Brandon Shumin
10/27/2020
ECE 6310
Lab 5

Reading Input Data and Input Image:

Contour points are read into 2 arrays, one for the row-coordinates and one for the column-coordinates from a .txt file. Then the image file is read in as a .ppm file and saved to an unsigned char pointer. By saving the contour points in 2 distinct arrays, the row and column coordinate for each given contour point can be easily accessed for any given contour point. The file read-ins function essentially the same in this lab as the methods I used in Lab 2 and Lab 3.

Plot Initial Contour Points:

The contour points read in on the previous step are now plotted onto the original image read in for the purposes of comparing the results of the contouring. The points are plotted by iterating through the number of points read in and then printing a 7x7 cross centered on the contour point location. The image generated from this is shown below in Figure 1.

Determine Sobel Gradient Image:

Determining the Sobel Gradient requires performing a convolution of the initial image on a Sobel matrix. I decided on a Sobel matrix of size 7x7 to conform to the window sizes for the contour points and internal energies used later in the contouring of this program. It should be noted that a smaller Sobel matrix could increase the detail on some edge detections, but I still decided to stick with a 7x7 matrix for consistency with other window sizes. The Sobel gradient is first determined in the x-direction, or left to right, and then in the y-direction, or top to bottom. These two gradients are then squared, added together, and then square rooted, providing a complete Sobel gradient representing edges in all directions. This final gradient is then normalized into 8-bits using the min and max values of the complete Sobel gradient un-normalized. This image is then saved, as shown below in Figure 2.

Contour the Image / Determine Internal and External Energies:

The main step of this program is to process the contour points given into an evenly spaced set of points that should squeeze around the hawk in the image, or any edges detected with regard to the shape of the main image. To accomplish this, two internal energies are first determined.

The first internal energy is found by determining the squared distance from every point in the 7x7 area around a contour point to the next contour point, looping around to the first contour point if the final contour point is currently be assessed. Then, the min and max of this internal energy should be calculated to normalize the image between 0 and 1.

The second internal energy is found by determining the square of the distance between the average distance between all points and the distance between the current contour point and the next contour point. To find the average distance, the distance between each point and the point after it is found and then the average of all these distances is determined. Then, the min and max of this internal energy should be calculated to normalize the image between 0 and 1.

The external energy was found by taking the 7x7 area around the given contour point from the Sobel image generated in Figure 2 and taking it out of 8-bit normalization and then renormalizing it between 0

and 1. This final normalization was then inverted and the absolute value was applied to this inverted array to make edges have high energies and non-edges have low energies. This acts to push contour points to the edges without letting them cross the edge.

The final step is to add these three normalized arrays together and find the minimum value contained in the array and then move the current contour point to this area of lowest energy. This should rubber band the contour points around an object contained within an image. The current program runs for 30 iterations of the above process to contour the hawk in the image correctly. The final contoured image is shown below in Figure 3. A list of the final contour points in the terminal output of the program can be seen below in Figure 4.

Brandon Shumin
10/27/2020
ECE 6310
Lab 5

Figure 1: Original image with initial contour points plotted

Brandon Shumin
10/27/2020
ECE 6310
Lab 5

Figure 2: Complete Sobel gradient applied to the original image

Brandon Shumin
10/27/2020
ECE 6310
Lab 5

Figure 3: Final contour points plotted onto the original image

Brandon Shumin
10/27/2020
ECE 6310
Lab 5

```
bshumin@DESKTOP-2F1JAJ7:~/compvis/lab5$ ./lab5 hawk.ppm contour_pts.txt
+------FINAL CONTOUR POINTS------+
Point  1: 274 118
Point  2: 276 127
Point  3: 278 137
Point  4: 278 147
Point  5: 278 157
Point  6: 274 168
Point  7: 271 177
Point  8: 268 186
Point  9: 264 195
Point 10: 260 204
Point 11: 256 212
Point 12: 253 223
Point 13: 246 236
Point 14: 234 234
Point 15: 227 245
Point 16: 221 260
Point 17: 210 267
Point 18: 198 265
Point 19: 195 253
Point 20: 185 243
Point 21: 177 238
Point 22: 173 227
Point 23: 177 218
Point 24: 181 209
Point 25: 182 196
Point 26: 183 186
Point 27: 184 176
Point 28: 185 166
Point 29: 187 156
Point 30: 189 146
Point 31: 192 137
Point 32: 195 127
Point 33: 198 118
Point 34: 207 108
Point 35: 216 104
Point 36: 224 99
Point 37: 235 89
Point 38: 243 85
Point 39: 253 85
Point 40: 260 90
Point 41: 266 101
Point 42: 268 111
bshumin@DESKTOP-2F1JAJ7:~/compvis/lab5$
```

Figure 4: Terminal output of the program with listed final contour points