Graham Fuller & Jacob Oakes
February 3, 2015

Lab 7

Description

Our algorithm finds circles of arbitrary radii using Hough transforms. Since we implemented it using arbitrary radii the Hough parameter space for voting is 3D and hard to visualize. An example of a single pixel voting space can be seen below. Edges are found using Sobel filters. Our algorithm goes through each edge and votes for circles of radii from 10 to a maximum radius which is determined to be half of the minimum of the height and width. Next, we find the local maximums using the votes. We also use a threshold of halfway between the mean and max number of votes for every circle so we don't include local maxima that are irrelevant. The local maxima found are what we consider all the possible circles. Next we go through every possible circle and determine if it similar to others. If it is we condense these similar circles by taking the mean center and max radii. The results from this are what we use for our final output.

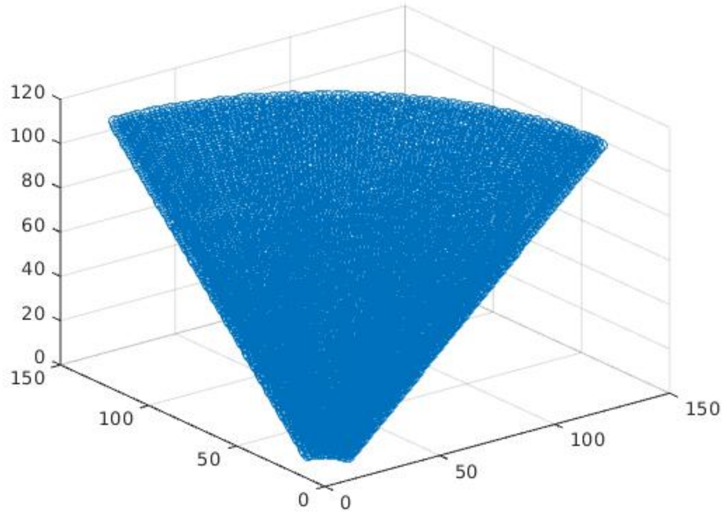

Figure 0: Example parameter space creating a cone for a single point

Results

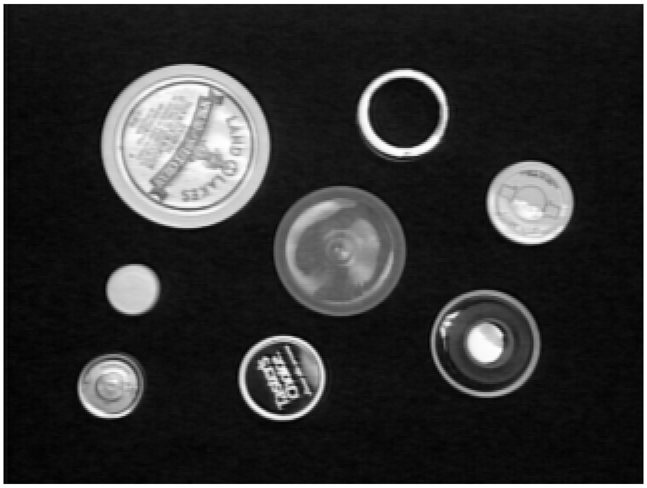

Figure 1: Original 1st Image

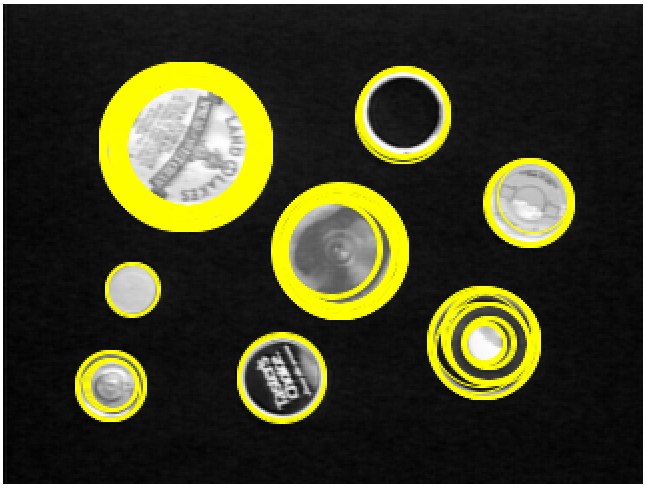

Figure 2: All circles that were local maxima and above threshold

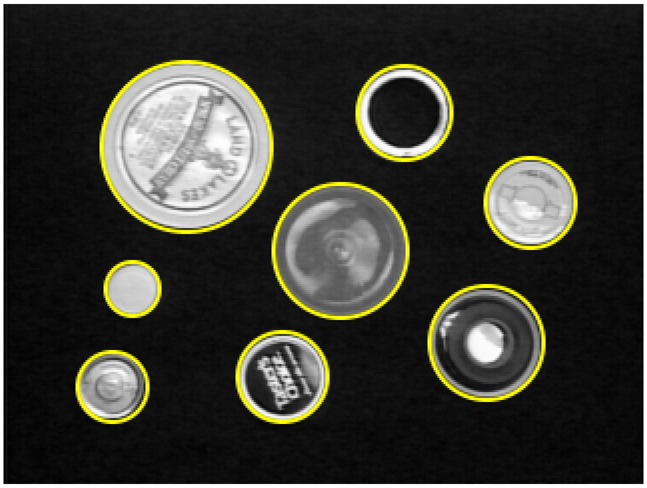

Figure 3: Final output

In this image the buttons are clearly distinguishable. There is no overlapping and there is a solid background. Our algorithm was able to easily separate the buttons and searched for a varying radius size to find all of the buttons. The circles created outline each button very well.

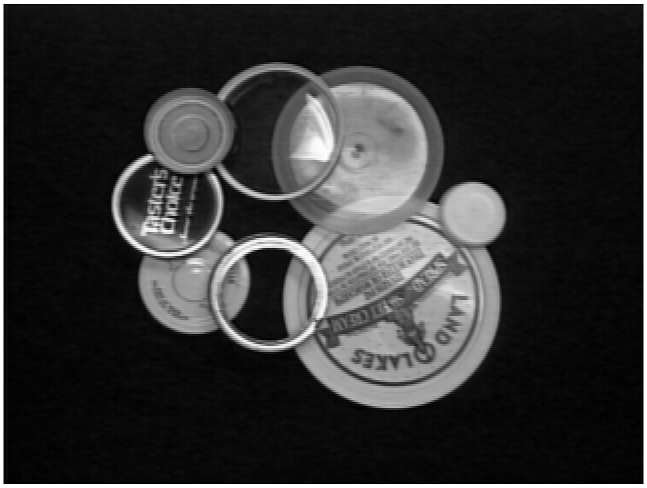

Figure 4: original 2nd image

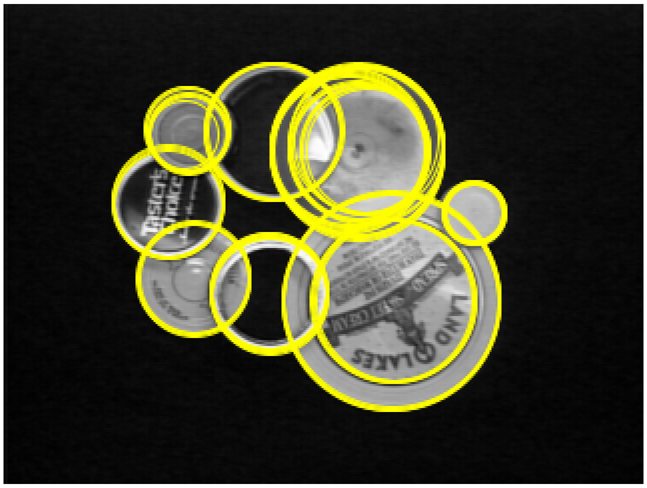

Figure 5: All circles that were local maxima and above threshold

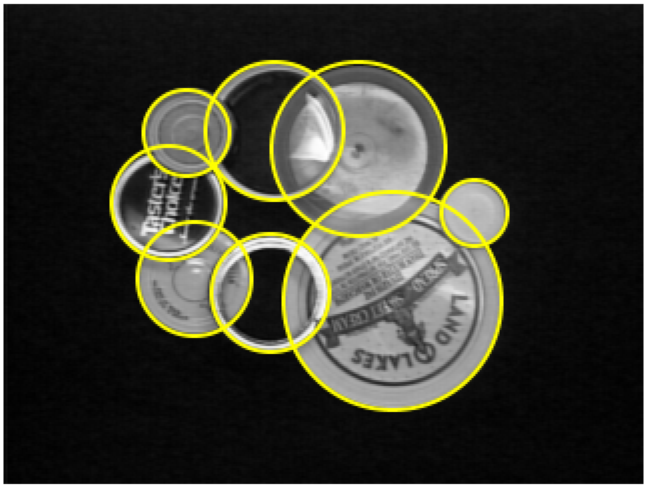

Figure 6: Final output

Although there were overlapping buttons in this image, our algorithm was still able to detect different edges of circles and use those unique edges to determine the center of each button. Our algorithm did a great job finding each button and encapsulating it with a yellow circle.

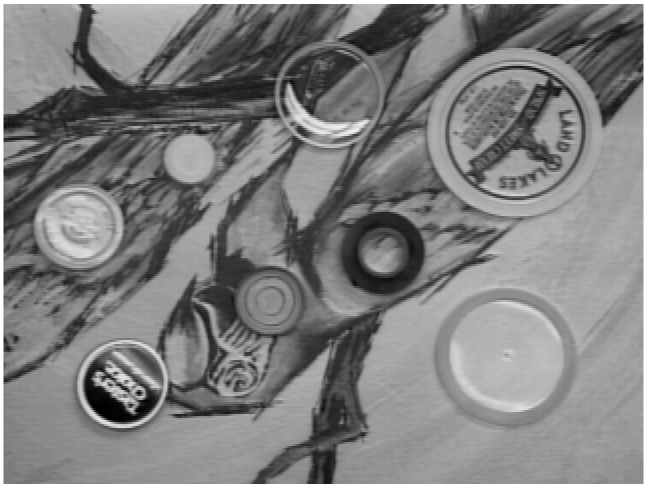

Figure 7: Original 3rd image

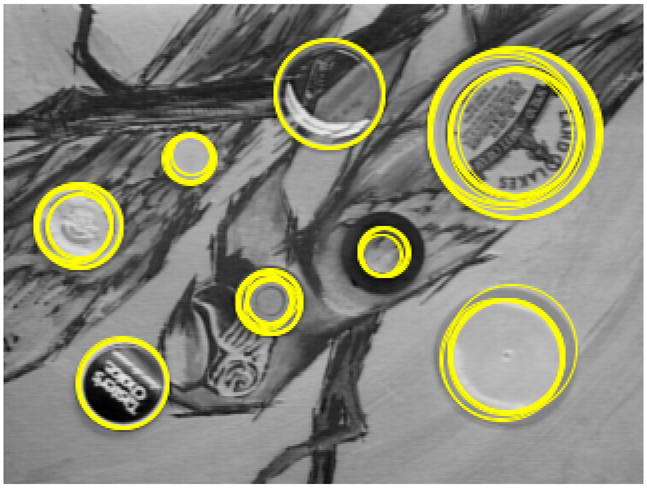

Figure 8: All circles that were local maxima and above threshold

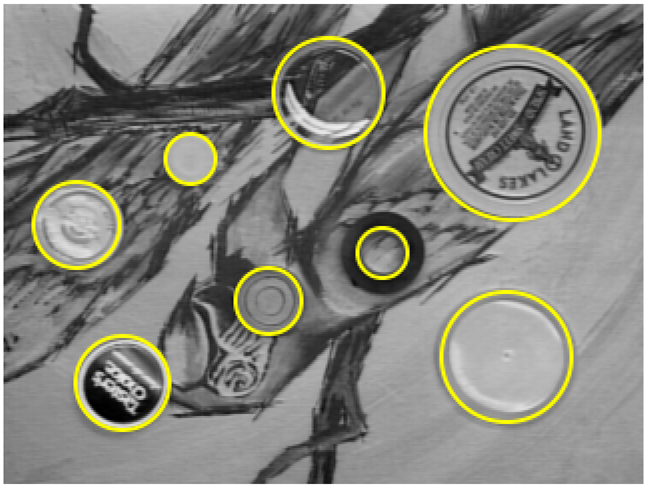

Figure 9: Final Output

In this image the buttons were more obstructed by a complicated background. Our algorithm was still able to locate each button on the complex background and find their centers. The one problem we ran into for this image was the dark button which only got its center identified and not its border. This could be due to the fact that the background is very similar color along the outside of the button.