Shape from Depth Discontinuities

Gabriel Taubin, Daniel Crispell, Douglas Lanman, Peter Sibley, and Yong Zhao

Division of Engineering, Brown University
Box D, Providence, RI 02912, USA
taubin@brown.edu
http://mesh.brown.edu

Abstract. We propose a new primal-dual framework for representation, capture, processing, and display of piecewise smooth surfaces, where the dual space is the space of oriented 3D lines, or *rays*, as opposed to the traditional dual space of planes. An image capture process detects points on a depth discontinuity sweep from a camera moving with respect to an object, or from a static camera and a moving object. A depth discontinuity sweep is a surface in dual space composed of the time-dependent family of depth discontinuity curves span as the camera pose describes a curved path in 3D space. Only part of this surface, which includes silhouettes, is visible and measurable from the camera. Locally convex points deep inside concavities can be estimated from the visible non-silhouette depth discontinuity points. Locally concave point laying at the bottom of concavities, which do not correspond to visible depth discontinuities, cannot be estimated, resulting in holes in the reconstructed surface. A first variational approach to fill the holes, based on fitting an implicit function to a reconstructed oriented point cloud, produces watertight models. We describe a first complete end-to-end system for acquiring models of shape and appearance. We use a single multi-flash camera and turntable for the data acquisition and represent the scanned objects as point clouds, with each point being described by a 3-D location, a surface normal, and a Phong appearance model.

Keywords: Multi-view reconstruction, appearance modeling, multi-flash, shape-from-silhouette.

1 Introduction

Because of the relative ease and robustness (particularly in controlled environments) of capturing object silhouettes, there exists a large body of work focused on reconstructing 3-D object shape based on silhouettes imaged from multiple viewpoints. All methods based purely on object silhouettes, however, face an inherent limitation: surface points which do not appear as part of the object silhouette from any viewpoint cannot be reconstructed. This limitation often leads to unsatisfactory results when the imaged objects contain details located within concavities that "protect" them from the occluding contour. Our method addresses this limitation by supplementing the silhouette information with additional depth discontinuity contours located on the object interior, providing a more complete and detailed reconstruction.

F. Nielsen (Ed.): ETVC 2008, LNCS 5416, pp. 216–237, 2009.

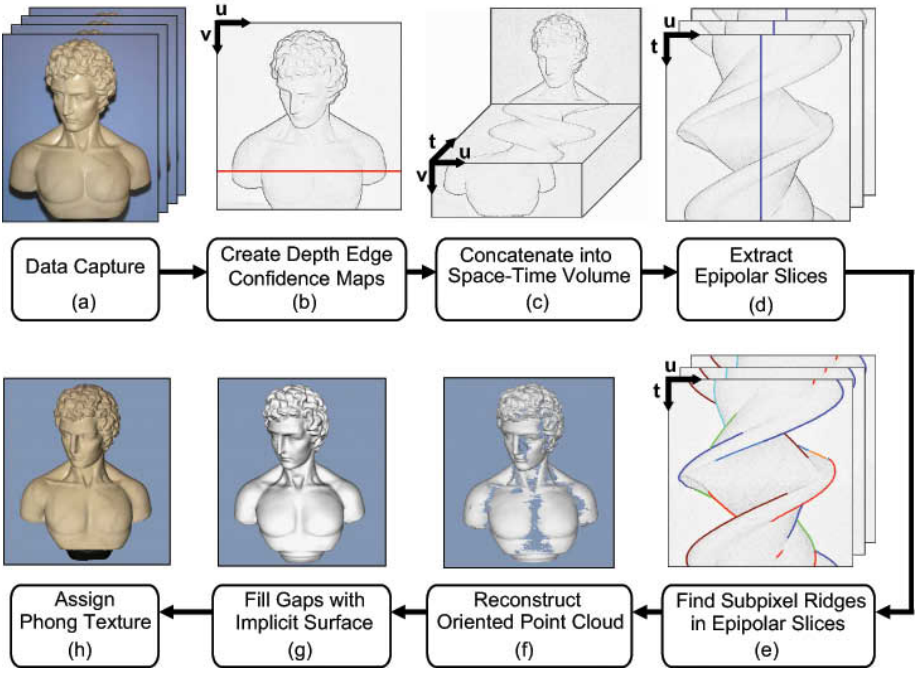


Fig. 1. The multi-flash 3-D photography pipeline. Data capture involves acquiring four images (using illumination from the top, left, bottom, and right) for each of 670 viewpoints of the object. Following data capture, a *depth edge confidence map* is estimated for each viewpoint. The confidence maps are concatenated to form a space-time volume. Each volume "slice" corresponding to an image scanline through all time is processed independently. After extracting subpixel ridges in the slices, differential reconstruction is applied to estimate an oriented point cloud. In order to fill sampling gaps, an implicit surface is fitted. Finally, for each point a Phong reflectance model (i.e., diffuse and specular colors) is estimated using 67 viewpoints.

We propose a new primal-dual framework for representation, capture, geometry processing, and display of piecewise smooth surfaces, with particular emphasis on implementing efficient digital data processing operations in dual space, and we describe our preliminary work based on multi-flash 3D photography [1,2] and vector field isosurface (VFIso) fitting to oriented point clouds [3].

Piecewise Smooth Surfaces: Piecewise smooth surfaces are a very popular way to describe the shape of solid objects, such as those that can be fabricated with machine tools. They are composed of smooth surface patches which meet along piecewise smooth patch boundary curves called feature lines. Across feature lines the vector field of surface normals can be discontinuous.

Surface Representations and Sampling: The family of piecewise smooth surfaces has infinite dimensionality. *Surface representations* with finite numbers of parameters must be used to operate on these surfaces in computers. Several popular surface

representations are in use: irregular polygon meshes, semi-regular subdivision surfaces, and disconnected point-sampled surfaces are some of them. The desired operations on surfaces must be translated into algorithms applicable on the corresponding surface representations. Since information such as surface normal discontinuities can be lost through the sampling processes which produce the surface representations for computer use, or just not explicitly representable, it is important to develop a theoretical framework to analyze and predict the behavior of different algorithms.

Depth Discontinuities: Current 3D shape measurement technologies based on triangulation capture points on smooth surface patches, but are unable to sample surface points along feature lines [4,5,6,7]. Several prior-art methods try to detect the feature lines lost in the point cloud obtained from one of these off-the-shelf sensors. We propose a new shape capture modality potentially able to directly detect feature lines. This capture process, which produces data complementary to triangulation based devices, is based on a new dual representation for piecewise smooth surfaces.

The Dual Space of Rays: The dual space considered here is the space of oriented lines in 3D, or *rays*

$$\{(q,v) : q,v \in \mathbb{R}^3, \|v\| = 1\} = \mathbb{R}^3 \times S^2$$

Points in this space correspond to rays defined in parametric form:

$$R_{qv} = \{p(\lambda) = q + \lambda\, v : \lambda \geq 0\}\,.$$

This space has been popularized by the image-based rendering literature: a light field [8] or lumigraph [9] is a function from the space of rays into the RGB color space. Image pixels correspond to points in $\mathbb{R}^3$ through the intrinsic equations of image formation which depend on the camera type, and the extrinsic camera pose. For an orthographic camera (which corresponds to a physical camera with a telecentric lens), the pixels correspond to a regular array of parallel rays; for a perspective (pinhole) camera, all the rays share a common origin: the optical center of the lens; catadioptric cameras may not have an optical center, and the mapping from pixels to rays may be more complex, as has been shown by many authors, including [10,11].

Representation of Surfaces in Dual Space: A smooth surface is represented as the set of all its tangent rays. This representation can be extended to piecewise smooth surfaces by considering the set of all its supporting rays (in the sense of convexity theory). We call this set the set of *depth discontinuities* of the surface. Note that locally concave points of the surface, deep inside concavities, do not correspond to visible depth discontinuities as seen from a camera located outside of the object bounded by the surface (Figure 2).

For example, let $S_F = \{p : f(p) = 0\} \subseteq \mathbb{R}^3$ be an implicit surface, with $f : \mathbb{R}^3 \to \mathbb{R}$ a smooth function which belongs to a family parameterized by a finite dimensional vector F (e.g. a polynomial of degree $\leq D$), and let $q \in \mathbb{R}^3$ be a point external to S_F. For every unit vector v we have a ray $R_{qv} = \{q + \lambda\, v : \lambda > 0\}$. The necessary and sufficient condition for the ray R_{qv} to be tangent to the surface S_F at some point is that:

$$\exists \lambda > 0 : \begin{cases} f(q + \lambda\, v) & = 0 \\ v^t \nabla f(q + \lambda\, v) & = 0 \end{cases} \tag{1}$$

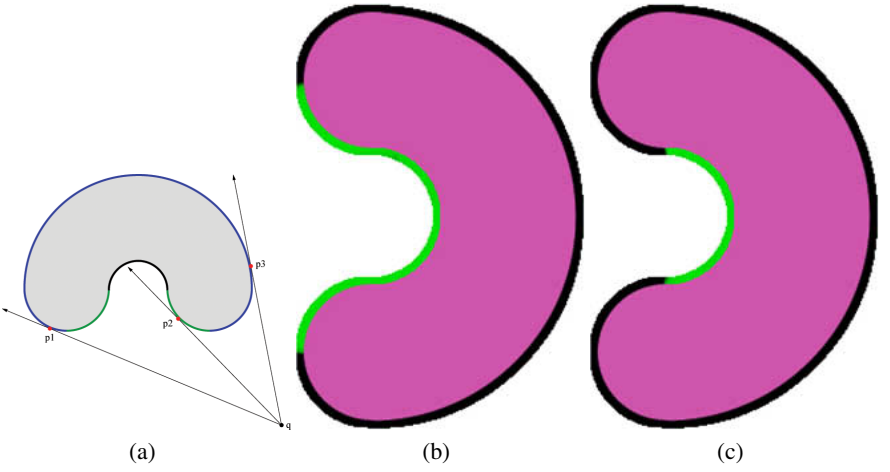


Fig. 2. (a) An object with concave surface points and the regions captured by: both depth discontinuity and silhouette-based reconstructions (blue), depth discontinuity-based reconstructions only (green), and neither (black). Points $p1$ and $p3$ are captured by both methods from camera position q, while $p2$ is only captured by depth discontinuity-based methods. (b) Visible silhouette points. (c) Visible depth discontinuity points.

Eliminating the variable λ from these two equations we obtain a single *resultant* equation

$$\phi_F(q,v) = 0 \tag{2}$$

which provides a necessary condition for tangency: in general if $\phi_F(q,v) = 0$ then the straight line supporting the ray is tangent to S at a point $p = q + \lambda v$, where the λ here is not necessarily positive (in which case the opposite ray satisfies the equation for positive λ because $q + \lambda v = q + (-\lambda)(-v)$). An expression for λ as a function of (F,q,v) is usually obtained as a byproduct of the elimination process, and can be used to determine the correct orientation for the ray. The set of depth discontinuities of the surface S_F is the set

$$\Phi_F = \{(q,v) : \phi_F(q,v) = 0\} \subseteq \mathbb{R}^3 \times S^2 \tag{3}$$

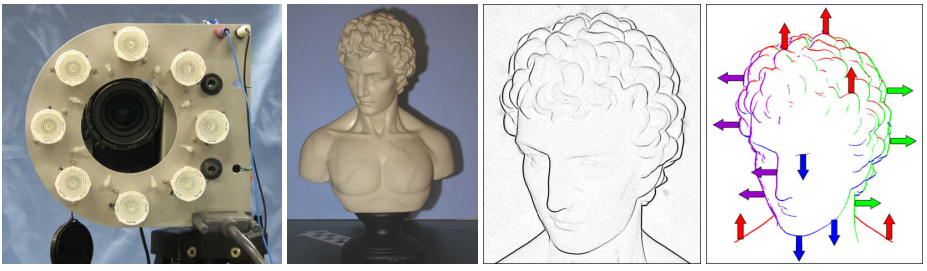
Most previous works based on duality (e.g. [12,13]) represent a smooth surface as the set of all its tangent planes.

Depth Discontinuity Sweeps: A *depth discontinuity sweep* is the time-dependent family of depth discontinuity curves span as the pose describes a curved path in 3D. This is a 2-surface in dual space, which typically includes self-intersections and cusps. For example, for a pinhole camera whose center of projection moves along a trajectory $q(\theta)$, corresponding to the points along a curve

$$C = \{q(\theta) : \theta \in \Theta \subseteq \mathbb{R}\}\,, \tag{4}$$

the corresponding depth discontinuity sweep is the set

$$\Phi_F^C = \{(q(\theta),v) : \theta \in \Theta\,,\ v \in S^2\,,\ \phi_F(q(\theta),v) = 0\}\,. \tag{5}$$



(a) (b) (c) (d)

Fig. 3. (a) Multi-flash camera. (b) Sample image acquired with flash located to the left of the camera's center of projection. (c) Depth edge confidence image produced by method in [14], with darker pixels representing a higher likelihood of a depth edge. (d) Approximate edge orientation corresponding to the flash with a maximum depth edge response. Up, down, left, and right edge orientations are shown in red, blue, purple, and green, respectively.

For a turntable sequence, the curve C is a circle of radius $r > 0$ in $\mathbb{R}^3$. As shown in figure 2, only part of depth discontinuity sweep is visible and measurable from a moving camera. Depth discontinuity pixels correspond to samples of the dual surface. The depth discontinuities visible from a particular camera pose are curves which include the silhouette visible from that pose, but convex points deep inside concavities can be estimated from the additional information, which is impossible just from silhouettes. Surface points laying at the bottom of concavities, however, do not correspond to depth discontinuities and cannot be measured, resulting in holes in the reconstructed surface. One of our future goals is to develop very efficient methods to fill these holes directly in dual space based on extrapolating the depth discontinuity curves to include the non-visible depth discontinuities. One method to fill these holes in primal space is described in section 4.5.

2 Multi-flash 3D Photography

We proceed to describe a first 3-D scanning system which exploits the depth discontinuity information captured by a multi-flash camera as an object being scanned is rotated on a turntable. Our method extends traditional shape-from-silhouette algorithms by utilizing the full set of visible depth discontinuities on the object surface. The resulting 3-D representation is an oriented point cloud which is, in general, unevenly sampled in primal space. We fit an implicit surface to the point cloud in order to generate additional points on the surface of the object in regions where sampling is sparse. Alternatively, the implicit surface can be regarded as the output of the reconstruction process. Finally, the appearance of each surface point is modeled by fitting a Phong reflectance model to the BRDF samples using the visibility information provided by the implicit surface. We present an overview of each step in the capture process and experimental results for a variety of scanned objects. The remainder of the article is structured as follows. In Section 3 we describe previous work related to both our reconstruction and appearance

modeling procedures. In Section 4 we describe in detail each stage of the reconstruction procedure, and discuss its inherent advantages and limitations in Section 5. In Section 6 we present results for a variety of scanned objects to demonstrate the accuracy and versatility of the proposed system. Finally, we conclude in Section 7.

3 Related Work

Our system draws upon several important works in both the surface reconstruction and appearance modeling fields of computer vision. We describe these works, their strengths and limitations, and how we extend and integrate them into our modeling system.

3.1 Surface Reconstruction

Surface reconstruction based on observing an object's silhouette as it undergoes motion has been extensively studied and is known broadly as *shape-from-silhouette* [15]. In general, shape-from-silhouette algorithms can be classified into two groups: those with volumetric, or global, approaches, and those which utilize differential, or local, information. Although our system falls under the category of the differential approach, we describe both here for completeness.

Space carving and visual hull algorithms [16] follow a global volumetric approach. A 3-D volume which completely encloses the object is defined, and the object is imaged from multiple viewpoints. The object silhouette is extracted in each of the images, and portions of the volume which project to locations outside of an object silhouette in any of the images are removed from the representation. Although robust, the quality of the results is somewhat limited, especially for complex objects containing concavities and curved surfaces.

An alternative differential approach uses the local deformation of the silhouettes as the camera moves relative to the object to estimate the depth of the points [17]. Related methods use a dual-space approach, where tangent planes to the object surface are represented as points in dual space, and surface estimates can be obtained by examining neighboring points in this space [18,19]. These systems provide a direct method for estimating depth based solely on a local region of camera motion, but are subject to singularities in degenerate cases. They also are not capable of modeling surface contours that do not appear as part of the object silhouette for any view, e.g. structures protected by concavities. Our method is similar in principle to these methods, but supplements the input silhouette information with all visible depth discontinuities. This extra information allows us to reconstruct structures protected by concavities that do not appear as part of the object silhouette in any view.

3.2 Multi-view Stereo Algorithms

In addition to purely silhouette-based approaches, multi-view stereo algorithms [20] are a class of hybrid approaches which combine image texture and color information with silhouette information [21,22,23]. These methods are capable of producing very accurate results, even recovering shape in areas protected by concavities. In most of

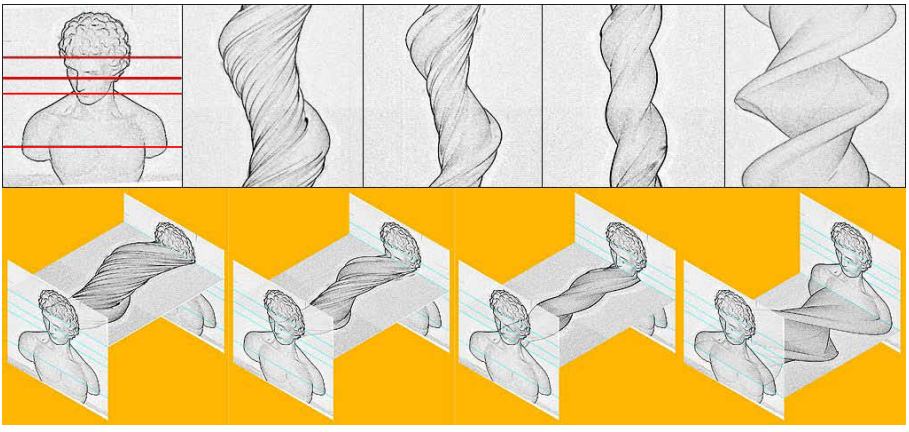


Fig. 4. In our multi-flash 3D photography system, depth edge confidence maps estimated for each viewpoint are concatenated to form a space-time volume, which is then sliced parallel to the image scan lines to produce *epipolar slices*

these algorithms the silhouette data is only used to construct an initial estimate of the visual hull surface represented as a polygon mesh, which is then iteratively deformed to minimize a properly formulated photo-consistency energy function. We look at these algorithms as operating mainly in primal space. Our system uses depth discontinuity information alone in order to produce the surface reconstruction. There is great potential to obtain more accurate surface reconstruction algorithms by combining multi-view stereo and depth discontinuities. We plan to follow this path in the near future. Again, what our multi-flash 3D photography algorithm shows is the 3D information contained only in the visible depth discontinuities.

3.3 Appearance Modeling

Appearance modeling has become an increasingly active area of research in both the computer vision and graphics communities. In [24], Lensch et al. introduced the notion of a *lumitexel*: a data structure composed of all available geometric and photometric information for a point on an object's surface. In addition, Lensch advocated lumitexel clustering to group similar surface components together and effectively increase the diversity of BRDF measurements. These methods were recently applied by Sadlo et al. to acquire point-based models using a structured light scanning system [25]. We apply a similar approach to assign a per-point reflectance model to the oriented point clouds obtained using our system.

4 System Architecture

The modeling system consists of a complete pipeline from data capture to appearance modeling (Figure 1). Here we describe the operation at each stage of the pipeline.

4.1 Data Capture

We use a turntable and stationary 8 megapixel digital camera to acquire data from up to 670 viewpoints in a circular path around the object (Figure 1(a)). We have constructed a camera rig similar to those used by Raskar et al. [14] consisting of eight 120 lumen LEDs positioned around the camera lens (Figure 3(a)) which are used as flashes. For each turntable position, we capture four images using illumination from the top, left, right, and bottom flashes, respectively. We have found that the four flashes positioned on the diagonals do not add a significant amount of extra information and are therefore not used in our experiments. The camera is intrinsically calibrated using Bouguet's camera calibration toolbox [26], and its position and orientation with respect to the turntable are determined using a calibration grid placed on the table. Once the data has been captured, we rectify each of the images to remove any radial distortion, and to align the camera's u axis with the direction of camera motion (i.e. perpendicular to the turntable axis of rotation and with zero translation in the u direction as shown in Figure 1(b)).

4.2 Depth Edge Estimation

Using the four images captured with different illumination at each turntable position, we are able to robustly compute depth edges in the images (Figure 1(b)) using the algorithms introduced by Raskar et al. [14] for non-photorealistic rendering. The distances between the camera center and the four flashes are small compared with the distance to the scene, so a narrow shadow can be observed adjacent to each depth discontinuity (Figure 3(b)) in at least one of the four images. As presented in [14], a simple method exists to extract both the position and orientation of the depth edges using the information encoded in these shadows. First, a *maximum composite* is formed by taking the largest intensity observed in each pixel over the multi-flash sequence. In general, this composite should be free of shadows created by the flashes. In order to amplify the shadowed pixels in each flash image (and attenuate texture edges), a *ratio image* is formed by dividing (per pixel) each flash image by the maximum composite. Afterwards, the depth edges can be detected by searching for negative transitions along the direction from the flash to the camera center (projected into the image plane) in each ratio image. With a sufficient distribution of flash positions and under some limiting assumptions on the baseline and material properties of the surface [14], this procedure will estimate a considerable subset of all depth discontinuities in the scene. A *depth edge confidence image* corresponding to the likelihood of a pixel being located near a depth discontinuity (see Figure 3(c)) is produced for each of the 670 turntable positions. Images encoding the flash positions which generated the greatest per-pixel responses are also stored in order to facilitate surface normal estimation in the reconstruction stage. By dividing the high resolution images between a cluster of 15 processors, we are able to complete the depth edge estimation for all 670 positions in under one hour.

4.3 Extracting Curves in Epipolar Slices

The *epipolar parameterization* for curved surfaces has been extensively studied in the past [27,17]. For two cameras with centers q_1 and q_2, an epipolar plane is defined as the

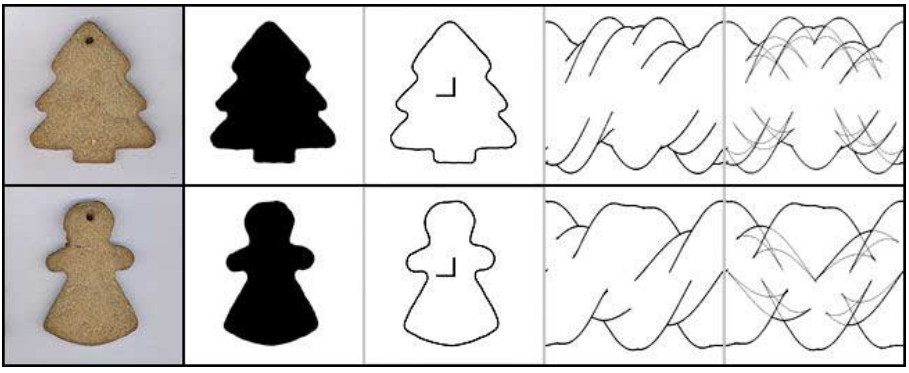


Fig. 5. Simulated orthographic epipolar slices showing invisible depth discontinuities

plane containing q_1, q_2, and a world point X being imaged. The epipolar planes slice the image planes, forming a pencil of *epipolar lines* in each image, and each point in one image corresponds to an epipolar line in another. A point x_1 along an apparent contour in one image is therefore matched to a point x_2 in the second image by intersecting the epipolar line defined by q_1,q_2, and x_1 with the corresponding apparent contour in the second image. For a continuous path of camera centers, $q(t)$, an epipolar plane at time t is spanned by the tangent vector $\dot{q}(t)$ to $q(t)$ and a viewing ray $r(t)$ from $q(t)$ to a world point p. So called *frontier points* occur when the epipolar plane is identical to the tangent plane of the surface.

Because we have rectified each input image so that the camera motion is parallel to the image u axis (Section 4.1), the depth edge confidence images exhibit the same property. By stacking the sequence of confidence images (Figure 1(c)) and "slicing" across a single scanline, we have an approximation to the epipolar constraint in local regions. We refer to these images containing a particular scanline from each image as *epipolar slices* (Figures 1(d) and 4). By tracking the motion of apparent contours in the slices, we are in effect implicitly utilizing the epipolar constraint for curve matching. The tracking problem can be solved using a form of edge following optimized to take advantage of properties of the slice images. The curve extraction stage is decomposed into three substages: subpixel edge detection, edge linking, and polynomial curve fitting. Although nearby slice images are strongly correlated, we treat them as independent in order to facilitate parallel processing. However, the inherent correlation between epipolar slices is exploited in the extraction of surface normals as described in section 4.4. So in fact, each estimated 3D point is a function of a 3D neighborhood of the corresponding depth discontinuity point in dual space.

Edge Detection. We begin by detecting the pixel-level position of the depth discontinuities by applying a two-level hysteresis threshold. Afterward, we estimate the subpixel position of each depth discontinuity by fitting a sixth order polynomial to the neighboring confidence values. Non-maximum suppression is applied to ensure that a single subpixel position is assigned to each depth edge.

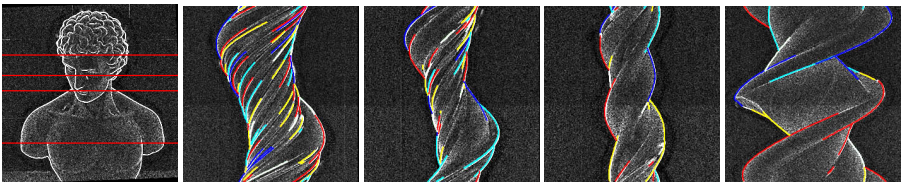


Fig. 6. Epipolar slice curve tracking and fitting

Edge Linking. As shown in Figures 1(d) and 5, the epipolar slices are complex and typically contain many junctions, indicating points of bi-tangency. These junctions emerge for a variety of reasons, including when external silhouettes becomes internal contours (and vice versa). Our edge linking algorithm follows edges through such transitions. We initialize the tracking process by finding the first detection to the left of the axis of rotation in an epipolar slice. Next, we search for the closest detection in the neighboring views within a small window. If any match is found, then we initiate a track using a linear prediction based on these two observations. We proceed to search for new detections within a neighborhood of the predicted edge position. The closest detection (if any) to the prediction is added to the track and neighboring detections are removed from future consideration. Once three or more detections have been linked, we predict the next position using a quadratic model. If a track ends, a new edge chain is initiated using the first available detection either to the left or right of the axis of rotation. This process continues until all detections have been considered. While simple, this tracking method consistently and accurately links depth discontinuities through junctions.

Curve Fitting. Once the subpixel detections have been linked, a sixth order polynomial is fit to each chain – providing an analytic model for the motion of depth discontinuities as a function of viewpoint. Sixth order polynomials were chosen because of their tendency to fit the chain points with low error, and no over-fitting in practice. RMS errors for the polynomial fits vary depending on the length and curvature of the chain, but are generally on the order of one pixel. Typical results achieved using this method are shown in Figure 1(e) and 6.

4.4 Point Cloud Generation

Once curves in the epipolar slice domain have been extracted, we can directly estimate the depth of the points on these curves and produce a point cloud representation of the object (Figure 1(f)).

The properties of surface shapes based on the apparent motion of their contours in images are well-studied [27,17]. In general, we represent a surface point p on a depth discontinuity edge as

$$p = q + \lambda r \tag{6}$$

where q is the camera center, r is the camera ray vector corresponding to a pixel $[u, v]$, and λ is the scaling factor that determines the depth. Cipolla and Giblin [17] showed that the parameter λ can be obtained from the following equation

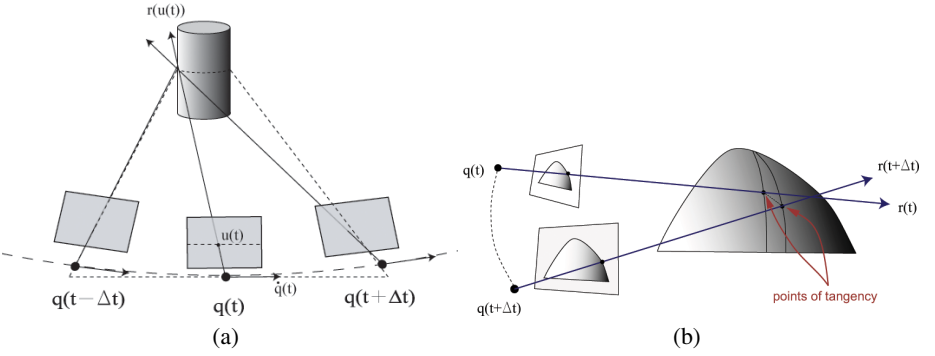


Fig. 7. (a) The epipolar plane (dotted line) used for curve parametrization is spanned by the viewing ray, r, and the camera's velocity vector, $\dot{q}$. The images are rectified such that the epipolar lines correspond to scan lines in the image. Unless the camera motion is linear, this plane is only an approximation for finite Δt, since the neighboring camera centers are, in general, not contained in the plane. (b) The tangent ray from the camera to the object slides over the surface as the camera moves. Depth can be estimated based on the apparent motion of the contour in the image plane relative to the camera motion in space.

$$\lambda = -\frac{n^t \dot{q}}{n^t \dot{r}} \tag{7}$$

where n is the normal vector to the surface at the point p, and $\dot{r}, \dot{q}$ are derivatives in time as the the camera moves with respect to the object and the camera ray r "slides over" the object (Figure 7-(b)). This method assumes that the functions $q(t)$, $r(t)$, and $n(t)$, as well as their derivatives with respect to t are known. The epipolar parametrization is then used to construct these curves from multiple silhouettes. Because the camera motion $q(t)$ is known from calibration, we effectively recover the function $r(t)$ by fitting analytic models to the curves in the epipolar slice images. For a given epipolar slice image, we have constant $v = v_s$ and image axes corresponding to u and t, where, for a given contour, u is function of t. We therefore express Equation 6 as:

$$p(u(t),t) = q(t) + \lambda\, r(u(t),t) \tag{8}$$

and Equation 7 as

$$\lambda = -\frac{n(u(t),t)^t\ \dot{q}(t)}{n(u(t),t)^t\ \frac{d}{dt}\{r(u(t),t)\}} \tag{9}$$

where

$$\frac{d}{dt}\{r(u(t),t)\} = \frac{\partial r}{\partial u}(u(t),t)\,\dot{u}(t)\ . \tag{10}$$

We use the standard pinhole camera model with projection matrix

$$P = K\begin{bmatrix} I & 0 \end{bmatrix}\begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \tag{11}$$

where R is a 3x3 rotation matrix and T is a 3x1 translation vector relating the world coordinate frame to that of the camera. K is a 3x3 matrix containing the camera's intrinsic

projection parameters. We recover these parameters along with 5 radial and tangential distortion coefficients using Bouguet's camera calibration toolbox [26]. We project image points in homogeneous coordinates to vectors in world space using the "inverse" projection matrix, $\hat{P}$.

$$\hat{P} = \begin{bmatrix} R^t & -R^t T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I \\ 0 \end{bmatrix} K^{-1} \tag{12}$$

The function $\frac{\partial r}{\partial u}(u(t),t)$ can then be calculated from the inverse projection matrix (Equation 12) associated with camera position $q(t)$:

$$\frac{\partial r}{\partial u}(u(t)) = \begin{bmatrix} \hat{P}_{1,1}(t) \\ \hat{P}_{2,1}(t) \\ \hat{P}_{3,1}(t) \end{bmatrix} \tag{13}$$

The contour path's motion in the u direction, $\dot{u}(t)$, can be obtained directly from the coefficients of the curve fit to the contour path (Section 4.3) in the slice image. We estimate the image normal $m(u(t),t)$ by performing principal component analysis (PCA) on a local region about the point $(u(t),v_s)$ in the original depth edge image corresponding to time t. There exists a sign ambiguity in this normal computation, so we compare m with the coarse normal information given by the flash with the maximum depth edge response (Section 4.2) and flip its direction as needed. The surface normal $n(u(t),t)$ in 3-D must then be perpendicular to the viewing ray $r(u(t),t)$, and contained in the plane spanned by $r(u(t),t)$ and the projection of $n(u(t),t)$ onto the image plane, $m(u(t),t)$.

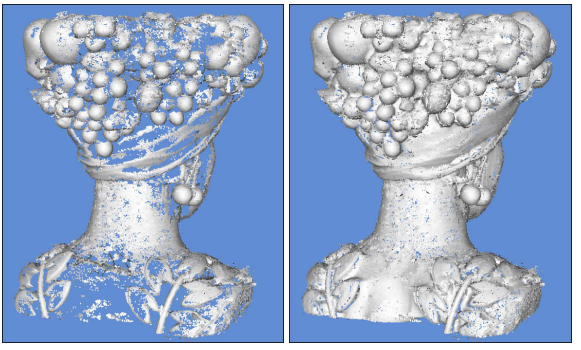
$$n(u(t),t) = (\hat{P}(t) \begin{bmatrix} m(u(t),t) \\ 0 \end{bmatrix} \times r(u(t),t)) \times r(u(t),t) \tag{14}$$

Substituting back in to Equation 9, we can now recover the depth of any point on the contour path, assuming known camera motion $\dot{q}(t)$. In our experiments, we dealt with the simple case of circular motion, so $\dot{q}(t)$ is well defined for all t.

Again dividing the computations between 15 processors, the curve extraction and depth estimation procedures take on the order of 20 minutes for our data sets.

4.5 Hole Filling

Each curve in each slice is processed independently, and sampled uniformly in t. This sampling in t causes the reconstructed points to be sampled very densely in areas of high curvature (since the viewing ray moves slowly over these regions) and conversely, very sparsely in areas of very low curvature, e.g. planes. The effects of this non-uniform sampling can be seen in Figure 1(f) in the form of gaps in the point cloud. Several approaches have been developed for resampling and filling holes in point clouds. Moving Least Square surfaces [28] provide resampling and filtering operations in terms of local projection operations, however these methods are not well-suited for filling large holes. Diffusion-based methods for meshes [29] and point clouds [30] have also been developed. As an alternative to these advanced methods, a standard approach is to fit an implicit surface or polygonal mesh to the point cloud and subsequently display this representation using the conventional graphics modeling and rendering pipeline.



(a) before resampling (b) after resampling

Fig. 8. Example of filling sampling gaps using the implicit surface as described in Section 4.5

We use a variant of this approach proposed by Sibley and Taubin [31] since we require both an intermediate surface for visibility computations as well as a method for introducing samples in regions that were not acquired using the multi-flash reconstruction (e.g., those shown in Figures 1(g) and 8). The surface fitting reduces to solving a linear least squares problem, and proceeds as follows: Given an oriented point cloud $\mathscr{D} = \{(p_1, n_1), \ldots, (p_m, n_m)\}$ sampled from a surface M, the method computes an implicit surface $M' = \{p | f(p) = 0\}$ where $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ is a scalar function, such that ideally $\nabla f(p_i) = n_i$, and $f(p_i) = 0$. If p_α denotes the position of a grid node, the problem reduces to the minimization of the following quadratic energy

$$E = \sum_i f(p_i)^2 + \mu \sum_i \|\nabla f(p_i) - n_i\|^2 + \lambda \sum_{(\alpha,\beta)} \|\nabla f(p_\alpha) - \nabla f(p_\beta)\|^2 \qquad (15)$$

where (α, β) are edges of the grid, and $\mu, \lambda > 0$ are a regularization constant. The scalar field f is represented as a linear combination of basis functions (e.g., trilinear) defined on a uniform Cartesian grid, $f(p) = \sum_\alpha f_\alpha \phi_\alpha(p)$, where $f_\alpha = f(p_\alpha)$. The gradient is approximated with finite differences.

Afterwards, we extract a triangular mesh with Marching Cubes (as shown in Figure 1(g)), and use it to resample the surface in regions where the original sampling was sparse.

Of course, since no information is captured from the invisible depth discontinuity points, the locally concave points at the bottom of concave areas are only hallucinated by this algorithm. Figure 9 is a simple illustration of some of the variability encountered in practice. The five shapes in this figure have identical visible depth discontinuities. The reconstruction produced by our algorithm is most probably close to the fourth example because the third terms in our energy function tends to minimize the variation of the function gradient, i.e., of the surface normal. So, holes tend to be filled with patches of relatively constant curvature. Additional primal space information, such as from triangulation-based sensors or multi-view stereo photometric information, is needed to differentiate amongst these shapes and to produce a more accurate reconstruction. In our view, the multi-view stereo approach, which is based on a similar variational formulation, seems to be the simplest to integrate with our system, as we already capture

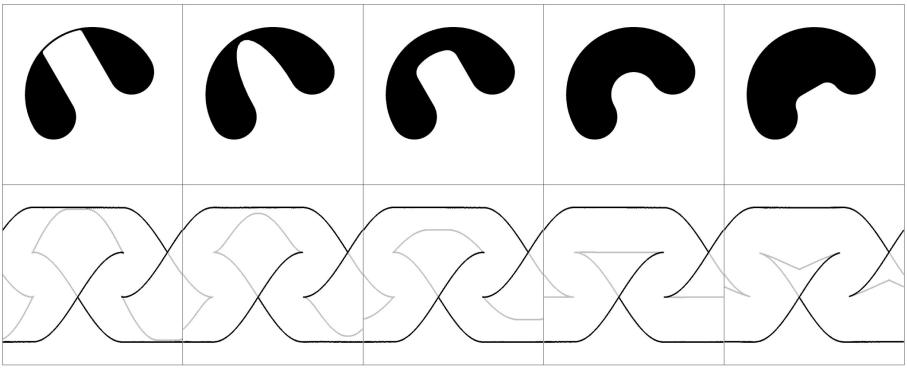


Fig. 9. Different shapes that produce the same visible depth discontinuity epipolar slices. Note the variability in the shape and location of the curves corresponding to invisible depth discontinuity points.

the necessary photometric information (currently ignored). As we mentioned before, we plan to explore these ideas.

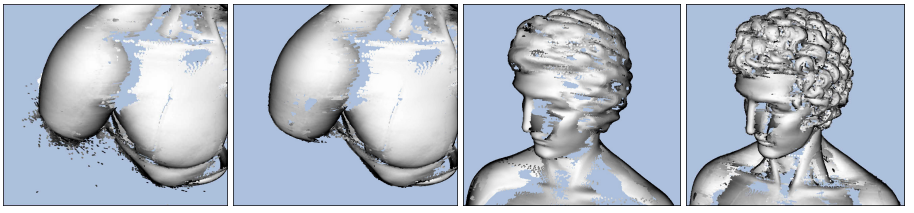
4.6 Appearance Modeling

As shown in Figures 1(g) and 8(b), the output of the gap-filling stage is a dense oriented point cloud. Given this representation of the surface shape, we assign a per-point appearance model using a subset of 67 images acquired from the turntable sequence. Note that, despite the relatively large number of available viewpoints, the BRDF remains sparsely-sampled since the illumination sources and camera are nearly coincident. As a result, we simply fit a Phong reflection model to the set of reflectance observations at each point. For simplicity, we assume that the surface does not exhibit significant subsurface scattering or transparency and can be represented by a linear combination of a diffuse term and a specular reflection lobe as described in the Phong model.

We begin the appearance modeling process by extracting a set of color observations for each point by back-projecting into each image. In order to determine the visibility of a point $p = q + \lambda r$, where q is the camera's center of projection, we perform a ray-mesh intersection test with the triangulated implicit surface. The first point of intersection is given by $p' = q + \lambda' r$. If p is outside some displacement ε from p' or if the point is facing away from the camera, then we mark the point as invisible, otherwise the color of the corresponding pixel is assigned to the observation table. Note that, unlike similar texture assignment methods such as [25], we can detect (or remove) shadows automatically using the *maximum composite* of the four images (described in Section 4.2) before assigning a color to the observation table. As a result, the combination of the implicit surface visibility test and the shadow removal afforded by the multi-flash system minimizes the set of erroneous color observations.

As described by Lensch et al. [24], we obtain a *lumitexel* representation for each point (i.e., a set color observations). We apply the Phong reflection model given by

$$I_\lambda = k_{a\lambda} + k_{d\lambda}\mathbf{n}\cdot\mathbf{l} + k_{s\lambda}(\mathbf{r}\cdot\mathbf{v})^n \tag{16}$$



(a) (b) (c) (d)

Fig. 10. (a) A portion of the bust point cloud, generated with no outlier rejection. An area of instability can be seen under the arm, where the surface is nearly perpendicular with the axis of rotation. (b) Outliers removed by back-projection validation using a small set of segmented images. (c) The generated point cloud using our algorithm with silhouette information only. (d) Reconstruction using all depth discontinuities. Notice increased detail in the eyes, hair, and neck concavities.

where I_λ is the wavelength-dependent irrandiance and $\{k_{a\lambda}, k_{d\lambda}, k_{s\lambda}, n\}$ are the ambient, diffuse, and specular coefficients, and the specular exponent, respectively. In this equation, the directions to the light source and camera are given by $\mathbf{l}$ and $\mathbf{v}$, whereas the direction of the peak of the specular reflection lobe is given by $\mathbf{r}$ and the surface normal is $\mathbf{n}$. Given the small baseline between the camera's center of projection and the flashes, we make the simplifying assumption that the flashes are coincident with the camera center (such that $\mathbf{l} = \mathbf{v}$).

We fit the per-point Phong reflectance model independently in each color channel. Following a similar approach as [25], we estimate the model parameters by applying Levenberg-Marquart nonlinear optimization. When insufficient data is available to fit the specular reflection component, we only estimate the diffuse albedo. Typical appearance modeling results are shown in Figure 12, where (c) and (d) illustrate typical diffuse and specular reconstructions, respectively. Note that, experimentally, we found that the *median diffuse albedo* (given by the median of the *lumitexel* values) was a computationally-efficient and visually-plausible substitute for the diffuse component of the Phong model. For applications in which only the diffuse albedo is required, the median diffuse albedo eliminates the need for applying a costly nonlinear estimation routine.

5 Analysis of Reconstruction Algorithm

5.1 Stability

One drawback to using Equation 7 to estimate depth is its dependence on usually noisy derivatives. In fact, in a previous implementation we used first order difference operators to estimate the derivatives and observed noisy and unstable depth estimates. By fitting polynomial curves to the contour samples in the epipolar slices, we essentially average out the noise and obtain accurate and stable derivative measurements as shown in our results.

A second drawback of our reconstruction algorithm is its ill-conditioning close to frontier points, where $n(t)^t \dot{r}(t) \approx 0$. In these cases, the denominator of Equation 7

approaches zero, causing unreliable depth estimates. Giblin and Weiss [27] have presented an alternate expression for depth that avoids this mathematical instability, but in our experiments the depth estimates remained unstable at frontier points. This is most likely due to the imprecision of matching when the epipolar lines are tangent to the surface contours. We combat this ill-conditioning in two ways. First, we reject reconstructed points with an infinitesimally small $n(t)^t \dot{r}(t)$ value (i.e. frontier points) outright, since they rarely provide meaningful reconstructions. Second, we deal with instability in the regions near these frontier points by performing the simple validation proposed by Liang and Wong [19]. We segment the object from the background in a small subset (15 views) of the original input images. We then back-project the reconstructed points into the images, making sure that each point lies within the image foreground. For the bust data set, 3.7% of points were removed in this way (Figure 10-(a,b)). One drawback of this approach is that points which are incorrectly reconstructed "inside" of the surface are not removed.

5.2 Surface Coverage

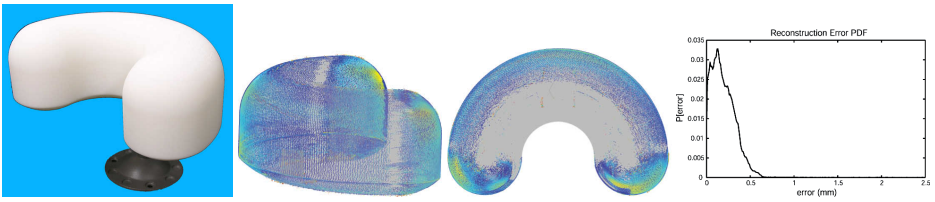
One key contribution of our reconstruction system is the use of the additional information provided by the full set of observable depth discontinuities. A typical example of the additional surface information that can be extracted can be seen in Figure 10-(c,d). Structure located in the concavities of the hair, eyes, and neck is successfully captured, while lacking in the silhouette-based reconstruction. Although a significant improvement in surface coverage can be seen, locally concave regions which do not produce visible depth discontinuities are not captured. Figure 2 demonstrates the theoretical limit of surface regions that can and cannot be captured with the two approaches.

6 Experimental Results

As presented, the multi-flash 3-D photography system represents a new, self-contained method for acquiring point-based models of both shape and appearance. In this section, we first present the reconstruction results for a 3-D test object with known dimensions in order to assess the system's accuracy. We then discuss (qualitatively) the reconstructions obtained for a variety of other physical objects in order to explore the system's versatility.

6.1 System Accuracy

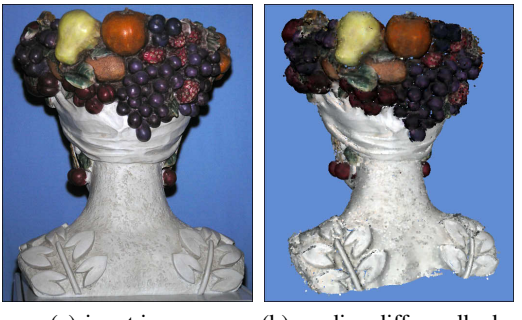
In order to experimentally verify the accuracy of the system, we designed and manufactured a test object using the SolidWorks 3-D modeling program and a rapid prototyping machine. The rapid prototyping machine specifications indicate that it is accurate to within 0.1 mm. We designed an object roughly in the shape of a half-torus, with varying curvature at different points on the surface (Figure 11-(a)). We then reconstructed a point cloud of the the model using the algorithm described in Sections 4.1 through 4.4, and aligned it with a rigid transformation to the original SolidWorks mesh using ICP. No segmentation-based outlier detection or surface fitting were used. Figure 11-(b,c) shows the aligned reconstructed point cloud, with points color-coded according



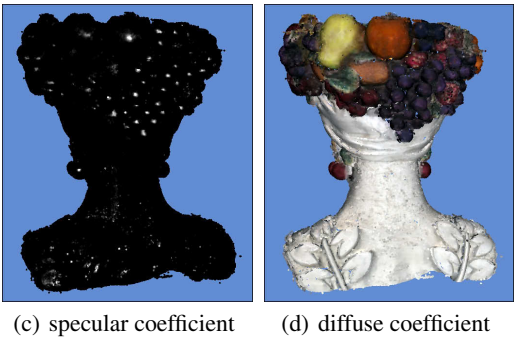
(a) (b) (c) (d)

Fig. 11. (a) The manufactured test object. (b) Reconstructed point cloud overlayed on original mesh. (c) The reconstructed points are color-coded according to their absolute reconstruction error in mm. (d) The probability distribution of the point cloud reconstruction errors.

to their absolute distance from the original mesh. As expected, concave surface points are not reconstructed, nor are regions close to frontier points. Scanning the object using multiple camera paths and merging the reconstructions could alleviate this deficiency. Figure 11-(d) shows the distribution of the reconstruction errors. Roughly 9% of the points had error greater than 2.5 mm and were considered outliers. These points were mainly due to reconstruction of surfaces not part of the CAD model, such as the base



(a) input image (b) median diffuse albedo



(c) specular coefficient (d) diffuse coefficient

(e) Phong appearance model

Fig. 12. Estimated appearance for "woman with fruit basket" using 67 images. Note how the *median diffuse albedo* provides a computationally-efficient and visually-plausible substitute for the diffuse component of the Phong model.

Fig. 13. Summary of reconstruction results. From left to right on each row: an input image, the reconstruction viewed under similar illumination conditions, another input image, and a corresponding view of the model. The first through fourth rows show the "woman with fruit basket", "bust", "pig chef", and "hand" models, respectively. Each model is represented by approximately $1,000,000$ points and was processed using a polygonal implicit surface with about $250,000$ faces. Note that, for the hand model, both the diffuse wood grain and highlights were reliably reconstructed. Similarly, the detailed geometric and color structure of the "women with fruit basket" were also captured.

used to hold the object. Disregarding the outliers, the mean reconstruction error was 0.20 mm, with a standard deviation of 0.16 mm. These results are very promising and suggest that the accuracy of the system is on par with commercially available scanning systems.

6.2 System Versatility

As shown in Figure 13, four objects were acquired with varying geometric and material complexities (from the top to bottom row: "woman with fruit basket", "bust", "pig chef", and "hand" models).

The "pig chef" model (shown on the third row of Figure 13) demonstrates low geometric complexity (i.e. its surface is nearly cylindrical with few self-occlusions). Similarly, its material properties are fairly benign – most of the surface is composed of a diffuse material and specularities are isolated to the jacket buttons and the spoon in the left arm). As with laser scanners or other active illumination systems, we find that highly specular surfaces cannot be reconstructed reliably. For example, consider the geometric and material modeling errors produced by the highly specular spoon in the left arm. Future work will examine methods to mitigate these errors, such as those presented in [32].

The "hand" model (shown on the last row of Figure 13) is challenging due to multiple self-occlusions and the moderately specular wood grain. For this example, we find the multi-flash approach has successfully reconstructed the fingers – regions that could not be reconstructed reliably using existing shape-from-silhouette or visual hull algorithms. In addition, the specular appearance was modeled in a visually-acceptable manner using the Phong appearance model. Note the "bust" model (shown on the second row of Figure 13) demonstrates similar self-occlusions in the hair and was also reconstructed successfully.

The "woman with fruit basket" model (shown on the first row of Figure 13) represents both material and geometric complexity with multiple self-occlusions and regions of greatly-varying material properties. As with other examples, we find the multi-flash approach has achieved a qualitatively acceptable model which accurately captures the surface shape and appearance of the original object.

7 Conclusions

We have presented in this article a fully self-contained system for acquiring point-based models of both shape and appearance using multi-flash photography. As demonstrated by the experimental results in Section 6, the proposed method accurately reconstructs points on objects with complex features, including those located within concavities. The geometric reconstruction algorithm is direct and does not require solving any non-linear optimization problems. In addition, the implicit surface fitted to the oriented point cloud provides an efficient proxy for filling holes in the surface, as well as determining the visibility of points. Finally, recent work in appearance modeling has been extended to the specific problem of texturing multi-flash image sequences.

While current results demonstrate the significant potential of this approach, we believe that the greatest benefit of multi-flash 3-D photography will be achieved by combining it with existing methods for shape recovery (e.g. laser scanners and structured light systems). These systems provide an efficient means to reconstruct regions of low-curvature, whereas the multi-flash reconstruction accurately models high-curvature regions and points of bi-tangency where these approaches have difficulties. Future work will explore the synergistic combination with existing approaches, especially with regard to planning optimal viewpoints for 3-D scanning.

7.1 Future Work

While sampling is regular for triangulation-based systems in primal space, in the proposed approach samples are highly concentrated in the vicinity of high curvature points. Feature line points, which are highly localized in primal space, are easy to estimate in dual space because they correspond to extended and smooth curve segments. We will implement hybrid systems combining depth discontinuities with triangulation-based systems, as well as multi-view photometric stereo, to achieve more accurate reconstructions of solid objects bound by piecewise smooth surfaces with accuracy guarantees for metrology applications. Applications to be explored range from reverse engineering to real-time 3D cinematography. Variational algorithms to fit watertight piecewise smooth implicit surfaces to the capture data, as well as isosurface algorithms to triangulate these implicit surfaces preserving feature lines will be developed as well.

References

1. Crispell, D., Lanman, D., Sibley, P., Zhao, Y., Taubin, G.: Beyond Silhouettes: Surface Reconstruction using Multi-Flash Photography. In: 3rd International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT 2006), UNC, Chapel Hill, USA (June 2006)
2. Lanman, D., Crispell, D., Sibley, P., Zhao, Y., Taubin, G.: Multi-flash 3d photography: Capturing shape and appearance. In: Siggraph 2006, Poster session, Boston, MA (July 2006)
3. Sibley, P.G., Taubin, G.: Vectorfield Isosurface-based Reconstruction from Oriented points. In: Siggraph 2005, Sketch (2005)
4. Chen, F., Brown, G.M., Song, M.: Overview of three-dimensional shape measurement using optical methods. Optical Engineering 39(1), 10–22 (2000)
5. Mada, S., Smith, M., Smith, L., Midha, S.: An overview of passive and active vision techniques for hand-held 3d data acquisition. In: Opto Ireland 2003: Optical Metrology, Imaging, and Machine Vision (2003)
6. Wu, H., Chen, Y., Wu, M., Guan, C., Yu, X.: 3d measurement technology by structured light using stripe-edge-based gray code. In: International Symposium on Instrumentation Science and Technology. Journal of Physics: Conference Series, vol. 48, pp. 537–541 (2006)
7. Zhang, L., Curless, B., Seitz, S.M.: Spacetime stereo: Shape recovery for dynamic scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 367–374 (June 2003)
8. Levoy, M.: Light Field Rendering. In: Siggraph 1996, Conference Proceedings, pp. 31–42 (1996)

9. Gortler, S., Grzeszczuk, R., Szeliski, R., Cohen, M.: The Lumigraph. In: Siggraph 1996, Conference Proceedings, pp. 43–54 (1996)
10. Lanman, D., Crispell, D., Wachs, M., Taubin, G.: Spherical Catadioptric Arrays: Construction, Geometry, and Calibration. In: 3rd International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT 2006), UNC, Chapel Hill, USA (June 2006)
11. Lanman, M., Wachs, D., Taubin, G., Cukierman, F.: Reconstructing a 3D Line from a Single Catadioptric Image. In: 3rd International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT 2006), UNC, Chapel Hill, USA (June 2006)
12. Kang, K., Tarel, J.-P., Fishman, R., Cooper, D.: A linear dual-space approach to 3d surface reconstruction from occluding contours using algebraic surfaces. In: IEEE International Conference on Computer Vision (ICCV 2001), vol. I, pp. 198–204 (2001)
13. Liang, C., Wong, K.-Y.K.: Complex 3d shape recovery using a dual-space approach. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2005) (2005)
14. Raskar, R., Tan, K.-H., Feris, R., Yu, J., Turk, M.: Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. ACM Trans. Graph. 23(3), 679–688 (2004)
15. Annotated computer vision bibliography on surface and shape from contours or silhouettes, `http://www.visionbib.com/bibliography/shapefrom408.html`
16. Matusik, W., Buehler, C., Raskar, R., Gortler, S., McMillan, L.: Image-based visual hulls. In: SIGGRAPH 2000 (2000)
17. Cipolla, R., Giblin, P.: Visual Motion of Curves and Surfaces. Cambridge University Press, Cambridge (2000)
18. Cross, G., Zisserman, A.: Quadric surface reconstruction from dual-space geometry. In: IEEE International Conference on Computer Vision (1998)
19. Liang, C., Wong, K.-Y.K.: Complex 3d shape recovery using a dual-space approach. In: IEEE Conference on Computer Vision and Pattern Recognition (2005)
20. Seitz, S., Curless, B., Diebel, J., Scarstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: IEEE Conference on Computer Vision and Pattern Recognition (2006)
21. Esteban, C.H., Schmitt, F.: Silhouette and stereo fusion for 3d object modeling. Comput. Vis. Image Underst. 96(3), 367–392 (2004)
22. Furukawa, Y., Ponce, J.: Carved visual hulls for image-based modeling. In: European Conference on Computer vision 2006 (2006)
23. Goesele, M., Seitz, S., Curless, B.: Multi-view stereo revisited. In: IEEE Conference on Computer Vision and Pattern Recognition (2006)
24. Lensch, H.P.A., Kautz, J., Goesele, M., Heidrich, W., Seidel, H.-P.: Image-based reconstruction of spatially varying materials. In: Proceedings of Eurographics Rendering Workshop (2001)
25. Sadlo, F., Weyrich, T., Peikert, R., Gross, M.: A practical structured light acquisition system for point-based geometry and texture. In: Eurographics Symposium on Point-Based Graphics, pp. 89–98 (June 2005)
26. Bouguet, J.-Y.: Complete camera calibration toolbox for matlab, `http://www.vision.caltech.edu/bouguetj/calib_doc`
27. Giblin, P.J., Weiss, R.S.: Epipolar curves on surfaces. Image and Vision Computing 13(1), 33–34 (1995)
28. Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, C.T.: Computing and rendering point set surfaces. IEEE Trans. on Visualization and Computer Graphics 9(1), 3–15 (2003)

29. Davis, J., Marschner, S., Garr, M., Levoy, M.: Filling holes in complex surfaces using volumetric diffusion. In: 3DPVT 2002 (2002)
30. Park, S., Guo, X., Shin, H., Qin, H.: Shape and appearance repair for incomplete point surfaces. In: IEEE International Conference on Computer Vision, vol. 2 (2005)
31. Sibley, P.G., Taubin, G.: Vectorfield Isosurface-based Reconstruction from Oriented Points. In: SIGGRAPH 2005, Sketch (2005)
32. Feris, R., Raskar, R., Tan, K.-H., Turk, M.: Specular reflection reduction with multi-flash imaging. In: 17th Brazilian Symposium on Computer Graphics and Image Processing (October 2004)