Determining Reflectance Parameters and Illumination Distribution from a Sparse Set of Images for View-dependent Image Synthesis

Ko Nishino[†], Zhengyou Zhang[‡] and Katsushi Ikeuchi[†]

[†]Dept. of Info. Science, Grad. School of Science
The University of Tokyo, JAPAN
{kon, ki}@cvl.iis.u-tokyo.ac.jp

[‡]Microsoft Research
Redmond WA, U.S.A.
zhang@microsoft.com

Abstract

A framework for photo-realistic view-dependent image synthesis of a shiny object from a sparse set of images and a geometric model is proposed. Each image is aligned with the 3D model and decomposed into two images with regards to the reflectance components based on the intensity variation of object surface points. The view-independent surface reflection (diffuse reflection) is stored as one texture map. The view-dependent reflection (specular reflection) images are used to recover the initial approximation of the illumination distribution, and then a two step numerical minimization algorithm utilizing a simplified Torrance-Sparrow reflection model is used to estimate the reflectance parameters and refine the illumination distribution. This provides a very compact representation of the data necessary to render synthetic images from arbitrary viewpoints. We have conducted experiments with real objects to synthesize photo-realistic view-dependent images within the proposed framework.

1. Introduction

Rendering photo-realistic virtual images from observations of real objects has been a major research topic in the computer vision and the computer graphics community for a while. Extensive amount of work in this area constitutes a few main research streams.

Image-based rendering is one of those major research streams. Taking only 2D images as the input, image-based methods rely on the fact that light rays can be parameterized as a 7D function called plenoptic function [1]. Considering each pixel in real images as samples of this plenoptic function, image-based methods synthesize virtual images by selecting the most appropriate sample of rays, or interpolating between the sampled rays. Levoy and Hanrahan [6] represents the light rays in free space (space free of occluders) in 4D by using two 2D planes, which enables the plenoptic function to be described in a more compact way. Gortler et al. [3] adopts a similar two 2D plane representation, but additionally they use rough geometric information derived from images to correct the basis function of the rays. Recently, especially for rendering scenes with the viewing direction inside-out, Shum et al. [12] have proposed a 3D representation of the plenoptic function. Since these plenoptic-function-based methods require only real images as the input, they provide high generality, i.e. they can be applied to a wide variety of objects and scenes. However, because of the principle of interpolation, these approaches tend to require a large amount of input images. Although the light rays can be represented efficiently in 4D or lower dimensionality, and compression techniques such as vector quantization or MPEG-based approaches can drastically reduce the total amount of information to be stored, they still require a dense sampling of the real object which means taking hundreds of images. Recently, Chai et al. [2] have proposed a framework to analytically compute the minimum sampling rate for these image-based approaches from spectral analysis with the use of sampling theorem. However, since they assume diffuse surface, view-dependent variance such as specularity, that plays an important role in photo-realisitic synthesis of images, is not taken into consideration. These view-dependent components occur in a high frequency domain compared to diffuse components, so that it excessively increases the sampling rate. Also, as long as relying on interpolation, it will be difficult to synthesize the movement of the view-dependent appearance effects. Basically they will just fade in and out.

Model-based methods or "Inverse Rendering" is another major research stream in this area. Model-based methods use both 2D images and 3D geometric model of the target object to estimate the BRDF of the object surface, by fitting a particular reflection model to the pixel values observed in input images [10] or by solving the inverse radiosity problem [17]. As model-based methods estimate the BRDF parameters of the object surface, view-dependent rendering and re-lighting can be accomplished with a very

To Appear in Proc. of ICCV 2001

compact representation of the object. In these methods the radiance and positions of the light sources need to be known to compute the BRDF, and direct information of lighting environment has to be provided in some way, e.g., with high dynamic range images of the light sources.

Recent researches in the so-called "3D photography" domain have proposed methods that go in between these two major streams. By taking advantage of the latest advances in 3D sensing equipments, such as laser range scanners and structured light scanners, these approaches try to make full use of the 3D geometry as well as the images. The work done by Nishino et al. [9] and Wood et al. [16] can be considered as setting one of the 2D planes in the light field approaches on to the object surface, in concrete, on the coarse triangular patches or dense surface points respectively. By deriving information from the geometry in this way, these approaches succeed in achieving higher compression ratio without losing smooth view-dependent variation such as the movement of highlights. However, these methods still rely on very dense observation of the objects.

In this paper, our goal is to accomplish photo-realistic view-dependent rendering with an efficient representation of the data, from a casually acquired image sequence as the input. By casual acquisition we mean a relatively sparse set of images taken by a hand-held camera. In this case, the input images are not enough to simply apply image-based rendering methods. Also we assume the 3D model and the camera parameters are known, while we do not rely on any direct information of the light sources, i.e. the radiance and the positions. To solve this problem, we first separate the view-dependent and view-independent components of the surface reflectance. The view-independent surface reflection will be stored as one texture map for the whole object. Then we use the view-dependent component images to derive an efficient representation of the lighting environment. We accomplish this by initially shooting back the pixels in the view-dependent images along the perfect mirror direction to form an *illumination hemisphere*. This is somewhat similar to what Wood et al. did to increase the correlation of lumispheres [16], except we build one global representation of the illumination. Using this illumination hemisphere as the initial estimation of the lighting environment, we estimate both the lighting environment and surface properties using a reflection model to approximate the reflectance mechanism.

The remainder of this paper consists as follows. In Section 2, we define a scenario we target, clarify the assumptions we make and overview our framework. In Section 3 we describe the representation we use for the view-independent reflection component of the object appearance. In Section 4, we introduce a method to represent the view-dependent reflection component efficiently, and how to refine the initial estimations of it. Based on the representation mentioned in these sections, we show how to render a view-dependent image in Section 5, and show experimental results, applying the proposed framework to real object, in Section 6. Finally, Section 7 concludes the paper.

2. Overview

2.1. Scenario and Assumptions

As mentioned in the previous section, extensive amount of research has been conducted and has succeeded in modeling and representing the appearance of objects to realize photo-realistic image synthesis. However, there is still a gap when we try to apply these techniques for practical use. For instance, consider a situation when a person wants to show his own object to a friend remotely, e.g. via internet, allowing his friend appreciate freely at any detail of the object, as if he had it in his own hand. This can directly apply to what people might want to do when they are purchasing objects online, i.e. e-commerce. Current techniques require the user to take a large amount of images or assume the scene structure like the lighting environment is known perfectly. Our ultimate goal is to enable people to do this with as less effort as possible with a representation that is as compact as possible. Setting this as our future goal, in this paper, we will concentrate on a constrained but still realistic scenario. We target a situation where a user takes several snapshots of an object in interest with a digital camera, moving around the object, and wants that information extended into some sort of representation, so that the user can see the object from arbitrary viewpoints. On tackling this problem, we make several assumptions while trying to keep the generality of the scenario.

Obviously the object and the light sources do not move while the image capturing period, and only the camera moves. The camera used to capture the input images can be pre-calibrated easily, for instance using techniques like [18, 15]. Also, the motion of the camera can be pre-estimated by applying low-level computer vision techniques. The major assumption we make is that we already have the 3D model of the target object. Considering recent developments in the laser scanning industry it has become relatively easy to obtain an accurate geometric model. In this case, the problem is aligning the images with the 3D model. Usually laser scanners provide low resolution texture together with its 3D point cloud, so the alignment of 2D images and 3D model can be achieved by solving the error minimization problem between 2D and 3D point correspondences. Also, when the target object has enough texture, computer vision techniques can be used to obtain the 3D model directly from the input color images as well as the camera motion.

2.2. Framework

Taking a geometric model and a sparse set of color images input, we propose a framework to derive an efficient

representation of the appearance of the target object and realize photo-realistic synthesis of view-dependent appearance. The framework will constitute of several steps as follows.

Separate the reflection components in input images based on intensity variation. We do this by taking the minimal pixel value for each point on the object surface throughout the image sequence. By separating the reflectance components, we obtain view-independent texture for each triangular patch of the 3D model. We refer to this view-independent texture as *global texture*.

Recover an initial approximation of the illumination distribution. We then compute residual images by subtracting the global-texture-mapped image for each camera position from the original images. These residual images contain the view-dependent reflected light such as highlights and inter-reflection, and the noise due to 3D-2D miss-alignment. By shooting each pixel values in each residual image along the perfect mirror direction with regards to the camera position till it hits a hemisphere covering the 3D model, we obtain an initial estimation of the illumination distribution. We refer to this representation of the illumination distribution as *illumination hemisphere*.

Refine illumination distribution and estimate surface reflectance parameters. Finally, we apply an iterative numerical method to estimate the reflectance parameter of the object surface according to a simplified Torrance-Sparrow model and simultaneously refine the illumination hemisphere.

Once we have the illumination hemisphere, the reflectance parameter of the object surface and the global texture for the view-independent reflection component, we are able to synthesize realistic images from arbitrary viewpoints.

3. Global Texture

To have an efficient representation, we start by decomposing input images into view-independent reflection component images and view-dependent reflection component images.

3.1. Reflectance Model

The light reflected on the object surface can be approximated as a linear combination of two reflection components: diffuse reflection component I_D and specular reflection component I_S [11, 5].

$$I = I_D + I_S \tag{1}$$

The mechanism of the diffuse reflection is explained as the internal scattering. When an incident light ray penetrates the object surface, it is reflected repeatedly at a boundary between small particles and medium of the object. The scattered light ray eventually reaches the object surface, and is reflected into the air in various direction. This phenomenon results in diffuse reflection. Since the directions of the reflected lights can be assumed to be evenly distributed in all directions, Lambertian model is widely used to approximate this diffuse reflection. The Lambertian model is described with the cosine of the angle θ_i between the illumination direction and the surface normal, multiplied by the diffuse color vector K_D, i.e.,

$$I_D = K_D \cos\theta_i \tag{2}$$

Recent studies on the diffuse reflection component have shown that in the case of surfaces with high macroscopic roughness, the diffuse reflection becomes view-dependent [8]. However, for many objects in our daily life, such as painted man-made objects, Lambertian model is good enough.

The other reflection component I_S called specular reflection explains the light directly reflected at an interface between the air and the surface medium. This specular reflection has a spike in the perfect mirror direction with relatively weak intensity spread (a lobe) around the perfect mirror direction. The lobe is caused by the microscopic roughness of the object surface. Since the spike of the specular reflection can only be observed on object surfaces smooth enough in comparison with the wavelength of the incident light and when the camera viewing direction is aligned with the perfect mirror direction, we can ignore the spike when approximating the specular reflection with a numerical model. The lobe of specular reflection can be approximated by an intensity distribution having the peak in the perfect mirror direction. The Torrance-Sparrow reflection model [14] uses a gaussian distribution with its mean in the perfect mirror direction:

$$I_S = \frac{1}{\cos\theta_r} K_S F G \exp[-\frac{\alpha^2}{2\sigma^2}] \tag{3}$$

where K_S is the color vector of the reflection which accounts for the gain of the sensor measuring intensity, the source strength, the normalization factor of the exponential function, and the reflectivity of the surface, F is the Fresnel's coefficient, G is the geometrical attenuation factor, θ_r is the angle between the viewing direction and the surface normal, α is the angle between the surface normal and the bisector of the viewing direction and the light source direction, and σ represents the surface roughness.

3.2. Recovering View-independent Reflectance Map

Since we assume that the object and the light source are fixed and only the camera is moving, only the viewing direction changes through the image sequence. This means that only the specular reflection component varies from image to image for each point on the object surface, while the diffuse reflection component is view-independent and constant. The curve in Figure 1 shows how the intensity value on a particular surface point should vary while the camera

moves around the object. The pixel value should be constant and have a peak if the highlight passes through that surface point.

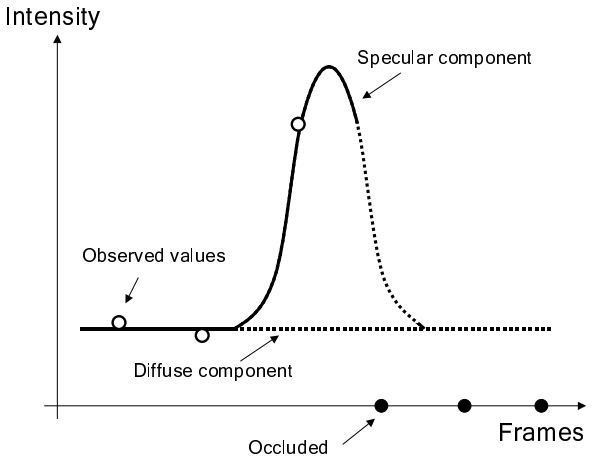

Figure 1. *Example of intensity variation at an object surface point.*

As the diffuse reflection is theoretically constant throughout the image sequence with the approximation of Lambertian reflection model, storing one value for each point on the object surface is enough to represent this diffuse reflection. In practice it is difficult to obtain this constant intensity value from the input image sequence. The circles on the left side of Figure 1 shows an example of the pixel values of a particular surface point through the image sequence. Due to the noise inherent in the imaging sensor, the intensity varies from image to image. If we have a large number of images as the input, it is possible to draw an intensity histogram for each surface point, and fit, e.g., a Gaussian model to the peak to derive the constant diffuse reflection component. However, as we have a sparse set of images, usually only a few sample points will be available, as depicted with white circles in Figure 1. The black circles in Figure 1 indicate that the surface point is invisible in those images due to occlusion. In consequence, we simply take the minimal pixel value for each surface point, and consider it as the initial estimation of the diffuse reflection component. Although the minimum pixel value is corrupted by the imaging noise, it should be close to the diffuse reflection component if the corresponding surface point is captured without surface reflection in at least one image. This condition will be the constraint to determine the lower bound on the number of input images, and this also means surface points which always have surface reflection through the input image sequence will not be treated correctly with our framework.

Since we do not have any continuous representation of the surface geometry, we represent this view-independent (diffuse) reflection component as textures of triangular patches. This can be achieved by first texture-mapping each input image onto the 3D model, then normalizing the texture of each triangular patch to a prefixed size (e.g. 20x20 pixels), and finally taking the minimal values for each pixel inside the normalized texture. This texture-based approach provides an efficient representation of the diffuse reflection component; we refer to this as global texture.

4. View-dependent Component

Our next step is to represent the view-dependent reflection component efficiently. We do this by recovering the lighting environment as a hemisphere over the object, and estimating the reflectance parameters of the surface and the illumination radiance using a simplified Torrance-Sparrow reflection model.

4.1. Illumination Hemisphere

The residual images, generated by subtracting the global-texture-mapped object image from each original input image, depicts the view-dependent appearance of the object surface. These images mainly consist of specular reflection (highlights) and some interreflection and noise. In our work, we ignore interreflection and consider it as noise. As described in Section 3, specular reflection is the light reflected at the surface interface, and its intensity peak is in the perfect mirror direction. In consequence, we can make a rough approximation of the illumination environment by mapping each pixel in the residual image onto a hemisphere covering the 3D object model. Each pixel in the residual image will be shot back towards the hemisphere in the perfect mirror direction, which can be computed from the surface normal and the viewing direction. An illumination hemisphere is represented by the upper half of a geodesic sphere, a subdivided icosahederon.

The illumination hemisphere generated from each residual image only covers partial regions of the true illumination environment, and we need to combine these partial illumination hemispheres to make a final illumination hemisphere approximating the real lighting environment. To deal with noise, we make a mask hemisphere that represents how many times each point was taken into account while making the partial illumination hemispheres. We then test for each point on the hemisphere whether the count in the mask hemisphere and the number of times they had an intensity value in the partial hemispheres are equal. Only those points that pass this check will be mapped on the final illumination hemisphere, and in this case we take the mean of the intensity values from the partial illumination hemispheres as its intensity value. This way of consistency checking also reduces the errors, e.g. introduced by 3D-2D miss-alignment. These errors are not view-dependent and would not stay in a particular region on the illumination hemisphere. Inter-

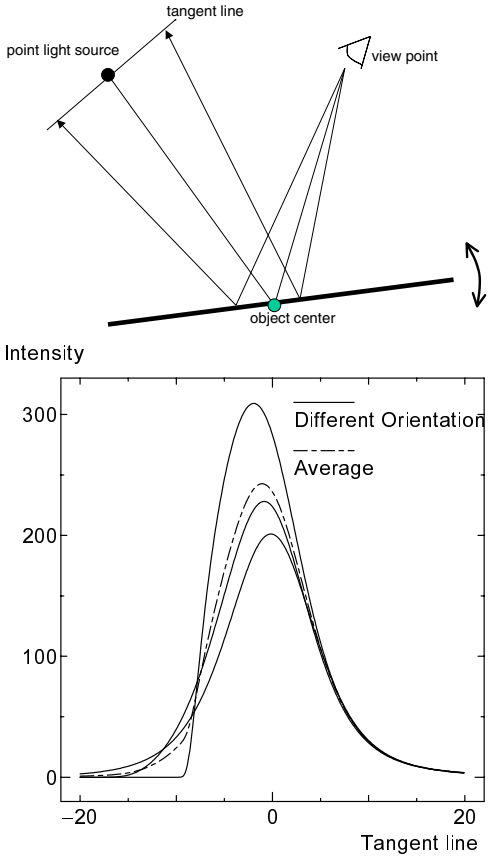

Figure 2. *A sample in 2D of intensity variation shot back to perfect mirror directions.*

reflection will also be faded out, since it can be considered as reflected light of a moving light sources.

4.2. What Illumination Hemisphere Represents

We will use a simple 2D example as shown in Figure 2 to examine what an illumination hemisphere represents. As depicted in the upper half of Figure 2, the viewpoint is set in a position so that the light source and the viewpoint becomes in a perfect mirror position with respect to the object center. When we vary the object line orientation while fixing the surface reflectance properties (K_s,F,G and σ in Equation (3)), point light source's position and brightness, the viewpoint changes accordingly to stay in a perfect mirror configuration. Consider a line perpendicular to the light source to object center direction. It can be considered as a section of the illumination hemisphere. If we shoot back the observed intensity values along the perfect mirror direction, we obtain a curve, referred to as *illumination curve*, as shown in the lower part of Figure 2. With varying the object line orientation, we obtain various distinct illumination curves as shown in Figure 2; this is equivalent to observing an object from different viewpoints in our real scenario.

If we consider one of the illumination curves in Figure 2 as the illumination distribution, and assume there is no roughness on the object surface, then the image rendered in the original viewpoint for that curve will be indistinguishable from the original image, but the images rendered for *other* viewpoints will not match the original images. This is because the illumination curves corresponding to what we observed in data images while assuming no surface roughness are different from each other. In other words, if we have multiple views of the same highlight, we should be able to differentiate the effect of the surface reflectance properties from the lighting distribution based on the input image data, i.e., we should be able to estimate both of them. Also, it is evident that we cannot estimate those parameters if we only have one observation of each highlight, because the surface roughness can be embedded in the lighting distribution without changing the appearance. Note that we do not necessarily need to estimate the true illumination radiance distribution. We just have to estimate the illumination distribution and the surface reflectance properties such that the highlights in the residual images can be explained.

Combining partial illumination hemispheres as described in the last subsection gives the average of overlapped curves (depicted in dashed line in Figure 2). Although it provides a fairly good approximation to the real illumination distribution, the effect of the surface reflectance property, i.e. roughness, is not taken into account. In order to achieve correct view-dependent image synthesis, we have to refine this illumination hemisphere and estimate the roughness in order to approximate the real illumination distribution better.

4.3. Illumination and Reflectance Parameter Estimation

Now we will try to separate the effect of the surface reflectance property from the illumination hemisphere. We will use a reflection model and estimate both the surface reflectance property and the intensities of the nodes on the geodesic illumination hemisphere using the illumination hemisphere obtained in section 4.1 as the initial guess.

We use the Torrance-Sparrow reflection model, except we assume the geometrical attenuation factor G in (3) is 1 and the Fresnel reflectance coefficient F is constant. These assumptions are valid for most dielectric and metal objects [13]. With this simplified Torrance-Sparrow reflection model, the specular reflection I_S is given by

$$I_{S,m} = K_{S,m}\frac{1}{\cos\theta_r}\exp[-\frac{\alpha^2}{2\sigma^2}] \quad (4)$$

$$\text{with} \quad K_{S,m} = k_S g\int_\lambda \tau_m(\lambda)s(\lambda)d\lambda \quad (5)$$

where m stands for each R, G and B component, g is the scene radiance to surface irradiance ratio, k_S is the spectral magnitude, τ is the spectral response, s is the surface irradiance on a plane perpendicular to the light source direction,

and λ is the wavelength.

In our implementation, the illumination hemisphere is uniformly sampled using a geodesic hemisphere (a subdivided icosahederon). Each node on the illumination hemisphere is considered as a point light source. Because linearity holds in surface irradiance, we can compute the specular reflection at point v on the object surface as

$$I_{S,m}(v) = k_{S,v} \sum_{l}^{N_L} \omega_l L_{l,m} \frac{1}{cos\theta_r} \exp[-\frac{\alpha^2}{2\sigma_v^2}] \quad (6)$$

where $L_{l,m}$ stands for the radiance of each point light source in each color band, and ω_l stands for the solid angle of the area that each point light source represents.

From (4) and (5), we see that the color vector direction of the specular reflection is the same as that of the light source. If we assume that all light sources in the environment have the same color, we can use the average color of the initial illumination hemisphere as the color of the light sources. In consequence, the specular reflection intensity on the surface of the 3D model in each image is reduced to

$$I_S = \frac{2\pi}{N_L} k_{S,v} \sum_{l}^{N_L} L_l \frac{1}{cos\theta_r} \exp[-\frac{\alpha^2}{2\sigma_v^2}] \quad (7)$$

Note we do not have m anymore, and L_l is now the magnitude of the color vector of each point light source.

Now, we can estimate the surface reflectance paramters k_S and σ, and refine the illumination hemisphere by estimating L_l through minimizing the following objective function

$$\min(I_v - I_{S,v})^2 \quad (8)$$

where I_v is the observed intensity value in residual images, and $I_{S,v}$ is the intensity value computed from Equation (7).

Since the value of the surface roughness parameter σ is usually much smaller than the light intensities L_l and spectral magnitude k_S, it will be difficult for the numerical minimization to converge if we try to estimate all the three simultaneously. Instead, we take an approach similar to that in [7, 4]. We estimate σ and the set of L_l and k_S alternatively in different iterations, and repeat the iteration by turns until both of them converge. We use the Y value in the YC_rC_b color coordinate as the initial estimation of L_l. Due to the possible error introduced in the construction of the global texture, we expect outliers in the specular images. To handle this, we solve this reflection parameter estimation problem in an M-estimator framework, and the objective function is

$$E(\mathbf{x}) = \frac{1}{N_V} \sum_{v}^{N_V} \rho(z_v(\mathbf{x})) \quad (9)$$

$$\text{with} \quad z_v(\mathbf{x}) = (\frac{I_v - I_{S,v}}{\sigma_{err}})^2 \quad (10)$$

where $\mathbf{x}$ is either $(L_l, k_{S,v})$ or (σ_v) depending on iteration,

Figure 3. *Two images out of eight input images.*

and we currently use the Lorentzian function

$$\rho(z) = \log(1 + \frac{1}{2}z^2) \quad (11)$$

In the above formulation, we assume the object has the same surface reflectance property k_S and σ for the entire surface. This can be a drawback of our framework, although this assumption can be made on many dielectric objects.

5. View-dependent Image Synthesis

After estimating the refined illumination hemisphere and the surface reflectance parameters, synthetic images from arbitrary viewpoints can be easily rendered through the following three steps.

1. Render a global-texture-mapped image with regards to the viewpoint.
2. Render a specular reflection image computing the pixel with the reflection model using the refined illumination hemisphere and surface reflectance parameters.
3. Composite two images by adding values at each pixel.

6. Results

We applied our framework to model the appearance of a real object. We took 8 color images, each from different position roughly on a circle around the target. The images were taken with a digital camera (Olympus C2000), while trying to keep the angle between each camera position similar, so that the object was observed uniformly. Figure 3 shows two of the input images. We used Cyberware laser stripe range scanner to obtain a geometric model of the object, and simplified it to a mesh model with 2834 triangles.

Figure 4 shows the texture variation for two different triangular patch, through the input image sequence and the corresponding global texture derived from them on the right side. Note, both were visible only in three images. The upper row is the successful case, showing by taking the minimal pixel values we can get the diffuse reflection component separated out. The middle in upper row has the highlight.

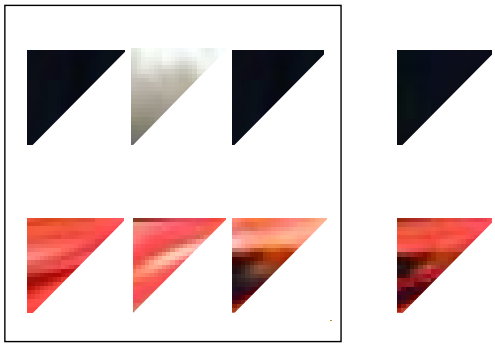

Figure 4. *In box: Two examples of texture variation in the input image sequence. Outside box: The corresponding global texture.*

Figure 5. *Global-texture-mapped image (left) and residual image (right) corresponding to the left image in Figure 3.*

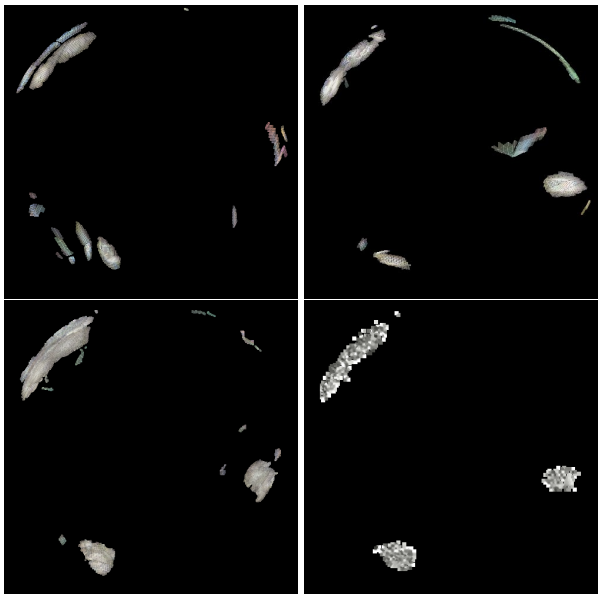

Figure 6. *Top row: Partial illumination hemispheres. Bottom row left: Initial combined illumination hemisphere. Bottom right: Refined illumination hemisphere (each point light source corresponding to a geodesic node is splatted for visualization). (All mapped on a 2D plane y=0)*

The lower row shows an example when noise is introduced into the global texture. The third texture in the lower row has some amount of the texture which should have corresponded to a neighboring triangular patch, and this results in the noise of global texture on the right hand side. The left side image in Figure 5 is a global-texture mapped object image, rendered from the view position corresponding to that of the left image in Figure 3. The errors can be seen on the frame of the sunglasses, due to the alignment of images and the 3D model. We will discuss how to treat this 3D-2D miss-alignment in Section 7 as future work. Currently, we do not try to refine this global texture.

The right image in Figure 5 shows the residual image corresponding to the left image in Figure 3. As mentioned above, the noise introduced in the former step, i.e. generating global texture, will appear in the residual images, and will be directly mapped onto the partial illumination hemispheres (top row in Figure 6). However, as described in Section 4.1, we can get rid of most of these noise values when combining these partial illumination hemispheres (bottom left in Figure 6). By sampling this initial illumination hemisphere uniformly, and through the two step numerical minimization described in Section 4.3 , we obtain a refined illumination hemisphere (bottom right in Figure 6) and the surface reflectance parameters, $k_S = 626.0$ and $\sigma = 0.0114$ for this object. Each residual image was thresholded with a certain brightness value and then sampled by picking up every other pixel. We used a geodesic hemisphere with 18000 nodes to sample the illumination hemisphere. In the initial illumination hemisphere, 1757 nodes had values assigned, and after the parameter estimation it decreased to 1162.

Figure 7 shows a side by side comparison of one of the input image and a synthetic image rendered using the refined illumination hemisphere and estimated reflectance parameters. Although the highlights are rendered close to the original in shape and brightness, their positions are slightly different from those in the original images. This is because we use a hemisphere for approximation, which means that assuming all light sources are in the same distance from the object center, the estimated distribution of lights is slightly different from that in real environment. Also since we use a simplified mesh model, the shape details are not well preserved, resulting in different shape highlight on the upper

Figure 7. *Left: Input image. Right: Synthetic image.*

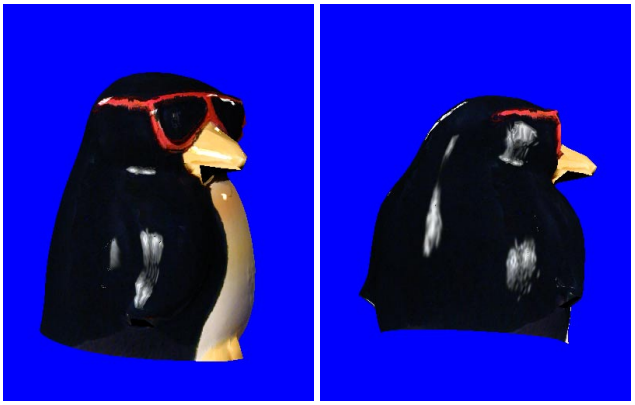

Figure 8. *Synthetic images rendered from viewpoints not included in input.*

left side of the penguin. Figure 8 shows two synthetic images rendered from a new viewpoint. Triangular patches under the bill of the penguin were not visible in the input images, so they are rendered in black.

7. Conclusion and Future Work

In this paper, we have proposed a framework to accomplish photo-realistic view-dependent image synthesis from a sparse image set and a geometric model. In particular, we proposed a method to separate the reflectance components in the input images based on the intensity variation of object surface points. We store the view-independent component as one texture map, and use the residual images to estimate a rough approximation of the illumination environment, using a hemisphere covering the object. Finally a two step numerical minimization technique utilizing a Torrance-Sparrow reflection model was introduced to refine this illumination distribution and estimate the surface reflectance parameters. This framework provides a very compact representation of the object appearance for view-dependent rendering. We have conducted experiments with real objects to show the effectiveness of our framework.

Future work includes approximating the illumination distribution with distance and integrating the estimation process of the cameras' pose parameters in the framework.

References

[1] E.H Adelson and J.R. Bergen. *The Plenoptic Function and the Elements of Early Vision*, pages 3–20. Computational Models of Visual Processing, M.Landy and J.A.Movshon(eds),MIT Press, 1991.

[2] J. Chai, X. Tong, S-C. Chan, and H-Y. Shum. Plenoptic Sampling. In *Computer Graphics Proceedings, ACM SIGGRAPH 00*, pages 307–318, Jul. 2000.

[3] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen. The Lumigraph. In *Computer Graphics Proceedings, ACM SIGGRAPH 96*, pages 43–54, Aug. 1996.

[4] K. Ikeuchi and K. Sato. Determining Reflectance Parameters using Range and Brightness Images. In *Proc. of Third International Conference on Computer Vision ICCV '90*, pages 12–20, Dec. 1990.

[5] G.J. Klinker, S.A. Shafer, and T. Kanade. A Physical Approach to Color Image Understanding. *International Journal of Computer Vision*, 4:7–38, 1990.

[6] M. Levoy and P. Hanrahan. Light Field Rendering. In *Computer Graphics Proceedings, ACM SIGGRAPH 96*, pages 31–42, Aug. 1996.

[7] S.K. Nayar, K. Ikeuchi, and T. Kanade. Shape from Interreflections. In *Proc. of Third International Conference on Computer Vision ICCV '90*, pages 2–11, Dec. 1990.

[8] S.K. Nayar and M. Oren. Generalization of the Lambertian Model and Implications for Machine Vision. *International Journal of Compute Vision*, 14(3):227–251, 1995.

[9] K. Nishino, Y. Sato, and K. Ikeuchi. Eigen-Texture Method: Appearance Compression based on 3D Model. In *Proc. of Computer Vision and Pattern Recognition '99*, volume 1, pages 618–624, Jun. 1999.

[10] Y. Sato, M.D. Wheeler, and K. Ikeuchi. Object shape and reflectance modeling from observation. In *Computer Graphics Proceedings, ACM SIGGRAPH 97*, pages 379–387, Aug. 1997.

[11] S.A. Shafer. Using Color to Separate Reflection Components. *COLOR Research and Application*, 10(4):210–218, 1985.

[12] H-Y. Shum and L-W.He. Rendering with Concentric Mosaics. In *Computer Graphics Proceedings, ACM SIGGRAPH 99*, pages 299–306, Aug. 1999.

[13] F. Solomon and K. Ikeuchi. Extracting the shape and roughness of specular lobe objects using four light photometric stereo. In *Proc. of Computer Vision and Pattern Recognition '92*, pages 466–471, 1992.

[14] K.E. Torrance and E.M. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, 57:1105–1114, 1967.

[15] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, Aug. 1987.

[16] D.N. Wood, D.I. Azuma, K. Aldinger, B. Curless, T. Duchamp, D.H. Salesin, and W. Stuetzle. Surface Light Fields for 3D Photography. In *Computer Graphics Proceedings, ACM SIGGRAPH 00*, pages 287–296, Jul. 2000.

[17] Y. Yu, P. Debevec, J. Malik, and T. Hawkins. Inverse Globall Illumination: Recovering Reflectance Models of Real Scenes From Photographs. In *Computer Graphics Proceedings, ACM SIGGRAPH 99*, pages 215–224, Aug. 1999.

[18] Z. Zhang. Flexible Camera Calibration By Viewing a Plane From Unkown Orientations. In *Proc. of Seventh International Conference on Computer Vision ICCV '99*, pages 666–673, Sep. 1999.