Robust Multi-View Stereo without Matching

Philippe Lambert and Patrick Hébert
Computer Vision and Systems Laboratory, Laval University, Quebec, Canada
philippe.lambert.1@ulaval.ca, hebert@gel.ulaval.ca

Abstract

This paper proposes a robust algorithm that finds a proxy surface from a series of calibrated pictures of an object without assuming any of its reflectance properties. This proxy is optimized to reduce view interpolation errors by globally minimizing the frequency criterion proposed in [1]. The generality of this setting makes robustness particularly difficult to achieve since no model from which to identify outliers or noise is available. Unfortunately, failing to achieve robustness results in unusable proxy for most of the datasets presented. The traditional method of identifying outliers by their discrepency from photoconsistency must somehow be replaced by a global analysis involving all viewpoints. The major contribution of this paper is to meet this requirement by proposing a robust estimation of the minimizer of the frequency criterion as well as a novel framework for merging the multiple depth hypotheses obtained. View interpolation results and proxies are shown for challenging datasets, both Lambertian and not.

1. Introduction

We strive to find a 3D proxy that reduces view interpolation error. Our starting point is a set of calibrated pictures of an object taken under fixed lighting conditions, as is common with most multi-view stereo algorithms. A distinguishing feature of this paper is removing any assumption regarding the reflectance properties of the object, which incidentally prohibits matching between pictures. Consequently, the primary role of the proxy becomes guiding view interpolation rather than describing the object shape, as illustrated in Figure (1). This distinction becomes crucial when no reflectance property is assumed because the proxy does not have to correspond to the actual object shape although it succeeds in diminishing view interpolation errors.

Recently, it was shown that the desired proxy must globally minimise the integration of the so-called frequency criterion over its surface[1], a conclusion we will review later. This criterion satisfies our requirement by avoiding photoconsistency measure in favor of enforcing smoothness among the colors of neighboring rays on the proxy. Interestingly, for shaded or textured Lambertian objects, a robust optimization of this criterion recovers the actual 3D shape of the object.

We will show why a direct approach to the optimization is not robust enough to converge when applied to datasets prone to visibility outliers. Achieving robustness is particularly difficult in this setting since no reflectance model is assumed from which to identify these outliers. Moreover, most forms of filtering are inappropriate since visibility outliers lead to erroneous values clustered together in a coherent way. Therefore, new techniques must be developed to deal with these sources of errors.

The main contribution of this paper is to provide a robust algorithm in spite of the difficulties exposed. We must emphasize that robustness in this setting is not just a slight improvement over a non robust algorithm. Without it, obtaining a usable proxy is not even possible for some of the datasets that will be presented. The proposed algorithm follows the two step approach of other multi-view stereo algorithms by first finding a depth estimate for each pixel of the pictures and then merging these depths to form the proxy.

Our contribution is supported by two main ideas used at each of these two steps respectively. The first and most important one is a proper decomposition of the computation leading to the location of the minimizer of the frequency criterion that ensures that outliers cannot bias its value. Outliers instead produce additional modes in the corresponding distribution and are later discarded. The second idea is a novel way to merge the multiple depths obtained from this multi-modal distribution, which also integrates a simple mechanism to handle dataless regions. This last step uses a scalar field of Radial Basis Functions that enforces and regularizes the multiple modes obtained, both within each depth map and then globally. Note that both ideas are general enough to apply to any other criterion. The proposed algorithm is shown to produce results comparable to the best ones recorded on the Middlebury datasets, despite the milder assumptions under which it operates. Results also improve those of existing approaches for non Lambertian datasets.
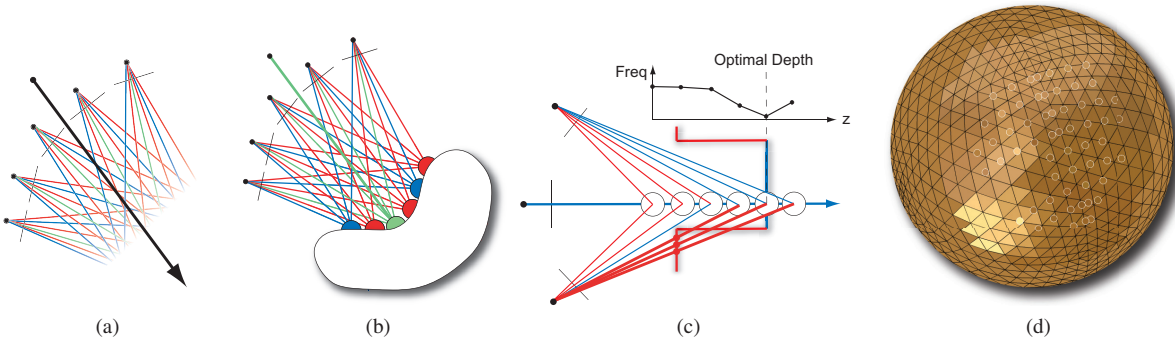
| (a) | (b) | (c) | (d) |

Figure 1: Interpolating the color of the unknown black ray in (a) implicitly requires a depth value along this ray to determine which neighboring rays to use. As shown in (b), a proxy provides this depth value for each ray emanating from the volume it encloses. When no reflectance property is assumed, the major difficulty in identifying the optimal depths arises from visibility outliers, like the three thick red rays shown in (c). In (d), outliers are visible on the left side of a lumisphere of the Temple dataset (depicted in Fig. (4) and (5)). Flat shading was used to enhance contrast.

2. Related Work

There is indeed a close relationship between the problem of finding a proxy and multi-view stereo algorithms. Recent such algorithms were reviewed in [2]. They operate by assuming specific reflectance properties, usually Lambertian, and optimize a criterion enforcing this assumption to obtain the object geometry. See also [3] for the application of this idea on the same non Lambertian objects as those used in this paper.

Both the Surface Light Fields [4] and Light Field Maps [5] proposed a strategy to perform view interpolation that removed the necessity of knowing the reflectance properties of the object. They used range scans to map the exiting radiance measured by calibrated pictures on the acquired object surface. This strategy requires the two additional steps of preparing the object for the scan using surface coverage and finally doing the registration that aligns the scan with the pictures.

Lambert et al. [1] proposed using a frequency criterion that is free from any reflectance property assumption yet operates from calibrated pictures alone. The surface that optimizes this criterion directly improves Surface Light Field rendering efficiency. Instead of relying on physical laws governing the interaction between light and surface, as in the multi-view stereo setting, the frequency criterion aims to decrease the variations in the Surface Light Field along its less densely sampled dimensions, which directly reduces interpolation error. A drawback to this approach is that it requires more pictures than in the multi-view stereo setting and it is highly sensitive to outliers.

State of the art in multi-view stereo has already identified the most common sources of difficulties in surface optimization and provided interesting strategies to handle outliers and noise. To mention a few among the most successful algorithms, Goesele et al.[6] and Campbell et al.[7] implement a simple yet effective outlier rejection by eliminating viewpoints whose normalised cross-correlation (NCC) score computed with a reference viewpoint is not high enough. This strategy is unfortunately prone to error with non Lambertian objects since the view dependant variations in colors can lead to a low NCC score. More generally, when the color of a point of the object varies in an unknown way with viewpoint, it is no longer possible to remove outliers based on a discrepancy from photoconsistency.

Campbell et al.[7] proposed to retain multiple depths corresponding to high NCC scores and merge them to produce each depth map using a Markov Random Field. In [6], merging of the depth maps relies on a discretised scalar field (VRIP). Inspired by these approaches, our merging step also handles multiple depth hypotheses and its field leverages on the following known properties of Radial Basis Functions (RBF): adaptation to non uniform sampling, hole filling and regularization. A simple yet novel mechanism to incorporate visual hull information is also proposed.

The organisation of the paper is as follows. Section 3 reviews the concept of a Surface Light Field and its lumispheres. In section 4 the motivation behind the frequency criterion are briefly revisited. The core of the paper then follows with the proposed implementation. It is first explained in a direct but easily understood manner followed by the mandatory modifications needed to achieve robustness. Finally, the proposed merging is presented along with results obtained on various challenging datasets.

3. Surface Light Fields and its Lumispheres

A Surface Light Field (SLF) is a provenly efficient way to store the light rays captured by the input pictures [4].

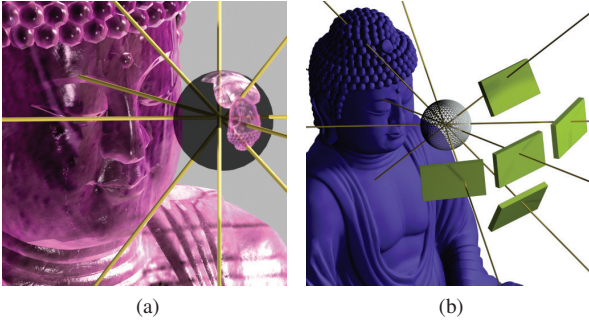(a)   (b)

Figure 2: In (a), a lumisphere depicts the colors of the rays traversing its center in all directions. In (b), the corresponding geometry is illustrated and shows how each picture provides a single sample of the lumisphere.

Each ray in a SLF is indexed by a point (u, v) on a given surface $K(u, v)$ and the angles (θ, ϕ) this ray makes when leaving the surface. A SLF is thus a 4D function $f : K \times S^2 \to [0, 1]^3, (u, v, \theta, \phi) \mapsto f(u, v, \theta, \phi)$ whose domain is the product of a 2D manifold K embedded in $\mathbb{R}^3$ and the two dimensional unit sphere S^2, while its range is the unit cube of RGB triplets. We will refer to the surface K as the *parameterization surface*, the combined (u, v) coordinates as the *spatial dimensions* and the remaining (θ, ϕ) coordinates as the *angular dimensions*.

In order to more easily understand the SLF, we use a central concept of this paper called a *lumisphere* [4]. It designates in the SLF the 2D function obtained by fixing a point of the spatial dimensions $P = (u_0, v_0)$ and varying only the angular ones: $L_P(\theta, \phi) = f(u_0, v_0, \theta, \phi)$. A lumisphere is graphically represented by a unit sphere at a specific location whose surface colors correspond to those of the light rays traversing at this location, as shown in Figure (2). The SLF can thus be conceived as a set of lumispheres located on the parameterization surface.

4. Frequency criterion

As illustrated in 2D in Figure 1(b), choosing a particular depth along a given ray determines which rays are used to interpolate its color. From this perpective, we are interested in knowing which parameterization surface is the best proxy. An answer to this question is proposed in [1] and can be summarized as follows using the terminology developed so far: *When interpolating an unknown ray, its color is obtained using the lumisphere of the SLF that the ray intersects. When there is a high amount of variations among the colors of this lumisphere, any slight shift in the angle of the unknown ray leads to a highly different color and thus produces visible artifacts. Motivated by this observation, it was proposed to use as a proxy the surface whose lumispheres*

exhibit less variations.

The measure of these variations in a lumisphere located at position P is given by the frequency criterion:

$$\|L_P\|_V = \iint_{S^2} \|\nabla L_P\|^2 \, dA, \qquad (1)$$

where the integration is performed over the lumisphere surface and dA is its surface element. The name frequency criterion comes from the alternative statement of this measure in the frequency domain that was first used in [1].

Why is the frequency criterion interesting? The presence of the gradient makes the frequency criterion sensitive to the ordering of rays within a lumisphere. This situation is not the same when computing the variance of the colors, for example, since a shuffling of the angular position of the rays does not affect the variance of the colors. The frequency criterion is thus a relevant measure when assessing interpolation quality because interpolation depends on this ordering. Other criteria like entropy and mutual information also lack this capability.

The frequency criterion is computed on a single lumisphere. It thus avoids having to use a window of many pixels in the pictures, a situation favorable to an increase in 3D resolution. The additional stability obtained by considering neighbouring pixels can still be used, but can be defered to the later stage of merging points, as will be done in the following.

5. Obtaining a proxy surface

The overall strategy to obtain the proxy follows two steps. First, for each pixel whose projector ray intersects with the object, we find the depth along the projector ray where the frequency criterion is minimal, what we call the optimal depth. Once these depths are found, they are merged together to obtain the proxy. These two steps are the subject of the next two sections.

5.1. Finding optimal depths

Evaluating the frequency criterion A fundamental operation required to find the proxy is computing the frequency criterion for a lumisphere located at any given position in 3D space, following equation (1). This computation is achieved by first obtaining the continuous function describing the colors on the lumisphere, what we call the reconstruction of a lumisphere, and then computing the frequency criterion for the function found by summing the norm of its squared gradients, which we now describe.

We are given the position P of a lumisphere in the world coordinate frame and we want first to obtain a continuous function describing the colors at its surface, as illustrated in Figure (1)(d). By projecting P in the ith picture and using
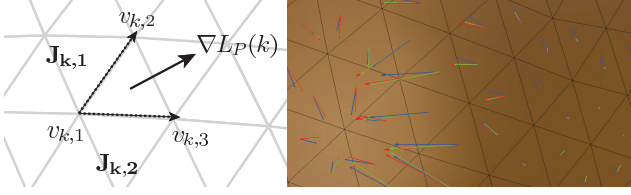
Figure 3: The geometry involved in the problem of finding a lumisphere's gradients on the left along with an example solution on the right. The color of the gradient indicates the corresponding color channel.

bilinear interpolation, the lumisphere color along the corresponding viewing direction is sampled. Using this time the world coordinate frame translated at P, these samples lead to data of the form $\{(x_i, y_i, z_i), (R_i, G_i, B_i)\}$, where $p_i = (x_i, y_i, z_i)$ is the point of the lumisphere unit-sphere along the ith viewing direction and $c_i = (R_i, G_i, B_i)$ its color. Note that we switched to cartesian coordinates to simplify future computations. Refer to Figure (2)(b) for an illustration of the geometry involved.

From the data obtained, many methods exist to obtain a continuous function giving the colors over the whole lumisphere. The particularly simple one we chose is justified principally by a specific usage in the robust approach to come. The function is first evaluated on the vertices of a triangular mesh of the lumisphere spherical surface. Specifically, to obtain the color of vertex v_j, we first compute its distance to each input viewing direction using $d(v_j, p_i) = \arccos(< v_j, p_i >)$. These distances then allow computing a weighted average of all sampled colors c_i. An exponential kernel gives more importance to closer data:

$$L_P(v_j) = \sum_i c_i \exp(-\alpha d(v_j, p_i)) / \sum_i \exp(-\alpha d(v_j, p_i)),$$
$$(2)$$

where $L_P(v_j)$ is the color of vertex v_j of the lumisphere and α is set empirically from the number of pictures available and serve to control the region over which a sample has influence (we used 100). The color of any point inside a triangle is obtained using the standard linear interpolation from the color of its vertices using barycentric coordinates. An example result of this computation along with the corresponding data, shown by circles of the corresponding color, is displayed in Figure (1)(d).

From the reconstructed lumisphere, we must compute the frequency criterion, as given by equation (1). This integral is divided over each triangle of the mesh since the gradient is constant within them. The gradient $\nabla L_P(k)$, corresponding to the triangle k of the mesh, is calculated using the colors of its three vertices. Refer to Figure (3) for the geometry involved. The two directional derivatives $\mathbf{J_{k,1}}, \mathbf{J_{k,2}}$ along the edges of each triangle are given by the

following expressions:

$$\begin{array}{rcl}
\mathbf{J_{k,1}} & = & (L_P(v_{k,2}) - L_P(v_{k,1}))/d(v_{k,1}, v_{k,2})\hat{\mathbf{v}}_{\mathbf{k,12}} \\
\mathbf{J_{k,2}} & = & (L_P(v_{k,3}) - L_P(v_{k,1}))/d(v_{k,1}, v_{k,3})\hat{\mathbf{v}}_{\mathbf{k,13}}
\end{array},$$
$$(3)$$

where $\mathbf{v_{k,ij}}$ is the vector joining vertices i and j of triangle k and $\hat{\mathbf{v}}_{\mathbf{k,ij}}$ is the unit vector in the same direction. A linear system is then solved to obtain the gradient of this triangle:

$$\begin{array}{rcl}
\|\mathbf{J_{k,1}}\| & = & < \nabla L_P(k), \hat{\mathbf{v}}_{\mathbf{k,12}} > \\
\|\mathbf{J_{k,2}}\| & = & < \nabla L_P(k), \hat{\mathbf{v}}_{\mathbf{k,13}} >.
\end{array}$$
$$(4)$$

The integration of the squared gradient of equation (1) is finally evaluated by summing the norm of the squared gradient inside each triangle, weighted by the triangle area, summed for the three color channels:

$$\|L_P\|_V = \sum_k \|\nabla L_P(k)\|^2 \cdot \frac{1}{2} \|\mathbf{v_{k,12}} \times \mathbf{v_{k,13}}\|. \quad (5)$$

Direct approach   Along a projector ray, the closest point of the proxy is more likely to be located where the lumisphere has the lowest value of the frequency criterion. Using equation (5), we can thus compute the criterion for a series of lumispheres located at regular interval along the projector ray and retain the position of the one having the lowest value as the corresponding location for the proxy. This procedure is illustrated in Figure (1)(c), where it is visible that the size of the interval used corresponds to the maximum resolution for the proxy.

Why is the direct approach inadequate?   A simple fact was overlooked in the direct approach, which leads to erroneous depths: when gathering the colors of a lumisphere from the acquired viewpoints, the point where the lumisphere is located might not be visible from all viewpoints. When such visibility outliers are present, locating the optimal depth will fail since the erroneous gradients added to equation (5) increase the value of the frequency criterion and shift the position of its minimum. A lumisphere of the Middlebury Temple dataset having such visibility outliers is shown in Figure (1)(d).

The exact visibility information is unfortunately not available since it depends on the geometry of the object. A common practice is to approximate it using the visual hull. To further reduce the occurence of visibility problems, only the viewpoints within 30 degrees of the viewing direction of the projector ray will be used. Although this strategy will remove potential visibility outliers, some will inevitably remain. The most common occurance happens when the object has a cavity not resolved by the visual hull, as illustrated in Figure (1)(c).
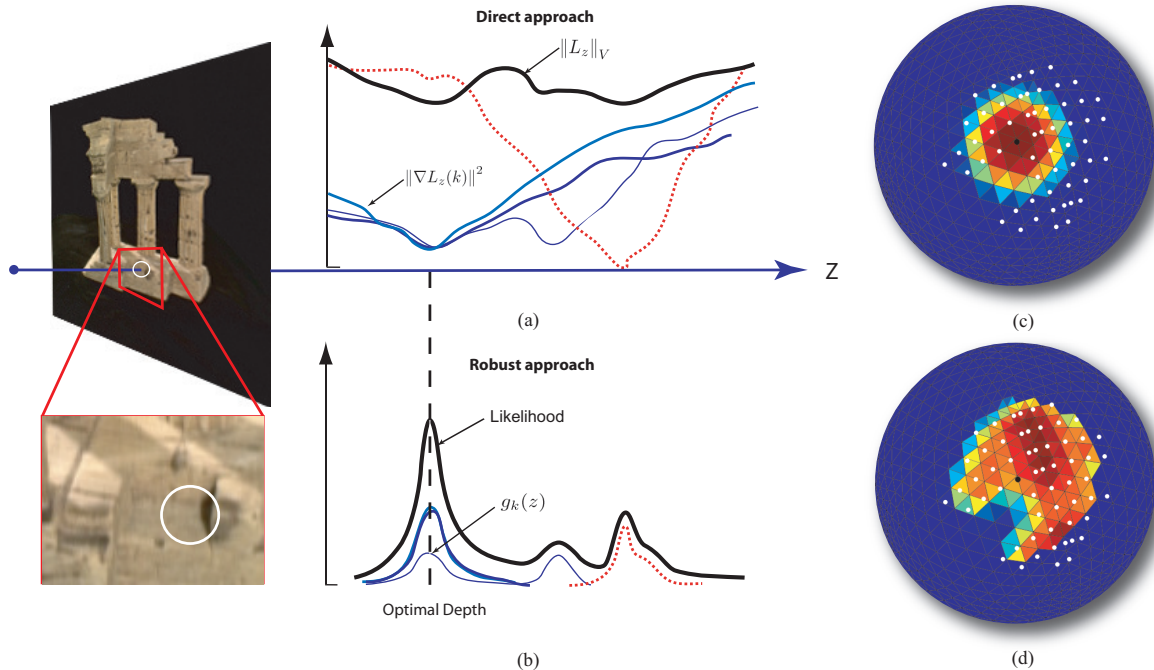
Figure 4: Using equation (5) to minimize the frequency criterion along a projector ray is highly influenced by visibility outliers. This situation is illustrated here along with the proposed solution for a pixel of the Temple dataset projecting inside a cavity. The vote of each triangle is weighted according to its distance from the projector ray considered (black dot), in (c), and from the sampling distribution (white dots) in (d). Warmer colors corrrespond to larger weights.

The robust approach The computation of equation (5) is illustrated in Figure (4)(a) to show where the problem originates. Each of the thinner lines of Figure (4)(a) represents the summand in equation (5) as P is moved through the depths z along a projector ray. The outlier is shown with a red dotted line. The computation of the criterion performed in equation (5) is the sum of all these lines and is displayed as the (offset) thicker black line. While there appears to be three candidate minima when looking at the thinner curves individually, their sum just barely conveys this information. Moreover, the consensus for the position of the minimum estimated by the three blue lines at the left of the graph is shifted by the influence of the red outlier whose gradient norm is much higher. Even doing a consensus pointwise is prone to fail because regions of consensus of high gradient values must not hide a possibly smaller consensus of low gradients which is of greater interest when doing a minimization.

These problems can be avoided by using a voting scheme. Each triangle of the lumisphere mesh votes for the position where it would more likely locate the minimum norm along z of its corresponding gradient. As noted previously, this vote must not be done for a single position. Instead, a probability distribution for the location of the minimum is constructed, as shown by the thin lines in Figure (4)(b). Formally, the likelihood for the position of the minimum obtained from the gradient k is calculated using the following expression:

$$g_k(z) = \exp(-(\|\nabla L_z(k)\|^2 - \|\nabla L_{z_{min}}(k)\|^2)^2/\sigma_k), \quad (6)$$

where σ_k is set for each lumisphere triangle k individually to a value such that g_k becomes negligible when $\|\nabla L_z(k)\|^2$ reaches its median value along the interval considered. This expression is built to correctly place a higher probability at regions where the norm of the gradient is close to its smallest value $\|\nabla L_{z_{min}}(k)\|^2$. Moreover, if multiple depths exhibit values close to the minimum, they all receive a part of the probability accounted for. These individual likelihoods are then normalized and summed for every triangle of the lumisphere. This sum is presented in Figure (4)(b) using a thick black line and shows how the red outlier does not interfere with the minimum on the left. The vote of each triangle is further weighted inversely according to its angular distance from the projector ray, as illustrated in Figure (4)(c). In this way, we have reached our stated goal of integrating all viewpoints in a global way without being biased by outliers.

Non-uniform sampling A final difficulty is caused by the non uniform sampling of the lumispheres. Because of the

varying visibility and non uniform placement of the viewpoints, lumispheres are in practice never regularly sampled. This situation is illustrated in Figure (1)(d) where each sample is represented by a circle of the corresponding color. Therefore, some regions of the lumisphere may contain less samples than others and the gradients in these regions are equally less significant. Furthermore, since they are extrapolated from the same data, these unreliable gradients unfortunately introduce detrimental modes in the likelihood function because they tend to vote for the same location.

To cope with this situation, the vote of each triangle is weighted by how close it is to the data, as illustrated in Figure (4)(d). This weight for each triangle is calculated by summing the distance from its center to the m nearest data (we use $m = 6$ for all our experiments) represented by white dots in the figure. It is thus set to

$$w_k = \exp(-(\sum_{i=1}^{m} d(t_k, p_i))^2/\gamma), \qquad (7)$$

where $t_k = (v_{k,1} + v_{k,2} + v_{k,3})/3$ is the center of the kth triangle and γ is fixed empirically according to the density of data on a typical lumisphere (we use 0.01).

5.2. Merging depths

The merging of the depths obtained from all pixels is a crucial step that selects the correct ones among the many candidates retained. This merging is done at two different levels: first within each viewpoint to obtain the corresponding depth maps and then globally using all viewpoints. A provenly efficient way to perform function approximation from scattered data is by Radial Basis Functions (RBFs) approximation [8]. This tool will be used similarly at both steps.

First, within each viewpoint, all local maxima along the likelihood curve of each pixel whose likelihood exceeds a threshold (we use 0.05) are used as samples for the approximation of the corresponding depth map function. The likelihood values are also used as weights to better fit depths having higher confidence. To add robustness, a Ransac-type procedure is used in the following way. A subset of the points belonging to a viewpoint are selected randomly and the Radial Basis Functions are used to fit a depth map. The depths of all points are then compared to the reconstructed depths and their absolute differences are stored. This process is repeated four times and the differences collected for a point are summed together. Afterwards, a point whose summed difference is above a threshold is considered an outlier.

When doing the final merging of all viewpoints, Radial Basis Functions are used this time to approximate an implicit function whose zero level is the proxy sought, as in [8]. Obtaining a mesh of the desired proxy will then sim-

ply amount to a zero level extraction using the well known marching cubes for example. The first step is thus to gather samples of this implicit function. The local maxima along the likelihood curves that passed the previous step are all potential zero valued points since they are supposed to lie on the proxy surface. Having only zero valued points is however not enough to determine the desired implicit function since a trivial implicit function evaluating to zero everywhere would satisfy all the points. A common technique to solve this problem is to add off-surface points, one on each side of a point of the proxy along the corresponding projector ray. While the point closer to the camera receives a positive value, the other is set negative to encode that they are respectively outside and inside the proxy. A subset of these points is shown in the upper right corner of Figure (5). Points outside are colored red, blue are inside and green are on the proxy. The same Ransac-type procedure is used to eliminate outliers using again likelihood values to weigh the points.

Uncertain regions   There are two types of regions where the data obtained so far is not enough to uniquely determine the shape of the proxy. Such regions will produce undesirable holes or the noise within them will lead to unnecessary curvature in the proxy. The first type is when a group of pixels does not provide a well defined position identifying the location of the proxy. This happens for textureless regions of the object for example. These pixels are easy to identify since no obvious maximum is found in the likelihood along the projector ray. The second type is for regions of the object that are not visible from any viewpoint, underneath the object for example.

Using the information given by the visual hull is a standard technique exploited by various successful algorithms reviewed in [2]. To avoid the complexity of enforcing a constraint during the optimization, we propose the following simple yet effective technique. For regions of the visual hull that are either not visible or not sufficiently well sampled, we add points of the visual hull along with their two off-surface points, forcing the proxy to follow the visual hull in these regions.

6. Results

Proxy of Lambertian Objects   For Lambertian textured or shaded objects, the proxy corresponds to the actual object surface since the frequency criterion cannot be lower than on the object where there is no variation. In this case, it is therefore possible to assess the quality of the implementation by looking at the surface obtained. Recall however that the Lambertian reflectance property was nowhere exploited in the optimization.

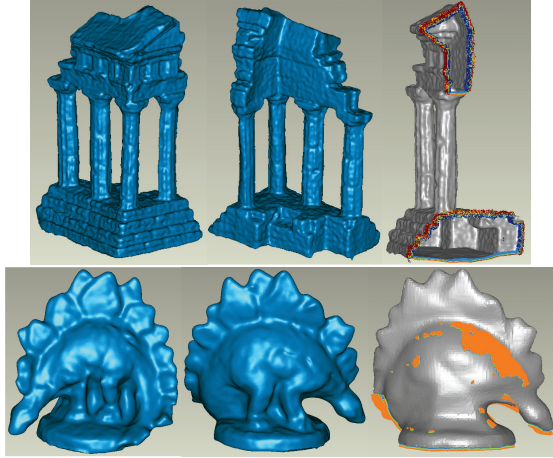Figure (5) shows the proxies obtained for the Temple

Figure 5: Results (accuracy at 90%/completeness at 1.25mm): Temple (0.55mm/99.7%), Dino (0.76mm/99.0%)

Figure 6: Both the depths of the virtual image of the light source and of the plane are found, as shown by the points on the right. The warmer the color the higher is the likelihood of the point.

(312 pictures) and Dino (363 pictures) of the Middlebury dataset[2]. The score obtained for both models compares favorably with the best ones recorded on the Middlebury website[1]. One challenge encountered with the temple is related to the problematic visibility caused by its complex topology and the sharp edges of the stairs and cavities. Consider the projector ray illustrated on the left of Figure (4). The lumisphere of Figure (1)(d) is the one located at the optimal depth along this ray, inside the cavity. The combined votes described in section 5.1 correctly locates its optimal position despite the numerous visibility outliers present. Note that the non-uniform sampling density of this lumisphere was also correctly taken into account and the corresponding vote weights are those visible in Figure (4)(d).

Some portions of the Temple are not visible from any viewpoint, particularly underneath it and below its entablature, between columns. Points of the visual hull that are automatically added to fill these dataless regions are visible in the upper right of Figure (5), colored according to their off-surface position: orange outside, light blue inside and green on the visual hull surface.

The color uniformity of some regions of the Dino produces uncertain regions, particularly on its back. Again, the added points are visible on the lower right of Figure (5), with the same color coding used previously, and correctly fill these regions.

Proxy of Non Lambertian Objects   Results for the challenging Bust and Buddha datasets of [5] are presented in Figure (7). Both objects have complex surface reflectance properties and are pictured 339 and 281 times respectively. The Bust is made from metallically-painted clay while the
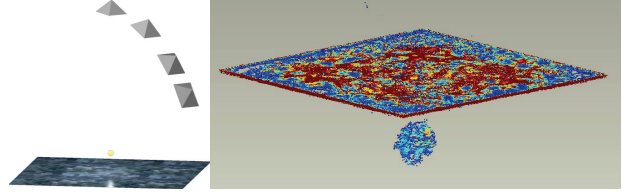
Buddha exhibits combined subsurface scattering and highly specular regions.

For non Lambertian objects, the shape of the proxy is less intuitive. Regions where the surface properties are still close to being Lambertian may correspond to the object surface, but in general this correspondence does not hold. Since the search for the proxy does not rely on the object's surface properties, this result is not surprising. This discrepancy is however not a problem because when interpolating between viewpoints, the proxy surface still produces low interpolation error by construction.

An interesting behavior emerges on regions where there are specularities. We observed that the optimal depths correspond to those of the virtual image of the light source instead of that of the object surface. This is consistent with the framework proposed since the importance of the signal from the virtual light source is superior to that of the object surface, which is hardly visible. These depths may however lead to a deviancy of the proxy from the actual shape, visible for example on the shoulder of the Buddha proxy.

To further support this finding, 196 rendered pictures of a glossy galvanized plane below a single light source were used as input pictures and the optimal depths of four depths maps were combined to produce the cloud points visible on the right in Figure (6). The reconstruction of the virtual image of the light source shows that the proposed approach could be generalized to produce a volumetric proxy, as opposed to a single surface.

View interpolation   The analysis presented so far did not depend on any particular choice of renderer for generating the interpolated views, the only output of the algorithm presented here is a mesh of the proxy. Therefore, as soon as the renderer supports surface light fields, e.g. [4, 5], we should see an increase in quality. We chose to use the Light Field Mapping software [5], which is freely available online.

To assess the quality of the viewpoints generated, we compare them to the corresponding input pictures, visible in the two last images of each group in Figure (7), and to the one obtained using as a proxy the geometry of the model for the synthetic Buddha and that of the scan for the Bust. Note

---

[1]http://vision.middlebury.edu/mview/eval/

Figure 7: Results obtained for two non Lambertian objects. For each group of four pictures, the proxy is shown first along with an input picture, then follow the rendered model using first the proposed proxy and then the actual object shape.

that contrast was enhanced to help comparison. Providing a perceptually meaningful metric to compare the results obtained to the ground truth is beyond the scope of this paper. We thus only rely on a visual assessment of the quality, which is nonetheless an important metric when assessing picture quality.

7. Conclusion

Finding a proxy optimized to reduce view interpolation error allows geometric information in the pictures to be extracted without using matching or more generally any assumption regarding the reflectance properties of the object. Through the robust implementation provided, the method applies to a wide variety of objects and produces results comparable with those of existing approaches that either use additional assumptions or rely on scanning. For objects that exhibit specularities, using a single surface as the proxy is the first limitation future reseach has to overcome.

References

[1] P. Lambert, J.-D. Deschênes and P. Hébert, "A Sampling Criterion for Optimizing a Surface Light Field", in Proc. of the International Conference on 3-D Digital Imaging and Modeling (3DIM 2007), Aug. 2007, pp. 47-54. 1, 2, 3

[2] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein and R. Szeliski, "A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms," in Proc. Conf. Computer Vision and Pattern Recognition (CVPR06), Vol. 1, June 2006, pp. 519-526. 2, 6, 7

[3] H. Jin, S. Soatto, A.J. Yezzi, "Multi-View Stereo Beyond Lambert," in Proc. Conf. Computer Vision and Pattern Recognition (CVPR03), Vol. 1, June 2003, pp. 171-178. 2

[4] D. Wood, D. Azuma, K. Aldinger, B. Curless, T. Duchamp, D. Salesin and W. Stuetzle, "Surface Light Fields for 3D Photography," in Computer Graphics (Proc. SIGGRAPH00), Aug. 2000, pp. 287-296. 2, 3, 7

[5] W.-C. Chen, J.-Y. Bouguet, M. H. Chu and R. Grzeszczuk, "Light Field Mapping: Efficient Representation and Hardware Rendering of Surface Light Fields," ACM Transactions on Graphics, Vol. 21, No. 3, 2002, pp. 447-456. 2, 7

[6] M. Goesele, S. M. Seitz and B. Curless, "Multi-View Stereo Revisited," in Proc. Conf. Computer Vision and Pattern Recognition (CVPR06), Vol. 2, June 2006, pp. 2402 - 2409. 2

[7] N. Campbell, G. Vogiatzis, C. Hernandez and R. Cipolla, "Using Multiple Hypotheses to Improve Depth-maps for Multi-View Stereo," in Proc. of the European Conf. on Computer Vision (ECCV08), Oct. 2008, pp. 766 - 779. 2

[8] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum and T. R. Evans, "Reconstruction and Representation of 3D Objects with Radial Basis Functions," in Proc. of the Conf. on Computer Graphics and Interactive Techniques, 2001, pp. 67-76. 6