Week 2 - Computer Vision

Tushar Nimbhorkar 11394110
Diede Rusticus 10909486

February 2017

Abstract

This report covers assignment 2 covering Photometric Stereo and Color Spaces.

1 Photometric Stereo

The task was to implement the photometric stereo algorithm as described in Algorithm 5.1 of Chapter 5 of the book "Computer Vision: A Modern Approach". Figure 1 shows the results for the Albedo, Integrability, Height-map and the Surface Normals.

- Load image (Provided)
  5 images of a sphere with different illuminations coming from 5 directions (front, bottom-left, top-left, bottom-right, top-right) are loaded.

- Matrix $\mathcal{V}$
  We determined the matrix V from the 5 light sources, by giving the unit vectors:

$$MatrixV = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 1 & -1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

  We normalise the matrix V into $\mathcal{V}$. We scale $\mathcal{V}$ with the constant k to connect the camera response to the input radiance.

- Create arrays matrix
  We create an i vector for each image point and construct the diagonal matrix $\mathcal{I}$ from that.

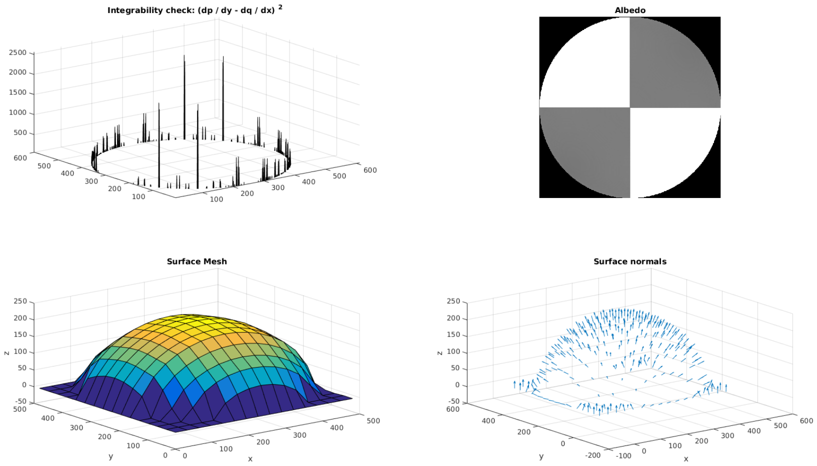
- Obtain g
  We multiplied $\mathcal{I}$ by $\mathcal{V}$ and took the pseudo-inverse. This, we multiplied by $\mathcal{I}i$ to get the value for g. Albedo at this image point is the norm of g. And the normal of this image point is $\frac{g}{|g|}$. The measured value of $p = \frac{df}{dx}$ is equal to $N1/N3$. The measured value of $q = \frac{df}{dy}$ is equal to $N2/N3$.

- Integrability
  To check if the surface gradient is acceptable, we take the second derivatives of p w.r.t. y and of q w.r.t. x.

- Height-map
  In order to construct the height map of the surface, we calculated first the most-left column to initialise the map:
  height value = previous height value + corresponding q value

  For the remaining columns:
  height value = previous height value + corresponding p value

Figure 1: Photometric Results



2 Color Spaces

For this task we converted an RGB-image into the (1) Opponent, (2) Normalized RGB, (3) HSV, (4) YCbCr Color Space and 4 types of Grayscale: lightness method, average method, luminosity method and matlab's build-in function for grayscale. It turned out that the build-in method is using the luminosity method, so those results were the same. As an example of our results, the four different Grayscale results and the scales of HSV Color Space are shown in Figure 2.

Figure 2: Left: Results for the Grayscales (top-bottom: lightness method, average method, luminosity method and matlab's build-in function for grayscale). Right: Results for the HSV Color Space (top-bottom: Hue, Saturation, Value, HSV combined in RGB space)