Fatemeh Darbehani [redacted]

ENSC474 – SFU – Spring 2017

Assignment 8

Part1: Saturn Rings

- In this part I had to clean the below noisy image of Saturn rings (figure1).

Figure1: Noisy Image

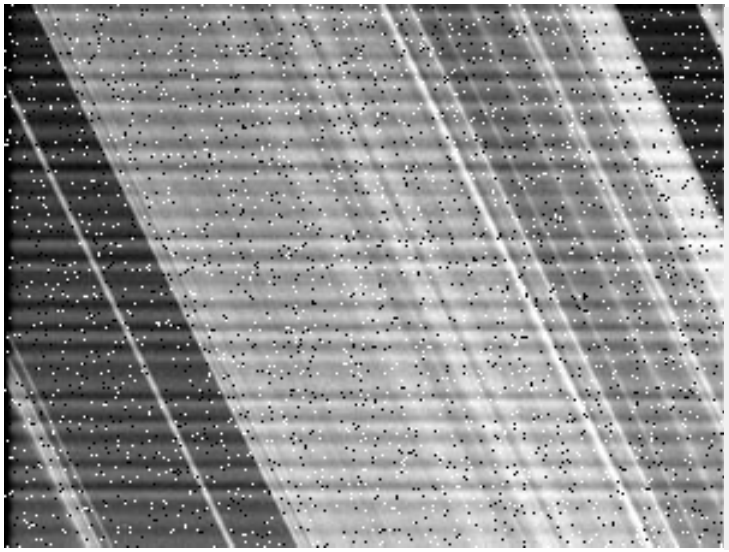
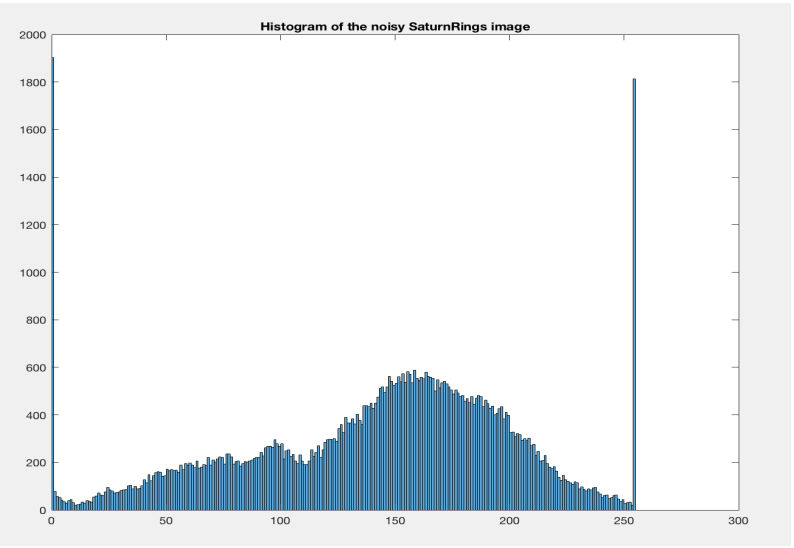


Figure2: Histogram of the noisy image



- By observation of the original image and by looking at its histogram, I could see the presence of Salt & Pepper noise.
  - Salt and pepper noise will cause high intensities at 0 and 255 in he histogram of the image as can be seen in figure 2

Removing Salt & Pepper Noise

- To remove the salt and pepper noise that was present in the image, I decided to use Median filter to reduce the effect of this noise. Median filters are a good choice for S&P noise because they provide effective noise reduction with less blurring compared to other linear smoothing filters such as 'mean' filters.

- I tried different sized Median filter to find the optimal size of the filter.

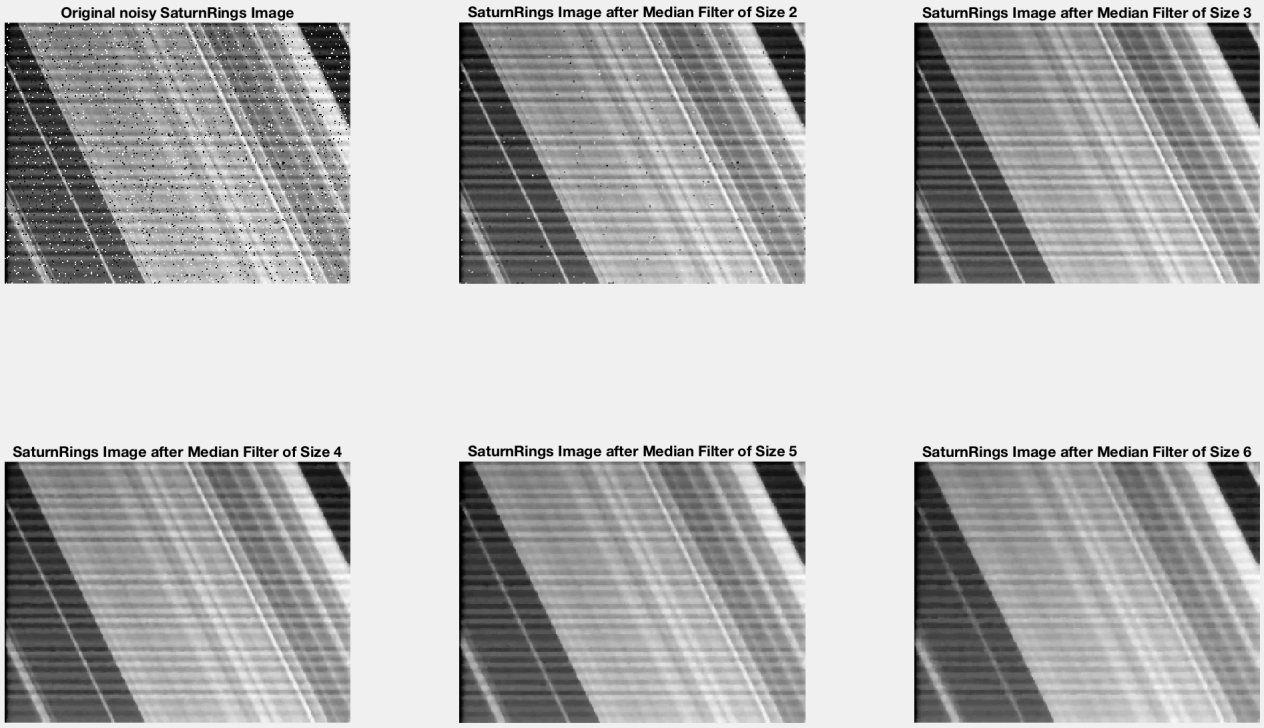


Figure3: Median Filtering the image with different filter sizes.

Removing Salt & Pepper Noise

- From the following figures we can see that after applying a 2x2 Median filter to the noisy image, there is still some noise found in the image. 3x3 Median filter was able to remove the noise better and there is no evidence of S&P noise in the image anymore. 4x4 Median filter removed the noise completely, however, it also made the image blurry. I observed that as the size of the filter increase, the image became more blurry and edges became less sharp.

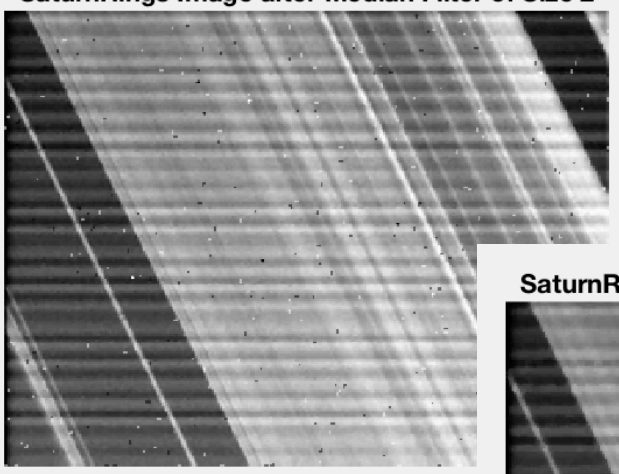


Figure4: Image After Applying 2x2 Median Filter

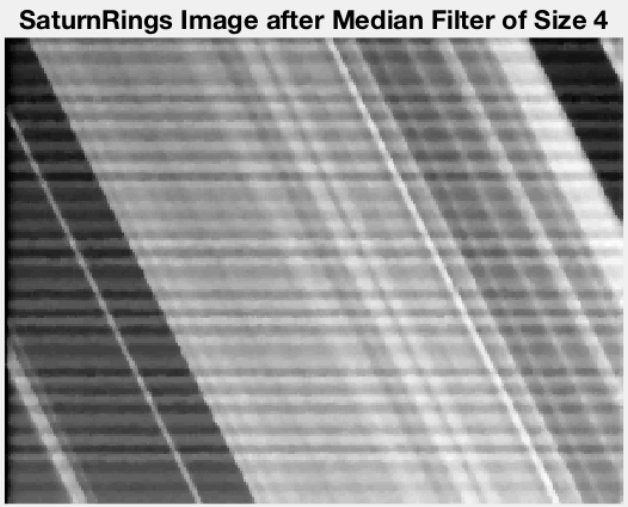


Figure5: Image After Applying 3x3 Median Filter

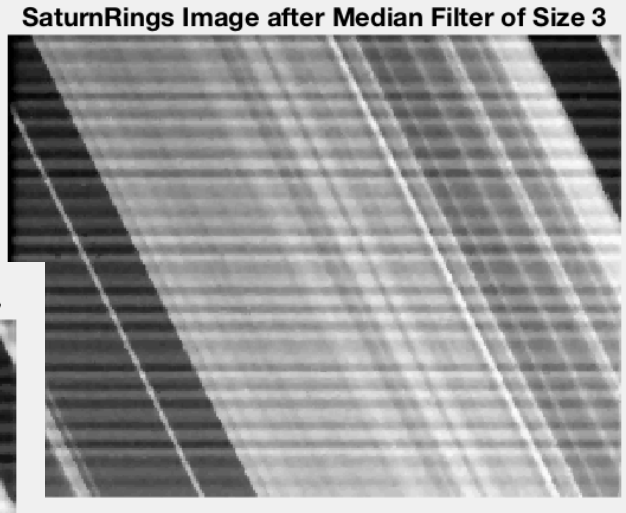
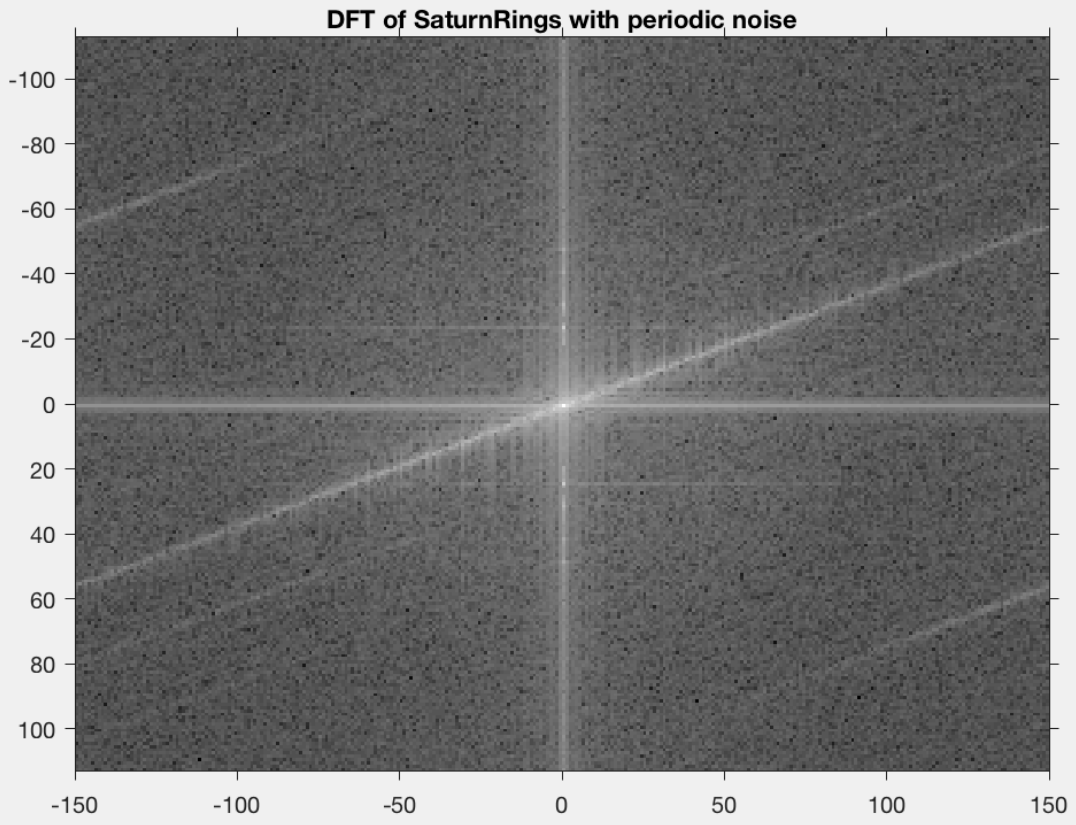


Figure6: Image After Applying 2x2 Median Filter

Removing Periodic Noise

- To remove this noise, I found the DFT of the image using MATLAB's inbuilt function fft2. Periodic noise is easier to filter in the frequency domain because it shows up as spikes at each frequency of the noise (bright values) in the frequency domain. I could see that periodic noise is along the horizontal axis of the image in spatial domain so I expected to see bright peaks along the vertical axis and in the center of the image in frequency domain.



- As expected, there are bright peaks along the vertical axis of the DFT of the image. Note that the diagonal spikes are expected and due to diagonal lines in the image itself.

- To remove these peaks I created rectangular notch filters with different widths in order to find the optimal filter. I decided to only remove frequencies below 60 along the vertical axis as the peaks are more obvious for those frequencies.

Figure7: DFT of the Saturn Rings Image

Removing Periodic Noise

- I created rectangular filters with widths 2, 6, and 10 and applied them to my image. Note that by filtering out frequencies in the frequency domain, we are removing the contribution of those frequencies to the image and we might even remove information along with the noise. Therefore, we should remove as small frequencies as possible.

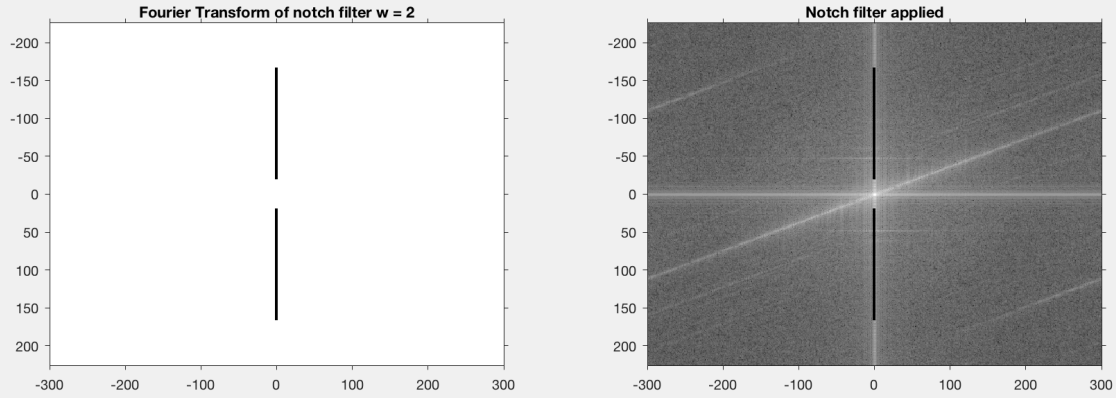


Figure8: Rectangular Notch filter with width of 2

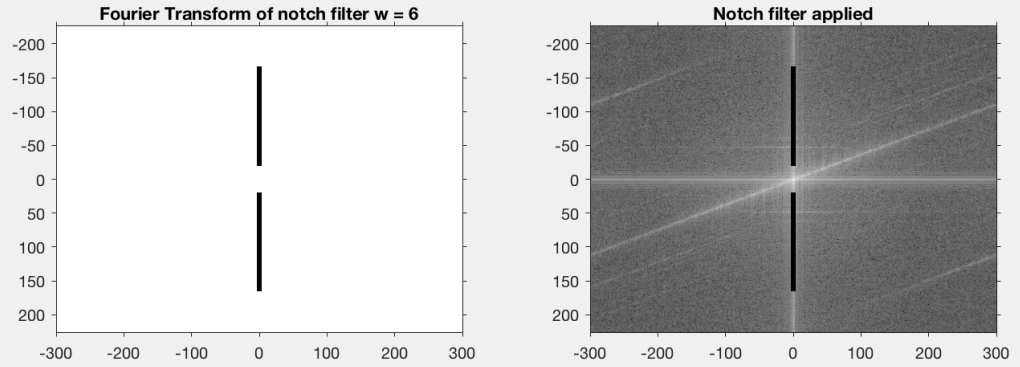
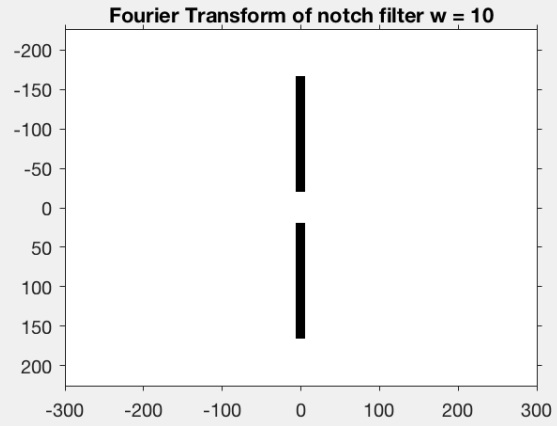


Figure9: Rectangular Notch filter with width of 6

Gaussian Noise



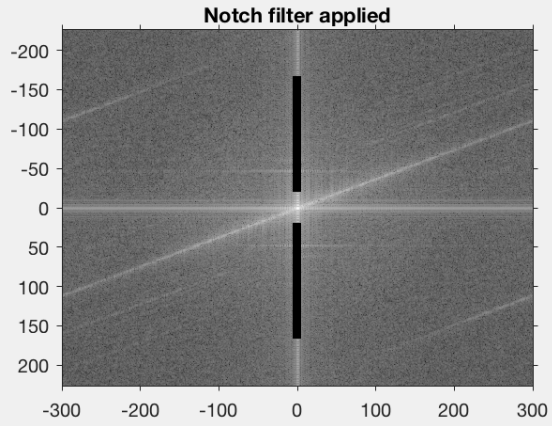
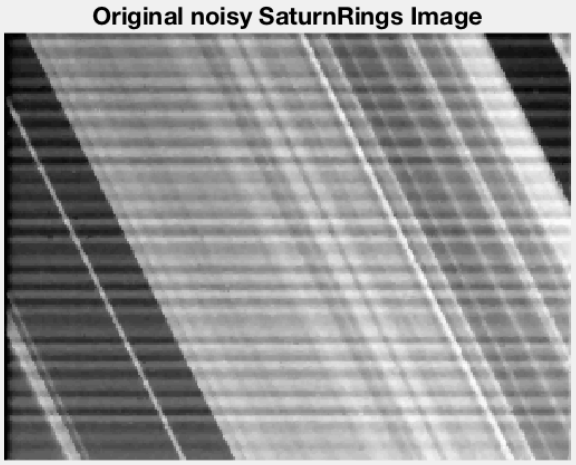


Figure10: Rectangular Notch filter with width of 10

- By using a wider notch filter we are removing more information from the image. Therefore, the image would be more blurry (Since we are removing higher frequencies).



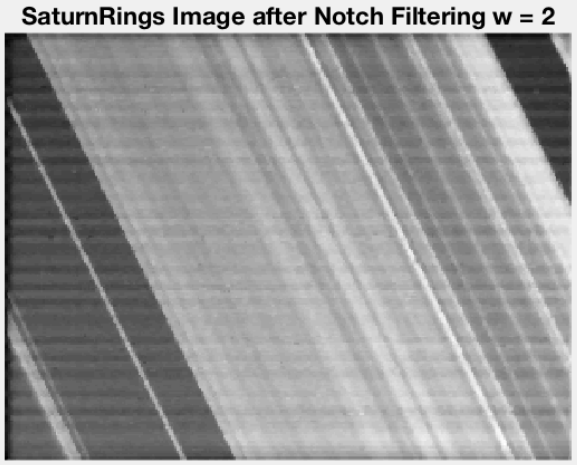


Figure11: Result of Applying Rectangular Notch filter with width of 2

Salt & Pepper Noise

- Notch filter with width of 2 couldn't sufficiently remove the periodic noise and notch filter with width of 10 removed most of the noise but made the image look blurry. So, I decided to use a notch filter with width of 6 to clean the image.

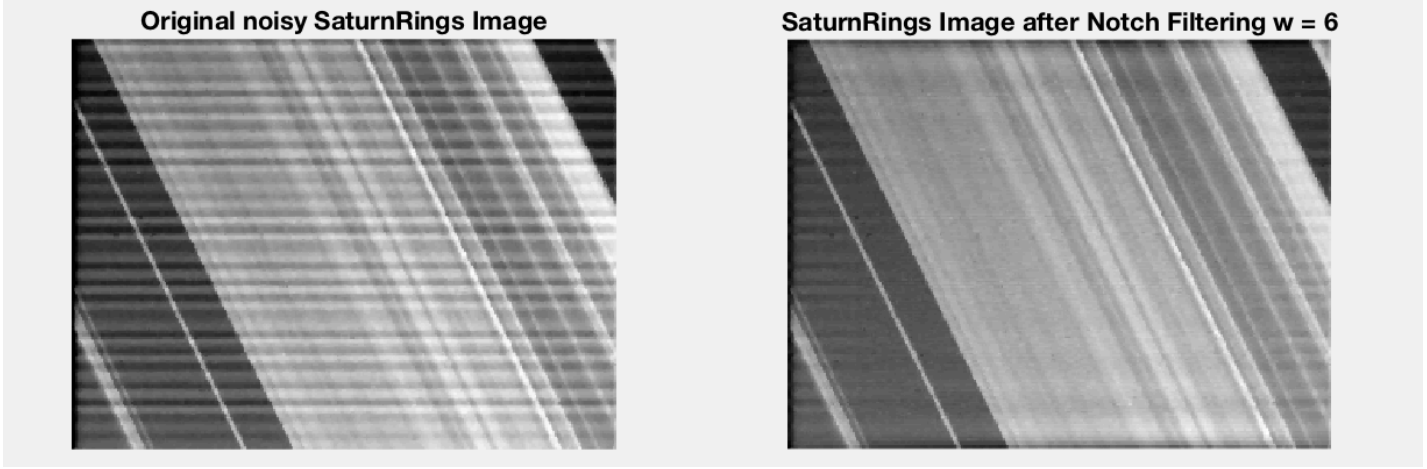


Figure12: Result of Applying Rectangular Notch filter with width of 6.

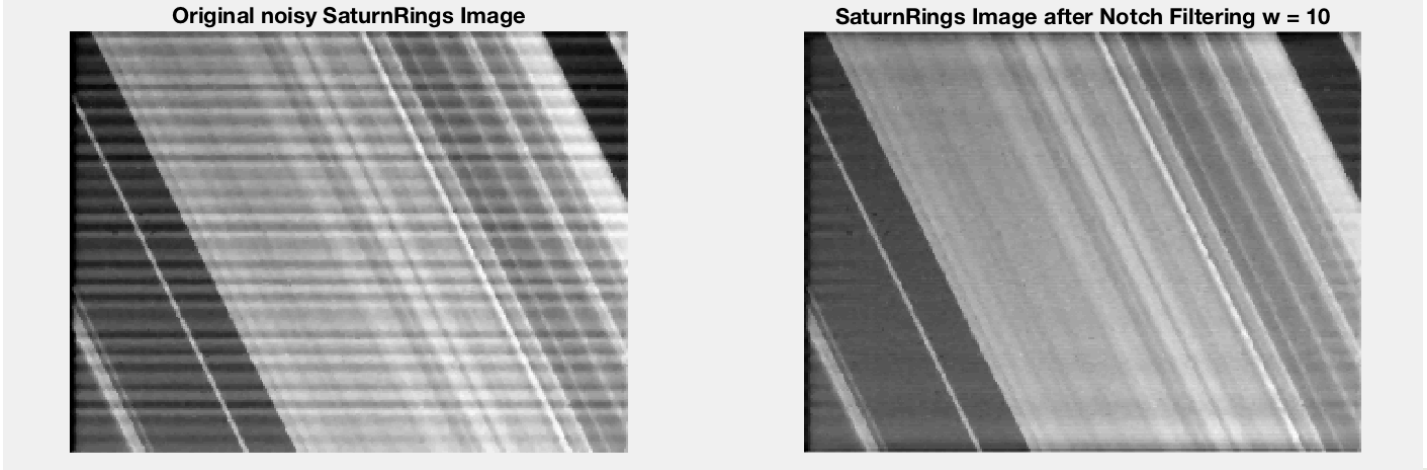


Figure13: Result of Applying Rectangular Notch filter with width of 10.

MATLAB Camera + Noise

- The thicker and lighter horizontal noise left in the image is due to lower frequency components so I increased the length of my notch filter to remove those as well.

Figure14: Notch filter closer to the center to remove lower frequency noise.

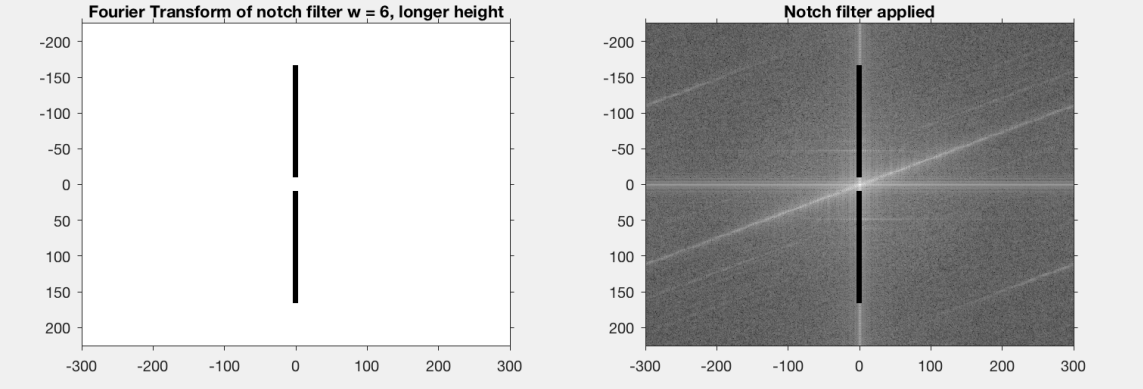
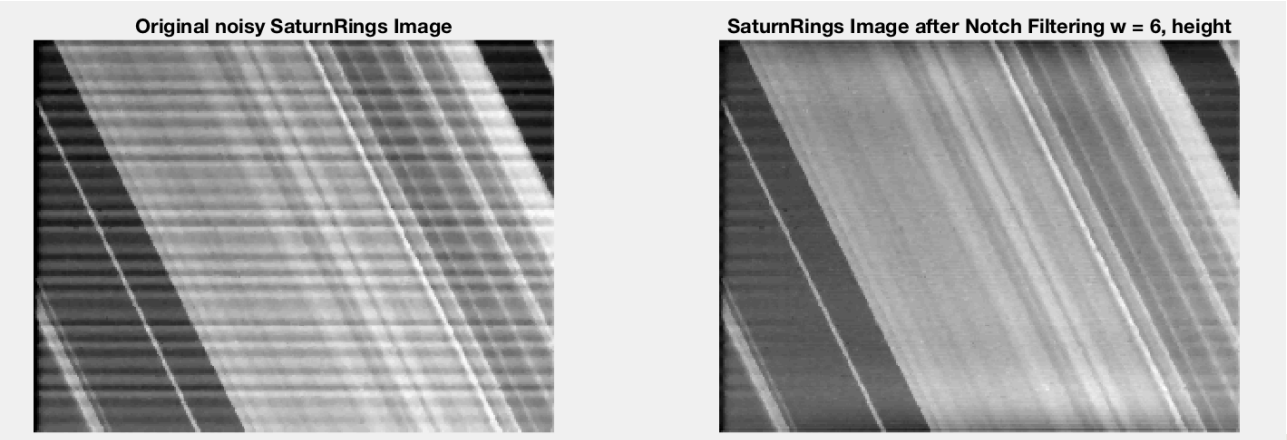


Figure15: Final clean image vs. the original image



Part2: Registration

- To register the rotated image with the original image I used the following landmarks.

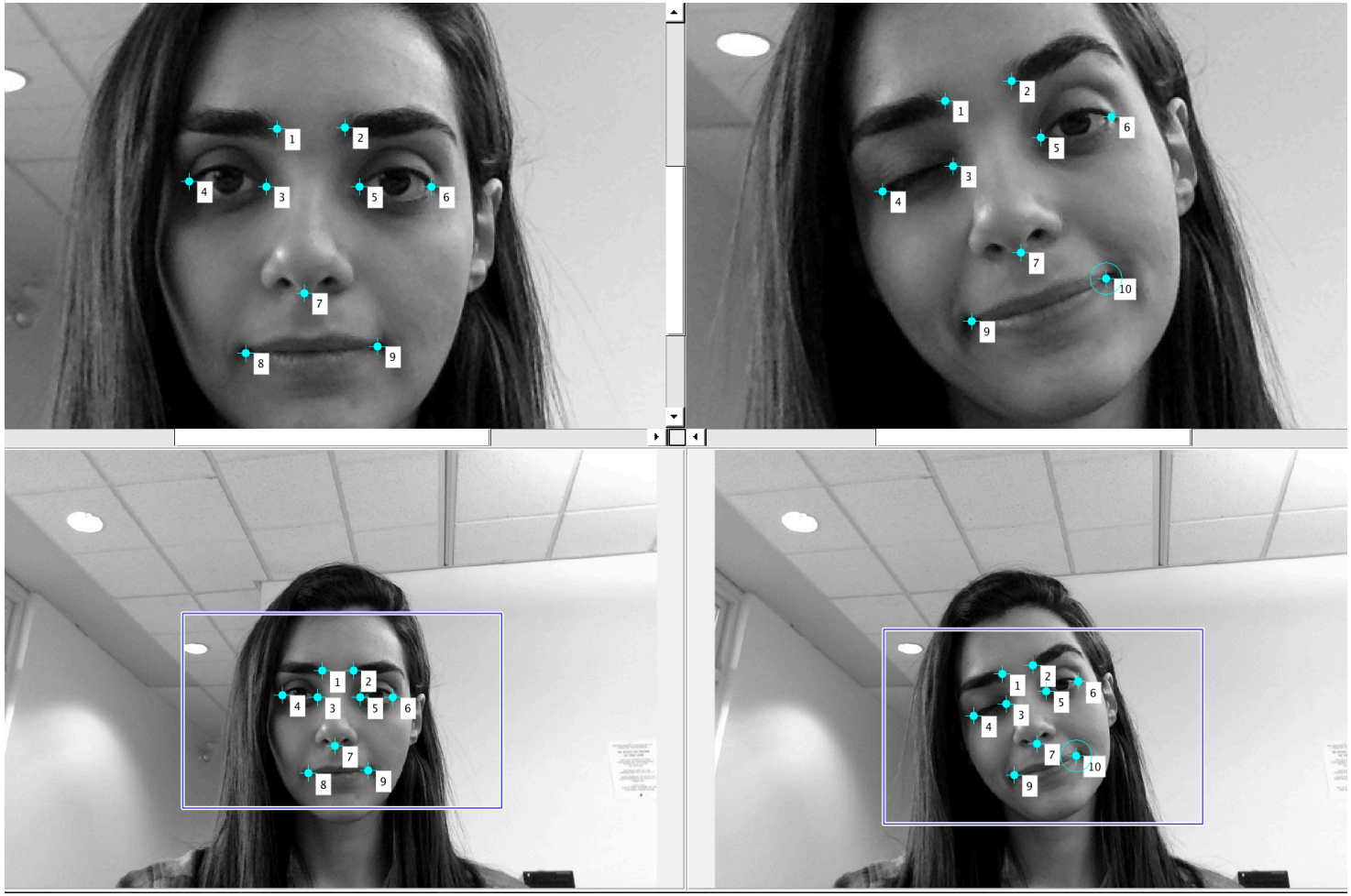


Figure16: Landmarks use in registration of the tilted image with the original image.

Part2: Registration

- To find the transform function that transformed the original image to the rotated image, I assumed a simple rotation and translation function.

$$\varphi(x) = Rx + T$$

- Then I found the covariance of the moving points and the fixed points. The rotation function and the translation function were found following the formulas given to us in class.

```
cov_h = (movingPoints' - moving_avg')*(fixedPoints - moving_avg)/size(movingPoints, 1);

[U, D, V] = svd(cov_h);

det_D = det(D);
if det_D >= 1
    S = [1, 0; 0, 1];
else
    S = [1, 0; 0, -1];
end

R_hat = U*S*transpose(V);
T_hat = transpose(moving_avg) - R_hat * transpose(fixed_avg);
```

Part2: Registration

- The transformation function found for my images is displayed in the following figure.

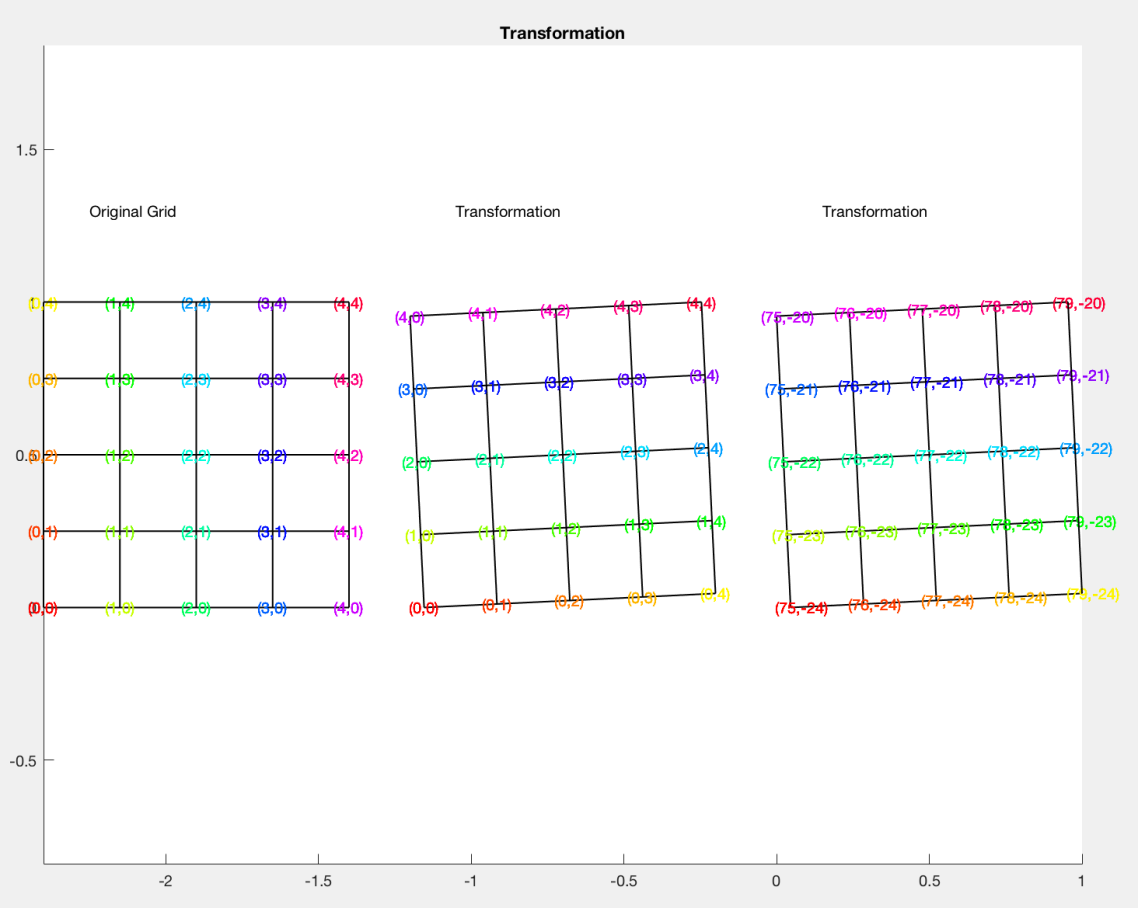


Figure 17: Transformation function from the original image to the tilted image.

Part2: Registration

- The tilted image then was transformed to find its registration with the original image. Next, I only transformed my head in the rotated image as my body wasn't rotated.

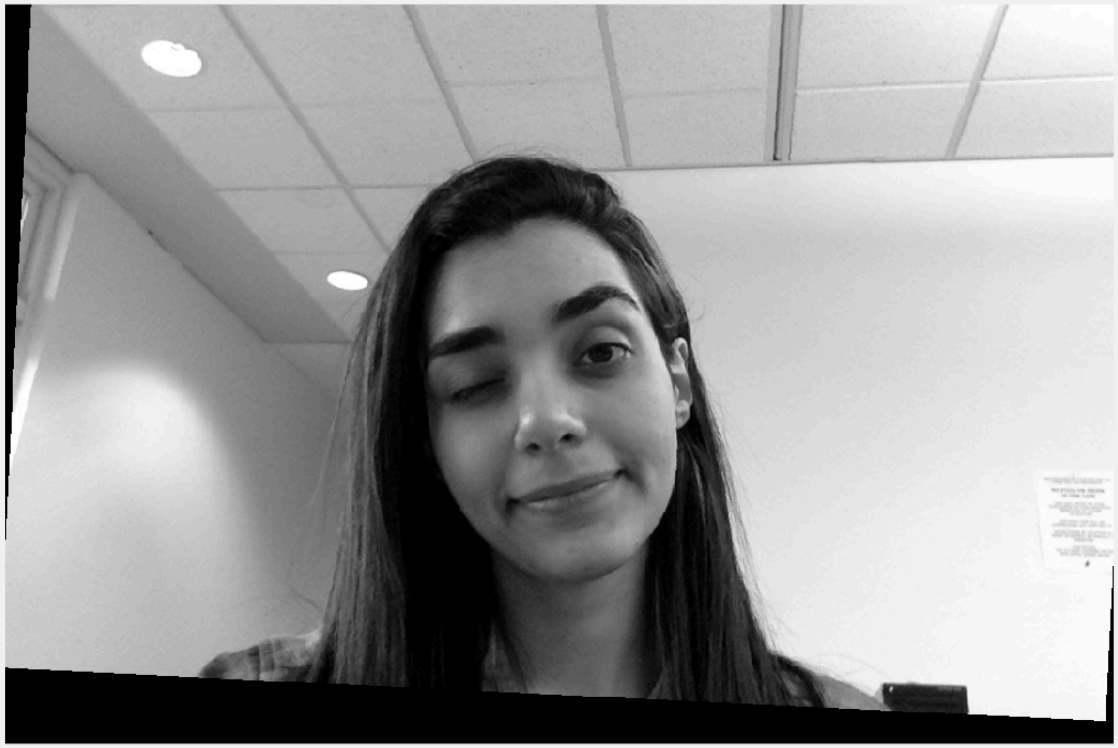


Figure 18: Transformed Image

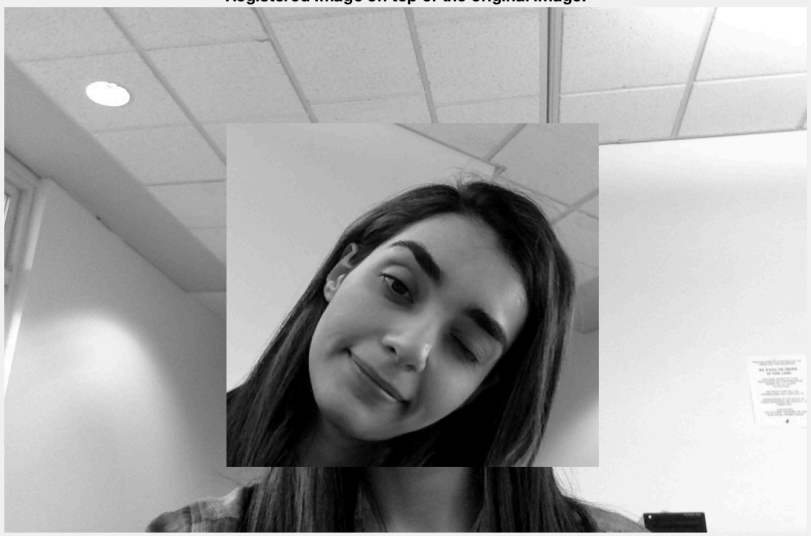
Part2: Registration

Figure 19: Comparison of transformed image with original image.

- I put my transformed face from the tilted image on top of the original image. As we can see, the tilted image was transformed back to its original position but the two images were not perfectly aligned. This is due to errors that exist in the formula.
- I could observe that as the angle of rotation increased, this error became bigger and the two images were less aligned. In fact the rotation and translation were both bigger than the actual ones. Results are shown in the next slide.

Part2: Registration

Registered Image on top of the original image.



Registered Image on top of the original image.