

Lab 8 - Region Segmentation

John Lawler

November 20, 2020

FUNCTIONALITY

This lab takes as input a 128 by 128 range image of a chair and thresholds a specified area in the image to segment with varying grayscale values. The image is first **thresholded at a value of 130** to select the floor and chair only, where anything above the value is turned to white and pixels of interest are set to black. Once thresholded, the 3D coordinates are calculated for each pixel in the image. These coordinates are used to determine the orientation of each “surface” in the image. They are used to calculate the surface normals on every given pixel. Once the 3D coordinates are found, the cross product is performed on every pixel’s 3D coordinates. Three sets of coordinates are needed; these pixels are selected (including the current pixel of interest) for each respective pixel calculation. Given a pixel of interest at column C and row R, the pixels 3 columns to the right (vector A) and 3 rows down (vector B) are used as the other two 3D coordinates for calculation (i.e. **pixel distance chosen was 3**). The cross product is calculated by subtracting the current pixel of interest from vectors A and B, and then performing the process on the two resultant vectors. The output of the cross product gives 3 new coordinates, which describe the orientation of the currently selected pixel with respect to the other two pixels’ coordinates. The resultant vector is the surface normal, and these are used for later determining which surfaces to grow a region in.

Once surface normals are calculated, the range image is iterated through for finding appropriate regions to segment. A 5 by 5 window is used for finding new regions to seed, and if every pixel in the 5 by 5 window shows as black in the thresholded image and is not currently in a region defined previously, then the pixel in the middle of the window is selected to start a new region. A count of verified pixels is used to keep track of each window, and if the count reaches 25, then the area will start seeding a new region and a region function is called. Using the queue C code provided for Lab 4 as a rough template, the region grow function from the code is modified to fit the parameters of this lab. The thresholded image, surface normals, and region are all used within the region growing function. The average of each surface normal is recalculated every time a new pixel is found to join the region, and this running average is what describes the average orientation throughout each region during growth. The predicate that follows for whether or not a pixel can join the region is determined by the angular difference between the current region's average orientation and the pixel of interest’s current orientation. To find the **angular difference** of two given vectors in 3D space, the dot product is found and divided by the multiple of the distance of both of the vectors (the current running surface normal average and the selected pixel). The arccosine is performed on the resultant value, and this value is thresholded to check whether or not the pixel can join. The value used for thresholding here was

a **value of 0.5**. The output table shown in Figure 1 represents the found regions in the image. The total pixel count on the left hand side, with the average surface normal for each X, Y, and Z shown to the right for each respective region. Figure 2 shows the range image segmented, with each region shaded as a different grayscale value.

DATA

# Pixels	Avg Sfnm:	X	Y	Z
65		-0.306071	-23.820669	7.739450
799		12.916608	0.863316	4.693754
470		-2.601533	2.322802	4.476250
5285		0.539222	-9.582164	3.060952
258		0.848150	-8.463701	2.420032
609		-36.930680	1.886817	12.536191

(base) lawler6@DESKTOP-66UA8JA:~/git/clemson/4310-vision/labs/8\$

Figure 1: Table of pixel counts and average surface normals

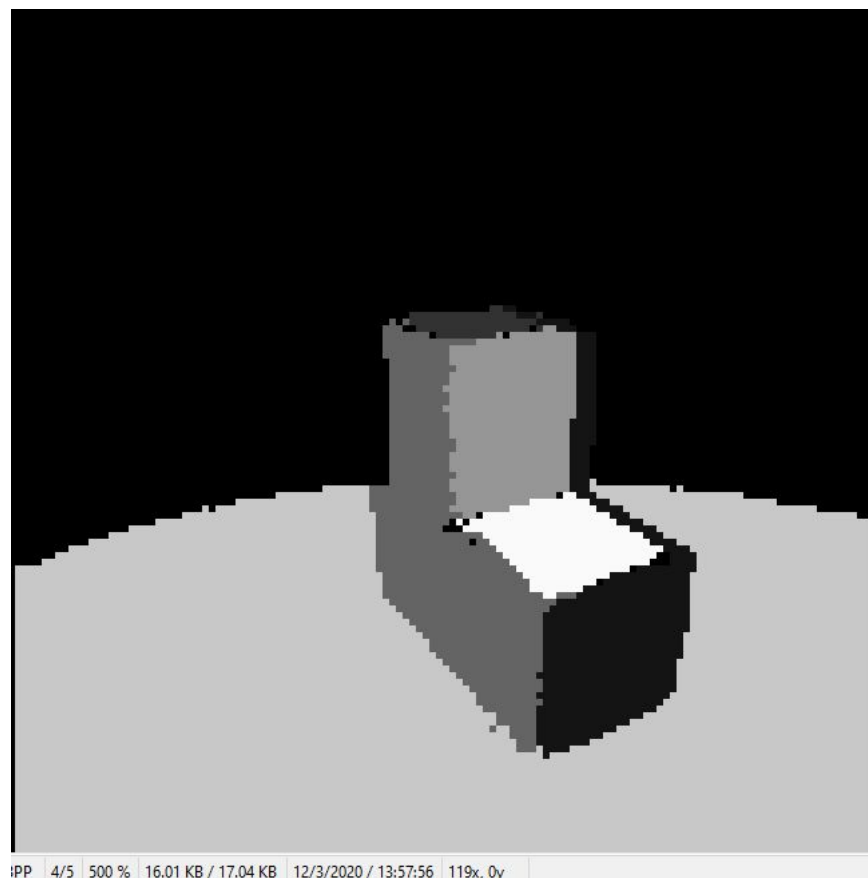


Figure 2: Image with region segmentation