**ELG 5163 – Machine Vision**

**Assignment 4**

**RGB-D RANGE SENSOR SIMULATOR**

**Name: Yusri Al-Sanaani**

**Student Number: 300216450**

Department of Electrical Engineering

Faculty of Engineering

University of Ottawa

**April 7, 2022**

# **Part I (Output type 1 - Textured point cloud)**

The virtual plane and the corresponding HSV color space were created according to the given coordinates in assignment 1 coded as multidimensional array of 2000 ×2000×6. So, the resulted virtual plane is 2000 ×2000×3 corresponding to (x, y, z). Also, the HSV color space is 2000 ×2000×3 corresponding to (H, S, V) which is converted into RGB space as well.

After obtaining the virtual plane, the second step is to transform the virtual plane coordinates into image plane according to Eq. 1. So, the textured point cloud was obtained by mapping these values with their corresponding RGB points as shown in Figure 1 that shows the 3D representation of our virtual plane in different views: Figure 1 (a) shows the 3D representation of textured point cloud, Figure 1 (b) shows the top view, and Figure 1 (c) shows the front view. As shown, the top view clearly displays the range of the depth coordinates, along Zcam.

|  |  |
| --- | --- |
|  | Eq. 1 |

The generated 3D point cloud is distributed in 3D model corresponding to actual shape of virtual plan. The color and (x, y) coordinates of the 3D point cloud remain same after transformation resulted with perfect representation for the virtual plane. The z coordinate values have been changed according to transformation matrix resulted with new range scale of [900,1100]. Therefore, I can confirm that undulated surface is distributed in the 3D model, the surface corresponds to the specifications of the virtual scene, and the points in the point cloud are properly textured.

# **Part II (Output type 2 - Cartesian depth map)**

According to the information and description for this part in the assignment documentation, the cartesian depth map has been obtained as follows:

**Step1:** the perspective projection can be represented as in Eq. 2:

|  |  |
| --- | --- |
|  | Eq. 2 |

perspective projection matrix P is given as in assignment 1:

**Step 2:** the projection operation introduces some distortion in the world representation that appears as a scaling factor. So, the perspective projection equations can be calculated using Eq. 3:

|  |  |
| --- | --- |
|  | Eq. 3 |

**Step 3:** calculating the pixel coordinates. Calculate the pixel coordinates of every projection over the pixel map by considering the image plane physical dimensions and the number of pixels per row and column in the output image. The size of image plane is (12.7 ×12.7) mm and the size of photosensitive plane is 480 × 640. So, the image width = image height = 12.7. The row pixel = 480 and the column pixel = 640. Since the resolution = physical size / pixel dimensions, the raw resolution = image width/row pixel whereas the column resolution = image height/column pixel. The pixel coordinates can be calculated using Eq. 4.

|  |  |
| --- | --- |
|  | Eq. 4 |

**Step 4:** computing the cartesian distance for every projection and store that distance value in the corresponding pixel of the depth map. So, this step includes propagating the z depth points along with the (x, y) coordinates of every sample point by initializing the background of the output image map with black and then retrieving the transformed coordination in image plane range: Columns between 0 and 641 and rows between 0 and 480.

**Step 5:** normalizing the resulting cartesian depth map with respect to the maximum depth measured.

The generated depth map image represents the front view of our virtual scene as shown in Figure 2. So, we can see the crests and troughs of the scene according to the gray level variation. The crest is darker corresponding to decreasing in depth where trough is lighter gray corresponding to increasing in depth.

# **Part III (Output type 3 - Radial depth map)**

According to the information and description for this part in the assignment documentation, the radial depth map has been obtained in the same manner as done in part 2 except for step 4 since we need to obtain the radial depth measurements corresponding to a specific pair of azimuth and elevation angles and store them in the depth map, where every pixel contains a radial distance instead of the Cartesian Z coordinate of a point with respect to the sensor’s reference frame. This can be obtained by transforming the cartesian coordinates into spherical (r, θ, ϕ) coordinates according to cartesian (x, y, z) to spherical transformation equations as in Eq. 5.

|  |  |
| --- | --- |
|  | Eq. 5 |

So, the depth for every 3D point over the virtual undulated surface point cloud was recalculated to estimate the radial distance measurement along the propagation line of the beam of energy emitted by an active range sensor. After calculating the radial depth r, and then mapping its values into depth map, the radial depth map obtained and then normalized as done in part 2.

The generated radial depth map image represents the front view of our virtual scene as shown in Figure 3. So, we can see the crests and troughs of the scene according to the gray level variation. The crest is darker corresponding to decreasing in depth where trough is lighter gray corresponding to increasing in depth. This is also noticed when inspecting the z depth values and their corresponding pixels in depth map.

For both cartesian depth map and radial depth map, the generated maps contain gray tones that vary in accordance with the measured cartesian and radial depth, where dark gray corresponds to surfaces closer to the range sensor (crest of sinusoidal surface) and light gray corresponds to surfaces farther away from the sensor (trough of sinusoidal surface). However, there are variations in the gray shading between the Cartesian and radial representations.

In radial depth map, the gray tones distributed over the surface according to the depth. So, the gray level distribution along the vertical axis looks equal, but inversely proportional to the depth when moving horizontally – resulted with gray level changes (the shorter depth, the darker gray and the longer depth, the lighter gray). This equal distribution along vertical axis and different in the horizontal axis forms a smooth vertical stripes seen clearly when zooming in over the image. This is expected since the cartesian depth was estimated based on distance measured parallel to the sensor’s optical axis (Zcam) with respect to the depth reference plane whose surface is parallel and perfectly aligned with the origin of the sensor’s image plane, that is Zcam = 0.

In radial depth map, the gray tones distributed over the surface according to the radial depth. So, the distribution along the vertical and horizontal axes looks not equal, but still inversely proportional to the depth when moving horizontally (the shorter depth, the darker gray and the longer depth, the lighter gray). This gray level changes in the vertical and horizontal axes forms a smooth circular stripes seen clearly when zooming in over the image. This is also expected since the radial depth estimated based on the measured distance from the projection center that changes according to the change in azimuth and elevation angles. Also, there is a difference in gray intensity values, the radial depth map has higher values compared to cartesian depth map, so the radial depth map looks darker than cartesian depth map.

Based on this analysis, I can confirm that the undulated surface shape is properly encoded in the depth maps. I came up with this comparison and conclusion based on my visual inspection for resulted images and examining the distribution of numerical values in the output (z,r) depth values/depth maps as well, which confirms the visual observation.

If additional object – such as planar surface with smaller area than our virtual plane – inserted in between the undulated surface and the range sensor, it will block the emitted energy of the sensor from reaching the undulated surface, this will affect the observability of the scene in both the textured point cloud and depth maps. In case of textured point cloud, it may capture both the undulated and planar surfaces, but the planar surface will appear as extrude (solid or surface) that would cause a cut through the undulated surface proportional to the size of planar surface. In case of depth maps, the planar surface will appear as darker area – proportional to its size – since it is close to the sensor. So, it will appear as if it is a solid or surface extrusion on the undulated surface. Figure 4 may help to deliver my point, Figure 4 (a) in case of point cloud case and Figure 4 (b) in case of depth maps case.

Based on my observation, the three representations are not equivalent in terms of knowledge about the scene, but we can get comparable knowledge regarding the depth in different ways. I think the textured point cloud provides a good representation since it captures the 3D structure of the undulated surface and reserves the RGB color map as well. The depth maps methods only provide depth information as mapped in gray levels. However, I think the cartesian depth map provides better depth information (as shown in Figure 2) compared to radial depth map (as shown in Figure 3). In term of computing, the depth map method takes more time compared with textured point cloud because in depth method we need to calculate the Perspective Projection and mapping the depth which takes more time. According to the hardware specifications of my laptop, the elapsed time for textured point cloud is around 13 seconds and for depth map is around 18 seconds.

I think the textured point cloud allow to rebuild the full 3D shape of objects, and reconstruct the location of all surfaces, so we could see them from any viewpoint around the object. This requires having all 3D information. Also, the thickness of 3D object may provide better representation toward 3D reconstruction.

The simulations done in this assignment are performed in ideal environments such that we have only the virtual plane, no thing else in the scene. In addition, the virtual plane does not have a thickness. So, the simulation results with good representations for 3D structure and depth. In the real RGB-D sensor, the output representations face some issues and artifacts such as holes, edge erosion, presence of other objects in the scene, etc.

# **Appendix**

|  |
| --- |
|  |
| **a)** |
|  |
| **b) Top View** |
|  |
| **c) Front View** |
| Figure 1: Output type 1 - Textured point cloud from different views. |

|  |
| --- |
| **Output\_Type\_2\_Cartesian\_Depth\_Map** |
|  |
| Figure 2: Output Type 2 Cartesian Depth Map. |

|  |
| --- |
| **Output\_Type\_3\_Radial\_Depth\_Map** |
|  |
| Figure 3: Output Type 3 Radial Depth Map |

|  |
| --- |
|  |
| 1. **Textured Point Cloud** |
|  |
| 1. **Depth Map Case** |
| Figure 4: Inserting a planar surface in between the undulated surface and sensor. |

The radial depth map generated in part 3 is done by recalculating the depth for every 3D point over the virtual undulated surface point cloud as stated in the assignment description and based on my understanding as well. So, the depth map has been calculated after obtaining the point cloud points by multiplying P\*Q by the virtual plane points. If we consider calculating the radial depth for every 3D point over the virtual undulated surface point cloud (before multiplying by P), I got radial depth map that almost has similar characteristics as the previous case, but it is darker as shown in Figure 5. There is no effect on the result of part 2 since the results are same in both cases.

|  |
| --- |
| **Output\_Type\_3\_Radial\_Depth\_Map** |
|  |
| Figure 5: Output Type 3 Radial Depth Map – depth without P. |