the surface (and surface normals) and imaging conditions. Therefore, the shape reconstruction problem is to recover a shape and its implied reflectance parameters that best agree with the input images. The shape and reflectance are estimated in an alternate fashion (see Section 4).

2.2 Shape Functional

We use a variational formulation for the shape recovery problem similar to the one from Faugeras and Keriven [2].

$$E(S) = \int_S g(\mathbf{x}, \mathbf{n}) dS = \int_v \int_u g(\mathbf{x}, \mathbf{n}) \|S_u \times S_v\| du dv \tag{1}$$

where $\mathbf{x} = (x(u, v), y(u, v), z(u, v))^T$ is a point on the surface and $\mathbf{n} = \frac{S_u \times S_v}{\|S_u \times S_v\|}$ is the surface normal at point $\mathbf{x}$.

The photo-consistency function g encodes the similarity between a point on the surface, and the images in which it is observed. We investigate a similarity function of the form:

$$g(\mathbf{x}, \mathbf{n}) = \sum_i h(\mathbf{x}, \mathbf{n}, P_i, L_i) \left(I_i(\Pi(P_i\mathbf{x})) - R(\mathbf{x}, \mathbf{n}, L_i)\right)^2 \tag{2}$$

where R is a rendering equation returning the color of point $\mathbf{x}$ under light conditions L_i. The function h is a weighting function that accounts for visibility and discrete sampling effects.

Rendering function The function R encodes the reflectance model at a point $\mathbf{x}$ on the surface. In fact, R is a function of the entire surface as it should account for inter-reflections and shadowing of a point $\mathbf{x}$. In our capture setup we minimized

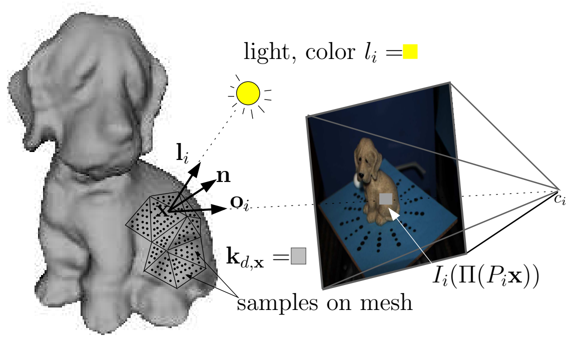

Fig. 1. An illustration of the sample points and the angles used in the shading equation.

self shadowing and inter-reflections and therefore ignored these subtleties. We model R with a parametric BRDF which is fitted to Eq. 2 (assuming known shape and imaging conditions).

For modeling the parametric BRDF we chose the Lambertian model to represent diffuse reflectance and the Blinn-Phong model [17] for the specular reflectance. The two models are briefly summarized below[3]. Refer to Fig. 1 for a explanation of our notations.

Lambertian model assumes constant BRDF and effectively models matte objects, such as clay, where the observed shading is a result of the foreshortening contribution of the light source. Integrating the Lambertian BRDF model into the reflectance equation we get the following expression for the observed color at a particular point $\mathbf{x}$ with normal $\mathbf{n}$:

$$R_{lamb}(\mathbf{x},\mathbf{n},L_i) = (\langle \mathbf{n}, \mathbf{l}_i \rangle l_i + a_i) k_{d,\mathbf{x}} \tag{3}$$

where $k_{d,\mathbf{x}}$ represents the Lambertian color (albedo). For better modeling light effects we incorporate an ambient term to capture the contribution of indirect light in each image a_i.

Specular reflectance is typically modeled as an additive component to the Lambertian model. We chose to represent the specular BRDF using the Blinn-Phong model [17]. Letting $\mathbf{o}_i$ be the outgoing direction from the point $\mathbf{x}$ to the center of the camera i (i.e., the view direction), and $\mathbf{h}_i$ the bisector of the angle between the view and the light directions $\mathbf{h}_i = \frac{\mathbf{o}_i + \mathbf{l}_i}{\|\mathbf{o}_i + \mathbf{l}_i\|}$ the shading model for a specular pixel is (refer to Fig. 1 for an illustration):

$$R_{spec}(\mathbf{x},\mathbf{n},L_i) = (\langle \mathbf{n}, \mathbf{l}_i \rangle l_i + a_i) k_{d,\mathbf{x}} + \langle \mathbf{n}, \mathbf{h}_{i,\mathbf{x}} \rangle^m l_i k_s \tag{4}$$

[3] The proposed method works with color images but for simplicity reasons we present the theory for one color channel. In practice all colors should be viewed as 3D vectors in RGB space.

where k_s is the specular color and m is the specular exponent. The specular parameters are not indexed per point due to the fact that several observations are needed for reliably estimating the BRDF. Instead (as discussed in Section 3) we compute the specular parameters for groups of points having similar diffuse component, thus likely to have the same material.

Weight function The similarity measure with respect to an image should be computed only for the visible points. This can be easily represented by setting the weight function, h, to the binary visibility function $V(\mathbf{x}, S, P_i)$.

Another issue to consider is image sampling. When the estimated surface is far from the actual surface widely separated views sample from different regions of the underlying surface, implying that the gradient may not provide meaningful information. In this case, it is helpful to restrict the number of views used in g. For choosing a subset of the observations, we select the $n_{cameras}$ closest cameras to the median visible camera [5], where the middle camera is currently based solely on the azimuthal angle. This modification fits into our framework by changing the visibility function into V', which returns 0 for all cameras that are not in the considered range.

Another sampling issue occurs even when the true surface is known. A patch on the true surface occupies a different number of pixels when projected into each of the images. In the views where the patch occupies few pixels, the sample points on the patch will project to overlapping pixels. This causes a blurred texture to appear on the patch. To account for this, each sample is weighted by $\langle \mathbf{n}, \mathbf{o}_i \rangle$. We note that this second sampling issue is a consequence of our error function being computed on the surface of the object. This problem can be avoided if the error is formulated in image space.

Cumulating visibility and sampling into the function h we get:

$$h(\mathbf{x}, \mathbf{n}, P_i, L) = \langle \mathbf{n}, \mathbf{o}_i \rangle V'(\mathbf{x}, S, P_i) \tag{5}$$

2.3 Surface Evolution

Optimizing the photo-consistency function in Eq. 1 with respect to the surface S results in a surface evolution problem. The gradient flow PDE is derived from the Euler-Lagrange equation of Eq. 1. The PDE contains higher order terms [2] resulting from the general form of g being a function of $\mathbf{n}$. Instead of using the full PDE, complete with the higher order terms, we use a simplified PDE containing only the first order terms. This flow is accurate for a g that is only a function of surface position $\mathbf{x}$. Similar PDE's were used by [18, 16] but with different g functions.

$$\frac{\partial S}{\partial t} = (2g\kappa - \langle \nabla g, \mathbf{n} \rangle)\mathbf{n} \tag{6}$$

where κ is the mean curvature. The flow will move each point along the current estimate for the normal. The first component of the motion in Eq. 6, $2g\kappa$, is essentially a smoothing term, reducing the mean curvature of the object, whereas the second component ensures the evolution decreases the error function on the surface.

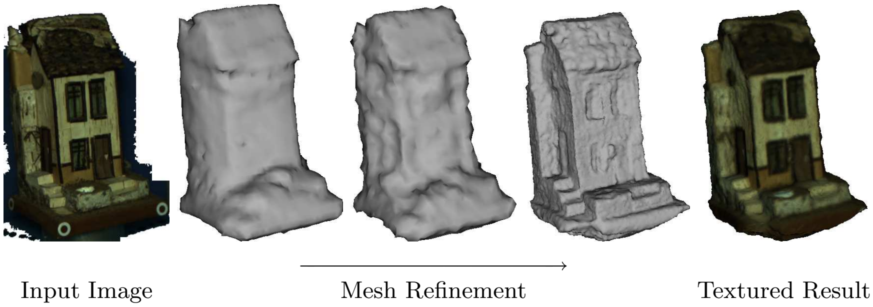

Fig. 2. An example of the mesh evolving to a refined shape (videos: face.avi & dog.avi)

The shape refinement then proceeds by iteratively updating the initial shape, S_0, using Eq. 6 until convergence. We stop the evolution when there is no significant change in the error function for several steps. Fig 2 gives an example of our surface evolution algorithm starting from the visual hull.

2.4 Discretization on the Triangular Mesh

The numerical solution for the surface evolution depends on the chosen representation. As we explore the use of a mesh based representation, it is useful to convert Eq. 1 into its discrete counterpart. The surface integral can first be broken down into a sum of integrals over piecewise continuous regions, namely triangles. Let $\mathit{\Delta} = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)$ be a triangle having vertices $\mathbf{v}_1, \mathbf{v}_2$ and $\mathbf{v}_3$. An interior point on the triangle can be expressed using the barycentric coordinates $\lambda_1, \lambda_2, \lambda_3$ satisfying $\lambda_1 + \lambda_2 + \lambda_3 = 1$ and $\lambda_k \geq 0$ for $k \in \{1, 2, 3\}$:

$$\mathbf{x} = \lambda_1 \mathbf{v}_1 + \lambda_1 \mathbf{v}_2 + \lambda_1 \mathbf{v}_3$$

The triangle normal $\mathbf{n}$ is then computed by smoothly interpolating the normals $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$ of the vertices:

$$\mathbf{n} = \lambda_1 \mathbf{n}_1 + \lambda_2 \mathbf{n}_2 + \lambda_3 \mathbf{n}_3$$

The integrals are then composed into a sum of regularly spaced sample points over the triangles, giving:

$$E(S) \approx \sum_{\{\mathbf{v}_1,\mathbf{v}_2,\mathbf{v}_3\} \in \mathit{\Delta}} \sum_{\{\lambda_1,\lambda_2,\lambda_3\}} g(\lambda_1 \mathbf{v}_1 + \lambda_2 \mathbf{v}_2 + \lambda_3 \mathbf{v}_3, \lambda_1 \mathbf{n}_1 + \lambda_2 \mathbf{n}_2 + \lambda_3 \mathbf{n}_3) \quad (7)$$

The method of computing the error on sampling points within the triangles relates our work to other mesh based approaches [6, 19, 11, 12]. An alternative approach, used in the work of Duan et al. [16], is to sample the error on the tangent plane of the mesh vertices.

To recover the shape of textureless surfaces, only a small number of sample points may be necessary. In such a case, the simplest sampling scheme is to use only the vertices of the mesh as sample points. On the other hand, a sampling resolution matching the image resolution may be necessary for highly textured objects. As we would like our method to work on textured and textureless surfaces, we choose the dense sampling. The only downside of the increased sampling is a proportional increase in computation time and used memory.

One way to implement the gradient flow given by Eq. 6 is to derive the analytic gradient of g. But, there are several problems with the analytic gradient. First, the visibility changes are not taken into account. While moving vertices it is possible that some parts of the surrounding triangles become occluded or visible (un-occluded), which is not taken into account by the analytic gradient. A second remark is that the formulas do not account for reflectance changes as the reflectance properties could only computed after taking the step. Similar to the visibility case, moving a vertex results in changes in the shading. As an alternative we implemented a numerical computation for the gradient of the similarity function.

Numerical Gradient The gradient of the similarity function along the direction of the normal, $\nabla g \cdot \mathbf{n}$, is computed numerically using central differences. Letting $g_{\mathbf{v}^+}$ (resp. $g_{\mathbf{v}^-}$) be the error computed on the mesh when a vertex $\mathbf{v}$ is replaced with $\mathbf{v}^+ = \mathbf{v} + \mathbf{n}\Delta n$ (resp. $\mathbf{v}^- = \mathbf{v} - \mathbf{n}\Delta n$), then:

$$\nabla g \cdot \mathbf{n} \approx \frac{g_{\mathbf{v}^+} - g_{\mathbf{v}^-}}{2\Delta n}$$

where $\Delta n = c_\Delta \sigma_{mesh}$ and $c_\Delta \in (0, 1]$, to ensure that the derivative step size is bounded by the minimum edge length (a tuning parameter σ_{mesh} explained in Section 4.1).

In order to compute the gradient efficiently, without displacing each vertex individually and computing the error over the entire mesh, we compute the gradient for a set of vertices simultaneously [20]. The idea is to partition the mesh into disjoint sets of vertices, U_i, such that moving a vertex from a set does not influence the error for the rest of the vertices in that set:

$$\forall \mathbf{u}, \mathbf{v} \in U_i, \mathcal{N}(\mathbf{v}) \cap \mathcal{N}(\mathbf{u}) = \emptyset$$

For finding the set U_i we studied the influence a single vertex has on the mesh. We will assume that displacing a vertex a small amount does not affect the visibility of the mesh, so only local effects need be considered. When displacing a vertex $\mathbf{v}$ along its normal, the directly connected triangles are altered, and therefore their projection into the images is also altered. Also, since the normal at a vertex is computed using a weighted average of the triangle normals, the normals at the neighboring vertices are also affected. Therefore, all triangles within distance two are affected because the error is also dependent on the normal. Under this reasoning, the gradient computation for a vertex $\mathbf{v}$ must do the reflectance fitting and error computation for all triangles within distance 2 of the vertex.

3 Reflectance Fitting

As previously mentioned, we assume that the reflectance function is implied by the shape and imaging conditions. We experimented with two parametric reflectance models briefly introduced in Section 2.2 : Lambertian, for diffuse and Blinn-Phong for specular surfaces. We describe here how we practically recover the reflectance parameters from a set of images given a shape S, illumination conditions L_i and calibration parameters P_i.

3.1 Lambertian Reflectance

Lambertian reflectance has only one parameter per point $\mathbf{x}$ (the albedo $k_{d,\mathbf{x}}$). To compute the albedo for each point $\mathbf{x}$ on the mesh, with normal $\mathbf{n}$ we minimize the similarity error in each image:

$$g_{lamb}(\mathbf{x},\mathbf{n}) = \sum_i \langle \mathbf{n}, \mathbf{o}_i \rangle V'(\mathbf{x}, P_i) \left(I_i(\Pi(P_i\mathbf{x})) - (\langle \mathbf{n}, \mathbf{l}_i \rangle l_i + a_i) k_{d,\mathbf{x}} \right)^2 \quad (8)$$

The albedo that minimizes the weighted least squares problem in Eq. 8 is obtained as:

$$k_{d,\mathbf{x}} = \frac{\sum_i \langle \mathbf{n}, \mathbf{o}_i \rangle V'(\mathbf{x}, P_i) (\langle \mathbf{n}, \mathbf{l}_i \rangle l_i + a_i) I_i(\Pi(P_i\mathbf{x}))}{\sum_i \langle \mathbf{n}, \mathbf{o}_i \rangle V'(\mathbf{x}, P_i) (\langle \mathbf{n}, \mathbf{l}_i \rangle l_i + a_i)^2} \quad (9)$$

3.2 Specular Reflectance

The parameters of the specular reflectance can be estimated given a set of input images, an object surface, and illumination information, by minimizing the similarity measure (Eq. 2). For a low parameter BRDF model, as the Phong model, given enough observations, the parameters can be estimated efficiently using an indirect iterated linear approach [21] or by a more direct non-linear method [22].

In practice, with only a limited number of input images, it is not always possible to fit a full reflectance model at each surface point. Instead of fitting the full model at each surface point, we chose to use an interpolation method that first attempts to fit the Phong model to the observations at each point. A reliable fitting is only possible when a point has several observations with a small angle between the surface normal and bisector of viewing and illumination direction. If there are not enough observations, the specular parameters will not be estimated correctly, leaving only a correctly fit Lambertian model. These points are assigned the specular parameters of a point where the specular fitting was successful. This assignment is based on the diffuse color of the point.

3.3 Filtering Specular Highlights

In practice, it is inefficient to fit a full reflectance model to each surface point during the optimization. Instead of fitting the full reflectance model, we choose

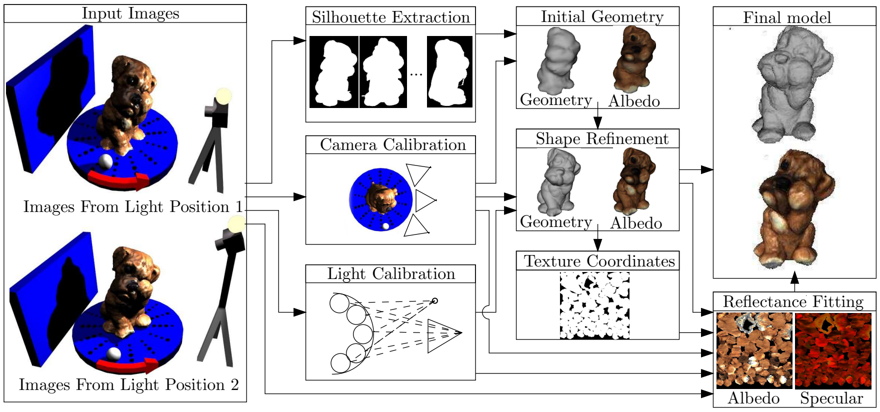

Fig. 3. Overview of the system used to scan objects

to filter out the specular highlights during the optimization and perform the shape refinement only for diffuse observations.

It is known that specular highlights occur at points having a large $\langle \mathbf{n}, \mathbf{h}_i \rangle$. As a consequence, one approach is to give smaller weights (in the h function) to those observations [23]. But, for a surface estimation method it is not the best approach as it relies on the current estimate of $\mathbf{n}$. Another approach, and the one used in this work, is to use the fact that specular highlights typically cause a bright image observation. Therefore, a fraction of the samples having the brightest intensity (typically 1/3) are excluded from the computation of the albedo and the g measure for a point. This type of filtering is essentially another binary function, like the visibility function V.

4 System and Implementation Details

Recall that our formulation of the shape refinement problem requires calibrated input images, a calibrated light source, and an initial shape. We use a turntable based capture setup as an easy way to capture many views of an object, while automatically providing light variation, and allowing for an initial shape to be computed from the object's silhouette.

Our particular capture setup consists of a single camera viewing an object rotating on a turntable (see Fig. 3 for an overview). Each set of images observes a full rotation of the object but has a different light position. In practice, the elevation of the light is varied between the two sets of images, and the light is positioned in a manner to avoid cast shadows (i.e., the source is placed close to the camera, implying that the camera also changes between the two sets of images).

The camera position is obtained through the automatic detection of a calibration pattern that is similar to the one used by Baumberg et al. [24]. A regular desk lamp is used as the light source and provides the majority of the illumination. The object rotates in front of a solid colored background, and a PCA based color segmentation is used to extract a set of silhouette images, which are used with shape from silhouette (SFS) to provide an initial shape.

The light source position and color are calibrated using a single glossy white sphere, which rotates along with the object on the turntable. Our approach is similar to other approaches that use a set of metallic spheres to calibrate a light source (e.g., [25]). The image of the specular highlight on the sphere in several views is used to triangulate the position of the source. As we used a white sphere, the non-specular pixels of the sphere are then used to calibrate the light source color.

In order to make the recovered model useful in computer graphics applications, the reflectance model is represented in texture maps. As a prerequisite, we first need to obtain texture coordinates for the refined model. We have implemented a method similar to that of Lévy et al. [26] to find these texture coordinates.

4.1 Overview of the shape refinement algorithm

The two components of the refinement in Eq. 6 are the gradient of the cost function and the regularizing component. The gradient is approximated per vertex using central differences, which was discussed in Section 2.4. The driving force behind the regularizing term is the mean curvature on the object, κ, which can be effectively approximated using a paraboloid method [27]. For a particular vertex, the mean curvature is computed by first finding the transformation taking the vertex to the origin and aligning its normal with the positive z axis. This transformation is applied to the neighboring vertices, and a paraboloid, $z = ax^2 + bxy + cy^2$, is then fit to the transformed points. The mean curvature at the vertex is $\kappa = a + c$.

Our mesh data structure is also capable of handling topological changes. In particular, we use the method proposed by Lachaud and Montanvert [28] to easily account for changes in the mesh topology. The mesh has a consistent global resolution, where edge lengths are confined to be within a certain range, i.e., if e is an edge in the mesh then $\sigma_{mesh} \leq \|e\| \leq 2.5\sigma_{mesh}$. A simple remesh operation ensures that the edges are indeed within this range and also performs the necessary operations related to topology changes. The global resolution of the mesh can be adjusted by altering this edge length parameter, σ_{mesh}.

The refinement starts with a low resolution mesh (i.e., large σ_{mesh}) and the corresponding low resolution images in a Gaussian pyramid. When the progress at a particular mesh resolution slows, the mesh resolution (and possibly the corresponding resolution in the Gaussian pyramid) is increased. This multi-resolution approach improves convergence, as there are fewer vertices (i.e., degrees of freedom), and enables the mesh to recover larger concavities.

5 Experiments

We have performed several experiments on synthetic and real image sequences to demonstrate the effectiveness of the method described in this paper. For the real sequences, the images were captured with either a consumer Canon Powershot

A85 digital camera or a Point Grey Research Scorpion firewire camera. In all the experiments, roughly 6 mesh resolutions were used during the refinement, and the total time for refinement was typically between 20 minutes and 1 hour. For the captures in this work, containing roughly 60 input images, we found $n_{cameras} = 12$ provided sufficient results for many of the sequences. In the final stages of the refinement this parameter was increased to 24.

The first experiment demonstrates the refinement of an object that a standard correlation based method would have problems with: a 3D printout of the Stanford bunny model, which has uniform Lambertian reflectance. Although an initial shape obtained from SFS is a good approximation to the bunny, several indentations near the legs of the bunny are not recovered (Fig. 4). These indentations are successfully recovered by our method, as illustrated by comparing the distance from the ground truth surface to the initial shape and the refined model (Fig. 5).

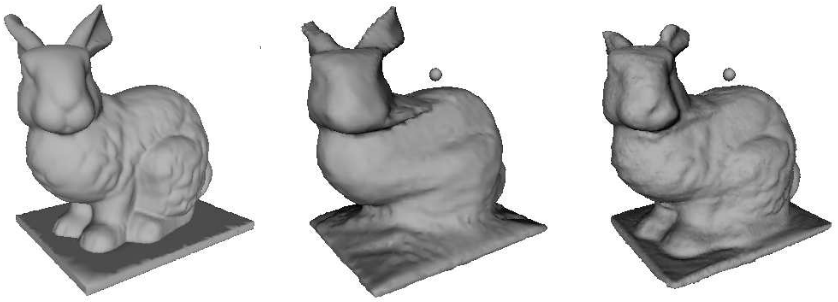

Fig. 4. From left to right a ground truth rendering, the recovered shape from SFS, and the refined model.

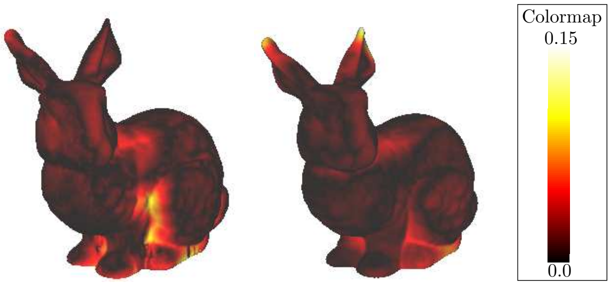

Fig. 5. An illustration of the distance from the ground truth object to the SFS model (left) and the refined model (right)

A second experiment, designed to test the effectiveness of the specular filtering, was performed on a synthetic object. The object has several concavities that were not reconstructed by the initial SFS shape (Fig. 6). The reconstruction obtained without specular filtering has artifacts. The most noticeable artifact is a sharp crease where the specularity was observed (second from the right of Fig.

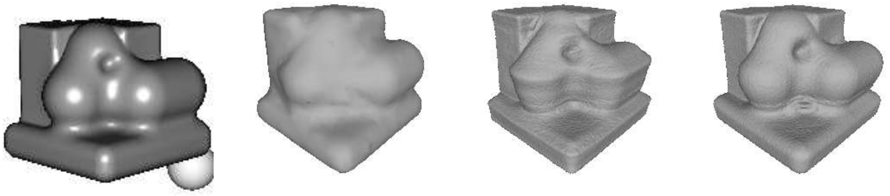

Fig. 6. From left to right, an input image of a synthetic specular object, the reconstruction from SFS, the reconstruction without specular filtering, and the reconstruction with specular filtering.

6). On the other hand, the refinement that used specular filtering successfully recovered the indentations.

We have also tested the method on several real objects with both textured and glossy surfaces (Fig. 7). Our method was capable of recovering an accurate geometry on all the objects. Notice the large concavity that was recovered in the house sequence. The fitted specular parameters give realistic highlights on the reconstructed results (see the sad dog and human head results). Unfortunately, the reconstructed specular component was not always as sharp as the true specular component, which is noticeable on the sad dog object (a similar observation was made by Yu et al. [12]).

Our high quality results are easily integrated into realistic computer graphics applications. To illustrate this, we have captured several real models of a chess game and combined them into a computer chess game (Fig. 8).

6 Discussion

We have presented a variational method that alternatively reconstructs shape and general reflectance from calibrated images under known light. The surface evolution is implemented on a deformable mesh at multiple resolutions. We have demonstrated the usefulness of the proposed method on controlled sequences, where an object was rotated relative to a light source. As demonstrated by the practical experiments, the results are quite accurate, proving that the method is able to reconstruct objects with large concavities and difficult specular reflections.

The capture setup used in this work provides an efficient way to capture a 3D model of an object, but currently we need to be able to rotate this object in front of the camera. As future work, we would like to extend our method to work on objects where this form of light variation cannot be obtained. For small outdoor statues, it may be sufficient to use the flash on the camera, or capture images on a sunny day at different times to obtain the light variation on the object. Of course a less restrictive method would be required for larger objects (e.g. buildings).

Other future directions include finding a more efficient way to utilize the information in specular highlights instead of filtering them out and to compare

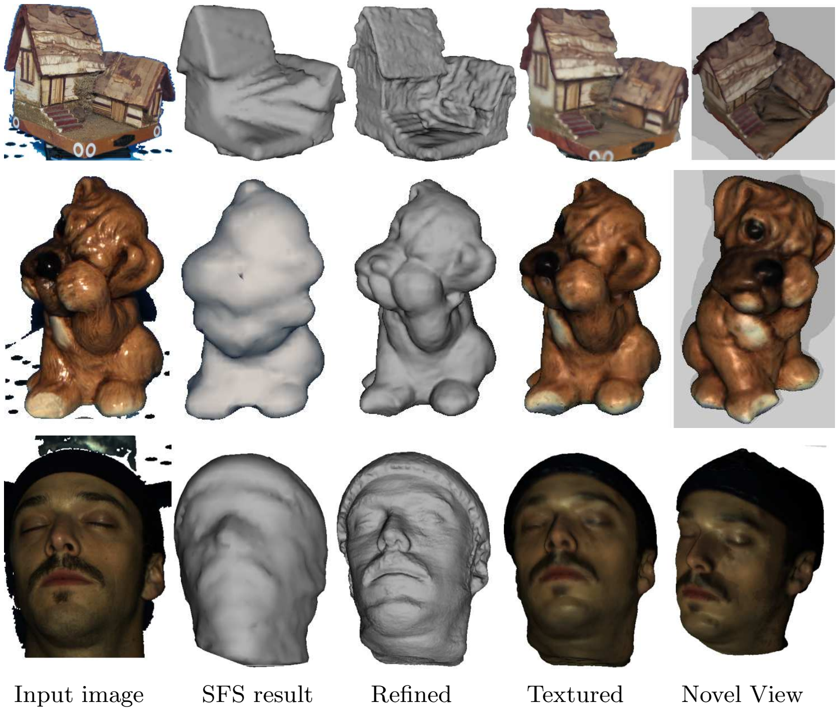

Fig. 7. Several reconstructed objects: a model house, a sad dog (see video: sad-dog_results.avi), and a real human head (see videos: face_results.avi and face_comp.avi)

the advantages of a level set implementation. We would also like to have some guarantee that the recovered surface is at (or at least near) a global minimum of the functional.

References

1. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vision **47** (2002) 7–42
2. Faugeras, O., Keriven, R.: Variational principles, surface evolution, pde's, level set methods and the stereo problem. IEEE Trans. Image Processing **7** (1998) 336–344
3. Robert, L., Deriche, R.: Dense depth map reconstruction: A minimization and regularization approach which preserves discontinuities. In: ECCV '96. (1996) 439–451
4. Yang, R., Pollefeys, M., Welch, G.: Dealing with textureless regions and specular highlights - a pregressive space carving scheme using a novel photo-consistency measure. In: ICCV. (2003)
5. Esteban, C.H., Schmitt, F.: Silhouette and stereo fusion for 3d object modeling. Computer Vision and Image Understanding **96** (2004) 367–392
6. Fua, P., Leclerc, Y.: Object-centered surface reconstruction: combining multi-image stereo shading. In: Image Understanding Workshop. (1993) 1097–1120
7. Jin, H., Yezzi, A., Soatto, S.: Stereoscopic shading: Integrating shape cues in a variational framework. In: CVPR. (2000) 169–176
8. Zhang, L., Curless, B., Hertzmann, A., Seitz, S.M.: Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multiview stereo. In: ICCV. (2003)

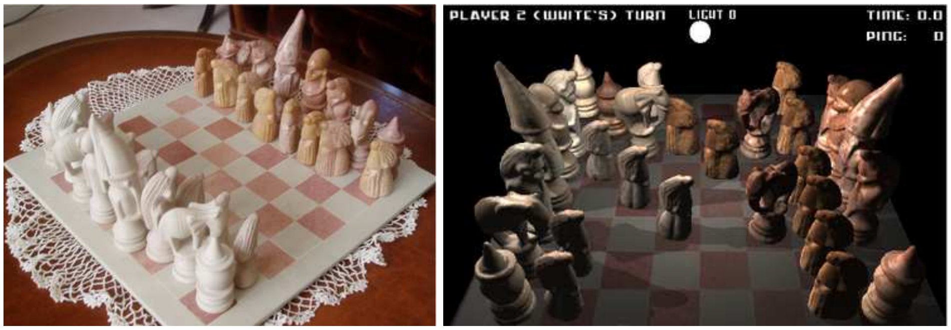

Fig. 8. An image of a real chess board (left), followed by a novel rendering of the captured models combined into a chess game (see video chess_results.avi).

9. Weber, M., Blake, A., Cipolla, R.: Towards a complete dense geometric and photometric reconstruction under varying pose and illumination. In: BMVC. (2002)
10. Zickler, T., Belhumeur, P.N., Kriegman, D.J.: Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. IJCV **49** (2003) 215–227
11. Yu, T., Xu, N., Ahuja, N.: Shape and view independent reflectance map from multiple views. In: ECCV. (2004)
12. Yu, T., Xu, N., Ahuja, N.: Recovering shape and reflectance model of non-lambertian. objects from multiple views. In: CVPR. (2004)
13. Debevec, P.E., Malik, J.: Recovering high dynamic range radiance maps from photographs. In: Siggraph. (1997)
14. Treuille, A., Hertzmann, A., Seitz, S.M.: Example-based stereo with general brdf. In: ECCV. (2004)
15. Sethian, J.: Level Set Methods. Cambridge University Press (1996)
16. Duan, Y., Yang, L., Qin, H., Samaras., D.: Shape reconstruction from 3d and 2d data using pde-based deformable surfaces. In: ECCV. (2004)
17. Blinn, J.F.: Models of light reflection for computer synthesized pictures. In: SIGGRAPH '77. (1977) 192–198
18. Caselles, V., Kimmel, R., Sapiro, G., Sbert, C.: Minimal surfaces based object segmentation. PAMI **19** (1997) 394–398
19. Zhang, L., Seitz, S.: Image-based multiresolution modeling by surface deformation. Technical Report CMU-RI-TR-00-07, Carnegie Mellon University (2000)
20. Zach, C., Klaus, A., Hadwiger, M., Karner, K.: Accurate dense stereo reconstruction using graphics hardware. In: Eurographics 2003. (2003) 227–234
21. Ikeuchi, K., Sato, K.: Determining reflectance properties of an object using range and brightness images. PAMI **13** (1991) 1139–1153
22. Lafortune, E.P., Foo, S.C., Torrance, K.E., Greenberg, D.P.: Non-linear approximation of reflectance functions. In: SIGGRAPH. (1997)
23. Marschner, S.R.: Inverse rendering for computer graphics. PhD thesis, Cornell University (1998)
24. Baumberg, A., Lyons, A., Taylor, R.: 3D S.O.M. - a commercial software solution to 3d scanning. In: Vision, Video, and Graphics (VVG'03). (2003) 41–48
25. Lensch, H.P.A., Goesele, M., Kautz, J., Heidrich, W., Seidel, H.P.: Image-based reconstruction of spatially varying materials. In: Eurographics Workshop on Rendering Techniques. (2001) 103–114
26. Lévy, B., Petitjean, S., Ray, N., Maillot, J.: Least squares conformal maps for automatic texture atlas generation. In: SIGGRAPH '02. (2002) 362–371
27. Surazhsky, T., Magid, E., Soldea, O., Elber, G., Rivlin, E.: A comparison of gaussian and mean curvatures triangular meshes. In: ICRA 2003. (2003) 1021–1026
28. Lachaud, J.O., Montanvert, A.: Deformable meshes with automated topology changes for coarse-to-fine 3D surface extraction. Medical Image Analysis **3** (1999) 187–207