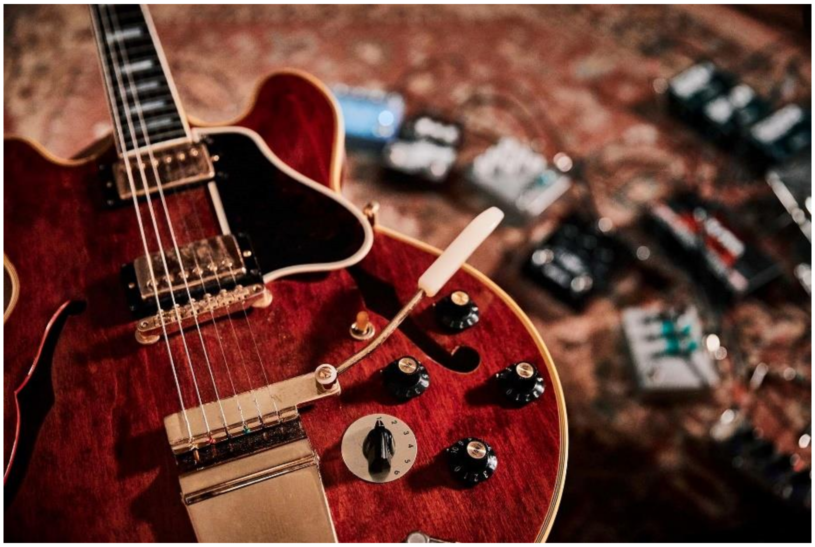
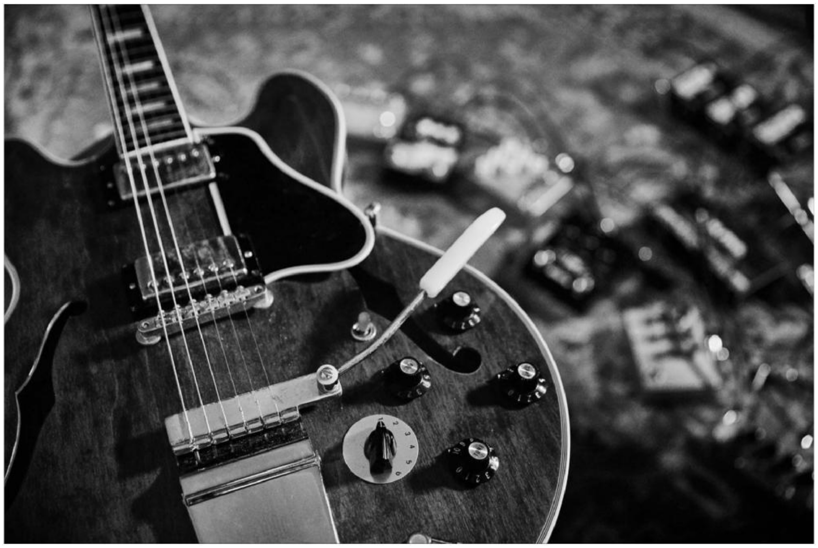
Gerom Pagaduan
CS390S
Homework 2
22 September, 2019

Homework 2 Report

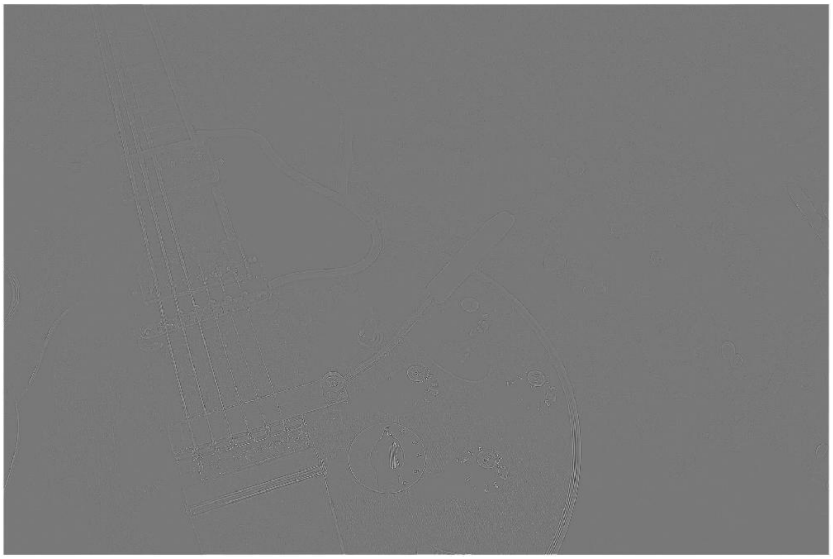
Input Image Output:



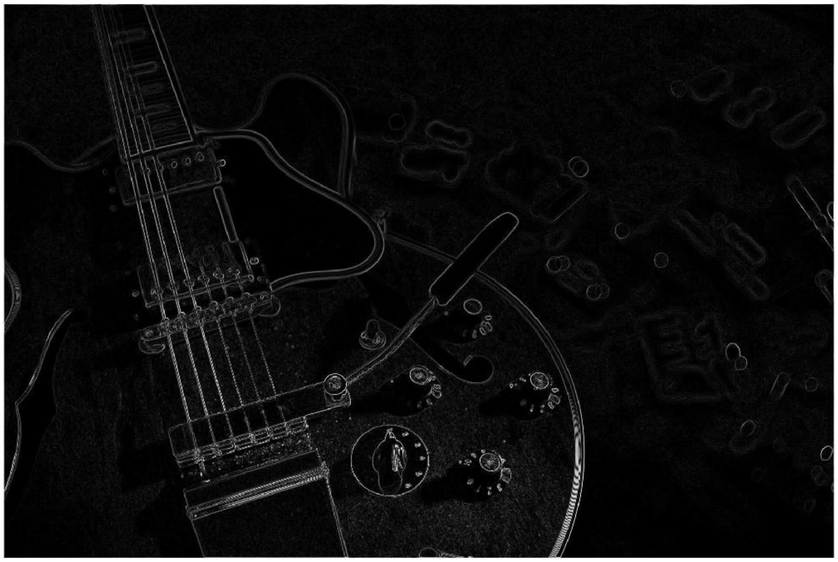
Average Filter Output:



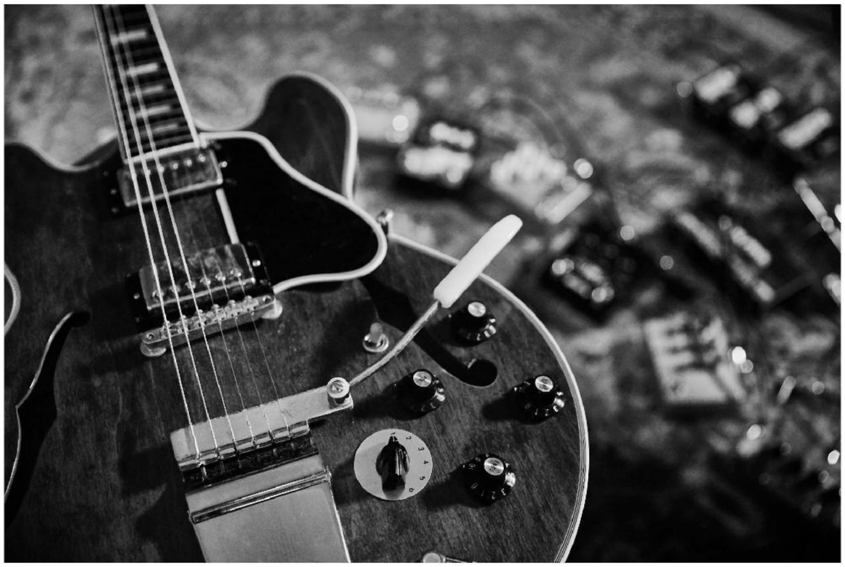
Laplacian Filter Output:



Sobel Filter Output:



Median Filter Output:



Code/Design Explanation

Average Filter: I created a 3x3 kernel of 1s, with a 9 in the middle and convolved it with the original image. This was the easiest filter to make.

Laplacian Filter: I created a 3x3 kernel of -1s, putting a -8 in the middle. This was based off of the in-class discussion about the Laplacian filter and I just transposed it into MATLAB's code.

Sobel Filter: This one was a challenging, but fun filter to make. I had trouble in figuring out how to apply my vertical and horizontal filter at the same time, even though I had my kernels set up correctly. I had to look online to discover that I needed these two lines to fix my code:

```
xResult = conv2(img, Gx, 'same');
yResult = conv2(img, Gy, 'same');
```

and after I included them, I just used the formula that was in the lecture's PowerPoint to get the output of the image.

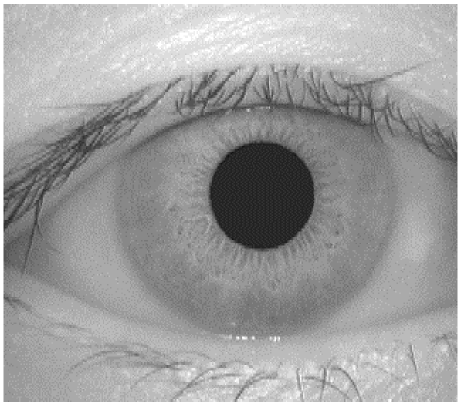
Median Filter: To me, this was the hardest filter to make. Even though I fully understood it conceptually, it was very hard for me to translate what I wanted into MATLAB. I looked online to see if there were examples of functional for-loops that could help me polish mine, but in the end, my classmate, Daniel Reuter, ended up assisting me on finalizing where to start and end my loops. I also had a hard time finding out how to properly pad the edges. However, once I got the filter to work, I was excited to see how much it emphasized the foreground compared to the background.

In short, the concept of this filter was to create a smaller 3x3 matrix for each pixel in the image, and changing its value to the median value of all the pixels in the matrix. To find the median, I created a 1-dimensional array and used index 5 since it's in the middle of all the values (i.e. the median).

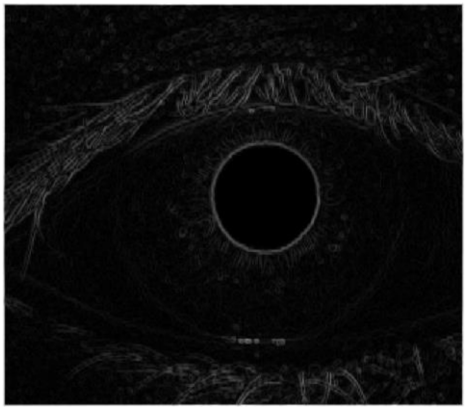
Pupil Detection

Although I've used my Sobel filter to detect the edges, I had difficulties properly masking the image and thresholding it to find the center of the pupil. Daniel Reuter and I worked together on finding the center of the pupil. We created a mask for the iris to make the center pupil an empty black, which we inverted to white. Then for the threshold, we made everything outside that circle black. Finally, using the distance of 45, we found the center of the circle, and made the difference from the center and the radius of 45 black, giving us the white circle.

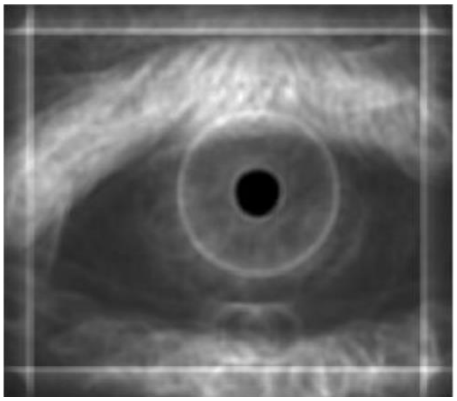
Input:



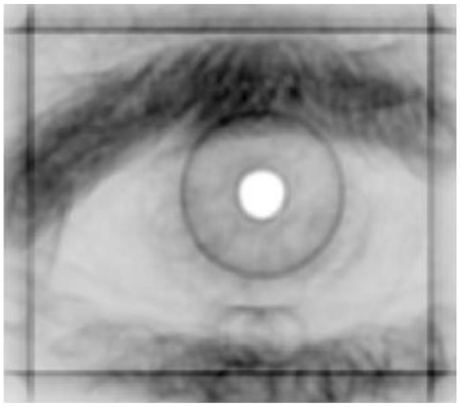
Sobel Filter Output:



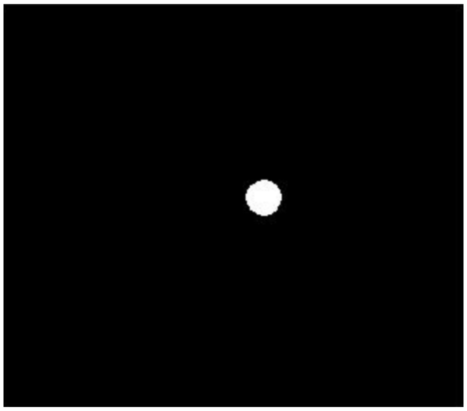
Iris Filter



Iris Filter (Inverse)



Thresholded Image



Thresholded Image when compared with Sobel image