University of Toronto

Photometric Stereo

CSC2503 Fall 2014
Foundations of Computer Vision

Alexander Hong
(c4hongal 997584706)

October 8, 2014

Contents

List of Figures

1 Calibration

The light source directions from the eight images of the chrome sphere was determined using the following algorithm:

i) Determining the center of sphere from the mask.

$$P_{center} = \begin{bmatrix} \frac{1}{M}\sum mx \\ \frac{1}{M}\sum my \end{bmatrix} \tag{1}$$

where M is the total of all mask values, m is the mask value at (x, y), and (x, y) is the pixel coordinates of the image. This formula assumes all values in the sphere are relatively the same and is analogous to determining the centre of mass of an object. The formula was chosen because it is a generalized method to determine the center of any shaped-object.

ii) Determining the radius of the sphere.

$$r = \frac{\texttt{maximum height} + \texttt{maximum width}}{4} \tag{2}$$

This formula takes the average of the maximum height and maximum width to determine the diameter. Dividing this value by 2 gives an approximation for the radius. This formulat was chosen because it takes the average radius for better approximation.

iii) Determining the brightest spot on the image. The brightest spot on the image is determined in a similar fashion as in step i) — by taking the center of mass of squared brightness values.

$$P_{brightest} = \begin{bmatrix} \frac{1}{B}\sum b^2x \\ \frac{1}{B}\sum b^2y \end{bmatrix} \tag{3}$$

where B is the total of all brightness values, b is the brightness value at (x, y), and (x, y) is the pixel coordinates of the image. The brightness values were squared to magnify weight of high intensity regions and reduce the weight of low intensity regions. This reduces the effects of noise or other specular reflections.

iv) Determining the Z value and surface normal of the brightest spot with respect to the center of the sphere.

$$\begin{bmatrix} n_x \\ n_y \end{bmatrix} = P_{brightest} - P_{center} \tag{4}$$

$$n_z = -\sqrt{r^2 - n_x^2 - n_y^2} \tag{5}$$

$$\vec{n} = \frac{\begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}}{\left\| \begin{bmatrix} n_x & n_y & n_z \end{bmatrix} \right\|} \tag{6}$$

The Z normal must be negative to indicate a visible normal. Equation 6 gives the normalized surface normal.

v) Determining the light source direction for each of the 8 images.

$$\vec{L_i} = 2(\vec{n} \cdot \vec{d_e})\vec{n} - \vec{d_e} \tag{7}$$

where $\vec{L_i}$ is the light source direction for image $i \in \{1, 2, ..., 8\}$, and $\vec{d_e}$ is the emitted direction of the camera ($\vec{d_e} = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}^T$).

Figure 1 shows the orthographic image of the 8 light source directions.

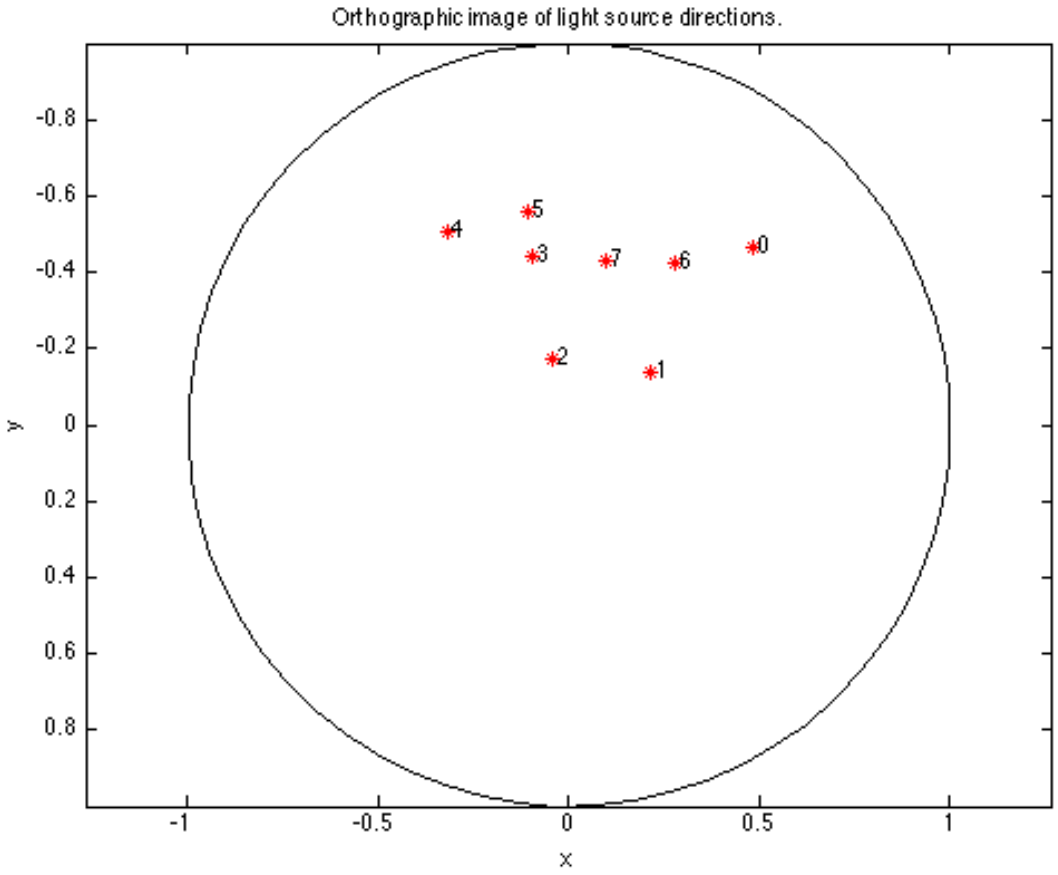


Figure 1: Light source directions from the 8 images for each different object

2 Computing Surface Normals and Grey Albedo

The least-squares error for pixel $\vec{x}$ is

$$E = \sum_{j=1}^{8} \left(I_j - \vec{g} \cdot \vec{L}_j \right)^2 \tag{8}$$

In order to find the minimum error with respect to $\vec{g}$, the derivative of E with respect to $\vec{g}$ is taken and set to zero.

$$\frac{dE}{d\vec{g}} = 2 \sum_{j=1}^{8} \left(\vec{g} \cdot \vec{L}_j - I_j \right) \vec{L}_j = \vec{0} \tag{9}$$

$$\implies \sum_{j=1}^{8} \left(\vec{g} \cdot \vec{L}_j \right) \vec{L}_j = \sum_{j=1}^{8} \left(I_j \vec{L}_j \right) \tag{10}$$

$$\implies (\mathbf{L^T L}) \vec{g} = \mathbf{L^T I} \tag{11}$$

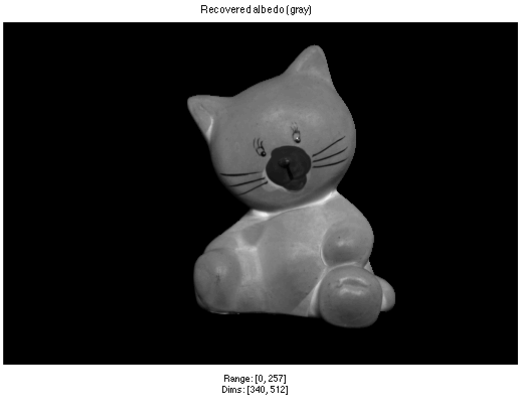
where $\mathbf{L} = \begin{bmatrix} \vec{L}_{j=1} & \vec{L}_{j=2} & \dots & \vec{L}_{j=8} \end{bmatrix}^T$ and $\mathbf{I} = \begin{bmatrix} I_{j=1} & I_{j=2} & \dots & I_{j=8} \end{bmatrix}^T$. Solving Equation 11 for $\vec{g}$ will yield the best least-squares approximation for $\vec{g}$. Seperating $\vec{g}$ into its magnitude, a, and direction, $\vec{n}$, will result in the grey-level albedo and the corresponding surface normal respectively.

$$a(\vec{x}) = \|\vec{g}(\vec{x})\| \tag{12}$$

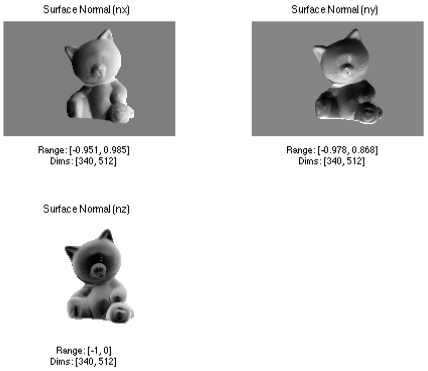
$$\vec{n}(\vec{x}) = \frac{\vec{g}(\vec{x})}{\|\vec{g}(\vec{x})\|} \tag{13}$$

Figure 2 shows the recovered normals, estimated grey albedo, image reconstuctions and their errors for the cat object. The results for the other figures are found in Appendix A.

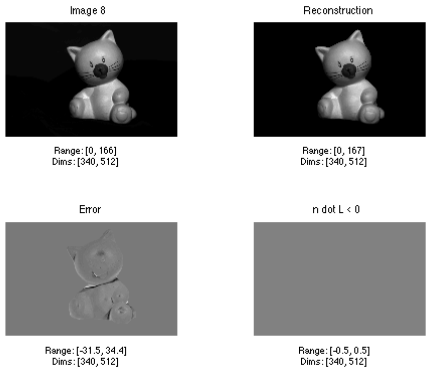
The grey-level image constructions look similar to the original images. The most apparent modelling errors show in the edges of the object, which indicates that the reconstuction did not capture the original depth of the image (i.e. not enough brightness in those areas). Generally, the range of the values in the reconstructed image is captured. However, large errors occur when the object has greater depth. This is apparant in the grey object and the zebra object (See Appendix A). Large errors occur around the gray sphere and the zebra's back leg. This is likely due to lighting source unable to capture all of the depth of the image. Less errors occur in brighter spots of the image.



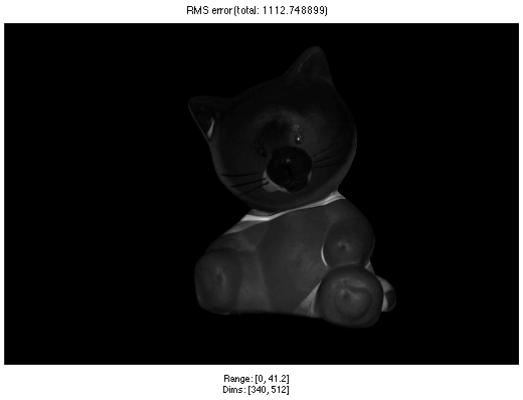
(a) Estimated grey albedo



(b) Components of the recovered normals



(c) Image reconstruction with their errors



(d) RMS error

Figure 2: Surface normals and grey albedo for Cat object

3 Computing RGB Albedo and Relighting

The least-squares error for the albedo $a_\nu(\vec{x})$ for pixel $\vec{x}$ and colour channel ν is

$$E_\nu = \sum_{j=1}^{8} \left(I_{j,\nu} - a_\nu \vec{n} \cdot \vec{L}_j \right)^2 \tag{14}$$

In order to find the minimum error with respect to a_ν, the derivative E_ν with respect to a_ν is taken and set to zero.

$$\frac{dE}{da_\nu} = \sum_{j=1}^{8} 2 \left(a_\nu \vec{n} \cdot \vec{L}_j - I_{j,\nu} \right) \left(\vec{n} \cdot \vec{L}_j \right) = 0 \tag{15}$$

$$\implies \sum_{j=1}^{8} \left(a_\nu \left(\vec{n} \cdot \vec{L}_j \right)^2 - I_{j,\nu} \left(\vec{n} \cdot \vec{L}_j \right) \right) = 0 \tag{16}$$

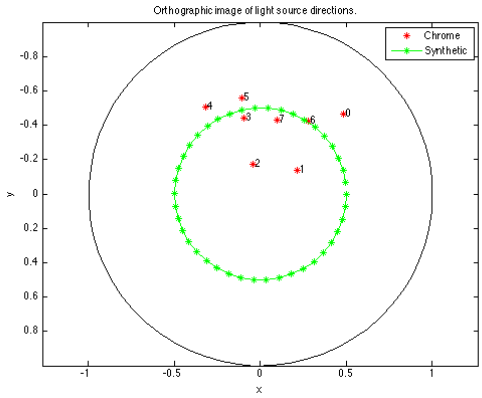
$$\implies a_\nu \sum_{j=1}^{8} \left(\vec{n} \cdot \vec{L}_j \right)^2 - \sum_{j=1}^{8} I_{j,\nu} \left(\vec{n} \cdot \vec{L}_j \right) = 0 \tag{17}$$

$$\implies a_\nu = \frac{\sum_{j=1}^{8} I_{j,\nu} \left(\vec{n} \cdot \vec{L}_j \right)}{\sum_{j=1}^{8} \left(\vec{n} \cdot \vec{L}_j \right)^2} \tag{18}$$

$$\implies \mathbf{a}_\nu = \frac{\mathbf{I}_\nu \cdot (\mathbf{nL})}{(\mathbf{nL}) \cdot (\mathbf{nL})} \tag{19}$$

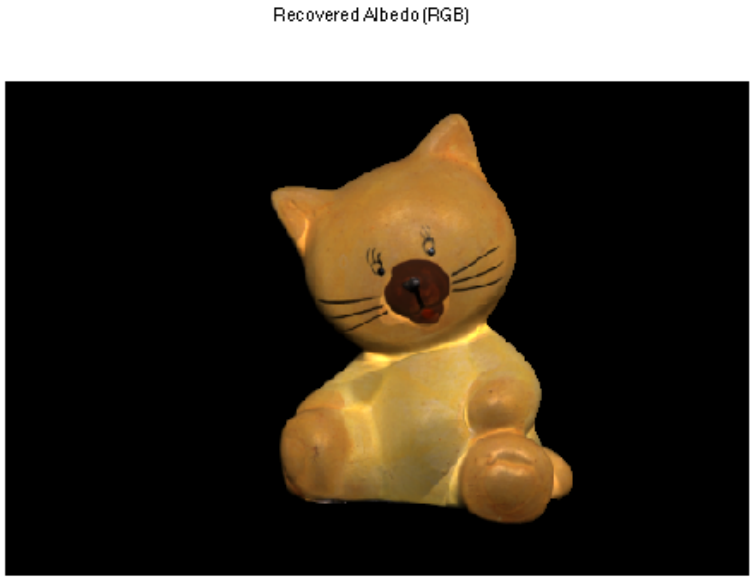
where $\mathbf{a}_\nu$ is the best least-squares approximation for albedo for each colour channel ν, $\mathbf{n}$ contains the surface normals for all pixels, $\mathbf{L} = \begin{bmatrix} \vec{L}_{j=1} & \vec{L}_{j=2} & \dots & \vec{L}_{j=8} \end{bmatrix}^T$, and $\mathbf{I}_\nu = \begin{bmatrix} I_{j=1,\nu} & I_{j=2,\nu} & \dots & I_{j=8,\nu} \end{bmatrix}^T$. Figure 3 shows the estimated albedo in colour and a synthesized image from a synthetic light source for the cat object. The results for the other objects are found in Appendix B.

The synthesized images appear realistic. However, there are modelling errors in generating these synthetically light images. There are some features, such as the neck of the cat (Figure 3), neck of Buddha (Figure 12), and the back of the zebra's leg (Figure 14), that appear to be unnatural. They should have shadows in these areas casted by the object's profile. These unnatural images correspond to the high intensity regions found in the RMS error figures from the last section. Hence, the errors stem from the albedo calculations. Supposing that the recovered albedos and surface normals are correct, the main remaining modelling errors is the initial assumption of Lambertian Surface. Although it gives simplicity and linearity of the model, it loses some realistic features. Errors occur at the lowest and highest intensity regions of the image because these ends are realistically non-linear.



(a) Synthetic light source positions

(b) Synthesized image from synthetic light



(c) Estimated RGB albedo

Figure 3: RGB albedo and relighting for Cat object

4 Surface Fitting

In this section, the surface fitting algorithm was implemented and executed for all the objects. Refer to the assignment page for details on the algorithm. Essentially, we are solving for $\vec{z}$ in

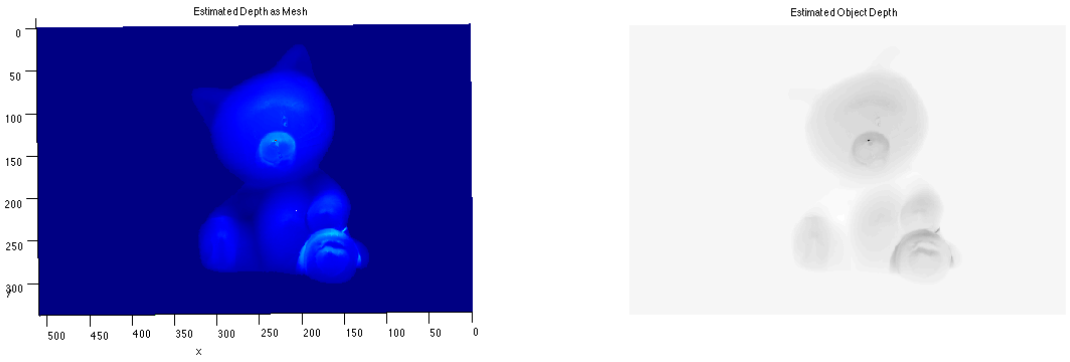
$$A\vec{z} = \vec{v} \tag{20}$$

Matrix A is constructed using the z-component of the normal vectors and represented as a sprase matrix. $\vec{v}$ is constructed by stacking all the x-components of the surface normals on top of all the y-components.

Figure 4 shows the estimated depth mesh and estimated object depth for the cat object. The results for the other objects are found in Appendix C.

The technique works well and produces reasonable depths for all the objects. One can identify the object and its profile by looking at either the depth mesh or the object depth figures. For instance, one can draw the conclusion that the cat object has its nose and left foot closer to the camera as indicated in Figure 4. Unfortunately, the technique fails when the albedo of the surface is extremely low (near zero). Figure 17 shows spikes in the depth for low albedo values. Low albedo indicates low reflectance ability and thus the algorithm is unable to produce accurate surface normals. The normals affect the construction of the matrices required for surface fitting.

One way to fix spikes in the depth map is to smoothen the albedo values. Lower values of the albedos should be replaced by the average of neighbouring acceptable albedos. Another way to smooth the depth map and remove the spikes is to use Guassian blur. Guassian blur is a low-pass filter and has the ability to reduce high values of depths.



(a) Depth mesh (b) Estimated object depth

Figure 4: Surface fitting for Cat object

5 Estimate Light Source Direction Given Scene Properties

Similar to Section 2 and 3, the least-squares error is found.

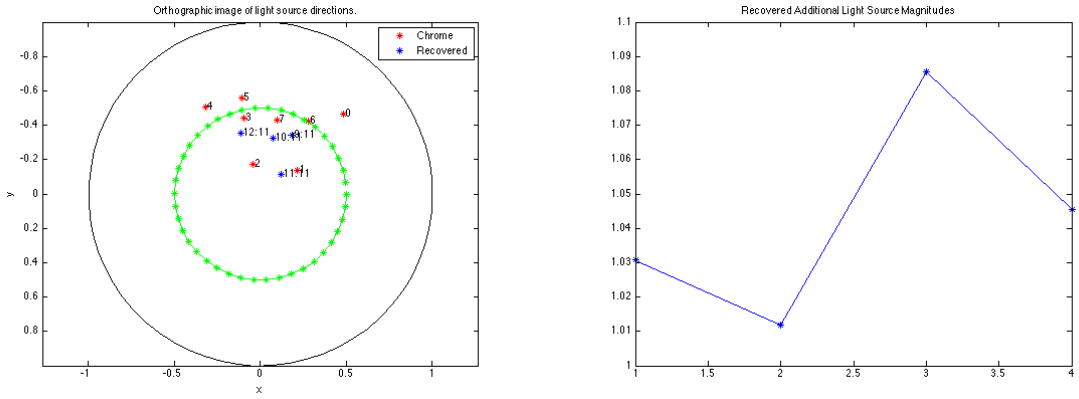
$$E = \sum_j \left(I_j - \vec{g} \cdot \vec{L}_j \right)^2 \tag{21}$$

In order to find the minimum error with respect to the light source, $\vec{L}_j$, the derivative E with respect to $\vec{L}_j$ is taken and set to zero.

$$\frac{dE}{\vec{L}_j} = 2 \sum_j \left(\vec{g} \cdot \vec{L}_j - I_j \right) \vec{g} = \vec{0} \tag{22}$$

$$\implies (\mathbf{g^T g}) \vec{L}_j = \mathbf{g^T I} \tag{23}$$

where $\mathbf{g}$ contains columns of $\vec{g}$ and $\mathbf{I} = \begin{bmatrix} I_{j=9} & I_{j=10} & \dots & I_{j=N} \end{bmatrix}^T$ up to N images. Solving Equation 23 for $\vec{L}_j$ will yield the best least-squares approximation for $\vec{L}_j$. Figure 5 shows the recovered light source directions and their magnitudes for the additional cat object images. The results for the other objects are found in Appendix D.



(a) Recovered light source directions (b) Light source magnitudes

Figure 5: Light source direction for Cat object for additional images

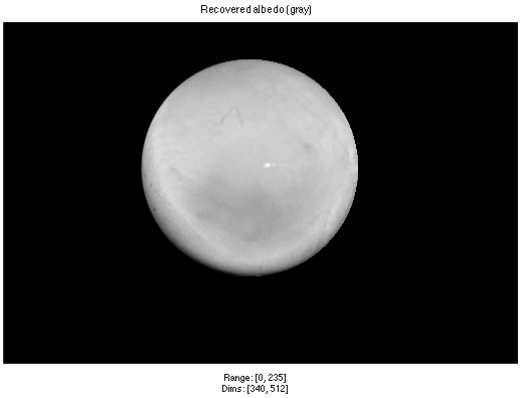
In terms of estimating 2 light source directions from an image that was actually illuminated by two seperate point light sources of reduced intensity, one can use principle of superposition.

$$I_v(\vec{x}) = a_\nu(\vec{x}) \lfloor \vec{n}(\vec{x}) \cdot \vec{L}_1 + \vec{n}(\vec{x}) \cdot \vec{L}_2 \rfloor \tag{24}$$

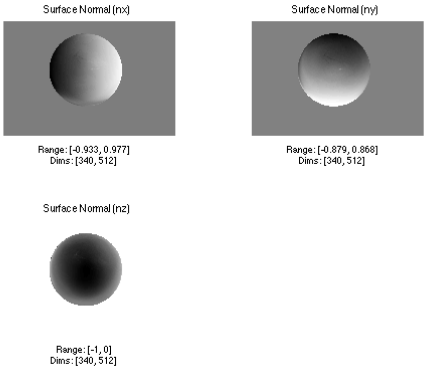
$$\implies I_v(\vec{x}) = a_\nu(\vec{x}) \lfloor \vec{n}(\vec{x}) \cdot (\vec{L}_1 + \vec{L}_2) \rfloor \tag{25}$$

This shows that one can estimate the combination of the two light sources with Equation 25. However, with the given information, it is insufficient to determine the two light sources **individually**. One would need more information about the image, such as regions with shadow, in order to deduce the light directions individually.

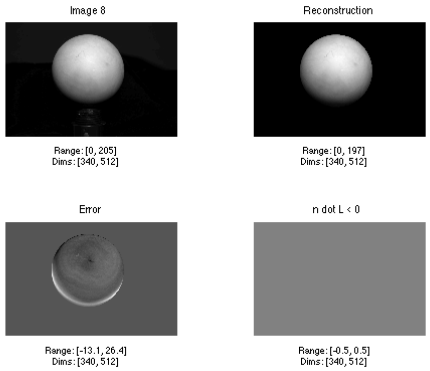
Appendix A Surface Normals and Grey Albedo



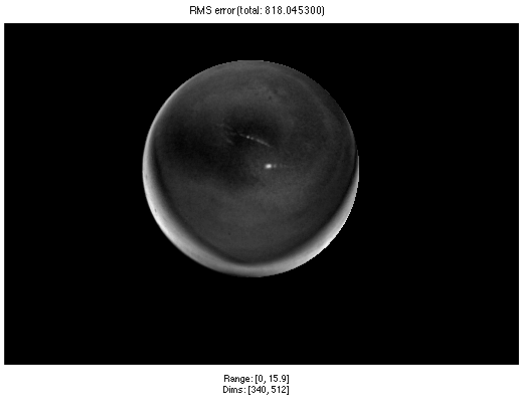
(a) Estimated grey albedo



(b) Components of the recovered normals



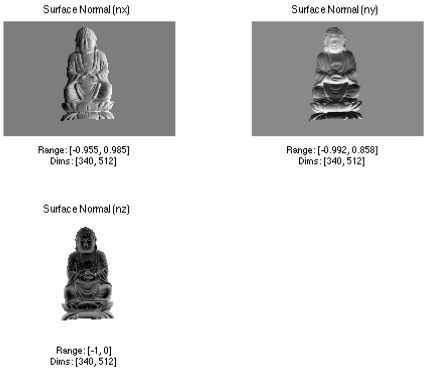
(c) Image reconstruction with their errors



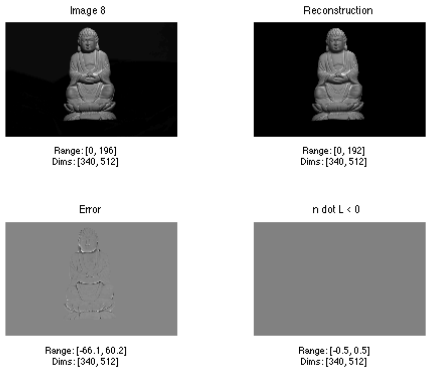
(d) RMS error

Figure 6: Surface normals and grey albedo for Sphere object

(a) Estimated grey albedo



(b) Components of the recovered normals

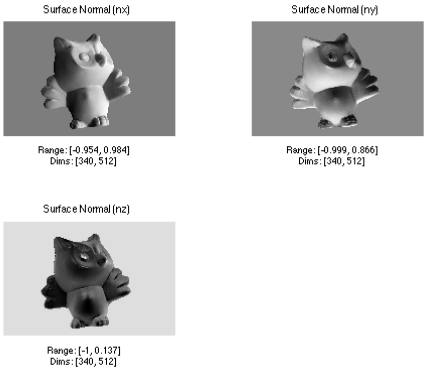


(c) Image reconstruction with their errors

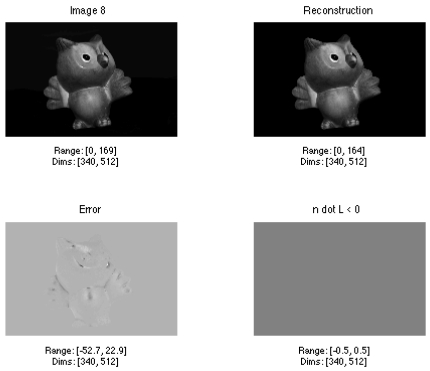
(d) RMS error

Figure 7: Surface normals and grey albedo for Buddha object

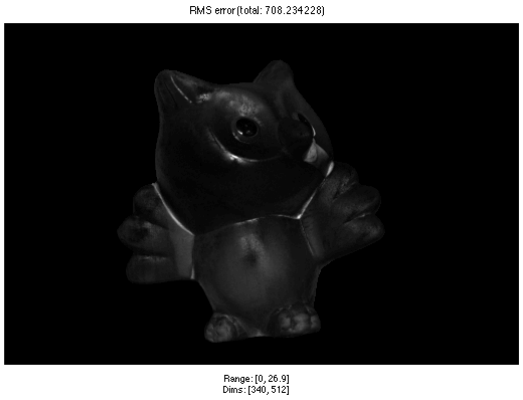
(a) Estimated grey albedo



(b) Components of the recovered normals

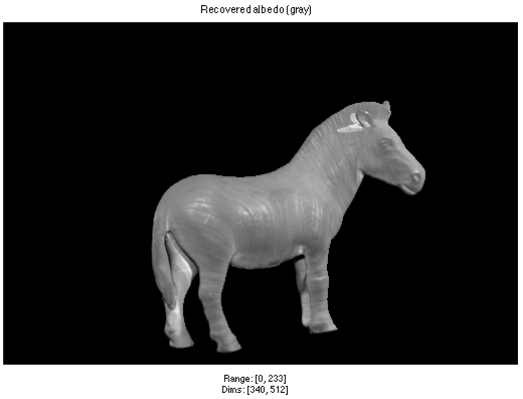


(c) Image reconstruction with their errors

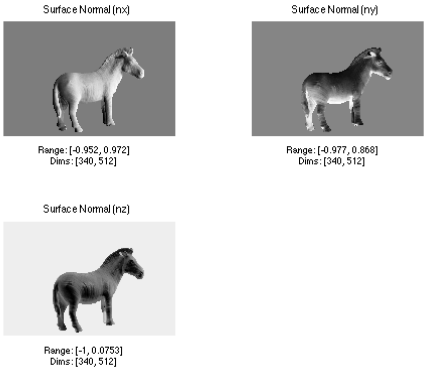


(d) RMS error

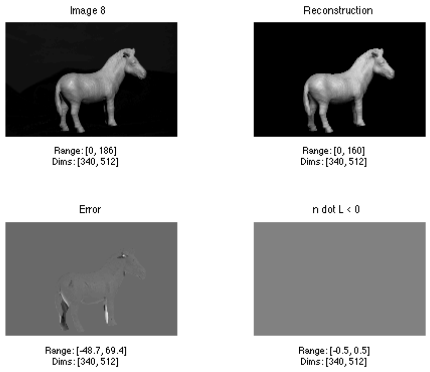
Figure 8: Surface normals and grey albedo for Owl object



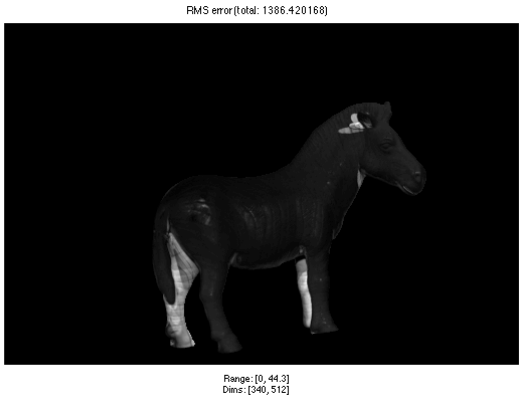
(a) Estimated grey albedo



(b) Components of the recovered normals



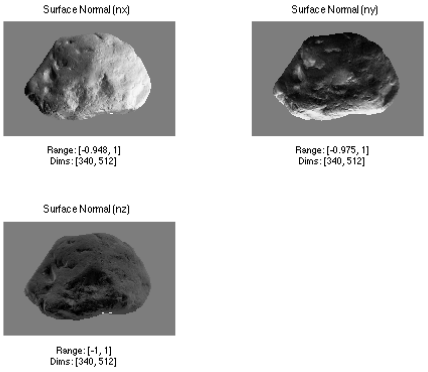
(c) Image reconstruction with their errors



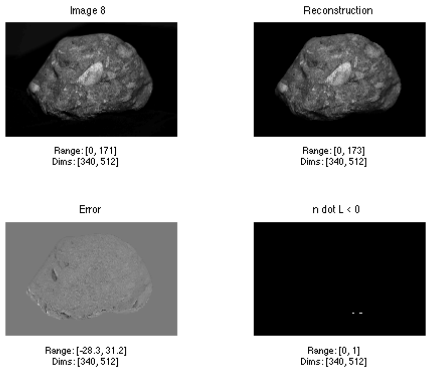
(d) RMS error

Figure 9: Surface normals and grey albedo for Zebra object

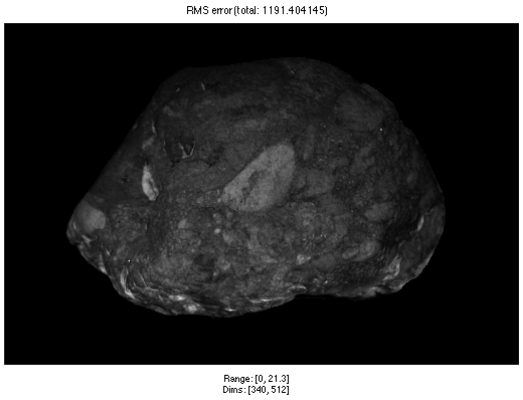
(a) Estimated grey albedo



(b) Components of the recovered normals



(c) Image reconstruction with their errors



(d) RMS error

Figure 10: Surface normals and grey albedo for Rock object

Appendix B RGB Albedo and Relighting

(a) Synthesized image from synthetic light (b) Estimated RGB albedo

Figure 11: RGB albedo and relighting for Grey object

(a) Synthesized image from synthetic light (b) Estimated RGB albedo

Figure 12: RGB albedo and relighting for Buddha object

(a) Synthesized image from synthetic light (b) Estimated RGB albedo

Figure 13: RGB albedo and relighting for Owl object

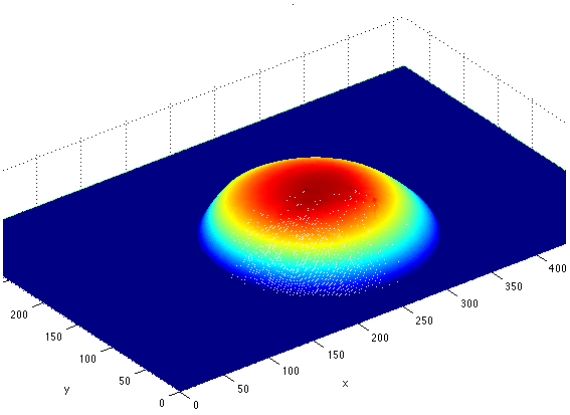
(a) Synthesized image from synthetic light (b) Estimated RGB albedo

Figure 14: RGB albedo and relighting for Zebra object

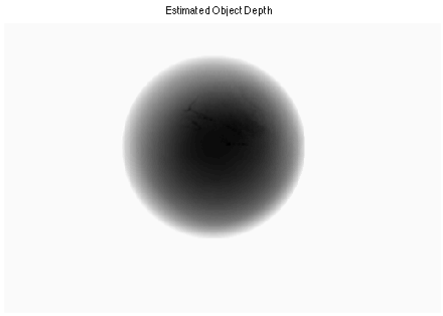
(a) Synthesized image from synthetic light (b) Estimated RGB albedo

Figure 15: RGB albedo and relighting for Rock object

Appendix C Surface Fitting

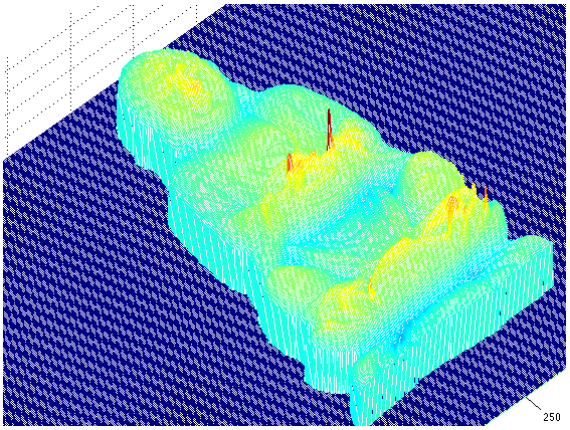


(a) Depth mesh

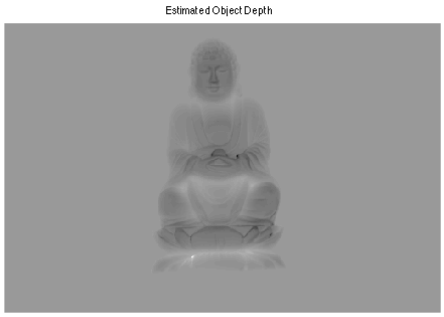


(b) Estimated object depth

Figure 16: Surface fitting for Grey object

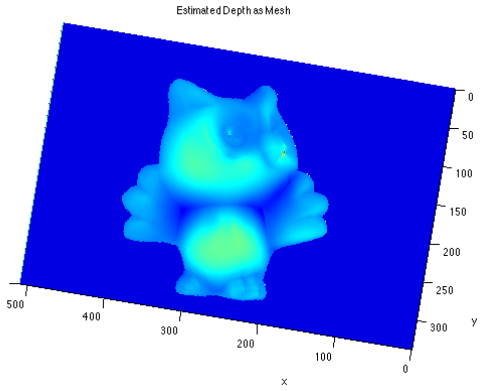


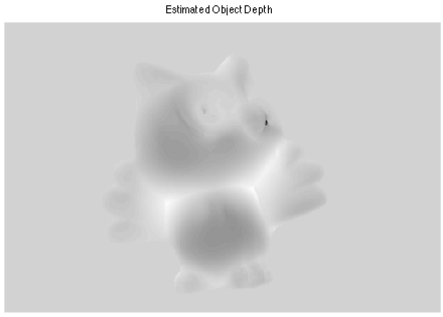
(a) Depth mesh



(b) Estimated object depth

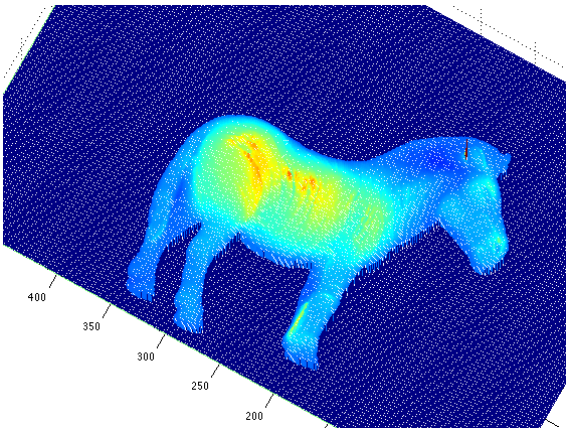
Figure 17: Surface fitting for Buddha object





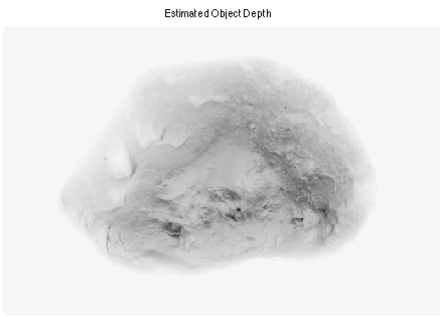
(a) Depth mesh (b) Estimated object depth

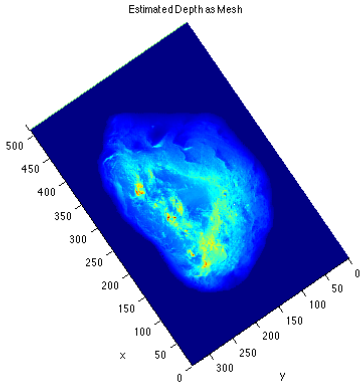
Figure 18: Surface fitting for Owl object



(a) Depth mesh (b) Estimated object depth

Figure 19: Surface fitting for Zebra object

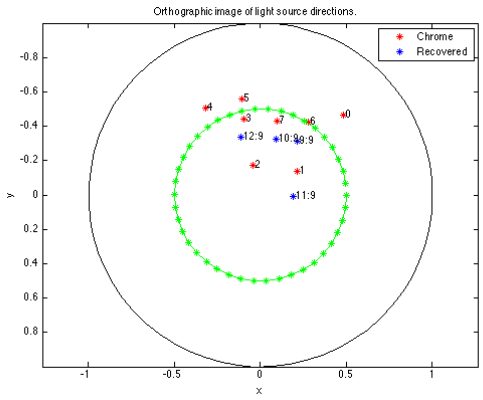


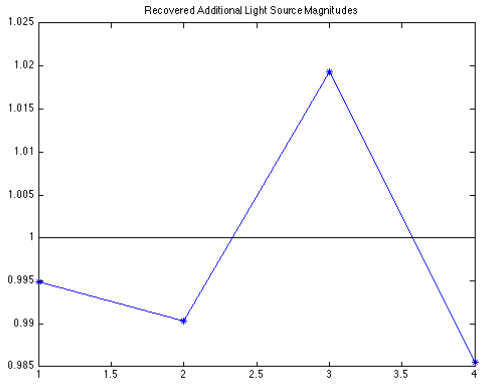


(a) Depth mesh (b) Estimated object depth

Figure 20: Surface fitting for Rock object

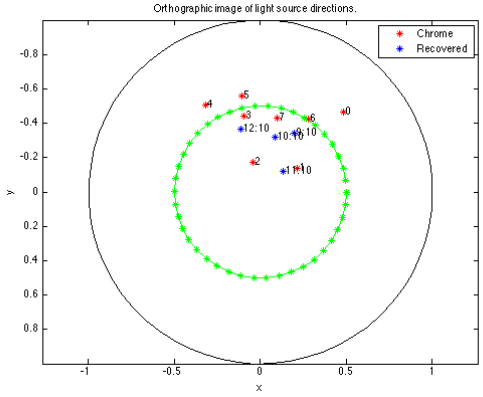
Appendix D Estimate Light Source Direction Given Scene Properties

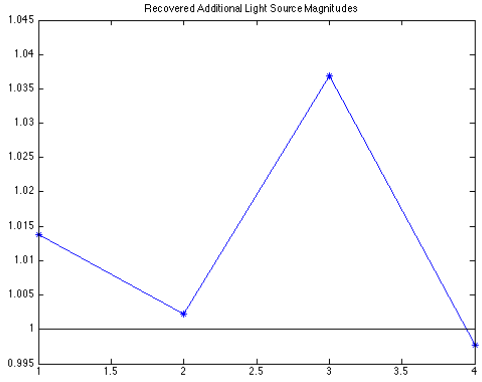




(a) Recovered light source directions (b) Light source magnitudes

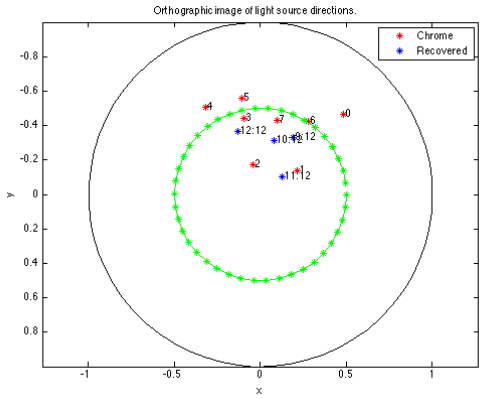
Figure 21: Light source direction for Grey object for additional images

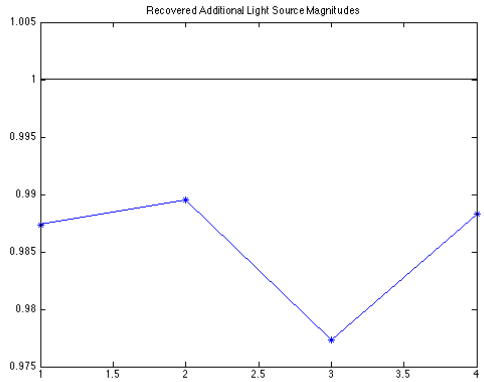




(a) Recovered light source directions (b) Light source magnitudes

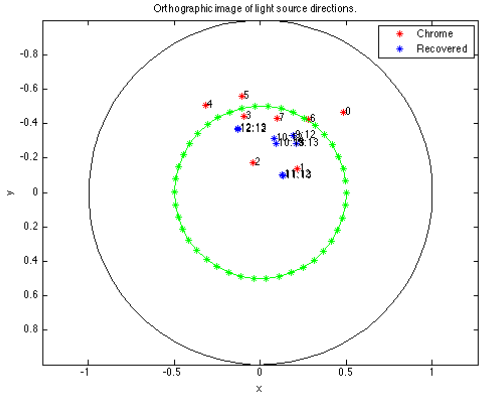
Figure 22: Light source direction for Buddha object for additional images

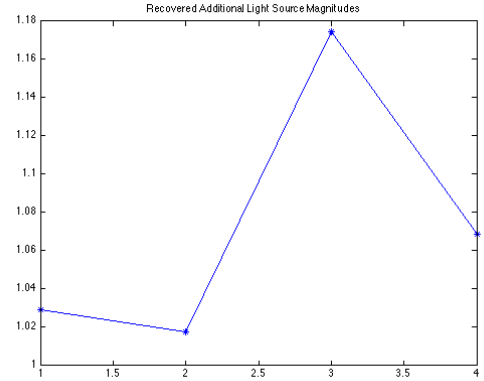




(a) Recovered light source directions (b) Light source magnitudes

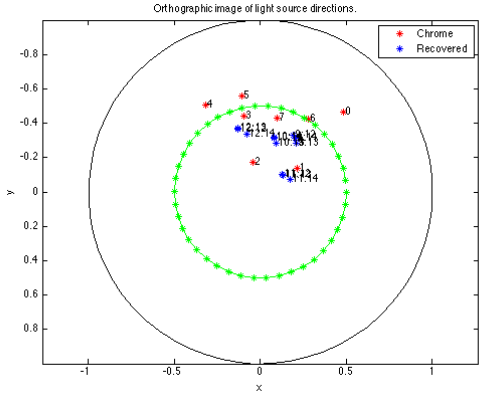
Figure 23: Light source direction for Owl object for additional images

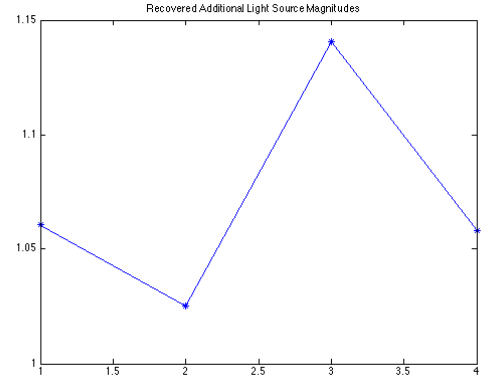




(a) Recovered light source directions (b) Light source magnitudes

Figure 24: Light source direction for Zebra object for additional images





(a) Recovered light source directions (b) Light source magnitudes

Figure 25: Light source direction for Rock object for additional images