Report CV1: Photometric Stereo and Color Spaces

Minh Ngo 10897402[1], Riaan Zoetmulder[2] 6072909

February 17, 2016

University of Amsterdam
[1]minh.ngole@student.uva.nl, [2] riaanzoetmulder@gmail.com

1 Photometric Stereo

Photometric Stereo is a method of recovering the surface from a series of pictures taken under different lighting conditions. In order to find the height map we had to assume (or estimate) the light source direction, after which albedo can be computed, correcting for contributions from shaded areas, and the surface normals. We also ensured that second order partial derivatives did not differ from each other significantly (test of integrability).

1.1 Albedo

We started out by calculating the vector $\vec{g}$ for each point. We did this by solving the following system of equations:

$$\mathcal{I}\mathbf{i} = \mathcal{I}\mathcal{V}\vec{g} \tag{1}$$

where $\mathbf{i}$ is the vector of intensity values for a pixel with the same coordinates on each of the pictures, $\mathcal{I}$ is a diagonal matrix with $\mathbf{i}$ on the diagonal and $\mathcal{V}$ is the matrix of the light source directions. Using the matrix $\mathcal{I}$ has the effect of zeroing out the contributions of shaded regions since these will have values of 0.

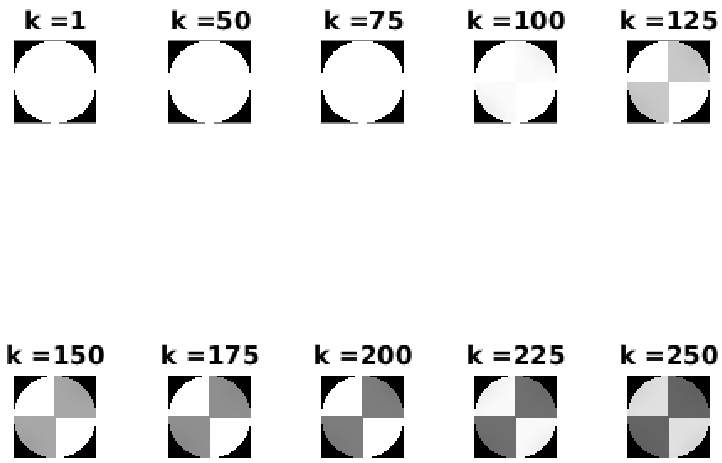

Figure 1: Variety of Albedos for different scales of the light source vectors.

Albedo is a norm of $\vec{g}$. By experimenting with scaling of $\mathcal{V}$ by changing the camera dependent scaling factor $\mathbf{k}$ we obtained different albedos [Fig 1]. It can be seen that the increase of $\mathbf{k}$ leads to the increase of intensity that give a better picture [Fig 2].

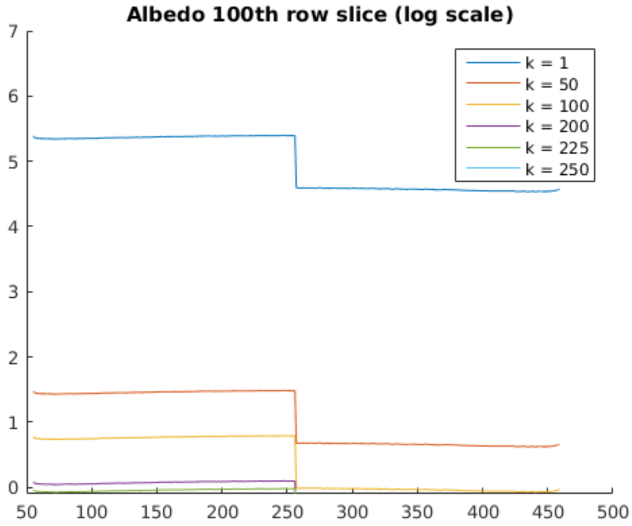

Figure 2: Albedo values in the 100th row slice in the logarithmic scale.

1.2 Surface Normals

The next step after calculating $\vec{g}$ is normalizing this vector to obtain the surface normals $\vec{N} = \{N_1, N_2, N_3\}$ [Fig 3].

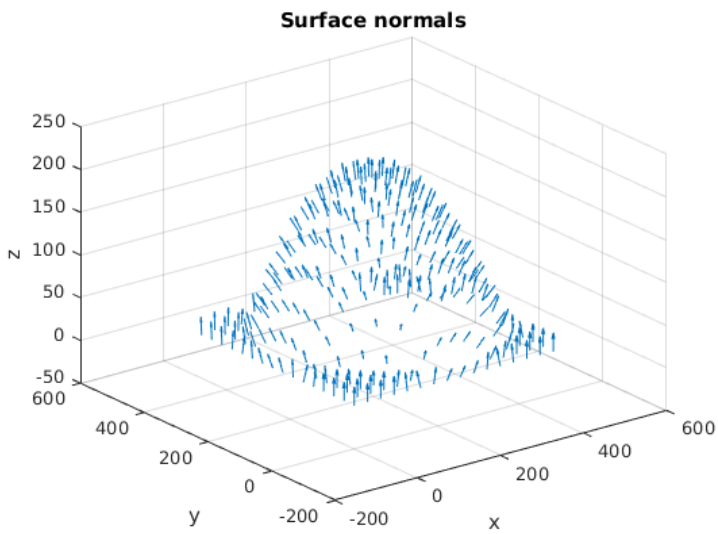

Figure 3: Surface Normals for the Object.

1.3 Recovered Surface

To recover the height map we computed surface derivatives in the x and y direction ($p = \frac{N_1}{N_3}$ and $q = \frac{N_2}{N_3}$). Next we do hill climbing from the border of the region regarding to derivatives and form a surface [Fig 4].

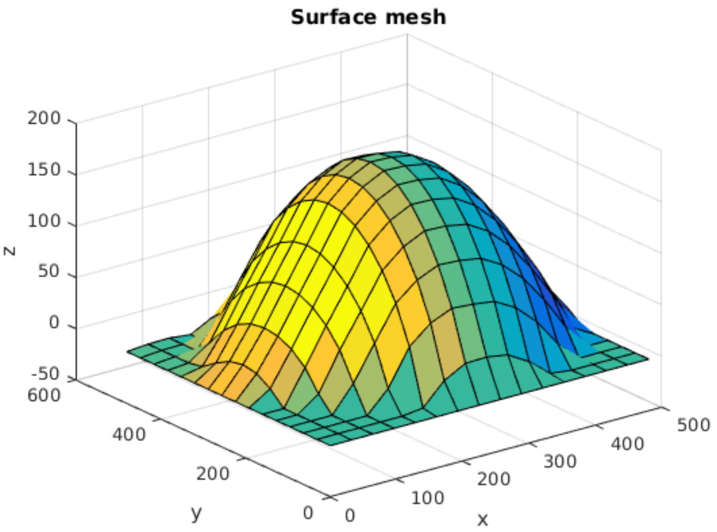

Figure 4: Surface for the object

1.4 Test of Integrability

Without the test of integrability, the surface that was plotted wouldn't be smooth. To detect outliers we compute the second derivatives $\frac{dp}{dy}$ and $\frac{dq}{dx}$ and check that the difference between them should be small, in ideal case - equal (Petrovic et al., 2001). If it's not, **p** and **q** are set to 0. A plot of the outliers is shown in figure 5.

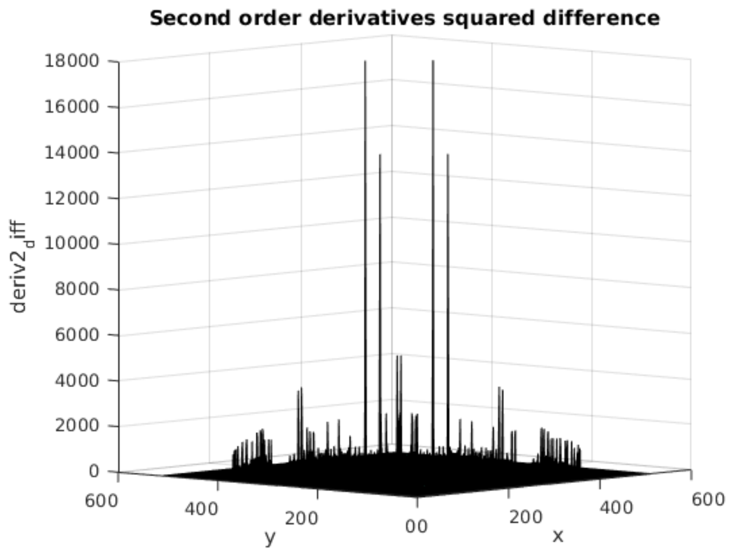

Figure 5: Second order derivatives squared difference. Peaks are outliers.

2 Color Spaces

2.1 Opponent Color Space

In order to calculate the opponent color space of a given image, we need to iterate over each pixel in the image and convert the R, G and B using the following formula.

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} \frac{R-G}{\sqrt{2}} \\ \frac{R+G-2B}{\sqrt{6}} \\ \frac{R+G+B}{\sqrt{3}} \end{pmatrix} \tag{2}$$

The results are shown in figure 6

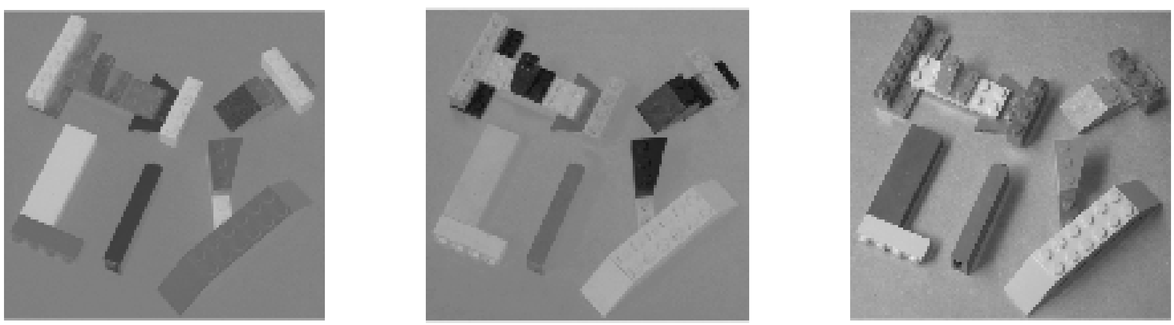

Figure 6: Opponent Color Space, depicting the three opponent channels.

When we plotted the different channels. We normalized the obtained values for O_1, O_2 and O_3 between 0 and 1.

2.2 Normalized RGB Color Space

In order to normalize the color space we iterated over each pixel in the image and use the following formula to calculate the normalized R, G and B values.

$$\begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} \frac{R}{R+G+B} \\ \frac{G}{R+G+B} \\ \frac{B}{R+G+B} \end{pmatrix} \tag{3}$$

The results are shown in figure 7.

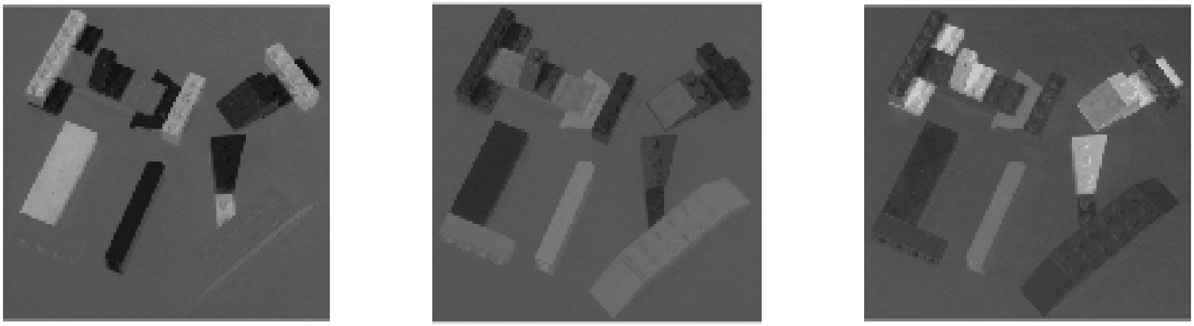

Figure 7: Normalized RGB Color space.

2.3 HSV Color Space

HSV color space is a non-linear transformation of RBG color space. Its main advantage is that it more closely resembles the way humans perceive color. **Hue** is color, depicted on a circle. Hue is therefore measured in degrees. **Saturation** is how gray the color is, when saturation is low the color is somewhat gray when saturation is high it is bright. Saturation has a value on the interval from 0 to 1. **Value** refers to the brightness of the color. Where 0 is black and 1 is white. In order to clarify this model, it is depicted in figure 8.

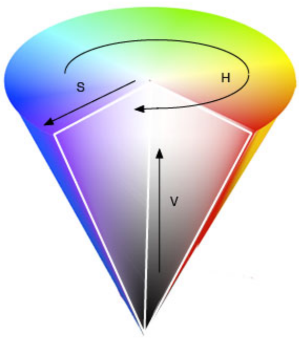

Figure 8: HSV Color Model (taken from: http://www.medicalmac.com/)

For the assignment we used the matlab function *rgb2hsv*. We than plotted the Hue, Saturation and Value channels individually. The results are shown in figure 9

Figure 9: HSV Color space translation results

References

N. Petrovic, I. Cohen, B. J. Frey, R. Koetter, and T. S. Huang. Enforcing integrability for surface reconstruction algorithms using belief propagation in graphical models. In *In: Proc. Conf. Computer Vision and Pattern Recognition*, pages 743–748, 2001.