UNIVERSITY OF TORONTO

Department of Computer Science

CSC2503H: Foundations of Computer Vision

Assignment #1:

Photometric Stereo

Sanjif Rajaratnam

999091986

rajara24

Course Instructor: Dr. Kyros Kutulakos

October 12th, 2016

Table of Contents

Table of Figures

List of Tables

1.0 Calibration

This results for the various models will be shown in the appendix. The first part of this project focused on finding the direction of the light sources. This is important because the direction of the light sources is required to determine the unknown surface normals of the various objects. In this case, an object with known surface normals was used to determine the light source directions. Firstly, the Phong reflectance model was used to determine the lighting direction (**Equation 1**).

$$L(\bar{p}, \overrightarrow{d_e}) = r_d I_d \max(0, \vec{s} \cdot \vec{n}) + r_a I_a + r_s I_s \max\left(0, \vec{m} \cdot \overrightarrow{d_e}\right)^{\alpha} \qquad (1)$$

In this equation, the diffuse and spectral components of reflection are due to the incident light from a point source. It was assumed that the spectral component dominated in this equation and that the diffuse and ambient term were negligible since the chrome ball could be approximated as a perfect mirror surface. The Phong model was then reduced to **Equation 2**.

$$L(\bar{p}, \overrightarrow{d_e}) = r_s I_s \max\left(0, \vec{m} \cdot \overrightarrow{d_e}\right)^{\alpha} \qquad (2)$$

The chrome ball was assumed to be a mirror surface so it can be assumed that the light experienced perfect specular reflection. Therefore, the emittant direction can be modelled using **Equation 3.**

$$\overrightarrow{d_e} = 2(\vec{n} \cdot \overrightarrow{d_\iota})\vec{n} - \overrightarrow{d_\iota} \qquad (3)$$

This equation requires normalized terms. In this case, the $\overrightarrow{d_e}$ is the direction of the camera, the $\overrightarrow{d_\iota}$ is the direction of the light source, and the $\vec{n}$ is the direction of the normal. In perfect specular reflection, $\overrightarrow{d_e}$ can be found by reflecting it across the normal since the angle between $\vec{n}$ and $\overrightarrow{d_\iota}$ is equal to the angle between $\vec{n}$ and $\overrightarrow{d_e}$. Therefore, **Equation 4** can be obtained.

$$\vec{d_\iota} = 2(\vec{n} \cdot \vec{d_e})\vec{n} - \vec{d_e} \qquad (4)$$

Here, it was assumed the point of perfect specular reflection occurred at the brightest point of the sphere. The next term to be determined was the surface normal of the chrome ball. Firstly, the pixel coordinate could be taken as the real world coordinate because of the orthographic projection. Next, since the object of interest was a sphere, its x, y, and z surface normal at any point can be found with **Equations 5, 6,** and **7**, respectively. Here negative signs were added to compensate for the coordinate system chosen.

$$n_x = \frac{(x - x_c)}{r} \qquad (5)$$

$$n_y = -\frac{(y - y_c)}{r} \qquad (6)$$

$$n_x^2 + n_y^2 + n_z^2 = 1 \quad \rightarrow \quad n_z = -\sqrt{1 - n_x^2 - n_y^2} \qquad (7)$$

Here, (x_c,y_c) is the coordinate of the center of the sphere and (x, y) is the coordinate of the brightest point of the sphere. This terms were solved for then plugged into **Equation 4** to get $\vec{L}$.

The direction of the camera was given and then the coordinate of the center and the coordinate of the brightest point can be found using MatLab's built-in 'imfindcircles' and 'max' functions. The 'imfindcircles' was used on the mask after finding an approximate radius using the 'imdistline' function. The 'max' function returns the coordinate of the first coordinate horizontally and vertically that has a max value of 255. This was assumed to be coordinate of perfect specular reflection since the patch of max values was small and the answer did not vary vastly when changing the coordinate by a small value. This also reduced coding time by simplifying the code. Here looping over pixels were avoided by doing the calculations within the same loop that read in the chrome ball images. The orthographic image of light source directions can be seen in **Figure 1.**

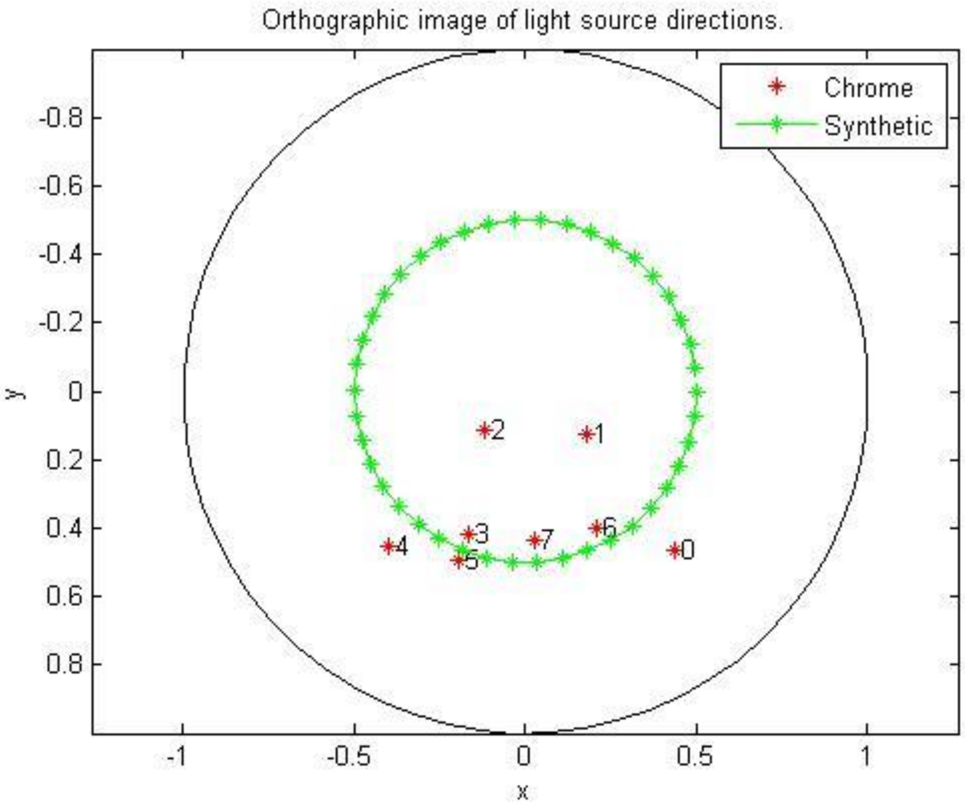


Figure 1: Orthographic Image of Light Source Directions

2.0 Surface Normals and Grey Albedo

Here, from Equation 3 from the Assignment 1 handout, the intensity values, $I(\vec{x})$, and light source directions, $\vec{L}$, are known but the albedo, $a(\vec{x})$ and the surface normals, $\vec{n}(\vec{x})$ are unknown. To create a linear system of equations, $a(\vec{x})\vec{n}(\vec{x})$ was set equal to $\vec{g}(\vec{x})$. It was assumed that the albedo $a(\vec{x})$ is the magnitude of the $\vec{g}(\vec{x})$ term, and $\vec{n}(\vec{x})$ is a unit vector in the direction of $\vec{g}(\vec{x})$ since the albedo is direction-invariant and the normal vector is a unit vector. They can be found using **Equation 8,** and **9**, respectively.

$$a(\vec{x}) = ||\vec{g}(\vec{x})|| \quad (8)$$

$$\vec{n}(\vec{x}) = \frac{\vec{g}(\vec{x})}{||\vec{g}(\vec{x})||} \quad (9)$$

Then, Equation 4 from the Assignment 1 handout can be differentiated with respect to $\vec{g}(\vec{x})$ and set equal to 0 to get **Equation 10**. Here the second term is the controlling term so the equation can be simplified to **Equation 11.**

$$0 = \sum_{j=1}^{8} 2\left(I_j(\vec{x}) - \vec{g}(\vec{x})\vec{\mathrm{L}_\mathrm{j}}\right)\left(\frac{dI_j(\vec{x})}{d\vec{g}(\vec{x})} - \vec{\mathrm{L}_\mathrm{j}}\right) \quad (10)$$

$$I_j(\vec{x}) = \vec{g}(\vec{x})\vec{\mathrm{L}_\mathrm{j}} \rightarrow I = GL \quad (11)$$

Equation 11 can then be rearranged to get **Equation 12**. The normal can be found using the ‘normr’ function within Matlab which normalizes the rows of G to a length of 1, and the albedo can be found using **Equation 13.** This operation can be done element wise to avoid looping through the pixels in the image to reduce computation time.

$$G = \frac{IL'}{LL'} \quad (12)$$

$$a(\vec{x}) = ||\vec{g}(\vec{x})|| = \sqrt{\vec{g}(x)^2 + \vec{g}(y)^2 + \vec{g}(z)^2} \quad (13)$$

The grey-level reconstructions for the eight images for each model can be found in their respective **Appendix** section. The resultant ranges found can be seen in **Table 1.** The estimated gray albedo can be seen in **Figure 2**. The calculated surface normals can be seen in Error! Reference source not found.. The average RMS error can be seen in **Figure 3**. The reconstruction for most of the images seemed visually accurate. Some images had much higher error (e.g. image 2). The highest being -83.8. In the case of the cat, the main region where the error occurred was the cat's neckline. The error could have been caused by light deflecting between the cat's head and torso due to the sharp corner or from self-shadowing. The final grayscale albedo seems to accurate represent the cat image. Also the total RMS value was 1125.8. Here the major points of concern are located around major bends on the bear where the surface changes more dramatically like at the ear, neck, where the arm meets the torso, and around the leg and tail. This again could be due to light deflecting back onto the object. It could also be due to light being blocked by self-shadowing.

Table 1: Ranges of Values for Gray Image Reconstructions

	Image	**Reconstruction**	**Error**	**n dot L < 0**
1	[0, 174]	[-17.8, 188]	[-23.8, 47.6]	[0, 1]
2	[0, 190]	[0, 157]	[-83.8, 27.7]	[-0.5, 0.5]
3	[0, 147]	[0, 150]	[-14.3, 73.1]	[-0.5, 0.5]
4	[0, 164]	[0, 170]	[-46.5, 28.4]	[-0.5, 0.5]
5	[0, 192]	[-25, 183]	[-46.7, 56.1]	[0, 1]
6	[0, 182]	[-8.33, 179]	[38.7, 77.7]	[0, 1]
7	[0, 175]	[0, 170]	[-42.3, 28.9]	[0, 1]
8	[0, 166]	[0, 170]	[-36.1, 35.3]	[-0.5, 0.5]

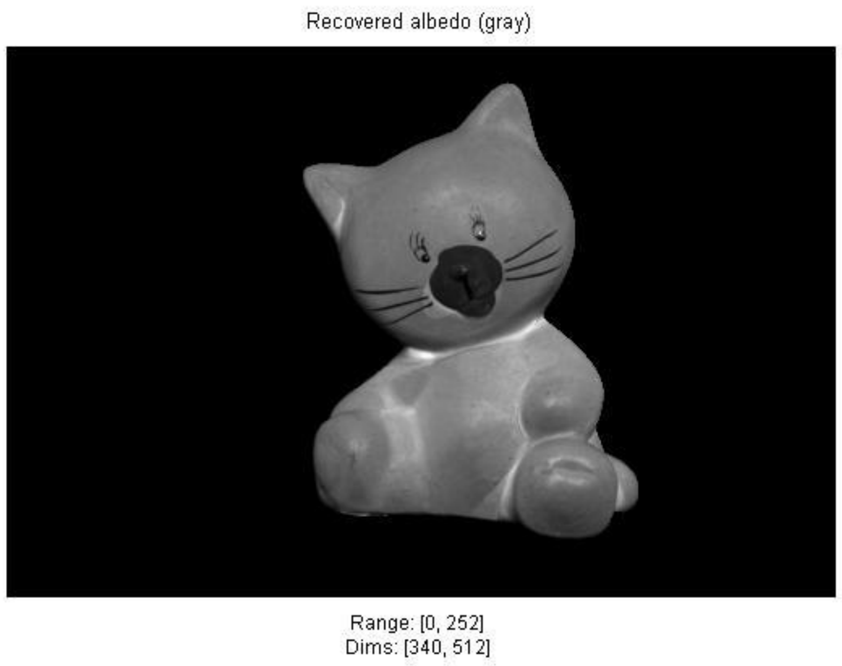


Figure 2: Recovered Albedo (Gray)

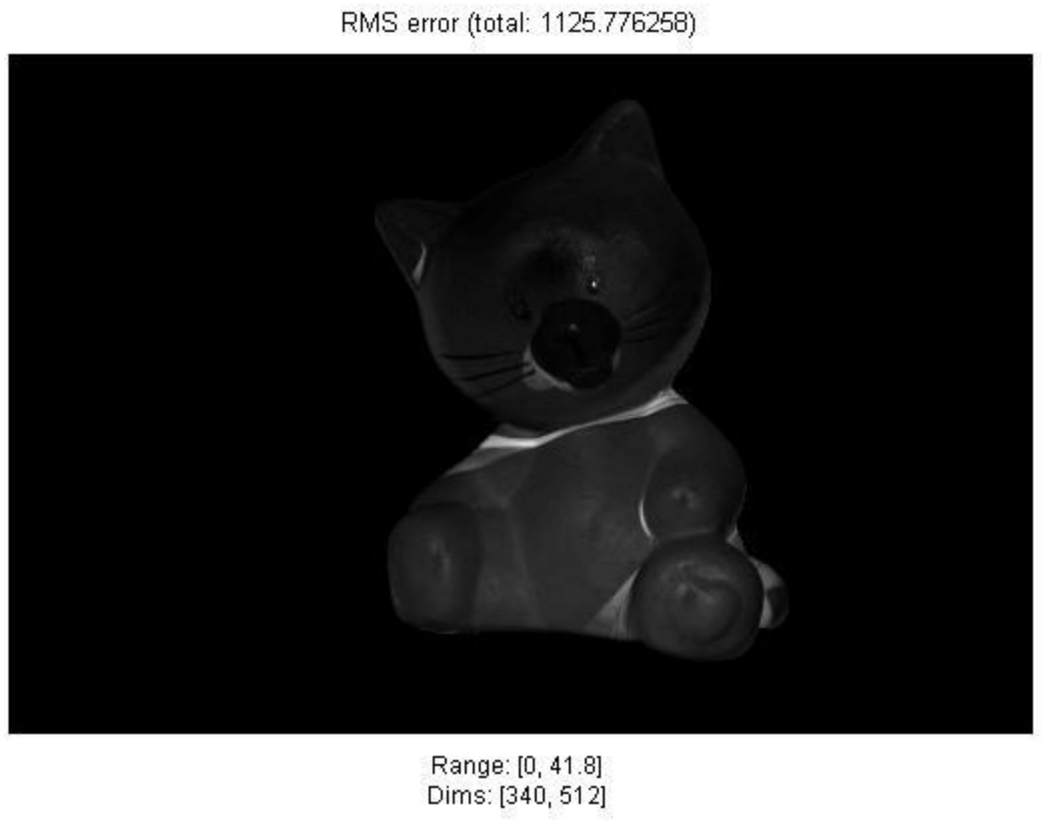


Figure 3: Average RMS Error (Cat)

The total RMS values for each of the objects can be seen in **Table 2**. The owl and the gray object had the lowest RMS whereas the horse had the highest RMS. Here the simpler the object, the lower the RMS value there was. RMS error regions existed mainly at bends, and places where self-shadowing exist. Self-shadowing is very evident in the horse's rear legs as seen in **Figure 4**. This could be because light is being blocked by the front leg.

Table 2: Total RMS for All Objects

Object	**Total RMS**
Buddha	1142
Cat	1125
Gray	887
Horse	1414
Owl	746
Rock	1188

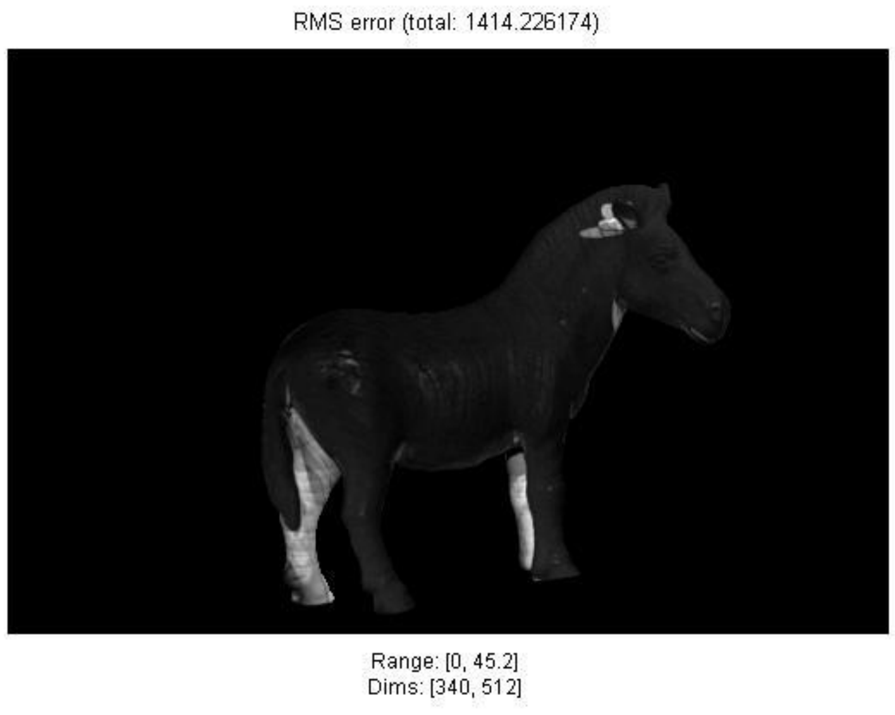


Figure 4: Average RMS Error (Horse)

3.0 RGB Albedo and Relighting

The surface normals is constant regardless of the colour channel used so the normals determined in the previous question can be used to solve for the various colour channel albedos. The calculations for this section work very similarly to the ones from the last section. Equation 5 from the Assignment 1 handout can be differentiated with respect to $a(\vec{x})$ and set equal to 0 to get **Equation 14**. The second term again is the controlling term so the equation can be simplified to **Equation 15.** Here I, N, and L are known, so the unknown A needs to solved for. This is a system of equations in the form of Ax = b.

$$0 = \sum_{j=1}^{8} 2\left(I_j(\vec{x}) - a(\vec{x})\vec{n}(\vec{x})\overrightarrow{\mathrm{L_j}}\right)\left(\frac{dI_j(\vec{x})}{da(\vec{x})} - \vec{n}(\vec{x})\overrightarrow{\mathrm{L_j}}\right) \qquad (14)$$

$$I_j(\vec{x}) = a(\vec{x})\vec{n}(\vec{x})\overrightarrow{\mathrm{L_j}} \;\rightarrow\; I = ANL \qquad (15)$$

The resulting RGB albedo for the cat can be seen in **Figure 5**. Here the RGB albedo seems to fairly accurately represent the object when compared to the actual image of the cat. The same error regions evident in the gray albedo are still prevalent in the RGB albedo. This could be due to these errors being invariant to colour. The area around the neck is still the region with the highest error.

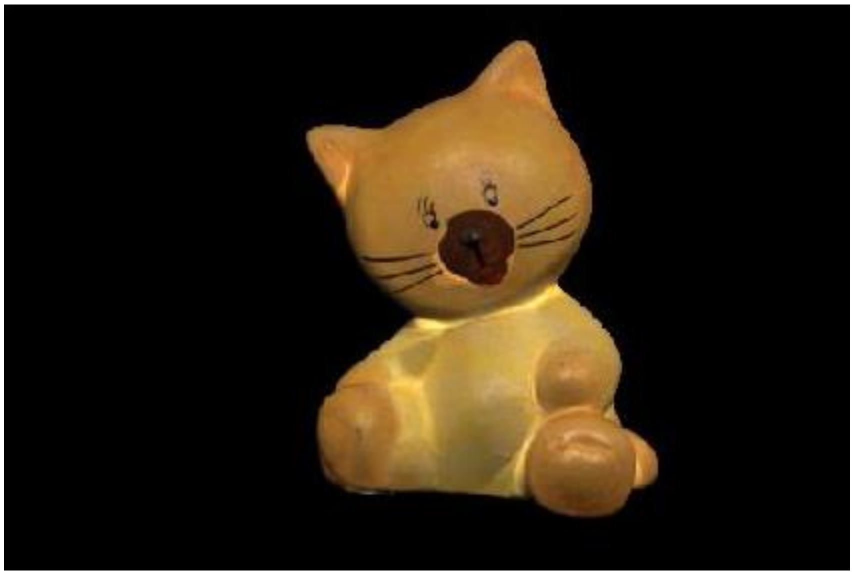


Figure 5: Recovered Albedo (RGB)

A still from the synthetically shaded image can be seen in **Figure 6**. The path the synthetic light travelled can be seen in **Figure 1.** The diffusion of light on the object seems fairly realistic except for the area around the neck. The light here is much brighter than it should be. This is due to it already being an error source in the calculations. The other regions that were errors in the albedos (ear, leg, and arm) do not seem to be issues in this image. They seem to be shaded more realistically. This could be due to how the light is oriented. The intensity values at these points are already low so an error on these values would not make a significant difference. The main modelling error for generating these synthetic light images, if the albedo and surface normal is correct, is the light and its properties. The light here is assumed to be a perfect sphere with a radius of 0.5 m that has a fixed flux of 1. This is not practical for a real light source. Light could diffuse over distances travelled, etc.

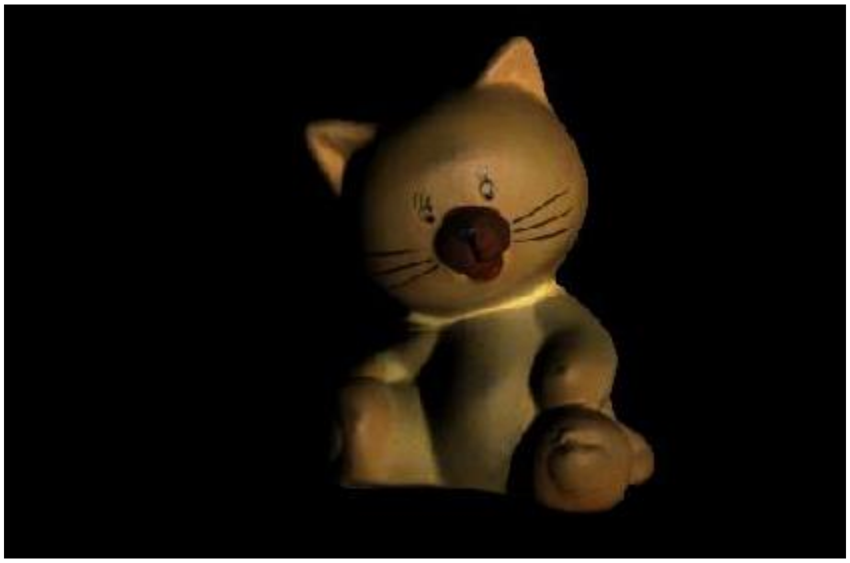


Figure 6: Still From Synthetically Shaded Image

4.0 Surface Fitting

The depth images were composed using the algorithm presented in the assignment 1 manual. This method relies heavily on the normal vectors being correct from previous parts so errors in determining the normal vector can result in inaccurate depth image reproductions.

The results for the various objects turned out fairly well. From a visual standpoint, the depth images looked accurate. For example, from **Figure 7**, the feet should be closer to the camera and the ears should be the furthest from the camera and this is evident in the gradient mapping. The gradient images seemed like a fairly accurate reproduction of a real depth image.

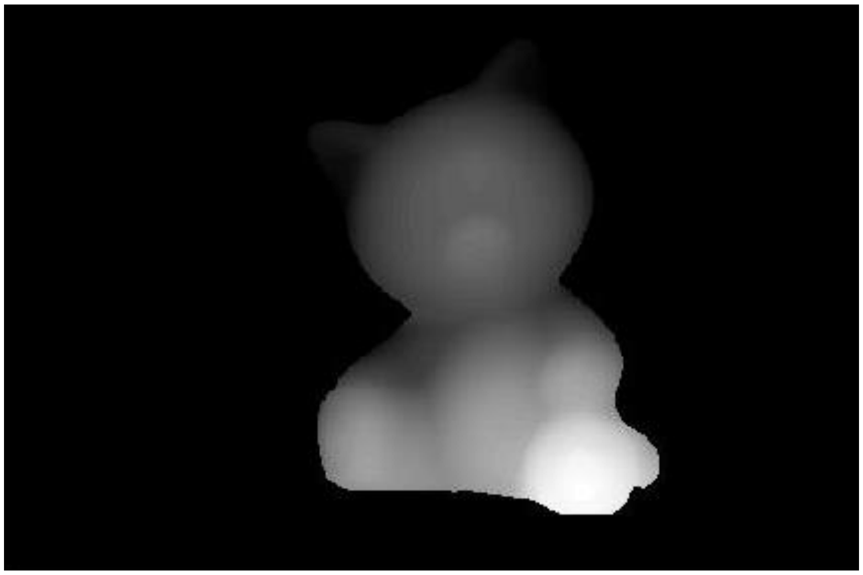


Figure 7: Estimated Depth Image (Cat)

There were some problems around the boundaries of objects as seen in **Figure 8**. Here instead of a fairly circular gradient going out, the shapes are more like ovals. The edges also do not have equal depth from the reproduction. The depth is higher at the left side and the bottom. This could be due to the finite difference method selected. The current algorithm calculates depth from only if the normal in the next pixel exists since this value is required for the algorithm to work. This can be fixed by implementing backwards finite differences at borders where the next normal value does not exist. This can also be further built upon by increasing the order of the finite differences to decrease the error and get a more accurate result.

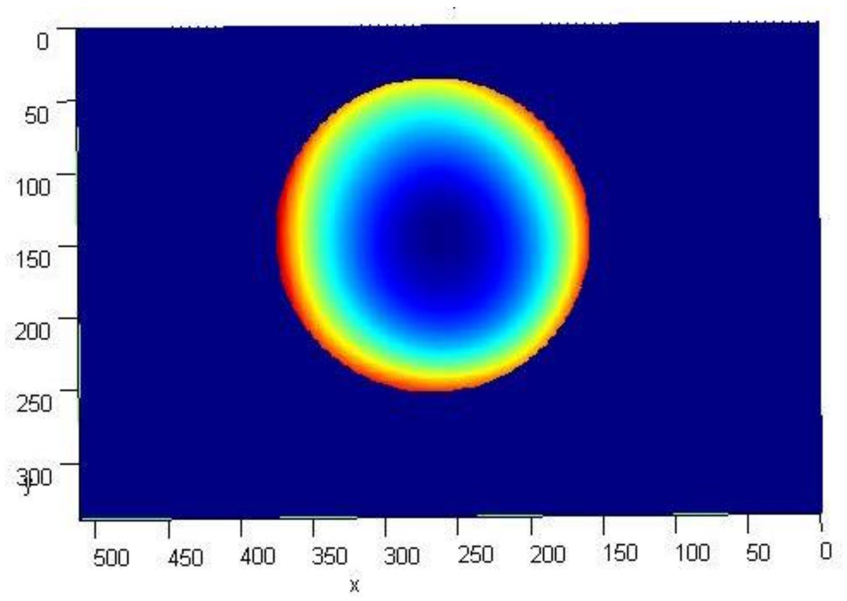


Figure 8: Estimated Depth Image as a Mesh (Grey)

The are also some problems around places where errors previously existed from albedo and normal calculations. For example, in **Figure 9**, at the neck line the depth should be higher than the torso but the gradient shows that it got closer. This error was likely caused by errors in the normal map around this area. This may be avoided by using either more source images with better lighting at the neck line to acquire a better normal map at this area.

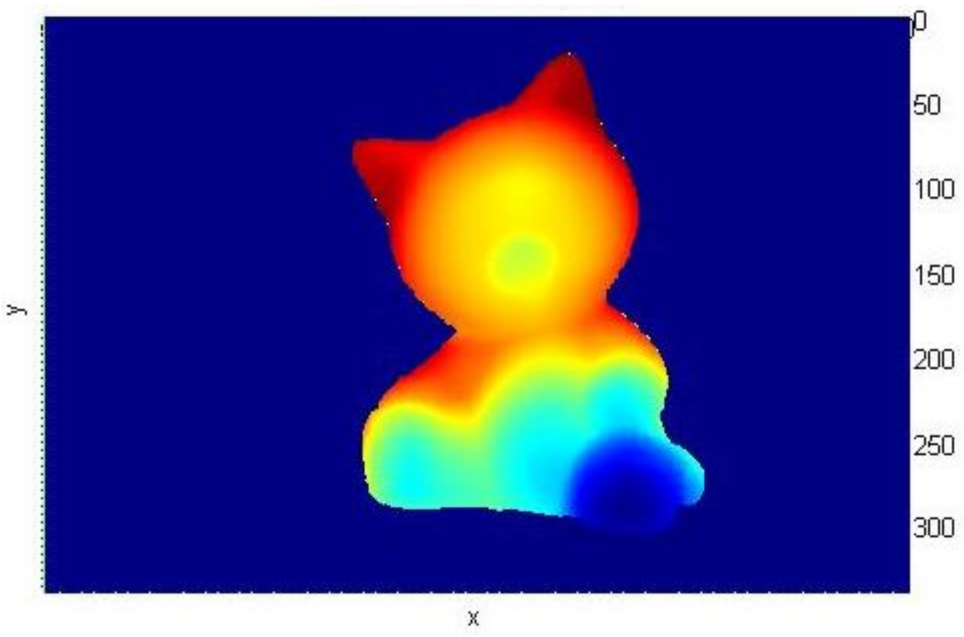


Figure 9: Estimated Depth Image as a Mesh (Cat)

From **Figure 10**, the second leg on the left side has an uneven gradient which should be more uniform. The hoof is further than the rest of the leg but this could be due to more self-shadowing. This is difficult to avoid due to the complexity of the object. It would be hard to orient a light source that would not result in self-shadowing on an object such as this. The front legs also look like it merged into one leg. This could be due to not enough lighting on the front left leg to obtain a reliable surface normal for that location. Finally, hard transitions in depth are shown as gradients, like between the tail and the leg. This could be due to the finite element method chosen. This method is treating everything like a surface so transitions look like gradients. This could be eliminated with pictures with higher resolutions so the spacing between adjacent pixels are smaller and more information regarding the normal map can be obtained.

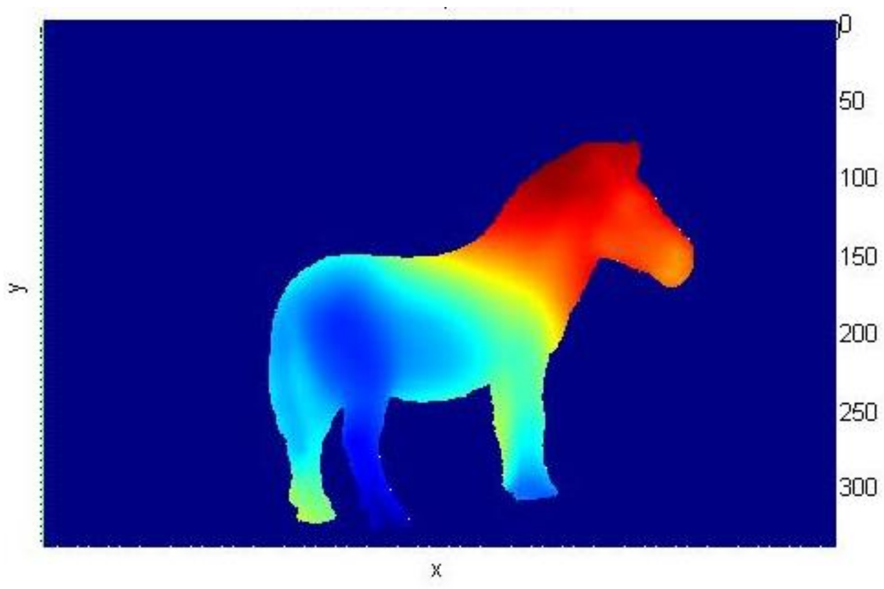


Figure 10: Estimated Depth Image as a Mesh (Horse)

5.0 Estimate Light Source Direction

It is possible to estimate both light source directions using one novel chrome ball image with two distant light sources. Each source should have its own point of perfect spectral reflection. If it is assumed that at a point of perfect spectral reflection, the effect on the other light source is negligible since the other source is dominant, then the above equations can be used to calculate the origin of two light sources using the above equations for perfect spectral reflection twice with two (x, y) coordinates corresponding to the two brightest locations.

Appendix A – Buddha

Reconstructions:

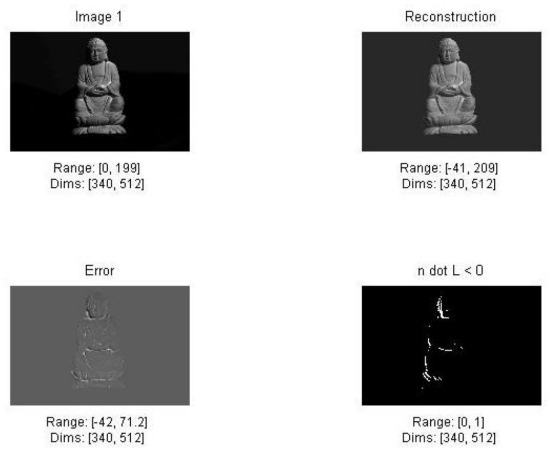


Image 1

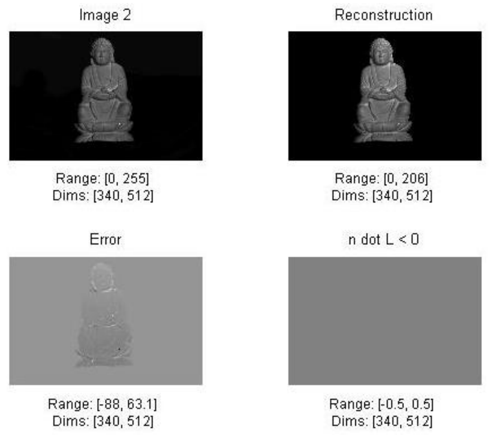


Image 2

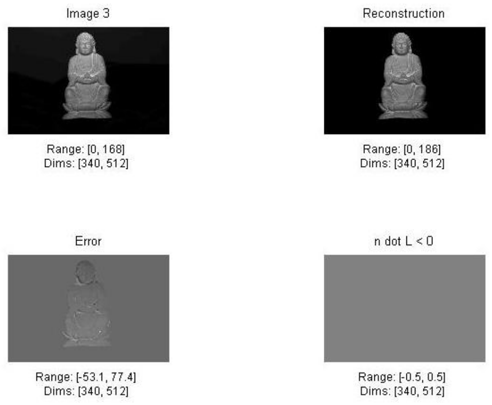


Image 3

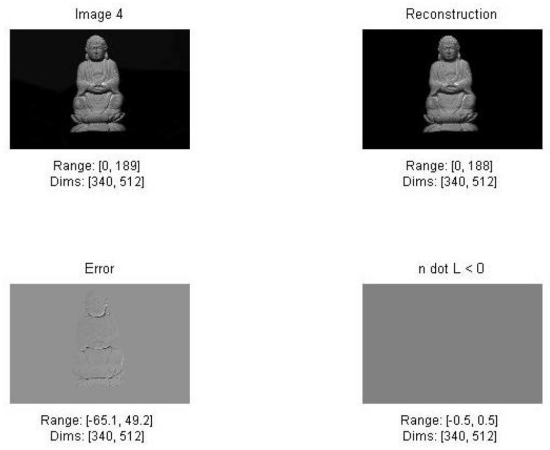


Image 4

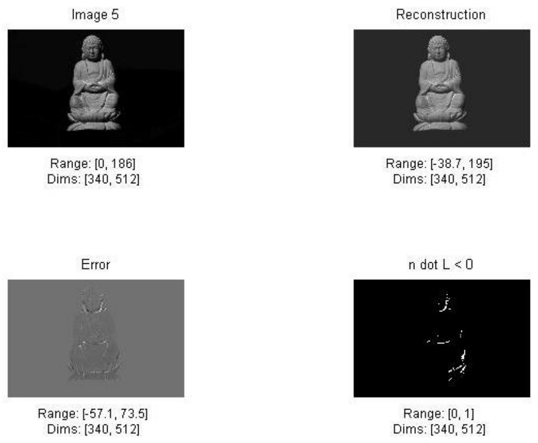


Image 5

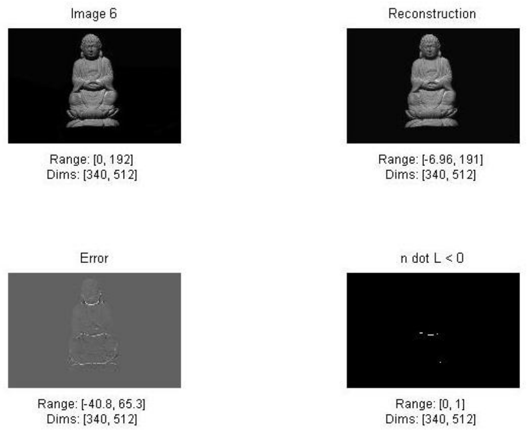


Image 6

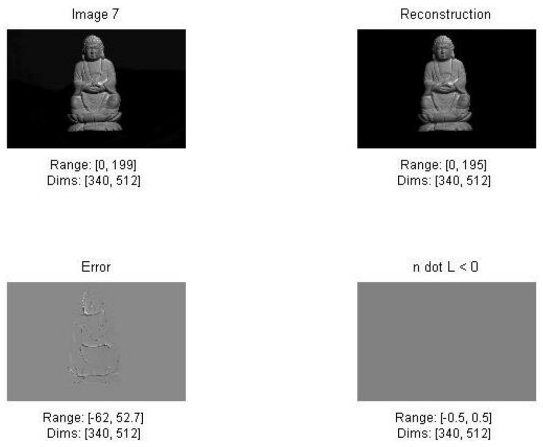


Image 7

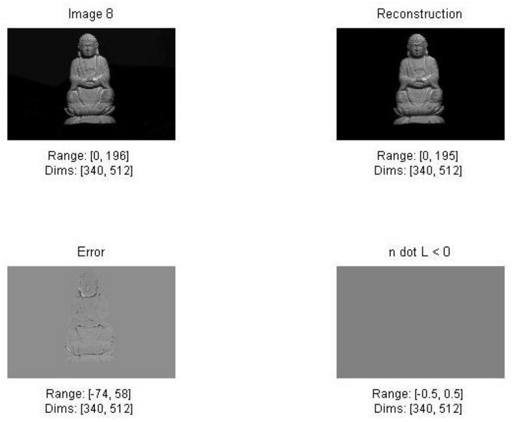
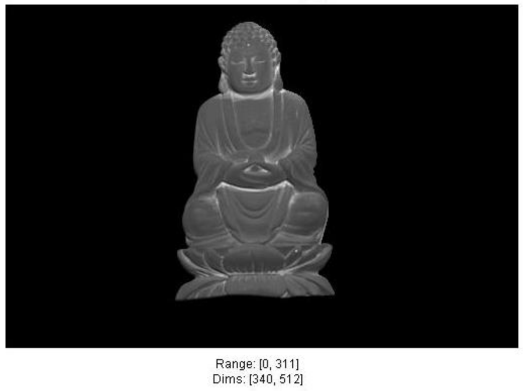
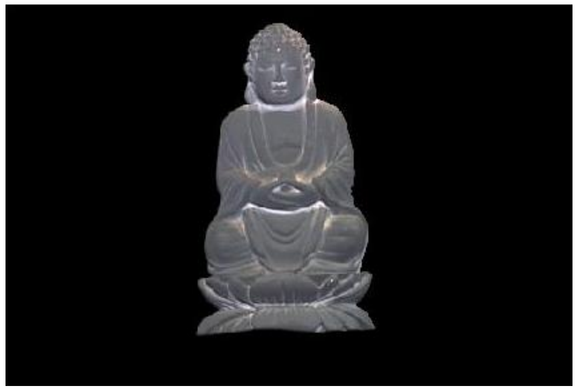


Image 8

Albedos:

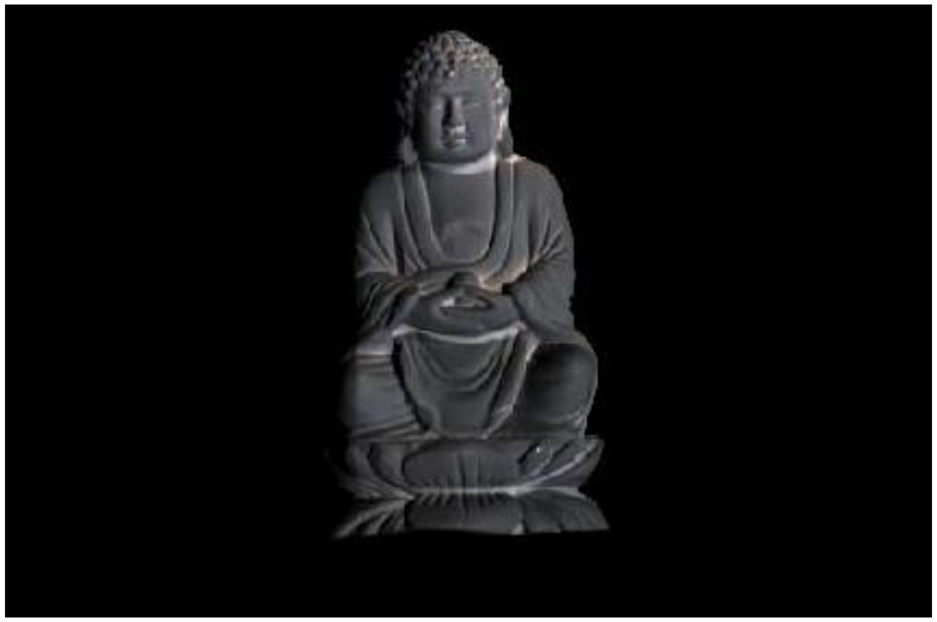


Gray Albedo



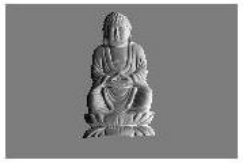
RGB Albedo

Synthetically Shaded Image Still:



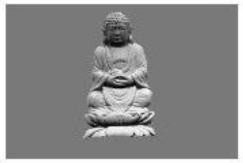
Surface Normals:

Surface Normal (nx)



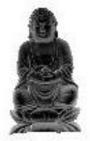
Range: [-0.965, 0.97]
Dims: [340, 512]

Surface Normal (ny)



Range: [-0.876, 0.983]
Dims: [340, 512]

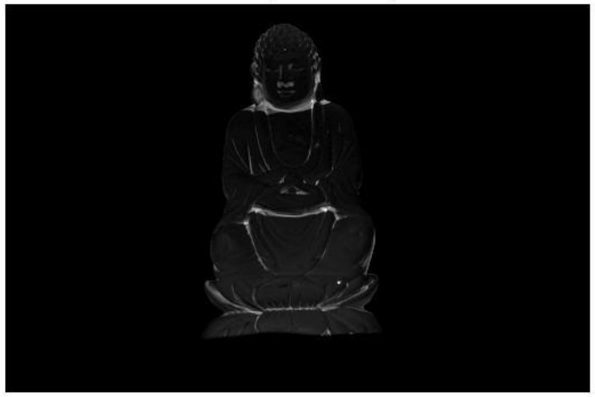
Surface Normal (nz)



Range: [-1, 0]
Dims: [340, 512]

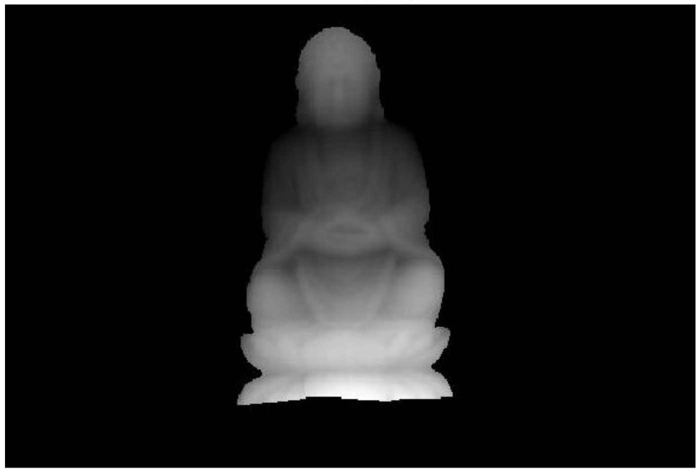
RMS:

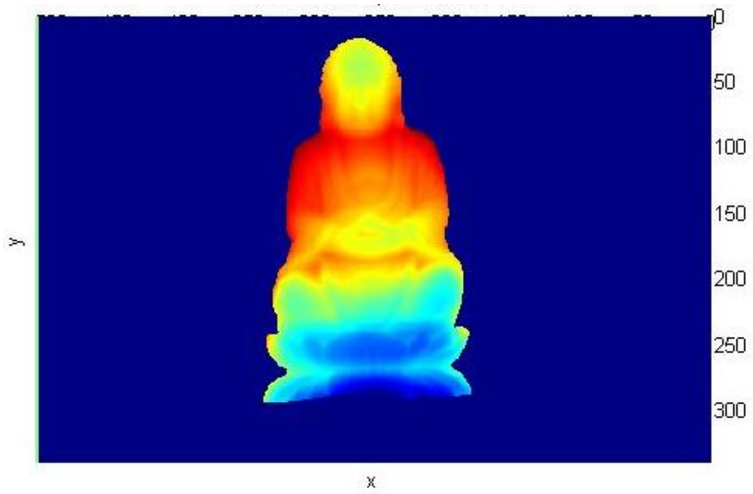
RMS error (total: 1142.396826)



Range: [0, 50.2]
Dims: [340, 512]

Depth Images:


Estimated Object Depth



Estimated Depth as Mesh

Appendix B – Cat

Reconstructions:

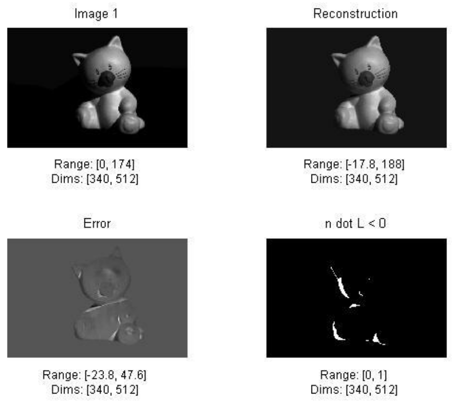


Image 1

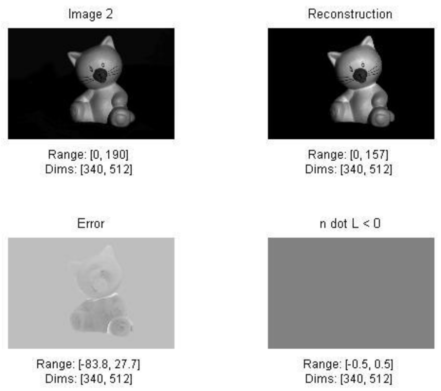


Image 2

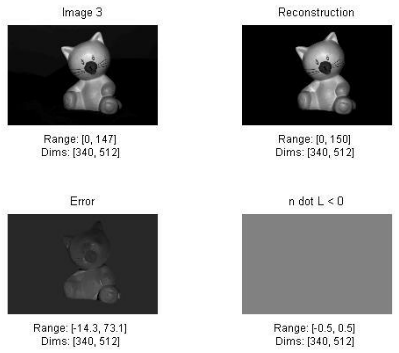


Image 3

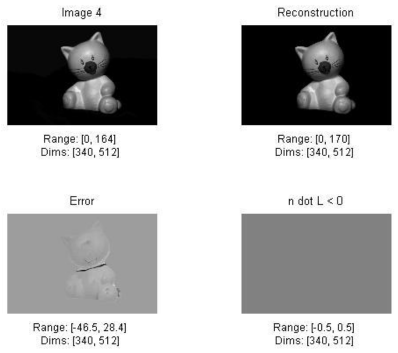


Image 4

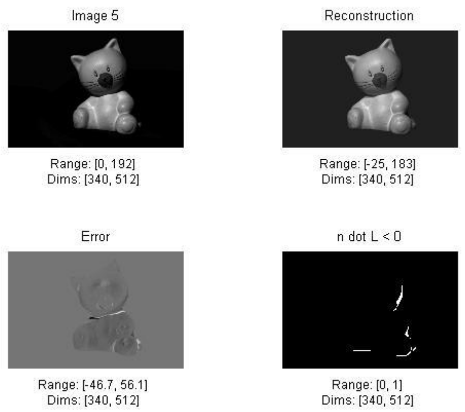


Image 5

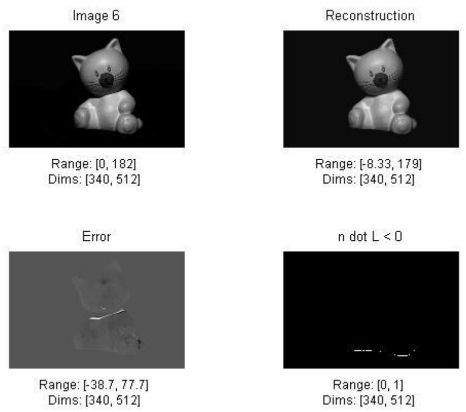


Image 6

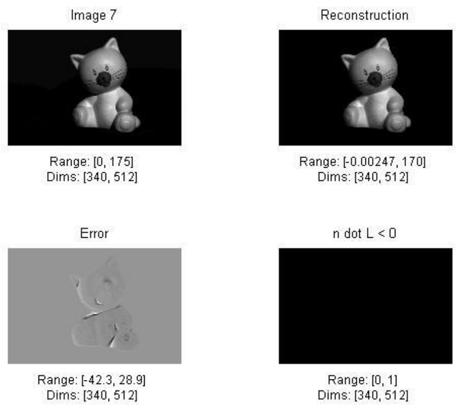


Image 7

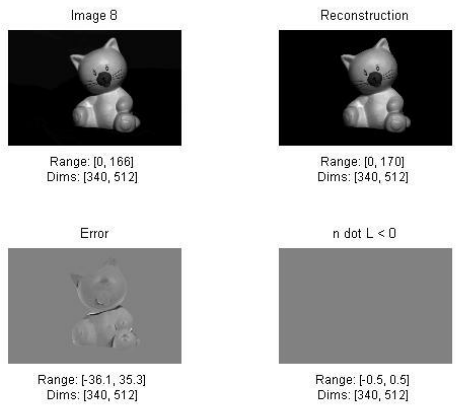
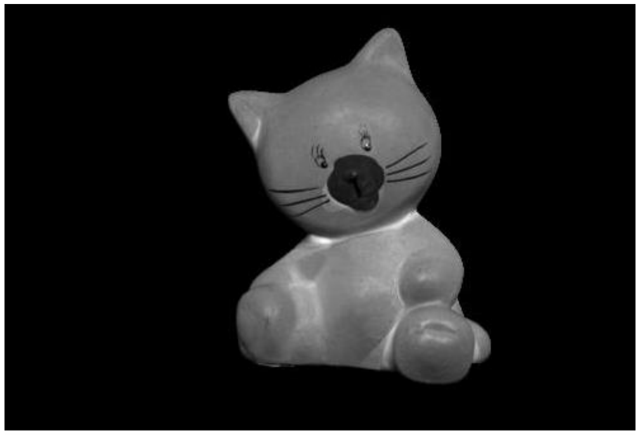
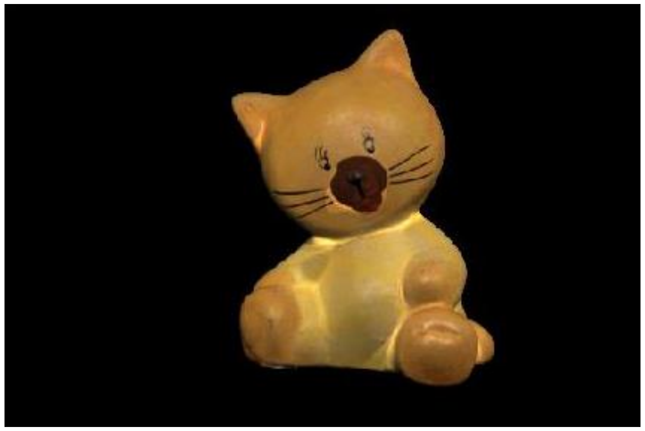


Image 8

Albedos:

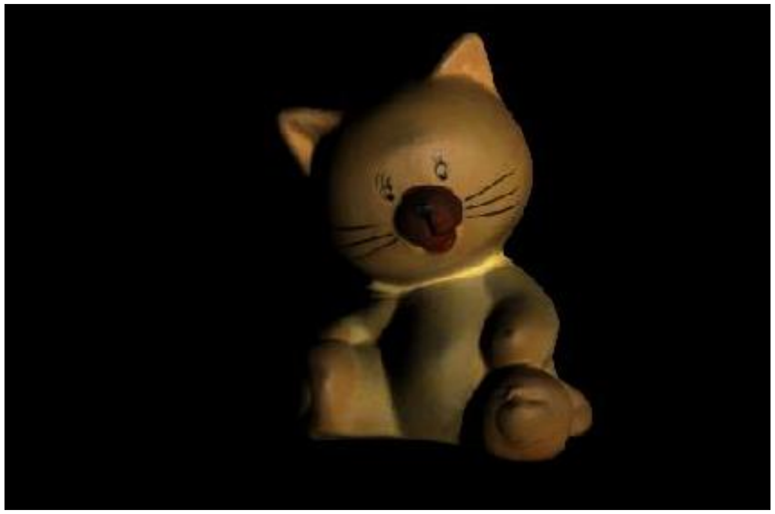


Gray Albedo

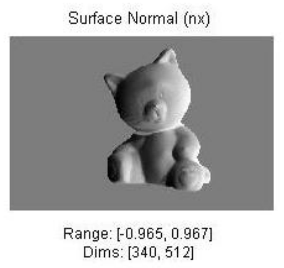


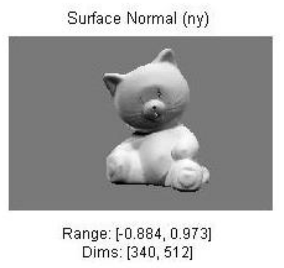
RGB Albedo

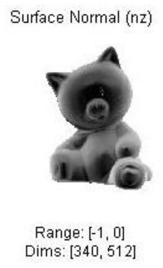
Synthetically Shaded Image Still:



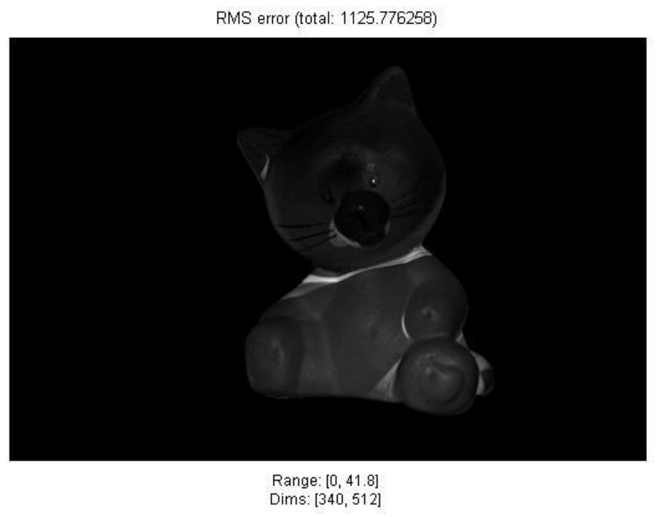
Surface Normals:



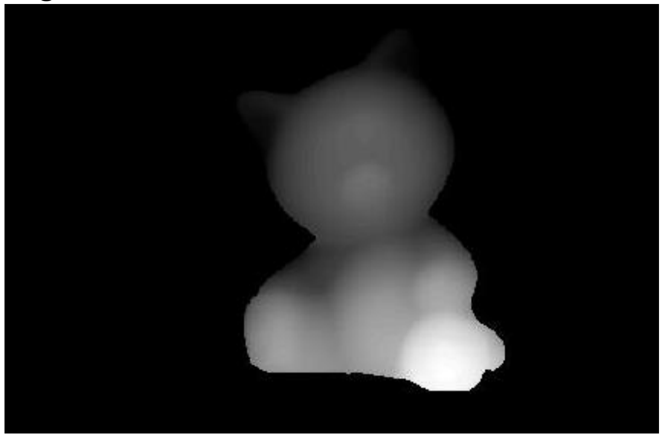




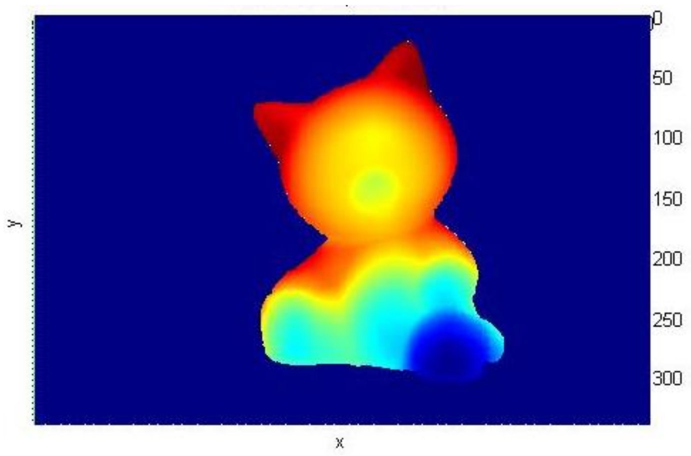
RMS:



Depth Images:



Estimated Object Depth



Estimated Depth as Mesh

Appendix C – Gray

Reconstructions:

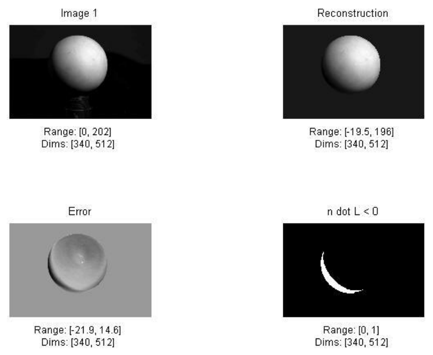


Image 1

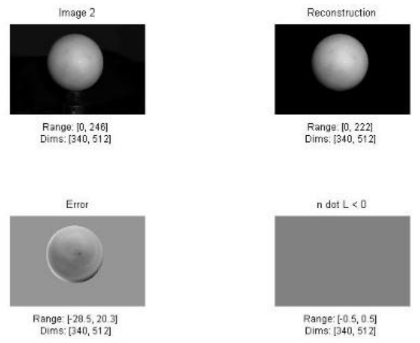


Image 2

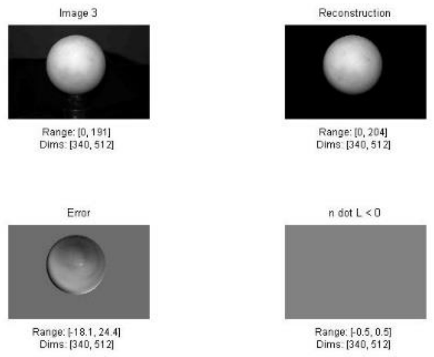


Image 3

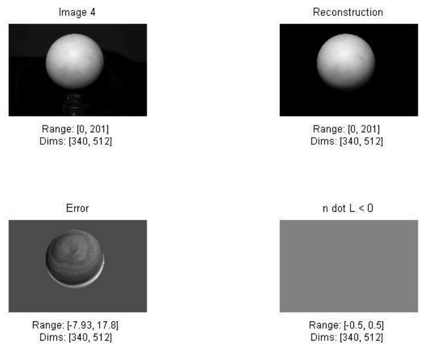


Image 4

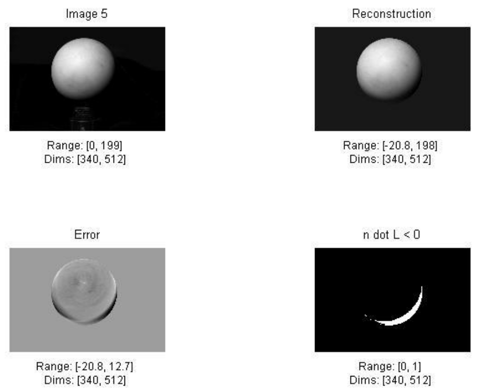


Image 5

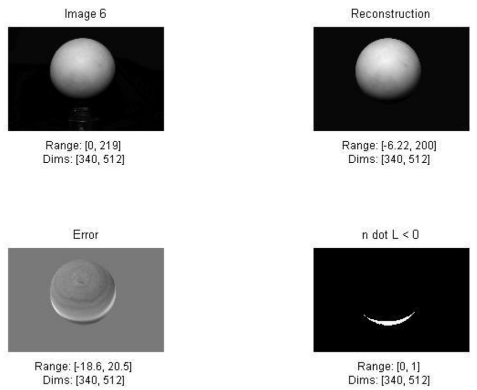


Image 6

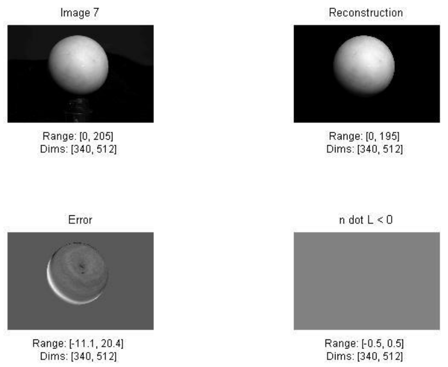


Image 7

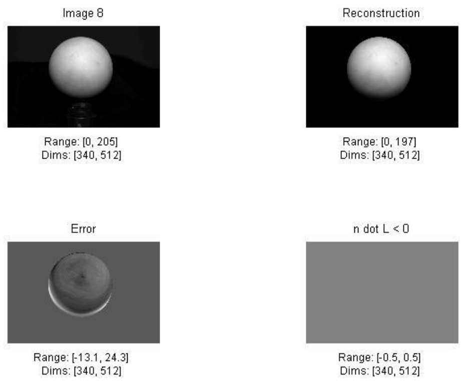


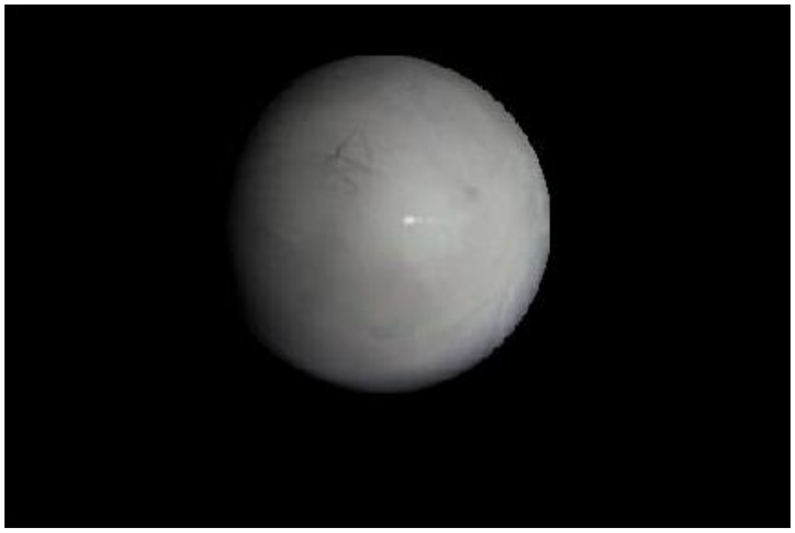
Image 8

Albedos:

Gray Albedo

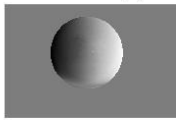
RGB Albedo

Synthetically Shaded Image Still:



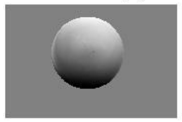
Surface Normals:

Surface Normal (nx)



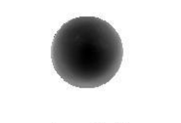
Range: [-0.952, 0.961]
Dims: [340, 512]

Surface Normal (ny)



Range: [-0.884, 0.854]
Dims: [340, 512]

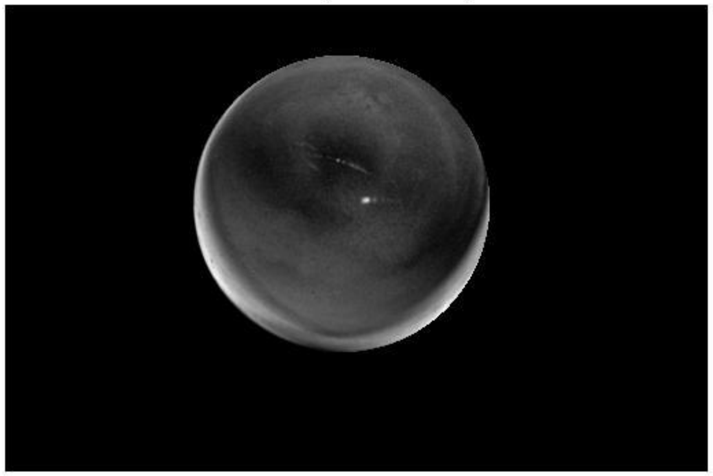
Surface Normal (nz)



Range: [-1, 0]
Dims: [340, 512]

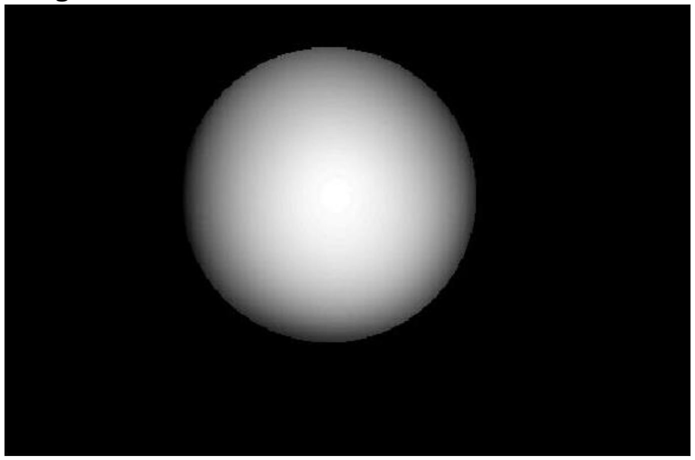
RMS:

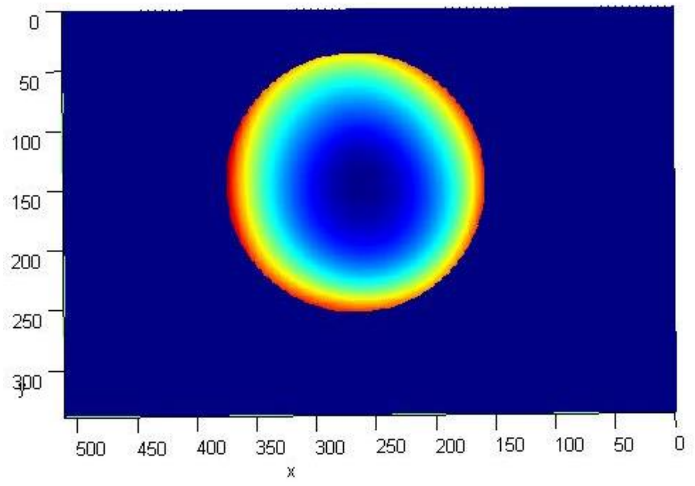
RMS error (total: 887.137402)



Range: [0, 15.6]
Dims: [340, 512]

Depth Images:


Estimated Object Depth



Estimated Depth as Mesh

Appendix D - Horse

Reconstructions:

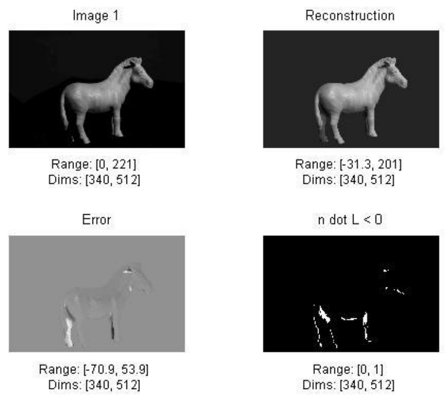


Image 1

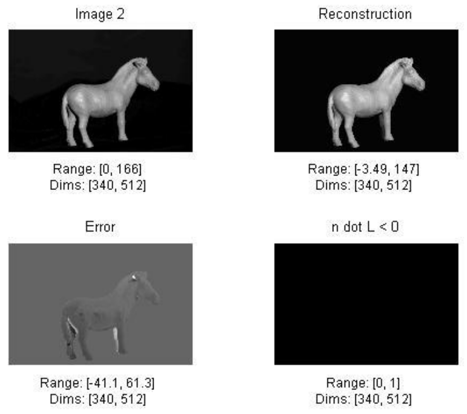


Image 2

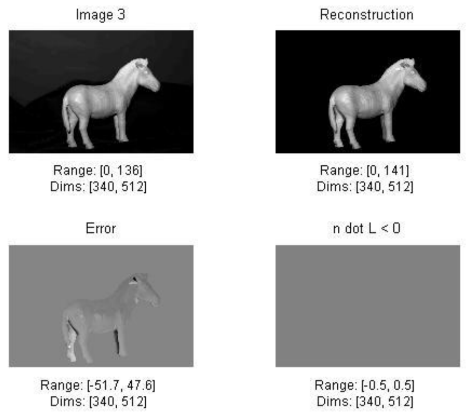


Image 3

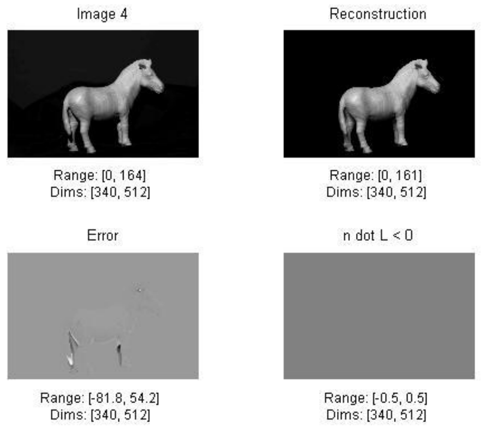


Image 4

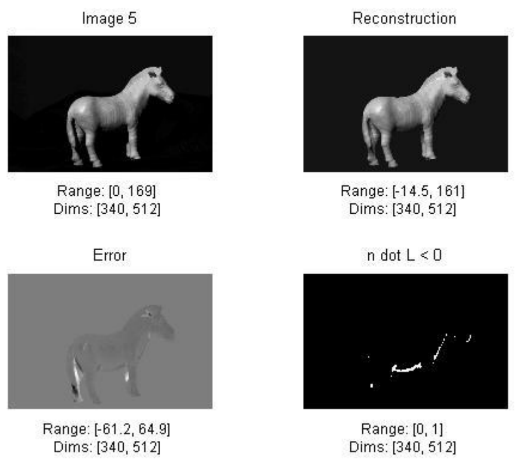


Image 5

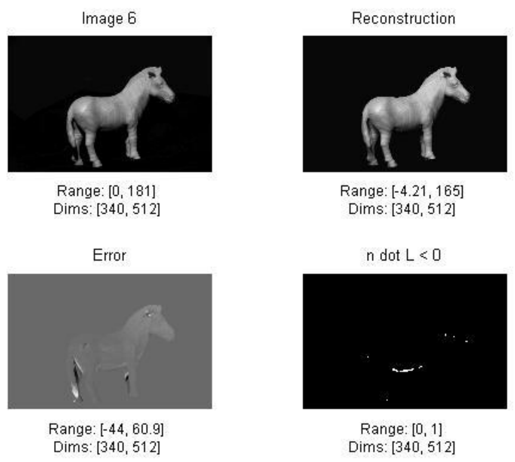


Image 6

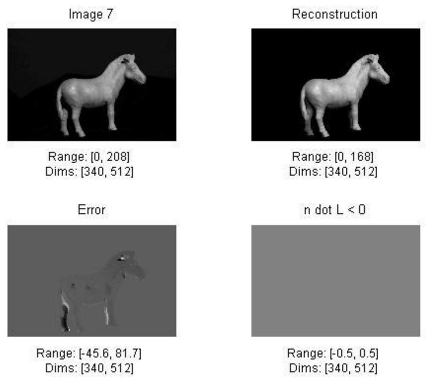


Image 7

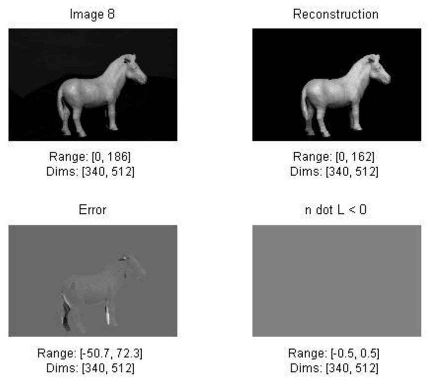
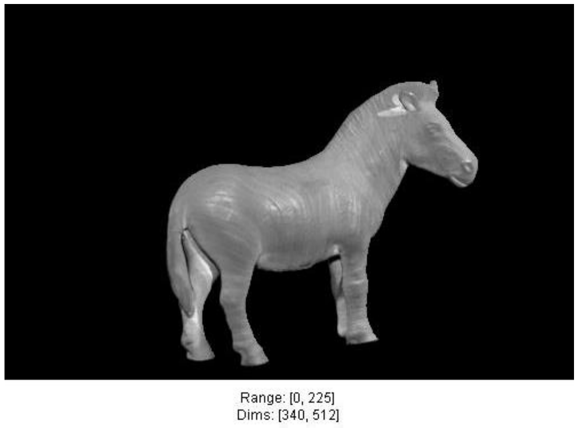
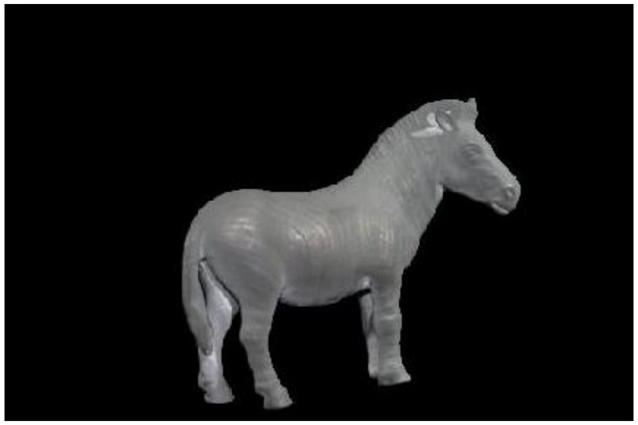


Image 8

Albedos:

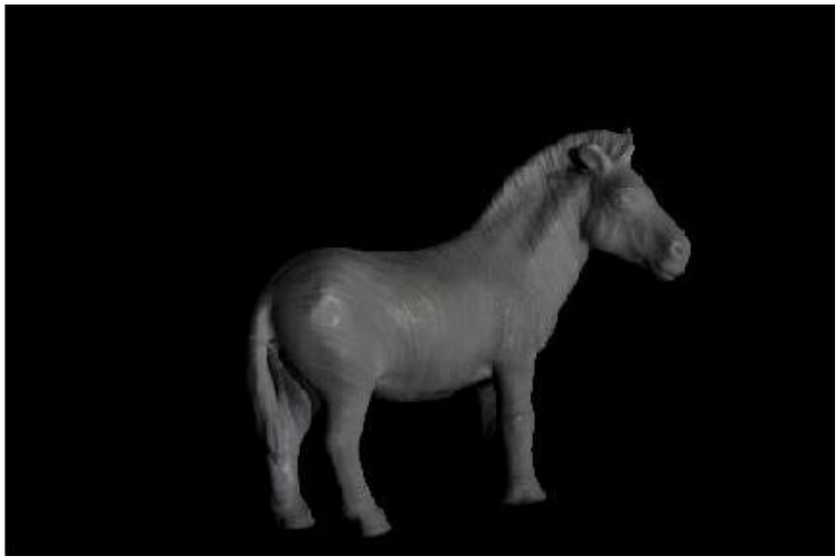


Gray Albedo

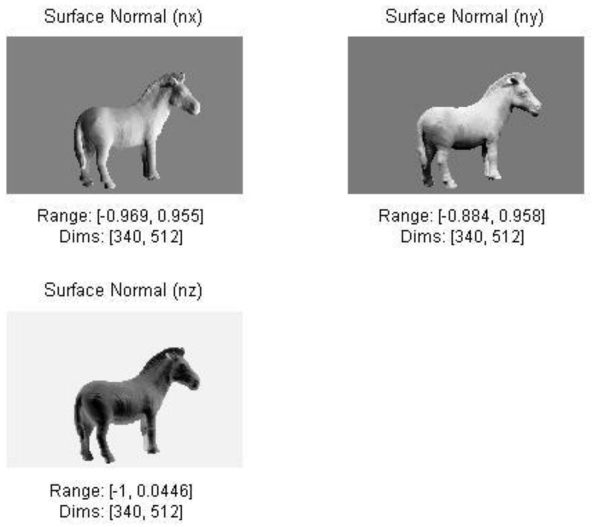


RGB Albedo

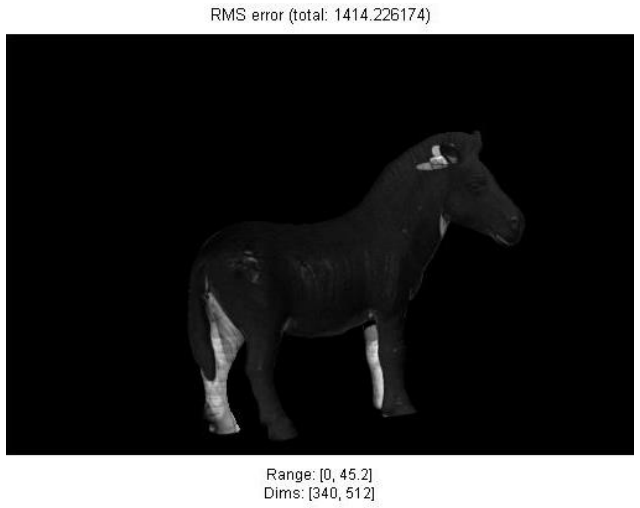
Synthetically Shaded Image Still:



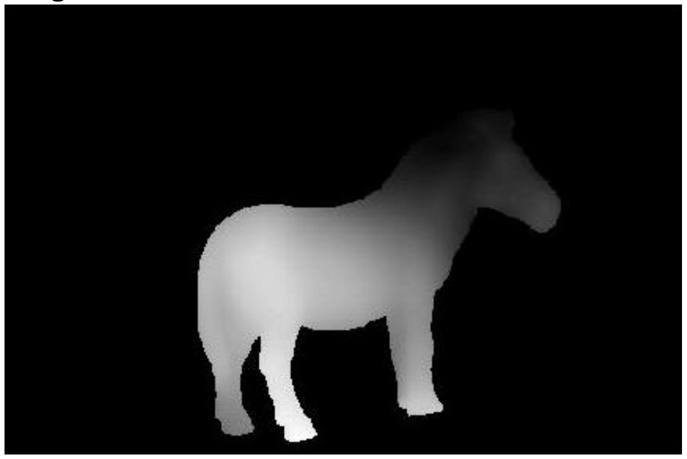
Surface Normals:



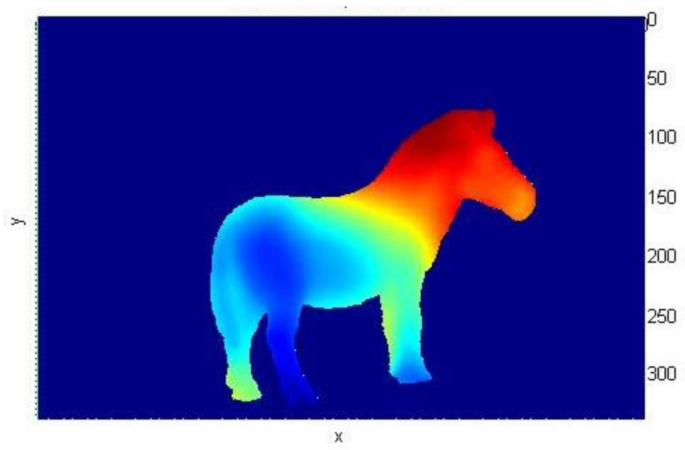
RMS:



Depth Images:



Estimated Object Depth



Estimated Depth as Mesh

Appendix E - Owl

Reconstructions:

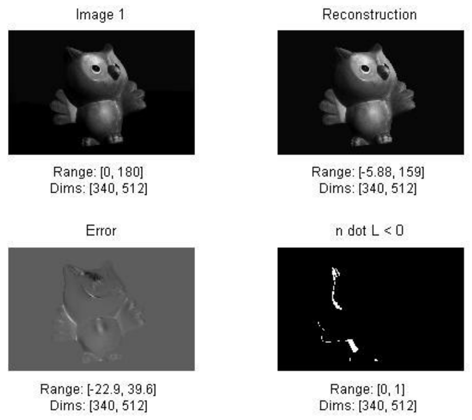


Image 1

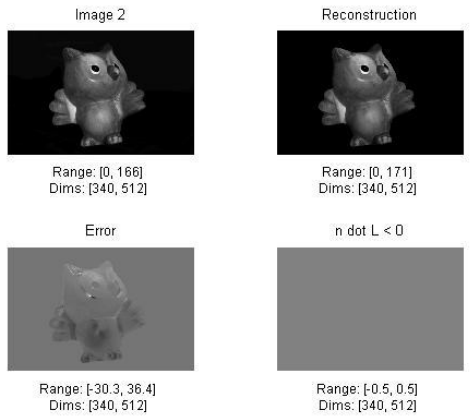


Image 2

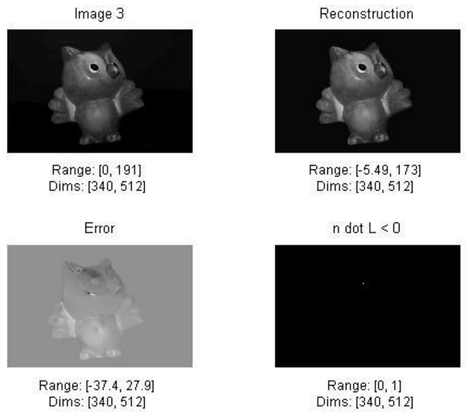


Image 3

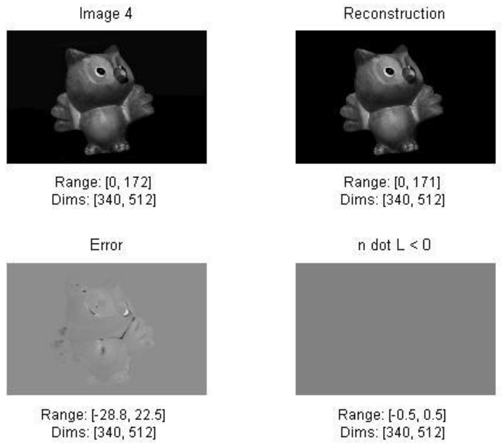


Image 4

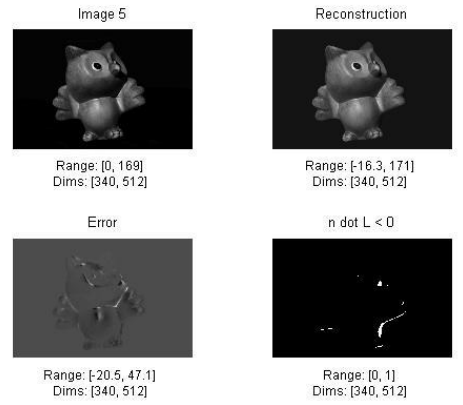


Image 5

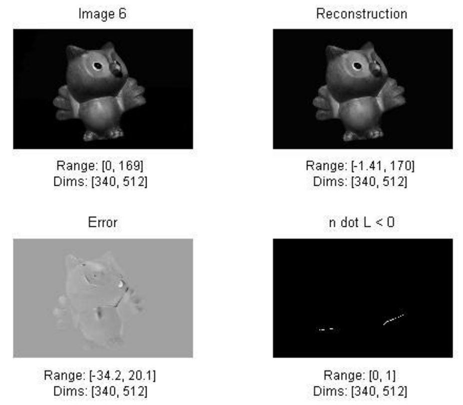


Image 6

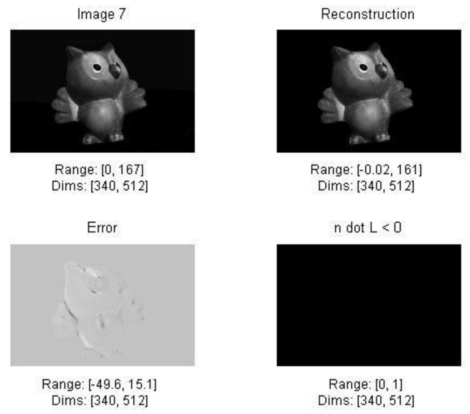


Image 7

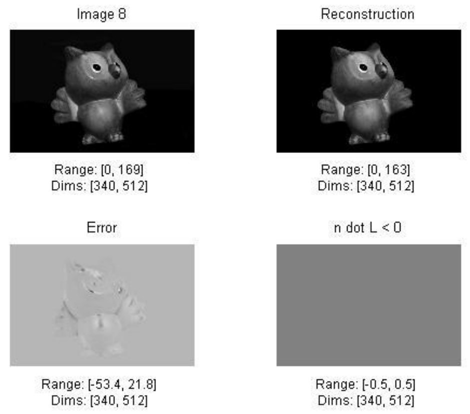
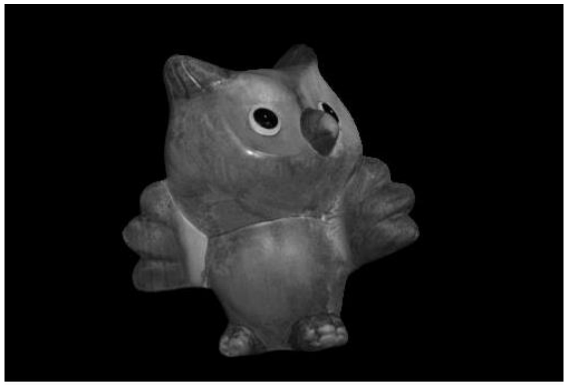


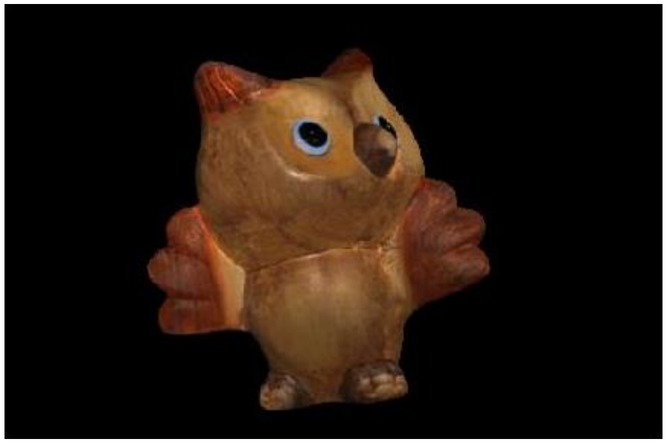
Image 8

Albedos:



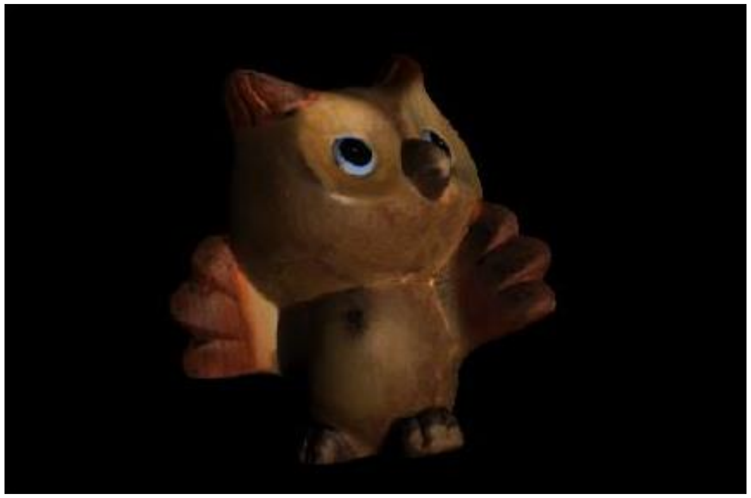
```
Range: [0, 232]
Dims: [340, 512]
```

Gray Albedo



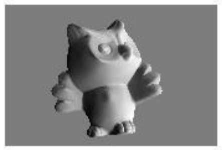
RGB Albedo

Synthetically Shaded Image Still:



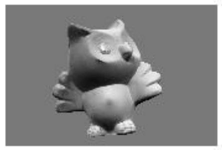
Surface Normals:

Surface Normal (nx)



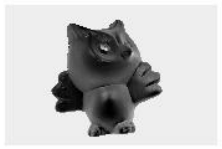
Range: [-0.965, 0.966]
Dims: [340, 512]

Surface Normal (ny)



Range: [-0.883, 0.994]
Dims: [340, 512]

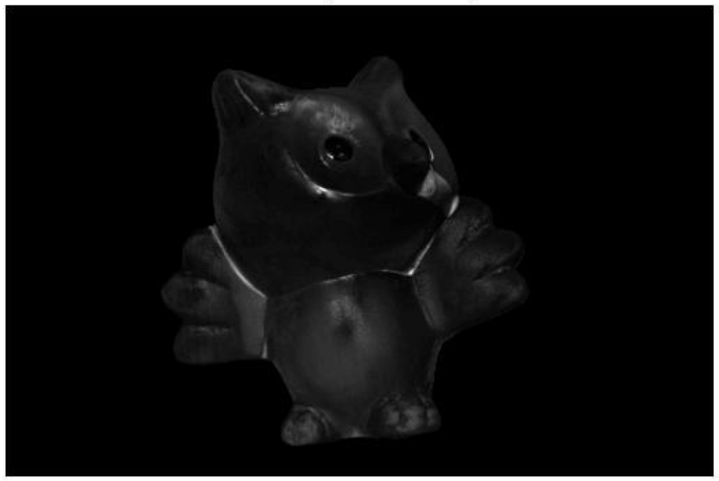
Surface Normal (nz)



Range: [-1, 0.0713]
Dims: [340, 512]

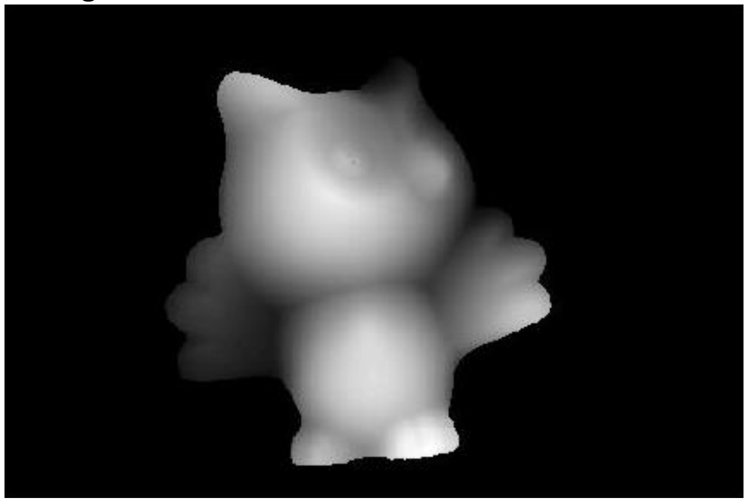
RMS:

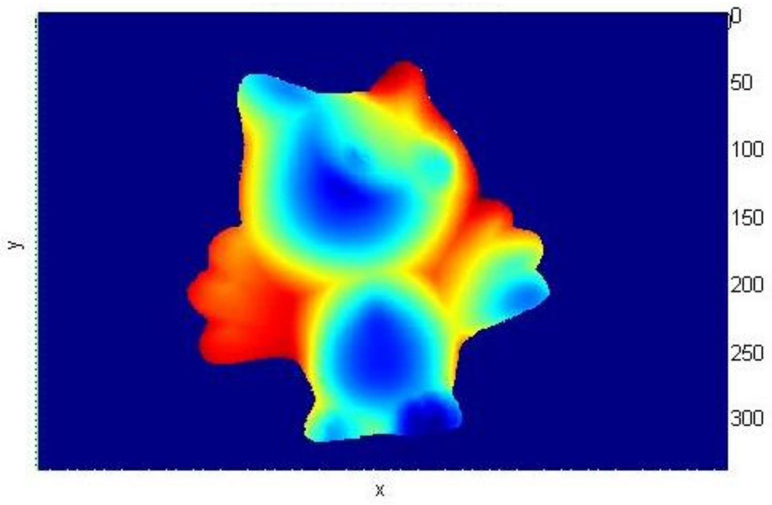
RMS error (total: 746.087649)



Range: [0, 26.9]
Dims: [340, 512]

Depth Images:


Estimated Object Depth



Estimated Depth as Mesh

Appendix F - Rock

Reconstructions:

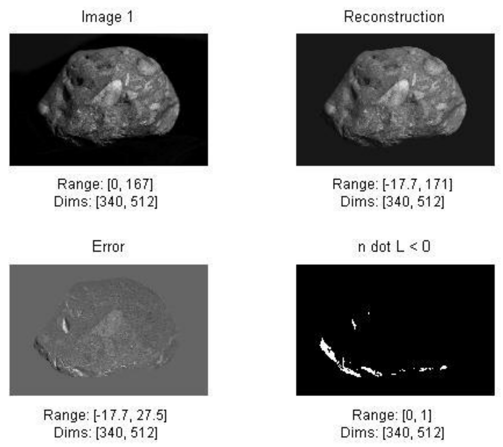


Image 1

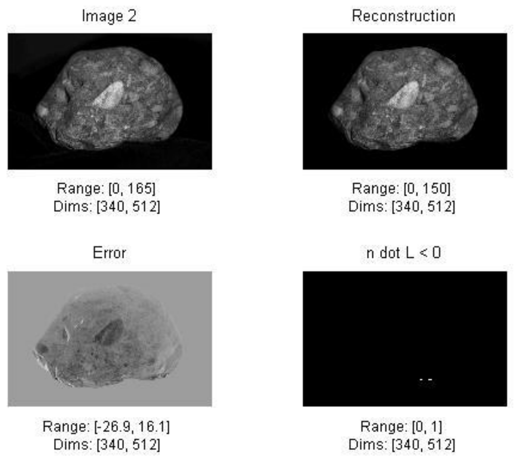


Image 2

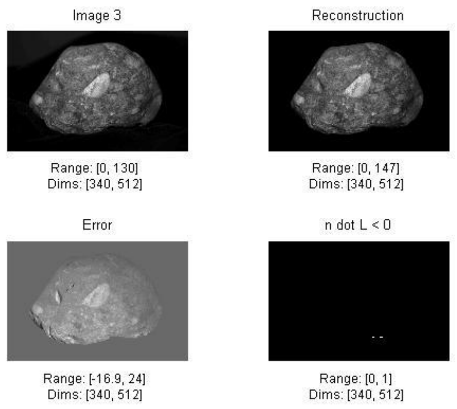


Image 3

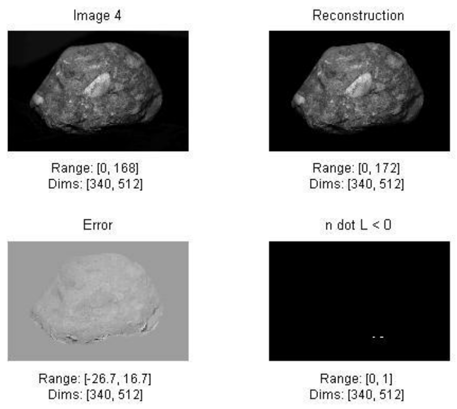


Image 4

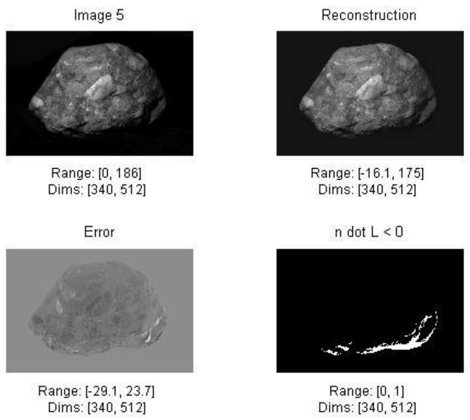


Image 5

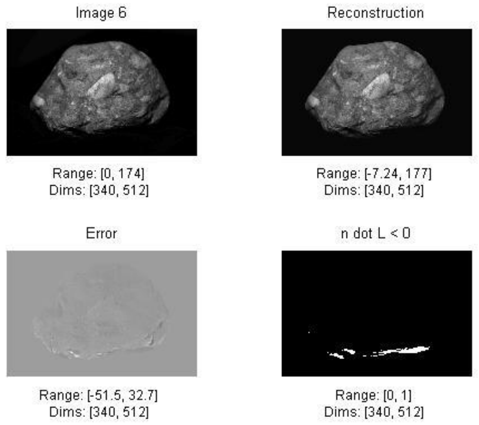


Image 6

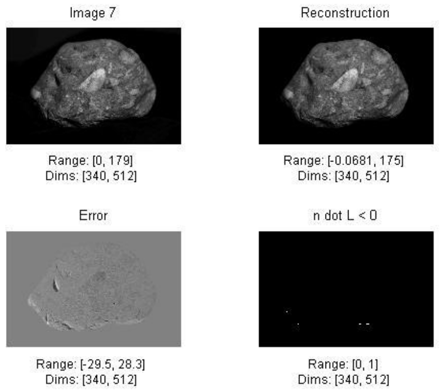


Image 7

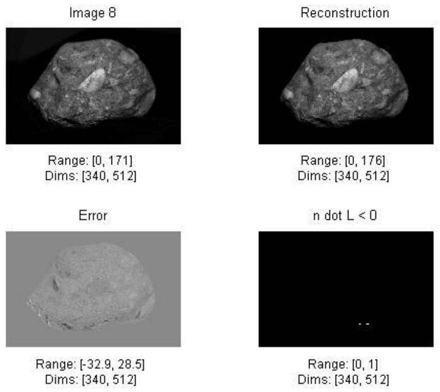
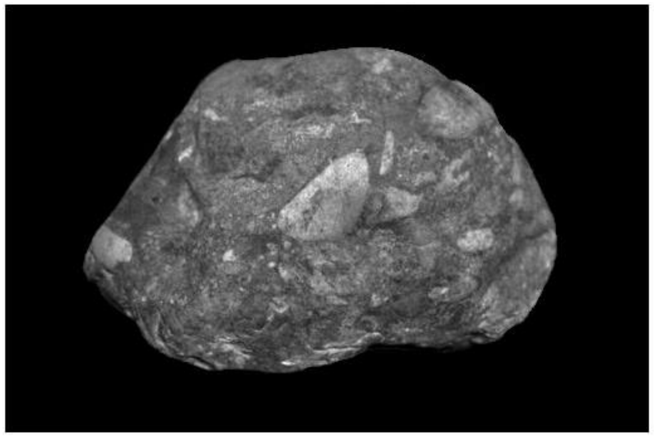
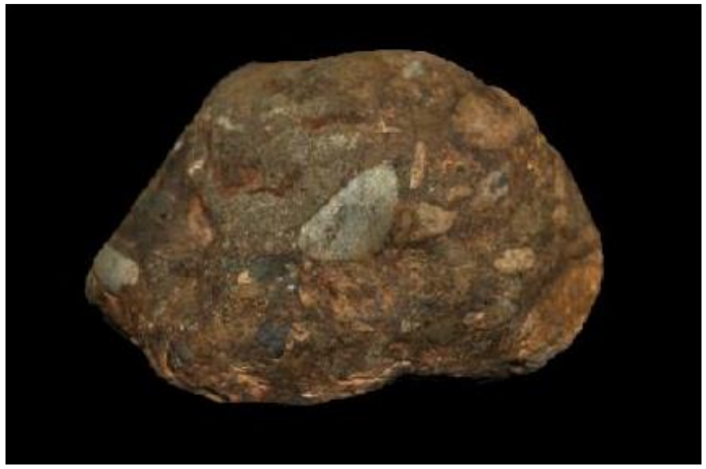


Image 8

Albedos:

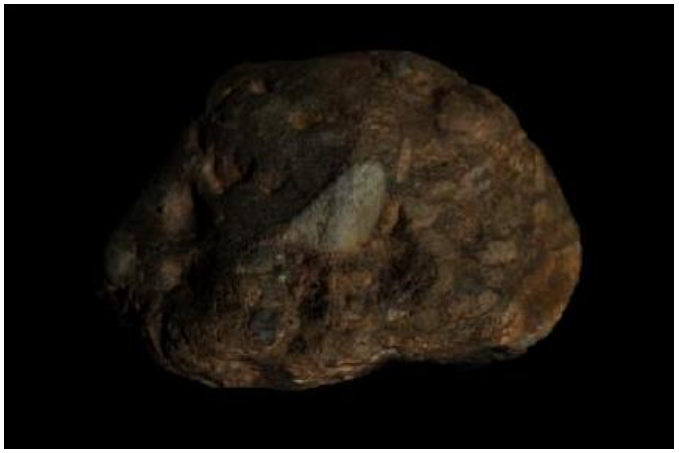


Gray Albedo



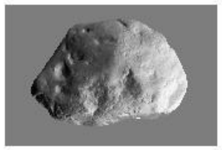
RGB Albedo

Synthetically Shaded Image Still:



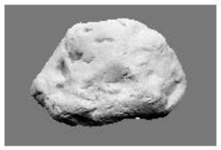
Surface Normals:

Surface Normal (nx)



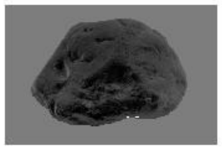
Range: [-0.965, 1]
Dims: [340, 512]

Surface Normal (ny)



Range: [-0.884, 1]
Dims: [340, 512]

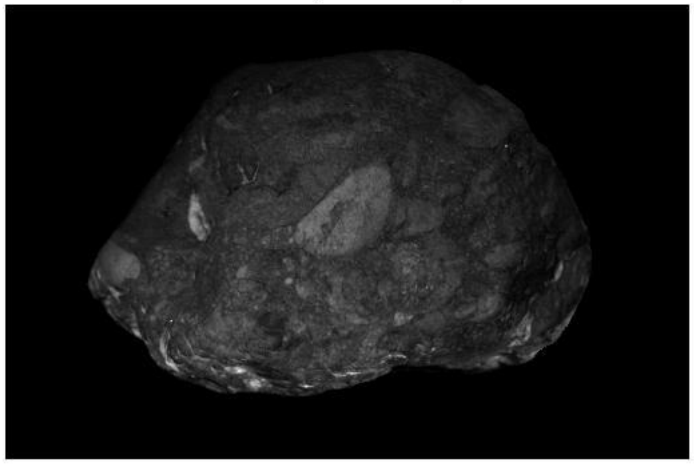
Surface Normal (nz)



Range: [-1, 1]
Dims: [340, 512]

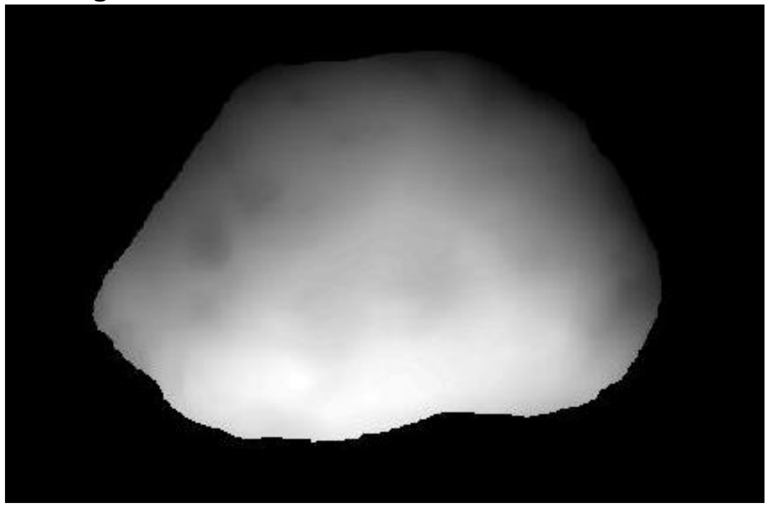
RMS:

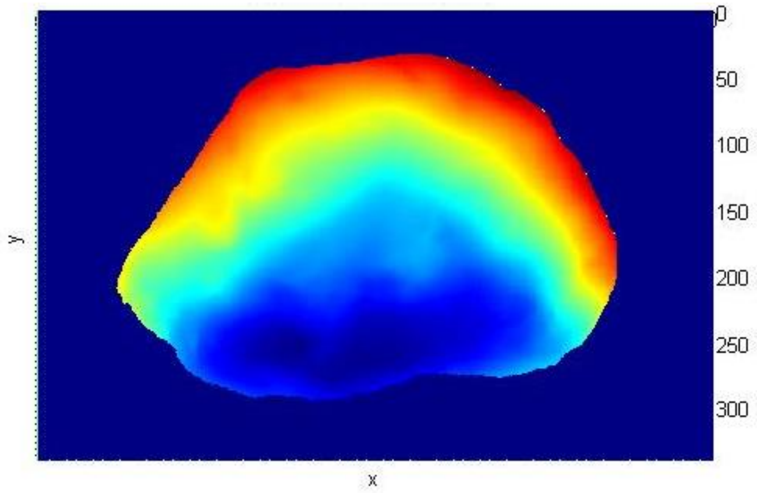
RMS error (total: 1188.055039)



Range: [0, 21.8]
Dims: [340, 512]

Depth Images:


Estimated Object Depth



Estimated Depth as Mesh