Adequate Reconstruction of Transparent Objects on a Shoestring Budget

Sai-Kit Yeung \(^1,2\)  
Tai-Pang Wu \(^2\)  
Chi-Keung Tang \(^2\)  
Tony F. Chan\(^1,2\)  
Stanley Osher \(^1\)

\(^1\) University of California, Los Angeles  
\(^2\) The Hong Kong University of Science and Technology

Abstract

Reconstructing transparent objects is a challenging problem. While producing reasonable results for quite complex objects, existing approaches require custom calibration or somewhat expensive labor to achieve high precision. On the other hand, when an overall shape preserving salient and fine details is sufficient, we show in this paper a significant step toward solving the problem on a shoestring budget, by using only a video camera, a moving spotlight, and a small chrome sphere. Specifically, the problem we address is to estimate the normal map of the exterior surface of a given solid transparent object, from which the surface depth can be integrated. Our technical contribution lies in relating this normal reconstruction problem to one of graph-cut segmentation. Unlike conventional formulations, however, our graph is dual-layered, since we can see a transparent object’s foreground as well as the background behind it. Quantitative and qualitative evaluation are performed to verify the efficacy of this practical solution.

1. Introduction

We address the problem of normal reconstruction for a transparent object, where the integrated surface is an overall shape of the target object that preserves salient features and fine structures if present on its exterior surface. Such detail-preserving exterior surface representation is adequate for vision and robotics applications where transparent objects are to be grabbed by a robotic arm, or avoided by a navigating robot in a cluttered scene. In our paper, our goal is different from photorealistic rendering or high-accuracy reconstruction of transparent objects, where custom equipment, calibrated and mechanical capture are often deemed necessary to achieve precision as high as to trace the complex refractive light-transport paths exhibited by the target object. On the other hand, when an adequate shape without this level of precision is sufficient, it is possible to propose a reconstruction approach that uses a simpler setup realizable using a smaller budget.

Without expensive or complicated setup while still supporting an adequate reconstruction, what visual cues concerning a transparent object can be utilized? Although some of us have had the unpleasant experience of smacking into a glass window without seeing it, we can still see a wide range of transparent objects despite their apparent transparency, because most of them refract and reflect incoming light. Tracing refractive light-transport paths using calibrated setup and capture had contributed to the success of techniques aiming at high-precision reconstruction. This paper on the other hand makes use of specularities directly reflected off an transparent object to produce an adequate reconstruction. Due to the low dynamic range of our inexpensive video camera, however, indirect reflection caused by complex light transport (e.g., caustics and total internal reflection) also produces strong highlights with intensity that appears as strong as direct specular highlights. Thus, the main problem to be solved is to identify at each pixel the subset of collected highlights that are caused by direct specular reflection.

1.1. Technical Overview

The working principle of our system is: given dense lighting directions, a pixel location within the data capture range has a high probability to observe a specular highlight directly reflected off the surface point being observed, see Figure 1. This can be used to obtain the surface normal from a specular object with known reference geometry. At first glance, given a dense collection of highlights at a pixel,
2. Related Work

The following classical works determine the shape of a transparent object by specularities. In specular stereo [1] a two-camera configuration and image trajectory were used. A theory of specular surface was developed in [16], where the relationship between specular surface geometry and image trajectories were studied and features were classified as real and virtual. Virtual features, which are reflections by a specular surface not limited to highlights, contain useful information on the shape of the object. In [11], two views were used to model the surface of a transparent object, by making use of the optical phenomenon that the degree of polarization of the light reflected from the object surface depends on the reflection angle, which in turn depends on surface normal. This approach utilizing light polarization, where the light transport paths were ray-traced, was further explored in [10] where one camera was used. In [8], a theory was developed on refractive and specular 3D shape by studying the light transport paths, which are restricted to undergo no more than two refractions. Two views were used for dynamic refraction stereo [12] where the notion of refractive disparity was introduced. In this work, a reference pattern and refractive liquid were used. Scatter trace photography [13] was then proposed to reconstruct transparent objects made of inhomogeneous materials, by using a well-calibrated capture device to distinguish direct scatter trace from other indirect optical observations. In [6], a transparent object was immersed into fluorescent liquid. Computer-controlled laser projector and multiview scans were available for merging. While the recovered normal maps of mesostructures look good in [2] and were demonstrated to be useful for relighting, their assumption on specular highlights does not apply to general transparent objects that exhibit complex refraction phenomena (such as total internal reflection and caustics). Our shape reconstruction method makes use of rough initial shape (normals), sparse normal cues, and dense specular highlights, which sets itself apart from the above approaches where mathematical theories were developed based on the physics of light transport, or simplifying assumptions were imposed on the transparent object.

3. Observations and Assumptions

Given a dense set of views of a solid transparent object captured by a static video camera under variable illumination, the problem is to reconstruct the normal map by utilizing specular highlights directly reflected off the surface.

The specular highlight directly reflected corresponds to the observed normals of the exterior surface, after applying orientation consistency using a reference shape of known geometry, which is a chrome sphere in our case. Transparent objects are more challenging, since indirect reflections caused by complex light transport and caustics can also produce highlight as strong as direct specular reflection. Using our inexpensive capture system and under the orthographic camera assumption, we have the following key observations:

Data capture range. The light direction is restricted such that those falling outside the 90-degree realm will be ignored; otherwise the light source would have been located
Figure 3. (a) Orientation consistency in the presence of transparency and indirect illumination effects. The reference sphere and the object have different material properties but they act like ideal specular objects when they reflect specular highlights (brightness and contrast were enhanced for visualization). (b) Under the orthographic camera assumption, we can directly obtain the normal orientation from the reference sphere. Notice that given a single view, only a subset of normals can be computed. ($L$ is light direction, $R$ reflection direction, $N$ is normal direction.)

Figure 4. [color figure] Three typical scenarios on the observed light emanating from a point on the exterior surface of a transparent and refractive object. The numbers in the figure indicate intensity magnitudes. (a) Nearly ideal specular reflection. In practice, the specular highlight spans a finite region in the image. (b) Non-specular reflection where the intensity is similar to that of a true specular highlight. Examples include caustics and total internal reflection reflected off from the back surface of the object. (c) Non-specular reflection where the intensity is attenuated making it easily discarded by thresholding. Orientation consistency should be applied to (a), but not to (b) or (c) because they are not specular reflections.

behind the transparent object. Under this lighting configuration and using a shiny chrome sphere (Figure 3(a)) as a reference geometry, surface normals falling outside the 45-degree realm measured from the upward direction cannot be recovered, Figure 3(b). The mathematical explanation can be readily derived using the law of specular reflection under orthographic projection.

**Highlight appearance.** Direct specular highlight is bright and concentrated (Figure 4(a)). However, some non-specular indirect reflection observed will also be classified as highlight. It happens when caustics and total internal reflection behind the surface appear as bright as specular reflection, especially a low-dynamic-range digital video camera is used in image capture (Figure 4(b)). Obvious true or false highlights can be marked up by user for disambiguation if needed.

**Data availability.** Given the limited data capture range, some pixels will have no highlight, or the observed intensity is not high enough to be considered a highlight (Figure 4(c)).

**Normal clusters.** The normals transferred from the chrome sphere after applying orientation consistency will form distinctive clusters at a pixel over the time frames (Figure 5), one of which corresponds to direct specular highlights whereas the rest are due to indirect illumination. In all of our experiments, we found that a pixel location observes at most two *salient* normals clusters over all the frames captured. A third cluster, if it exists, is too weak to be detected as highlight. This two-cluster observation, however, is not critical: as long as the strongest is detected or identified as direct specular highlight, the exact number of clusters does not matter.
4. Normal Optimization

We propose to solve the optimization problem by integrating the rough shape given by an object’s silhouette, and the normals transferred via orientation consistency. To resolve severe highlight ambiguity as stated in the previous section, a limited amount of user specification is employed. This improves not only the initialization but also the optimization process. The optimization can be mapped to one similar to image segmentation and solved via graph-cut optimization.

4.1. Initialization

Since an object’s silhouette on an image is easily available, a rough overall shape can be derived from shape from silhouette. Normals derived from the resulting shape give a reasonable initial guess.

To improve the quality of initial normals, the user may indicate true highlights on keyframes. Automatic tracking technique is applied to trace the corresponding locations in-between frames. With the true highlights being tracked, we can transfer the normals from a chrome sphere, and these normals are considered as hard constraints in shape-from-silhouette for computing the initial rough shape.

Given a sparse set of normals, consisting of transferred normals obtained from user’s markups and the silhouette normals, we compute the surface normal \( n^F \) for all pixels within the object’s silhouette via a standard MRF formulation for which a stable implementation is available [14].

4.2. Shape Refinement by Graph Cuts

After estimating initial normals, the next step is to integrate the information given by the dense highlight collected. As discussed in previous section, given a total of \( T \) image frames, the possible situations at a given pixel are: 1) no highlight, 2) highlights form a single normal cluster, 3) highlights form two or more normal clusters. So our problem is translated into a labeling problem, that is, given a normal cluster, determine if it corresponds to one that produces direct specular reflection on the exterior surface. Our idea is to utilize the data measurement given by applying orientation consistency to refine the initial surface. The initial shape also gives us relevant cues for rejecting wrong measurements due to false highlights.

Normal clustering. The shape optimization problem can be posed as a binary labeling problem, by assigning every cluster as exterior surface normal or otherwise. Since each pixel can observe direct and indirect reflections, we adopt the following method to extract the two representative normals: Given \( T \) observations per pixel, we threshold the pixel intensities to discard weak intensities. It results in \( M \) usable observations, where \( 0 \leq M \leq T \). If \( M = 0 \), it means that this pixel contains no useful information and initial normal will be used instead. Otherwise, we apply \( K \)-means clustering to extract all normal clusters. The process with \( K = 2 \) is illustrated in Figure 6(a). Using Minimum Description Length principle [4, 3] the value of \( K \) can be estimated automatically.

Graph formulation. Next, we construct a graph \( G = (V, E) \), where \( V \) is the set of all nodes and \( E \) is the set of all edges connecting adjacent nodes. In our case, the nodes contain labels to the two most salient normals clustered, and the edges represent adjacency relationships. The graph can have up to \( 2N \) nodes for \( N \) processing pixels and every node can have 9 edges as shown in Figure 6(b). For pixel locations with only one cluster, we duplicate the cluster in the other node to simplify the implementation (Figure 6(c)).

The labeling problem is to assign a unique label \( s_i \) for each node \( i \in V \), i.e., \( s_i \) \{exterior surface normal \( = 1 \), otherwise \( = 0 \)\}. The solution \( S = \{s_i\} \) corresponding to the final frontal surface normals can be obtained by minimizing a Gibbs energy \( E(S) \) [9]:

\[
E(S) = \sum_{i \in V} E_1(s_i) + \sum_{(i,j) \in E} E_2(s_i, s_j)
\] (1)

where \( E_1(s_i) \) is the likelihood energy, denoting the cost...
is similar to a normal that is their neighboring nodes. Pixel locations with no observation will be completed by their initial estimation. Providing a normal, in order to define the affinity and hence the energy of should be available for processing. We introduce the idea of assigning an observed normal to non-exterior surface. Since the information on the non-exterior surface should not belong to the exterior surface, it will not be similar to the initial normal \( n_i^F \). In other words, we may say \( n_i \) is similar to a normal that is dissimilar to the initial normal. Let us call it counter normal. Since our captured normals must be facing upward, it is meaningless to consider normals with \( n^<_z < 0 \). Therefore, we set the corresponding counter normal by flipping the gradient of \( n^<_F \), i.e., \( \{ p^<_F, q^<_F \} = \{ -p^<_F, -q^<_F \} \), where \( p^<_F \) and \( q^<_F \) are the \( x \) and \( y \) gradient of \( n^<_F \), and \( p^<_F \) and \( q^<_F \) are the \( x \) and \( y \) gradient of \( n^<_F \). Basically, this strategy adopts the most dissimilar surface normal (with \( z \)-axis as the reference) with upward direction as the counter normal.

With the initial normal and the counter normal, we can now define our energy term \( E_1 \). For each node \( i \), we compute the difference of the corresponding gradients with the frontal gradient and the counter (non-frontal) gradient by \( d^F_i = |p_i - p^F_i| + |q_i - q^F_i| \) and \( d^C_i = |p_i - p^C_i| + |q_i - q^C_i| \) respectively. Notice that for pixel locations with no highlight observation, we set the labels of two nodes at each of these pixels to correspond to the initial \( p^F_i \) values and set \( E_1(0) = E_2(0) = 0.5 \) so the smoothness term takes over in the optimal labeling problem. Therefore, \( E_1(s_i) \) can be defined as following:

\[
\begin{align*}
E_1(s_i = 1) &= 0 & E_1(s_i = 0) &= \infty & \forall i \in \mathcal{F} \\
E_1(s_i = 1) &= \infty & E_1(s_i = 0) &= 0 & \forall i \in \mathcal{C} \\
E_1(s_i = 1) &= \frac{d^F_i}{d^F_i + d^C_i} & E_1(s_i = 0) &= \frac{d^C_i}{d^F_i + d^C_i} & \forall i \in \mathcal{U}_1 \\
E_1(s_i = 1) &= 0.5 & E_1(s_i = 0) &= 0.5 & \forall i \in \mathcal{U}_2
\end{align*}
\]

where \( \mathcal{U}_1 \) and \( \mathcal{U}_2 \) are the set of nodes from region with and without normal observation, respectively, and \( \{ \mathcal{U}_1 \cup \mathcal{U}_2 \} = \mathcal{V} - \{ \mathcal{F} \cup \mathcal{C} \} \) is the set of uncertain nodes to be labeled. Eqn (2) is similar to [9] except for \( i \in \mathcal{U}_2 \), \( \mathcal{F} \) is the set of nodes labeled as exterior surface normals, which are available in the initialization. \( \mathcal{C} \) is the set of nodes corresponding to non-exterior normal observations, which are specified by the user in a similar manner as the exterior surface normals.
e.g., by marking up false instead of true highlights.

Notice that since our graph is dual-layered, nodes in \( F \) and \( C \) can have the same pixel location. This is the main difference in the graph construction as compared to works in image segmentation [17, 9] where nodes in \( F \) and \( C \) must have different locations. The first two equations guarantee the nodes in \( F \) or \( C \) always have the label consistent with user inputs. If we ignore the \( E_2 \) term in Eqn (1), minimizing the energy \( E_1 \) produces a “winner-takes-all” labeling strategy based on normal similarity only.

**Prior energy.** We use \( E_2 \) to encode the smoothness constraint between neighboring nodes. Define the normal similarity function between two nodes \( i \) and \( j \) as an inverse function of the smoothness constraint:

\[
E_2(s_i, s_j) = |s_i - s_j| \cdot g(|p_i - p_j| + |q_i - q_j|)
\]

where \( g(\xi) = (\xi + 1)^{-1} \). Note that \( |s_i - s_j| \) allows us to capture the smoothness information when the adjacent nodes have different labels. In other words, \( E_2 \) is a penalty term when neighboring nodes are assigned with different labels. So if the neighboring normals are similar, assigning them with different labels will increase the energy of the graph and vice versa. This energy term encourages integrable normals to be grouped into the same surface.

We use the max-flow algorithm [7] to minimize the energy \( E(S) \) in Eqn (1). Readers may notice that we do not enforce the two nodes at the same pixel location (recall our graph is dual-layered) to have different labels in our graph formulation. Although in our examples, pixel originally having two clusters will always have one labeled as exterior surface normal, we cannot guarantee that the two clusters may both come from false highlights.

### 5. Experimental Results

The data sets tested and running times of normal map reconstruction are summarized in Table 2. The running times shown exclude those of surface integration, where we use the source codes from [14] to produce the final surfaces.

**Setup.** To capture a dense image set, similar in fashion to [5, 2], we use an off-the-shelf DV camera with a fixed viewpoint to simultaneously capture the reference chrome sphere and the target object. A moving spotlight is used to mimic a distant light source at varying directions.

**Quantitative Evaluation.** In order to evaluate the performance of our system, we capture two real data sets whose analytical geometries are known (namely, a hemisphere and a cylinder). They can be served as the ground truth for quantitative comparison. We compute the average difference of our computed surface normals with \textit{Sphere} and \textit{Cylinder} respectively. The results are depicted in Table 3. We also generate the surface normals by using a winner-takes-all strategy such as [2]. We implemented this strategy as the sum of second order moments followed by eigen-decomposition. The corresponding visual results are shown in Figure 7. From the results, we can see that our final reconstructed shape is more faithful compared with the one using the winner-takes-all strategy. Without proper labeling, the normals from the true highlight will be mixed up with the wrong normals transferred due to false highlights, making the final optimized normals fail to integrate into a reasonable surface as shown in Figure 7(c).

**Qualitative Evaluation.** Figure 8 shows the single-view reconstruction result on a transparent glass \textit{Jug}. This example is simple as the initial shape is already close to the final solution. We show one view of the surface reconstruction alongside with the real object in a similar view.

Figure 9 shows the results a glass figurine \textit{Fish}, which is very similar to the one used in [13] and our result looks comparable. Note that the image sequence contains a lot of shadows and highlight. The figurine contains internal structures of varied colors. However, it is highly specular and thus can produce sufficient specular observations. The zoom-in views illustrate the details preserved in our recon-
<table>
<thead>
<tr>
<th>no. of images</th>
<th>image dimensions</th>
<th>total running time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPHERE</td>
<td>8000</td>
<td>251 × 221</td>
</tr>
<tr>
<td>CYLINDER</td>
<td>4226</td>
<td>146 × 384</td>
</tr>
<tr>
<td>JUG</td>
<td>3870</td>
<td>279 × 419</td>
</tr>
<tr>
<td>FISH</td>
<td>1512</td>
<td>259 × 252</td>
</tr>
<tr>
<td>WINE GLASS</td>
<td>3133</td>
<td>153 × 324</td>
</tr>
<tr>
<td>WATER GLASS</td>
<td>3383</td>
<td>153 × 279</td>
</tr>
</tbody>
</table>

Table 2. Running times are measured on a desktop computer with Dual-Core 2.6 GHz CPU and 3.0 GB RAM.

<table>
<thead>
<tr>
<th>SPHERE</th>
<th>CYLINDER</th>
</tr>
</thead>
<tbody>
<tr>
<td>data region</td>
<td>whole region</td>
</tr>
<tr>
<td>Our method</td>
<td>0.096</td>
</tr>
<tr>
<td>[2]</td>
<td>0.738</td>
</tr>
<tr>
<td>data region</td>
<td>whole region</td>
</tr>
<tr>
<td>Our method</td>
<td>0.071</td>
</tr>
<tr>
<td>[2]</td>
<td>0.386</td>
</tr>
</tbody>
</table>

Table 3. The mean errors of the computed surface normals using our method and [2] are shown (where the error is defined as sum of the squared difference of three normal components). Data region consists of pixel locations with highlight observations. Whole region includes all processing pixels.

Figure 9. **FISH**. The top shows the normal maps $\mathbf{N}$ displayed as $\mathbf{N} \cdot \mathbf{L}$. The lighting directions are: (a)–(c) $\mathbf{L} = (- \frac{1}{\sqrt{2}}, - \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T$, (d) $\mathbf{L} = (0, 0, 1)^T$, (e) $\mathbf{L} = (\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, - \frac{1}{\sqrt{3}})^T$. The middle row shows the reconstructed surface. (a) “winner-takes-all” strategy [2]. (b) user-supplied normal cues and silhouettes [15]. (c) Our result in frontal view. (d)–(e) Our results in other views. The bottom shows the comparison of the reconstructed surface with the real object at similar viewpoint. (h) Zoom-in view of (f). (i) Zoom-in view of (g). The figurine is a solid transparent object with complex colors inside the object.

6. Conclusion and Future Work

This paper presents a practical approach for reconstructing the normal map of the exterior surface of a transparent object. While inadequate for high-precision graphics rendering, our detail-preserving output is a faithful reconstruction of the transparent object, as demonstrated by our convincing results, and should be useful in a range of vision applications. Our approach makes use of an initial shape, normals transferred using orientation consistencies, and sparse user markups if needed. The problem was translated into one similar to image segmentation, and the optimization can be formulated using graph cuts on a dual-layered graph. This alternative approach is desirable for quick 3D prototyping of existing transparent objects with details adequately preserved. Our current system produces a depth map for the exterior surface of a transparent object. To generate a full 3D reconstruction, it is possible to merge overlapping depth maps which will be the future work.
Figure 10. WATER GLASS. The top shows ten captured input images and normal maps $N$ displayed as $N \cdot L$. The lighting directions respectively are: (a)–(c) $L = (-\sqrt{3}/3, 1/\sqrt{3}, 1/\sqrt{3})^T$, (d) $L = (0, 0, 1)^T$, (e) $L = (\sqrt{3}/3, 1/\sqrt{3}, 1/\sqrt{3})^T$. (a) “winner-takes-all” strategy [2]. (b) user-supplied normal cues and silhouettes [15]. (c)-(e) our results. (f)–(i) show the comparison of our reconstructed surfaces with the real object at novel viewpoints, along-size with the corresponding zoom-in views.

References