*Submitted to SIGGRAPH 2001*

# Image-based Modeling and Rendering of Surfaces with Arbitrary BRDFs

Melissa Koudelka
Peter N. Belhumeur
Center for Comp. Vision & Control
EE and CS
Yale University
New Haven, CT 06520-8267

Sebastian Magda
David J. Kriegman
Beckman Institute
Computer Science
University of Illinois, Urbana-Champaign
Urbana, IL 61801

## Abstract

*A challenge of computer graphics is to render* as realistically as possible *manmade and natural objects. This paper presents a method for image-based modeling and rendering of objects with arbitrary (possibly anisotropic and spatially varying) BRDFs. An object is modeled by sampling the surface's incident light field to reconstruct a non-parametric apparent BRDF at each visible point on the surface. This can be used to render the object under arbitrarily specified illumination. We demonstrate how these image-based objects can be embedded in synthetic scenes and rendered under global illumination or composited into video footage.*

## 1 Introduction

A goal within many computer graphics and video applications is to render as accurately as possible scenes composed of natural and synthetic objects. Advances in computation and global rendering techniques can now effectively simulate the most significant radiometric phenomena to produce accurate renderings so long as the geometric and reflectance models are accurate. Yet while researchers and practitioners have succeeded in developing accurate reflectance models for manufactured objects such as plastics and metals, they have stumbled in the development of reflectance models for natural objects. Consider the challenges of modeling and accurately rendering materials like leather, wrinkled human skin, shag carpeting, the fur on an old mare, or a plate of greasy French fries. These objects have extremely complex reflectance properties, including spatial nonhomogeneity, anisotropy, and subsurface scattering.

Consequently, image-based modeling and rendering techniques have recently emerged for renderings of natural or complex scenes. Yet most efforts focus on viewpoint variation often at the expense of the ability to control lighting. For example, image-based texture maps are effective when the synthetic lighting during rendering is similar to that during acquisition. This makes it difficult to embed these and other image-based object models (e.g., Lumigraphs/light fields) in either synthetic or natural scenes unless the lighting in these composed scenes is identical to that in the image-based object model acquisition. To more effectively handle arbitrary lighting, recent methods have attempted to recover reflectance properties of objects [7, 27, 20] though they generally assume that the reflectance properties at each point can be characterized by a few parameters and that their variation across the surface is also characterized by a few parameters (e.g., albedo variation).

It has long been recognized that while ad-hoc reflectance models such as Phong can be used to represent certain materials (e.g., smooth plastics), these models do not effectively capture the reflectance of materials such as metals and glazed ceramics. As a result, a number of physics-based reflectance models have been developed to more accurately capture the reflectance of rough metals [23, 6], matte surfaces [18] or objects with anisotropic reflectance [24]. Important to developing these physical models is the understanding of how materials such as dialectrics reflect light, but perhaps more important is the understanding of how to characterize the micro-structure of the surface and the impact of shadowing, interreflection, masking and foreshortening. More elaborate models of the surface geometry (micro-facets) have been developed [2, 13, 18] yielding more realistic reflectance functions. Yet each of these only characterizes a limited class of surfaces and none of them address the nonhomogeneity of reflectance functions over the surfaces of objects.

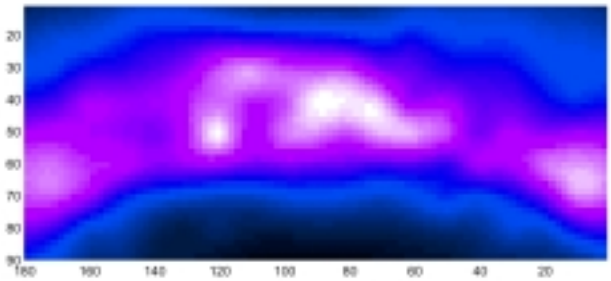


Figure 1: Not all BRDFs have a simple lobe structure. Images of a small teddy bear were acquired as a point light source was moved over a quarter of a sphere. The plot shows the measured intensity of one pixel as a function of light source direction, with the direction specified in degrees of spherical angles.

In this paper, we present a method for rendering images of objects with arbitrary reflectance functions. The reflectance functions may be anisotropic and spatially varying. The illumination of the object is likewise unrestricted. We adopt a common assumption that the reflectance of an object can be locally modeled by a bidirectional reflectance distribution function (BRDF). A general BRDF at a point on the surface is the ratio of the outgoing radiance to the incident irradiance. The BRDF can be represented as a positive 4-D function $\rho(\hat{\mathbf{i}}, \hat{\mathbf{e}})$ where $\hat{\mathbf{i}}$ is the direction of an incident light ray, and $\hat{\mathbf{e}}$ is the direction of the outgoing ray.

As a simple empirical illustration of the complexity of the BRDFs of real surfaces, consider the plot shown in Fig. 1. For a small teddy bear, covered with a velour-like material, this plot shows the measured intensity of one pixel on a convex portion of the object as an isotropic point source is moved over a quarter of a sphere at approximately a constant distance from the surface. Notice the broad, bending band and the multiple peaks – this is qualitatively very different than the BRDFs arising from reflectance models such as Phong.

Our method for image-based modeling uses only a single viewpoint of the object, but *many* images of the object illuminated by a point light source moved over some surface about the object. This surface should be star-shaped (e.g., convex) with respect to all surface points, and in practice we use a sphere. If a second set of images, obtained by moving a point source over a second star-shaped surface, is available, then the method in [1] can be used to reconstruct the surface's geometry. We use the recovered shape and sampling of the object's light field to efficiently and accurately render

images of real objects under novel illumination conditions. In particular, we estimate an apparent BRDF for the surface patches corresponding to every pixel. This apparent BRDF differs from the true BRDF in that it is expressed in a global coordinate system and it includes (1) shadows that the object might cast upon itself, (2) the cosine foreshortening term, and (3) the effects of interreflection from the object onto itself. When using this apparent BRDF for rendering, it will exactly account for self shadowing and foreshortening, and the interrelationship will often closely approximate those that would occur in a real scene.

Consider other approaches to image-based modeling and rendering. In [3, 9, 11, 12, 14, 16, 21], images of a real object taken from multiple viewpoints are used to render synthetic images of the object from arbitrary viewpoints. In [3, 9, 11, 14, 21], few real images are needed for the synthetic renderings, but the method first must determine a 3-D model of the scene by establishing the correspondence of feature pixels in the real images. A radical departure from reconstruction or correspondence-based approaches to image-based rendering is the 4-D lumigraph [12] or light field [16]. (See also [26]). In these methods, renderings of the object from novel viewpoints can be synthesized without any 3-D model of the scene; however, thousands of real images are needed for accurate renderings. See the discussion in Section 2.

In contrast to this work, we present a method for rendering synthetic images of an object or scene from a fixed viewpoint, but under arbitrary illumination conditions. The method uses many images of an object illuminated by point light sources, to recover the object's shape and then to estimate an apparent BRDF. The problem of synthesizing images for Lambertian surfaces with light sources at infinity without shadows is considered in [22] and with shadows in [4]. Methods for re-rendering images with diffuse linear combinations of images formed under diffuse light is considered in [17]. In [25, 7], a method is proposed for performing image-based rendering under variable illumination by estimating an apparent BRDF (for a fixed viewing direction) associated with each scene point by systematically moving light sources at infinity. However, to synthesize images for nearby light sources, 3-D scene geometry as well as the apparent BRDF at each point is needed, and it is assumed that geometry has been acquired by some other means (e.g., a range finder).

The rest of this paper is organized as follows. In the next section, we detail our method for image-based modeling and rendering, presenting the results of rendering isolated objects with complex BRDFs under novel lighting conditions. In Section 3, we present a method for embedding real objects in complex (possibly synthetic) scenes. In Section 4, we present a new method for, and results from, compositing these image-based objects into a video stream. Finally in Section 5, we summarize our method [1] for reconstructing the object's geometry using a double covering of the surface's incident light field.

## 2 Image-based Modeling and Rendering

At SIGGRAPH 1996, two papers introduced a novel approach to image-based rendering of natural 3-D scenes from arbitrary viewpoints [12, 16]. Rather than modeling object geometry and the reflectance function across the surface as is traditionally done in computer graphics, the approach is based on directly representing the radiance in all directions emanating from a scene under fixed illumination. As discussed in [15], the set of light rays is a four-dimensional manifold. Under static illumination, the radiance along a ray in free space is constant. Note that this reduces the 5-D plenoptic function to 4-D [10]. Now consider surrounding a scene by a closed smooth convex surface. By moving a camera with its two-dimensional image plane over the entire surface (a 2-D manifold), one can sample the intensity along every ray emanating from the surface's convex hull. In doing this, one obtains a function on the 4-D ray space $\mathcal{L}$ which has been called the Lumigraph [12] or light field [16].

For any viewpoint $\mathbf{o}$ outside of the surface, an image can be synthesized by considering the radiance of all of the rays passing through $\mathbf{o}$. The set of rays passing through $\mathbf{o}$ is simply a two-dimensional subset of $\mathcal{L}$, and the radiance of those rays that intersect the image plane are used to compute the irradiance at each point of the synthesized image. This turns rendering into a problem of simply indexing into a representation of $\mathcal{L}$ rather than ray tracing, for example. The advantages of such an approach are that the representation is constructed directly from images without needing reconstruction or correspondence, that no assumptions about the surface BRDF are required, and that interreflections do not need to be computed since they occurred physically when the images were acquired. The main challenges and issues of this approach are interpolation from the finite number of sample images, compression of the lumigraph/light field representation which is constructed from a very large number of images, and accurate camera localization during modeling.

In [12, 16] the illumination must be fixed during modeling, and all synthesized images are valid only under the same illumination; this complicates the rendering of scenes composed of both traditional geometric models and lumigraphs/light fields under general lighting. It is natural to ask whether one could "turn the lumigraph/light field around" and synthesize images under fixed pose, but variable lighting. As described in [15], the space of source rays illuminating a scene is also four-dimensional. Like the rays passing through a camera's optical center, the set of light rays emanating from a point light source is two-dimensional. Hence, by moving an isotropic point source over a closed surface (a 2-D manifold) bounding a scene, images can be acquired for all possible source rays crossing this surface. Let the surface of point source locations be given by $\mathbf{s}(\phi, \psi)$ and the corresponding images be given by $\mathbf{I}(\phi, \psi)$. We call the collection of images denoted $\mathbf{I}(\phi, \psi)$ the object's illumination dataset.

Now consider synthesizing an image from the same viewpoint, but under completely different lighting conditions. The applied lighting is a function on the 4-D light ray space. For a single illumination ray $\mathbf{g}$ arising from some source $\mathbf{s}_{syn}$, we can find the intersection of the ray with the surface of point sources. The intersection is a point light source location $\mathbf{s}' = \mathbf{s}(\bar{\phi}, \bar{\psi})$ and there is a corresponding image $\mathbf{I}(\bar{\phi}, \bar{\psi})$. There exists a light ray emanating from the point source $\mathbf{s}(\bar{\phi}, \bar{\psi})$ coincident with $\mathbf{g}$ which intersects the scene, and sheds light onto some image pixel. However, it is not evident which pixel of $\mathbf{I}(\bar{\phi}, \bar{\psi})$ corresponds to the intersection of the illuminating ray and the surface – unless the 3-D position of the point $\mathbf{p}$ is known. See Fig. 2.

These observations lead us to a method for rendering an image of a scene illuminated by a synthetic point light source $\mathbf{s}_{syn}$ that does not necessarily lie on the surface defined by $\mathbf{s}(\phi, \psi)$. The method can clearly be extended to render scenes under any illumination, e.g., area sources, strip sources, point sources at infinity, etc. First, the 3-D positions of points $\mathbf{p}$ on the surface must be obtained and pixel registered with the image set. (We summarize such a method in Section 5 for objects with arbitrary BRDFs.)

Next, the surface of light source locations $\mathbf{s}(\phi, \psi)$, is triangulated. To determine the intensity of a pixel $\mathbf{q}$ for point light source $\mathbf{s}_{syn}$, we find the intersection $\mathbf{s}'$ of the ray defined by $\mathbf{p}$ (the surface point corresponding to pixel $\mathbf{q}$) and $\mathbf{s}$ with one of the triangles in the triangulation. See again Fig. 2. If the intersection happens to be a vertex, then the intensity of the pixel in the corresponding measurement image could be used. Since this is rarely the case, we instead interpolate the intensities of corresponding pixels associated with the three vertices to estimate the intensity $i'(\mathbf{q})$ for a fictitious point light source at $\mathbf{s}'$. Because the solid angle of the surface corresponding to $\mathbf{q}$ as seen by $\mathbf{s}_{syn}$ and $\mathbf{s}'$ depends on the squared distance, the pixel is rendered with

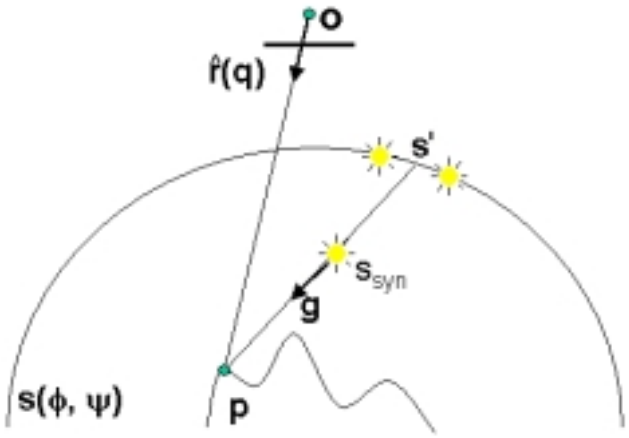


Figure 2: To determine the intensity $i(\mathbf{q})$ of pixel $\mathbf{q}$ for a point light source $\mathbf{s}_{syn}$ that does not lie on $\mathbf{s}(\phi, \psi)$, knowledge of the 3-D position of the corresponding surface point $\mathbf{p}$ is required. (Note that the surface point $\mathbf{p}$ is viewed from the direction $\hat{\mathbf{r}}(\mathbf{q})$ by the pixel $\mathbf{q}$.) If the 3-D position of $\mathbf{p}$ is known, the intersection $\mathbf{s}'$ of the ray from $\mathbf{p}$ through $\mathbf{s}_{syn}$ with the triangulated surface of light sources can be determined. Based on the vertices (sample light sources) of the triangle containing $\mathbf{s}'$, the image intensity $i(\mathbf{q})$ is computed by interpolating the measured pixel intensities in the images formed under light sources located at the vertices. Here the light source located at the vertices are represented by the light source icons on either side of $\mathbf{s}'$.

$$i(\mathbf{q}) = \frac{||\mathbf{p}(\mathbf{q}) - \mathbf{s}'||^2}{||\mathbf{p}(\mathbf{q}) - \mathbf{s}_{syn}||^2} i'(\mathbf{q}). \tag{1}$$

This process is repeated for each pixel $\mathbf{q}$ in the rendered image.

Note that the 3-D position of $\mathbf{p}$ is used in two places. First it is used to index into the set of triangles, and secondly it is used to determine the $1/r^2$ loss. For each pixel in the synthesized images, this procedure can be performed independently. Images for multiple point light sources can be synthesized simply through superposition of the images formed for each light source, weighted by the relative strength of the light sources.

Figure 5: To render the image of the cup, top, from a point light source located 23 cm away, intensities from a number of images in the object's dataset were interpolated. The lower left image color codes each pixel in the rendered image with the triplet of light sources whose corresponding images were used to render that pixel. On the bottom right, the corresponding triplets of light source positions are shown. The dots on the sphere denote the light source positions. The color triangles denote triplets of light source positions. The three images acquired by the three light source positions in the triplet are used to render the intensities of the like-colored regions in the synthetic image, bottom left and top.

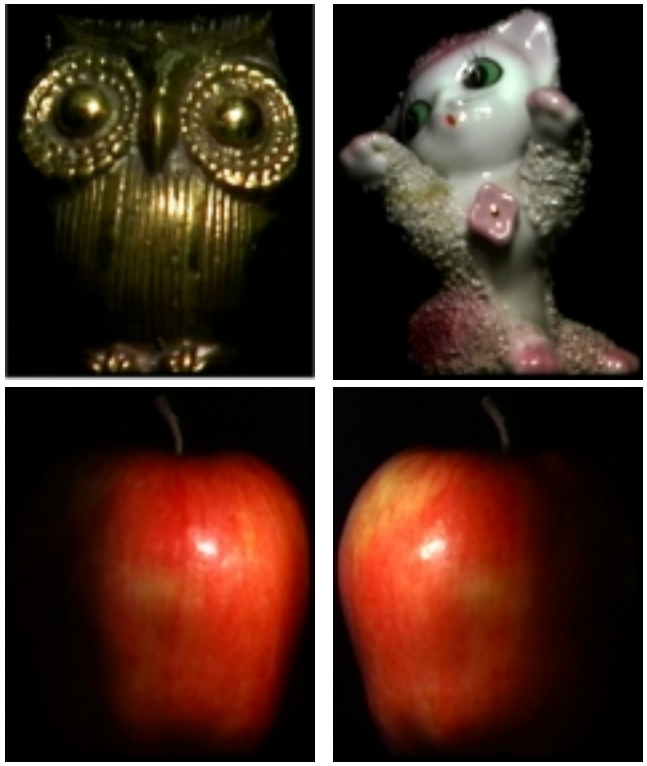

Figure 6: Synthetic images of three different objects were rendered for this figure: a dirty brass owl, top left; a ceramic figurine, top right; and a red delicious apple, bottom left and right. The objects are rendered under point light sources with locations significantly different from those in the objects' respective illumination datasets. The rendering technique used is that described above and in Fig. 5.

shows two examples of a scene containing a pitcher (the "real" object) under two different lighting conditions. For shadow and reflection ray calculations, the pitcher is represented by a 3-D model. The surface of the model was rendered using a custom surface shader, which uses the array of images of the pitcher as shown in Fig. 4 to perform image-based rendering. For each incoming light ray from the scene, the three images with light directions closest to the ray direction are used to interpolate the color of the surface.

## 4 Compositing Real Objects in Video

Here, we introduce a method for automatically compositing a real object into a video sequence using a set of images of the object captured with fixed pose and variable illumination. Similar systems for compositing synthetic objects into real scenes have been proposed in [8, 19]. Their techniques address inserting synthetic objects with known geometry and reflectance properties. In contrast, our technique is used to insert a real object with an arbitrary geometry and BRDF by indexing into a previously captured illumination dataset. In particular, this method is designed (1) to composite real objects onto real video such that the illumination on the composited object matches the illumination in the real scene, (2) to provide realistic shadowing onto the background scene under multiple illuminates, (3) to capture the effects of interreflection between the composited object and the background scene.

To convincingly composite an object within a scene captured in a video sequence requires knowledge of how the scene is illuminated – not at just one moment in time but across the whole temporal sequence. If we could measure, as a function of time, the light field surrounding the location where the object is to be composited, then we could render, using the method described in Section 2, images of the object with the correct illumination. Yet due to the difficulties in dynamically gathering light fields, we employ a simpler alternative. Instead of measuring the light field, we dynamically measure the radiance at the location where the object is to be composited. Thus, we simultaneously gather two video sequences: one of the actual scene and the other of the radiance at the location within the scene where the object is to be composited. For each frame in the sequence, we use the gathered radiance maps to select the appropriate collection of images from the illumination dataset of the object to be composited. If the radiance map for a video frame contains a single point light source, only one image from the illumination dataset is used. If the radiance map contains multiple or extended sources, the image of the object is rendered as the superposition of the singularly illuminated images.

This approximation works quite well for pixels on the object, but it ignores the effects of shadows and interreflections cast from the object onto the scene background. To overcome this, we gather our illumination datasets in a manner that allows us to model these effects. In particular, in addition to the illumination dataset we gather a background illumination dataset; the background illumination dataset is identical to the illumination dataset except that the object has been removed from the scene. The addition of the back-

Figure 7: In this scene rendered using the Blue Moon Rendering Tool Kit, the background, supporting plane, and apple are synthetic while the ceramic cup is real.

ground image allows us to compute – through simple ratios of the two images – how the intensity and color of the background (outside of the boundary of the object) is altered by the presence of the object. This effect is expressed by a radiance scale factor. In shadow regions, the background will be darkened and the radiance scale factor is less than one; in regions of interrefletion, the background will be lightened and the radiance scale factor is greater than one. The radiance scale factor is then used to darken or brighten the region surrounding the object in the composite frame. The method is exact if the background in the scene and the background in the illumination dataset have the same geometry and reflectance and is an approximation otherwise.

We present our complete method through the following compositing example. A video sequence was filmed using a Sony HandyCam mini digital video camera. Simultaneously, the radiance at the point where the object will be composited was captured using a Nikon CoolPix 990 digital camera with a fisheye lens attachment. The camera's MPEG capture capability was used to acquire a video sequence; the field of view of the fisheye was slightly larger than a hemisphere; and the camera was radiometrically calibrated. Together, this allowed us to construct a sequence of radiance maps that captured how the lighting environment varied over time. (Our radiance maps are similar to those obtained with metallic spheres in [8], though the price paid for obtaining a time sequence of radiance maps through video acquisition was limited dynamic range.) In each frame, the pixel coordinates of the light sources in the fisheye radiance maps were determined automatically and used to select the images in the illumination dataset corresponding to the closest captured light source locations. The object was then composited into the video frame using a superposition of the selected images and the radiance scale factor adjustment described in more detail at the end of this section. Note that the illumination datasets were acquired with light sources located at every two degrees in azimuth and elevation over the upper front quadrant of a sphere.

A one-time segmentation was performed to label the pixels as corresponding to either the object or the background; this process was done by hand, but it would be a simple extension to automate this process as well. Since the viewpoint was fixed, it was sufficient to segment a single image within each illumination dataset. A large "active" region encompassing the object, shadows, and interreflections in the entire illumination dataset was also hand selected, although only a rough boundary was needed. The radiance scale factor $m(\mathbf{q})$ was computed for each pixel $\mathbf{q}$ in the active region as follows

$$m(\mathbf{q}) = i_{object}(\mathbf{q}) / i_{back}(\mathbf{q}) \tag{3}$$

where $i_{object}(\mathbf{q})$ was the intensity of the pixel $\mathbf{q}$ of the appropriate illumination dataset image with the object present, and $i_{back}(\mathbf{q})$ was the intensity of the pixel $\mathbf{q}$ of the corresponding illumination dataset

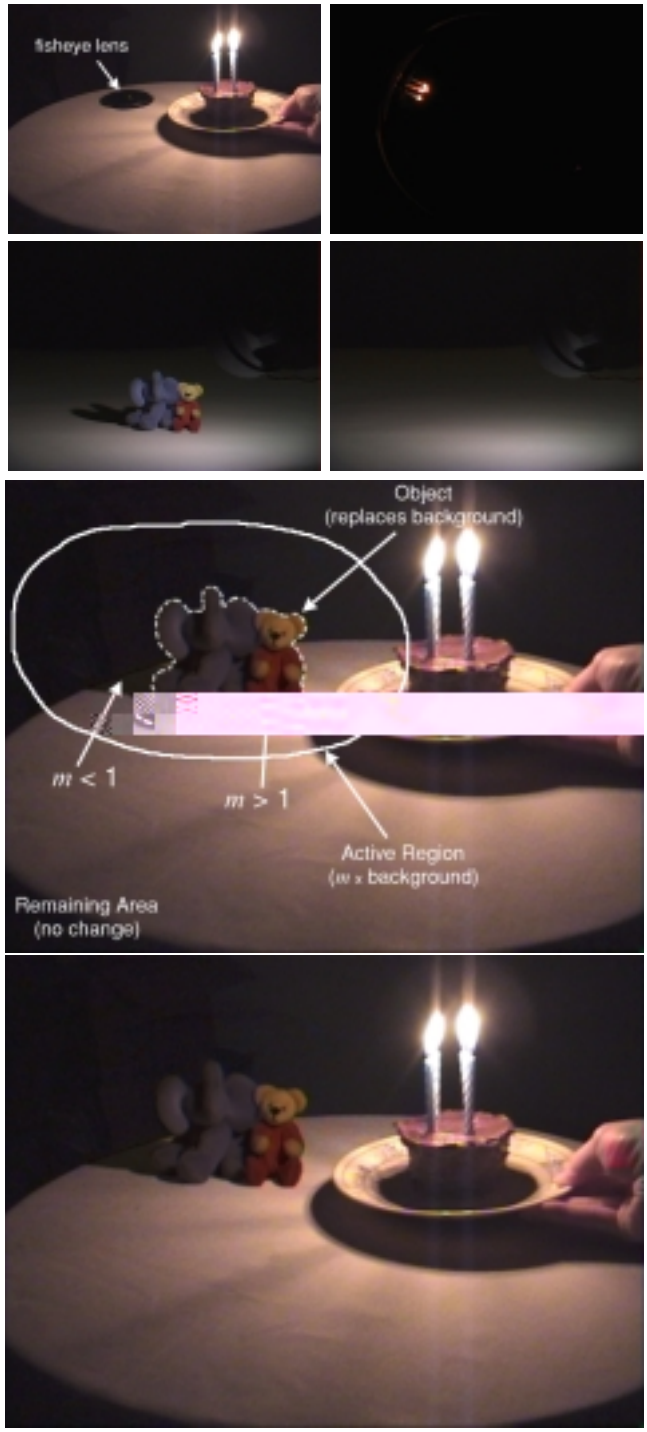


Figure 8: This figure shows how images of an object are composited into video footage. The top row shows a still frame from the video sequence, left, and the recorded radiance map, right. Note that the dark circle, top left, is the fisheye lens for the Nikon camera recording the radiance map, top right. The second row shows the image from the illumination dataset, left, and the image from the background illumination dataset corresponding to one of the light sources detected in the radiance map. The third row shows a diagram outlining the segmented object boundary and the active region in the composite image. The fourth row shows the composite image.

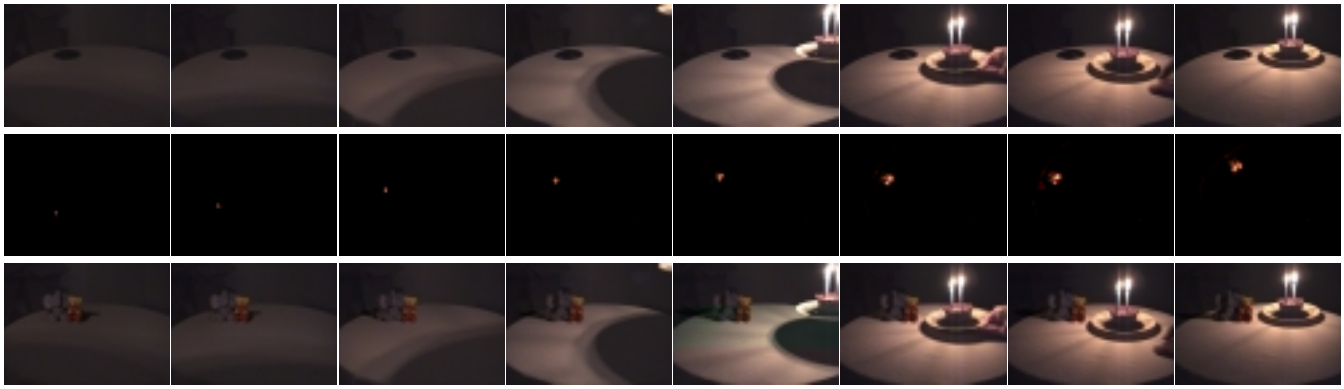

Figure 9: The above images show 8 still frames from a video compositing sequence of a birthday party. The top row shows the still frames before images of an object are composited. The middle row shows the still frames recording the incident radiance at the point on the table where the object is to be composited. The radiance is measured by a Nikon Coolpix 990 camera operating in video mode. The Nikon camera is outfitted with a fisheye lens with a 183 degree field of view. The radiance images from the Nikon camera are automatically processed and used to render – frame by frame – the appropriate images of the object to be composited. The bottom shows the still frames after images of the object have been composited.

background image. For color images, this factor can be computed separately for each of the three color channels.

The pixel value in the final composite image was computed as follows: if the pixel was on the object to be composited, we assigned the object image value to the composite image; if the pixel was in the active region but not on the object, we scaled the video frame value by the radiance scale factor $m$ and assigned it to the composite image; and if the pixel was outside of the active region, we assigned the video frame value to the composite image.

The above procedure is illustrated in Fig. 8. The top row shows a still frame from the video sequence, left, and the recorded radiance map, right. Note that the dark circle, top left, is the fisheye lens for the Nikon camera that recorded the radiance map, top right. The second row shows the image from the illumination dataset, left, and the image from the background illumination dataset, right, corresponding to one of the light sources detected in the radiance map. The third row shows a diagram outlining the segmented object boundary and the active region in the composite image. The fourth row shows the composite image. Note the consistent shadow cast by the animals onto the supporting table.

Figure 9 shows still frames from a composite video sequence. The top row shows the original frames, without compositing. The middle row shows the corresponding radiance maps. The bottom row shows the composited frames. The sixth column from the left has the frames described in Fig. 8.

## 5 Reconstruction

In this section, we overview a method for surface reconstruction that is described in detail in [1]. We include this method only for completeness. The method resembles photometric stereo in that a single viewpoint and multiple lighting directions are used, yet differs significantly in that depth is directly estimated, and no assumptions are made about the surface BRDF.

Consider a fixed calibrated pinhole camera observing a static scene. Let the coordinates of a point on the image plane be given by $\mathbf{q} \in \mathbb{R}^2$. For every $\mathbf{q}$, there is a line passing through the optical center $\mathbf{o}$ in the direction $\hat{\mathbf{r}}(\mathbf{q})$ (i.e., the line of sight of pixel $\mathbf{q}$). See again Fig. 2. The image point $\mathbf{q}$ is the projection of a scene point $\mathbf{p}$ lying on the line defined by $\mathbf{o}$ and $\hat{\mathbf{r}}(\mathbf{q})$. The depth $\lambda(\mathbf{q})$ of $\mathbf{p}$ from $\mathbf{o}$ is unknown, and the relation can be expressed as

$$\mathbf{p}(\mathbf{q}, \lambda) = \lambda(\mathbf{q})\, \hat{\mathbf{r}}(\mathbf{q}) + \mathbf{o}. \tag{4}$$

The process of reconstruction is to estimate the depth map $\lambda(\mathbf{q})$, in this case from images gathered under different lighting conditions. Since we will be able to independently estimate $\lambda$ for each $\mathbf{q}$, we will drop $\mathbf{q}$ from our notation.

Consider the scene to be illuminated by an isotropic point light source (not at infinity) whose location $\mathbf{s} \in \mathbb{R}^3$ is known. The direction of the light ray from $\mathbf{s}$ to $\mathbf{p}$ is $\hat{\mathbf{d}}(\mathbf{s}, \lambda) = \frac{1}{||\mathbf{p}(\lambda) - \mathbf{s}||}[\mathbf{p}(\lambda) - \mathbf{s}]$, while the distance between $\mathbf{s}$ and $\mathbf{p}$ is $d(\mathbf{s}, \lambda) = ||\mathbf{p}(\lambda) - \mathbf{s}||$. Recall from Section 2, we consider an apparent BRDF $\rho(\hat{\mathbf{d}}, \hat{\mathbf{r}})$ in a global coordinate system as a function of the incoming light ray $\hat{\mathbf{d}}$ and the outgoing direction $\hat{\mathbf{r}}$.

The image intensity measured at $\mathbf{q}$ is a function of the light source intensity, $d^2(\mathbf{s}, \lambda)$ and $\rho(\hat{\mathbf{d}}, \hat{\mathbf{r}})$. Let all images be acquired with the same light source which can be taken to have unit intensity. As in Section 2, the measured image intensity (irradiance) for image point $\mathbf{q}$ corresponding to a surface point at depth $\lambda$ illuminated by light source $\mathbf{s}$ can be expressed as

$$i(\hat{\mathbf{d}}(\mathbf{s}, \lambda)) = \frac{1}{d^2(\mathbf{s}, \lambda)} \rho(\hat{\mathbf{d}}(\mathbf{s}, \lambda), \hat{\mathbf{r}}). \tag{5}$$

We now consider a method for simultaneously estimating the apparent BRDF and the depth. Consider moving a point light source over two non-intersecting star-shaped surfaces. For i=1,2, parameterize each surface by $(\phi_i, \psi_i)$, and the position of the light source on surface $i$ is expressed as $\mathbf{s}_i(\phi_i, \psi_i)$. For every light source position $\mathbf{s}_i(\phi_i, \psi_i)$, an image $i_i(\phi_i, \psi_i)$ is measured.

For every light source position from the first (inner) surface $\mathbf{s}_1(\phi_1, \psi_1)$, there is a light source on the second (outer) surface $\mathbf{s}_2(\phi_2, \psi_2)$ where the ray from $\mathbf{p}$ through $\mathbf{s}_1(\phi_1, \psi_1)$ is identical to the ray from $\mathbf{p}$ through $\mathbf{s}_2(\phi_2, \psi_2)$. We can express this correspondence of light sources on the two surfaces as a change of coordinates $\phi_2(\phi_1, \psi_1; \lambda)$ and $\psi_2(\phi_1, \psi_1; \lambda)$. This change of coordinates depends on the unknown location of $\mathbf{p}$, and so it is parameterized by the depth $\lambda$.

For such a pair of light sources $\mathbf{s}_1(\phi_1, \psi_1)$ and $\mathbf{s}_2(\phi_2, \psi_2)$, the value of the apparent BRDF $\rho(\hat{\mathbf{d}}, \hat{\mathbf{r}})$ is the same, and so the image intensities are related by

$$\begin{aligned} i_2(\phi_2, \psi_2) &= \frac{[d_1(\mathbf{s}_1(\phi_1, \psi_1), \lambda)]^2}{[d_2(\mathbf{s}_2(\phi_2, \psi_2), \lambda)]^2} i_1(\phi_1, \psi_1) \\ &= i_2(\phi_2(\phi_1, \psi_1; \lambda), \psi_2(\phi_1, \psi_1; \lambda)) \end{aligned} \tag{6}$$

This relation between the intensities for corresponding light sources

can be used to form an objective function in the depth $\lambda$:

$$\mathcal{O}(\lambda) = \int\int [d_2^2(\lambda) i_2(\phi_2(\phi_1, \psi_1; \lambda), \psi_2(\phi_1, \psi_1; \lambda)) - d_1^2(\lambda) i_1(\phi_1, \psi_1)]^2 d\phi_1 d\psi_1. \quad (7)$$

The depth $\lambda$ is then found by minimizing $\mathcal{O}(\lambda)$, which would be zero without any image noise. At such a minimum, we have found a depth where the correspondence between light sources on the two surfaces leads to image intensities which are consistent. Further details and examples of reconstructions based on this method can be found in [1].

## 6 Discussion

We have presented a method for rendering novel images of an object under arbitrary lighting conditions. The method correctly handles shadowing without the need for ray tracing and can synthesize point, anisotropic, extended, or any other type of light source.

There are, of course, many issues to explore. As in the lumigraph work [5], what is the relation of the BRDF and geometry to the sampling rate of light sources that yields effective renderings? What are efficient ways to compress the presumably redundant information for most scenes? What are fast ways to render images using the resulting representation? How can such methods be extended to handle different viewpoints as well as illumination? Note that we only really recover a 2-D slice of the BRDF at each point. Are there principled means to extrapolate the apparent 4-D BRDF from the 2-D slice so we can correctly render novel viewpoints?

## References


[1] Anonymous. Beyond Lambert: Reconstructing surfaces with arbitrary BRDFs. In *Int. Conf. on Computer Vision*, 2001. submitted.

[2] M. Ashikhmin, S. Premoze, and P. Shirley. A microfacet-based brdf generator. In *SIGGRAPH*, pages 65–74, 2000.

[3] S. Avidan and A. Shashua. Novel view synthesis in tensor space. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 1034–1040, 1997.

[4] P. N. Belhumeur and D. J. Kriegman. What is the set of images of an object under all possible lighting conditions. *Int. J. Computer Vision*, 28(3):245–260, 1998.

[5] Jin-Xiang Chai, Xin Tong, Shing-Chow Chan, and Heung-Yeung Shum. Plenoptic sampling. *SIGGRAPH*, pages 307–318, July 2000.

[6] R.L. Cook and K.E. Torrance. A reflectance model for computer graphics. In *SIGGRAPH*, pages 307–=316, 1981.

[7] P. Debevec, T. Hawkins, C. Tchou, H-P Duiker, Westley Sarokin, and M. Sagar. Acquiring the reflectance field of a human face. In *SIGGRAPH*, pages 145–156, 2000.

[8] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. *Proceedings of SIGGRAPH 98*, pages 189–198, July 1998.

[9] P.E. Debevec, C.J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *SIGGRAPH*, 1996.

[10] E.H. Edelson and J.R. Bergen. Computational models of visual processing. In Landy and Movshon, editors, *The Plenoptic Function*. MIT Press, 1991.

[11] Y. Genc and J. Ponce. Parameterized image varieties: A novel approach to the analysis and synthesis of image sequences. In *Int. Conf. on Computer Vision*, pages 11–16, 1998.

[12] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The lumigraph. In *SIGGRAPH*, pages 43–54, 1996.

[13] J.J. Koenderink, A.J. vanDoorn, K.J. Dana, and S.K. Nayar. Bidirectional reflection distribution function of thoroughly pitted surfaces. *Int. J. Computer Vision*, 31(2/3):129–144, April 1999.

[14] K. Kutulakos and J. Vallino. Calibration-free augmented reality. *IEEE Trans. Visualization and Computer Graphics*, 4(1):1–20, 1998.

[15] M.S. Langer and S.W. Zucker. What is a light source? In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 172–178, San Jaun, PR, 1997.

[16] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH*, pages 31–42, 1996.

[17] J. Nimeroff, E. Simoncelli, and J. Dorsey. Efficient rerendering of naturally illuminated environments. In *Eurographics Symposium on Rendering*, Darmstadt, Germnay, June 1994.

[18] M. Oren and S.K. Nayar. Generalization of the Lambertian model and implications for machine vision. *Int. J. Computer Vision*, 14:227–251, 1996.

[19] Imari Sato, Yoichi Sato, and Katsushi Ikeuchi. Acquiring a radiance distribution to superimpose virtual objects onto a real scene. *IEEE Transactions on Visualization and Computer Graphics*, 5(1):1–12, Jan.-Mar. 1999.

[20] Yoichi Sato, Mark D. Wheeler, and Katsushi Ikeuchi. Object shape and reflectance modeling from observation. *SIGGRAPH*, pages 379–388, August 1997.

[21] S.M. Seitz and C.R. Dyer. View morphing. In *SIGGRAPH*, pages 21–30, 1996.

[22] A Shashua. On photometric issues in 3D visual recognition from a single image. *Int. J. Computer Vision*, 21:99–122, 1997.

[23] K.E. Torrance and E.M. Sparrow. Theory for off-specular reflection from roughened surfaces. *JOSA*, 57:1105–1114, 1967.

[24] G. J. Ward. Measuring and modelling anisotropic reflection. In *SIGGRAPH*, pages 265–272, 1992.

[25] T.T. Wong, P.A. Heng, S.H Or, and W.Y. Ng. Illuminating image-based objects. In *Proceedings of Pacific Graphics*, pages 69–78, Seoul, October 1997.

[26] Daniel N. Wood, Daniel I. Azuma, Ken Aldinger, Brian Curless, Tom Duchamp, David H. Salesin, and Werner Stuetzle. Surface light fields for 3d photography. *SIGGRAPH*, pages 287–296, July 2000.

[27] Yizhou Yu, Paul Debevec, Jitendra Malik, and Tim Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. *Proceedings of SIGGRAPH 99*, pages 215–224, August 1999.