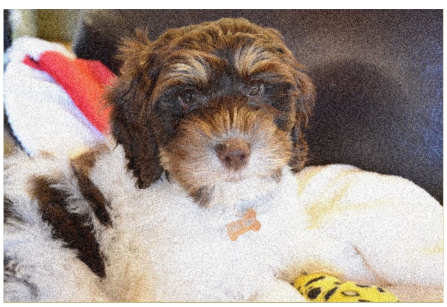
Project #1: Spatial and Frequency Filtering
CSC391: Introduction to Computer Vision
Grace Newman
Due date: Wednesday, February 6

3 Spatial Filtering

Original Noisy

Gaussian (3x3)

Median (3x3)



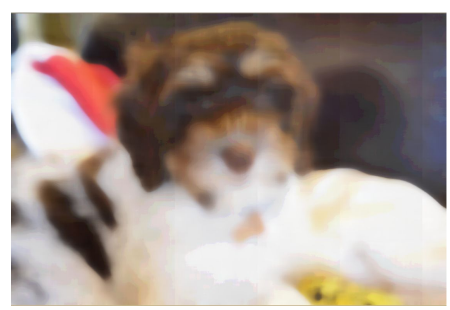
Original Noisy

Gaussian (9x9)

Median(9x9)

Original Noisy

Gaussian(27x27)



Median(27x27)

Based off of my experimentation using Gaussian blur and Median blur functions, it appears that Median in each case got rid of more of the noise by smoothing out the image, however the Gaussian filter kept more clarity and smoothed the image out less than the Median. I think that using k=3 as the filter size did not make much of an impact on the noisiness of the photo, however k=27 really compromised the subject of the photo by smoothing it out so much. I think that using k=9 was the best filter size option, and it depends on whether it is better to have less noise or a clearer photo whether to choose Gaussian or Median filter.

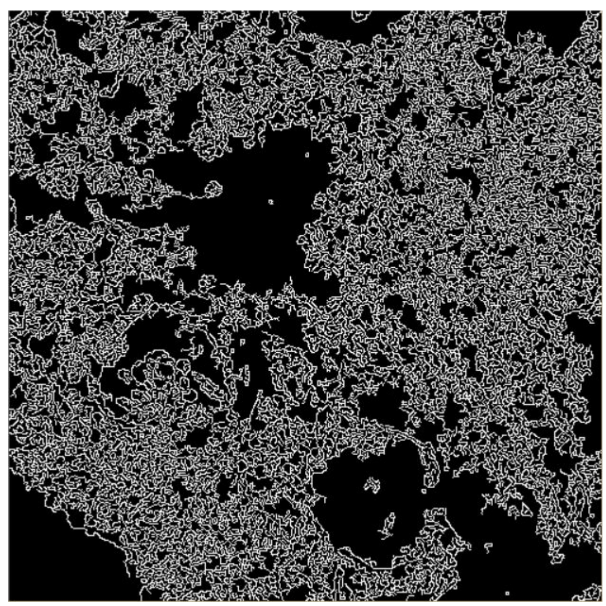
Noisy Image

Edges

Not noisy Image

Edges

window-05-05.JPG

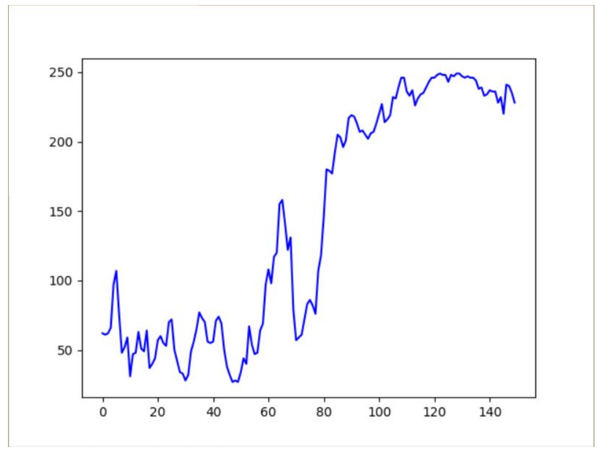


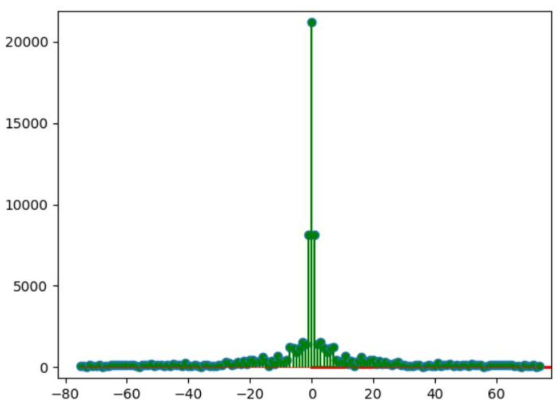
Edges

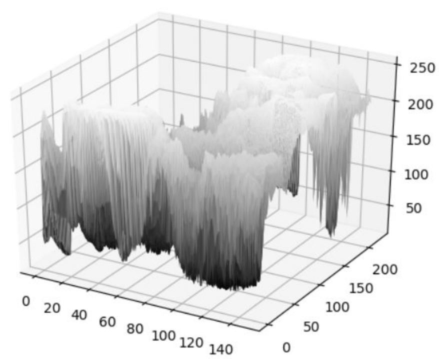
I used cv2 Canny function to obtain these results. On the noisy image of the dog, there edges are more erratic and looks more like static, than having any real outline. The noise distracts the image from the edges and it becomes less clear what the image pattern is. However, in the not noisy image of the puppy, there are more distinct edges of the outlines of the puppy. Also with the other dataset field image used, the main outlines of where the greenery is are distinct, but within those outlines the edges are all over the places because there is more noise in the image data.

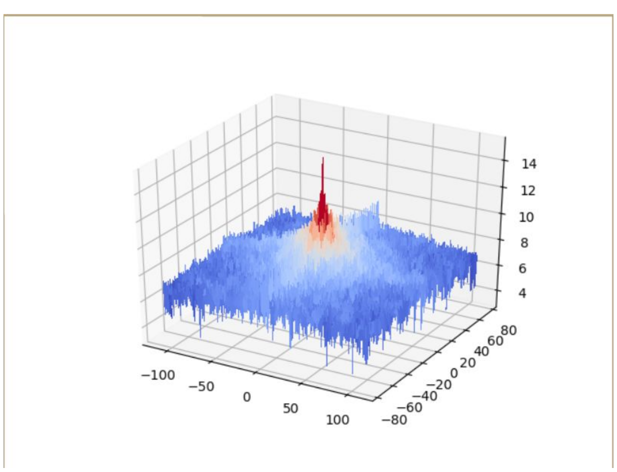
4 Frequency Analysis

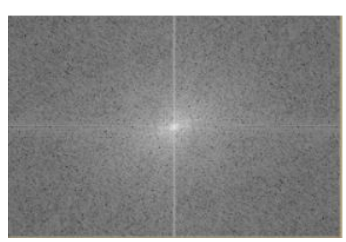
DSC_9259

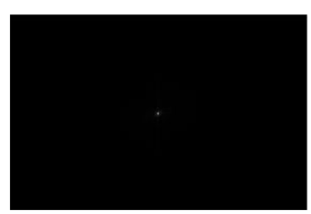






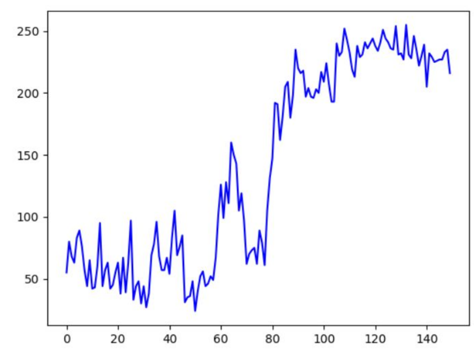


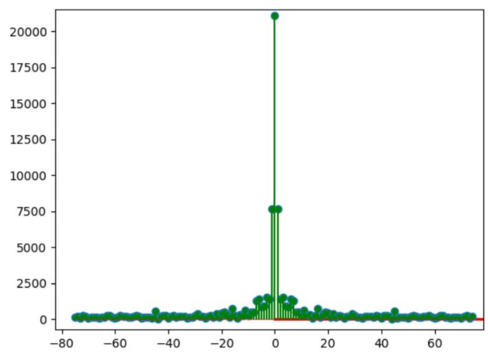


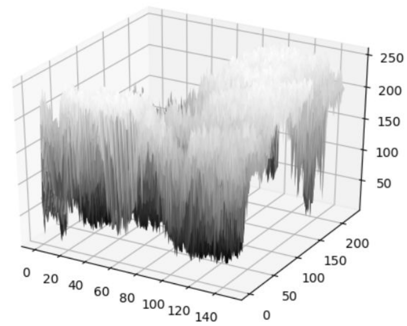


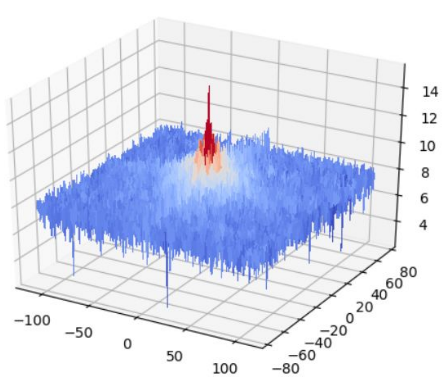
2D image of magnitude and log magnitude + 1 of the Fourier coefficients

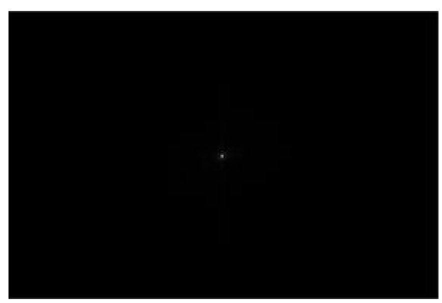
DSC_9259-0.50



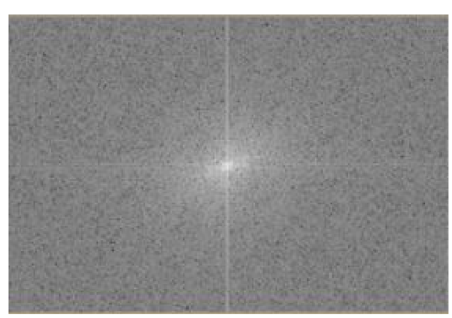




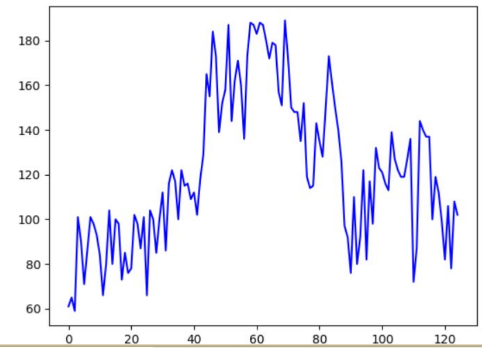


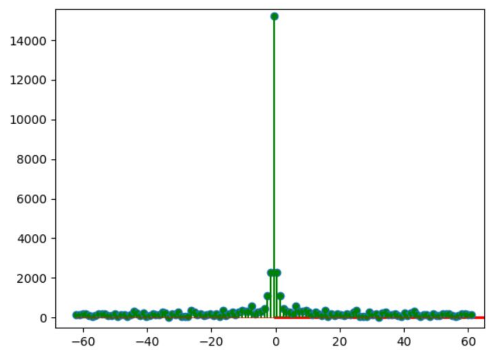


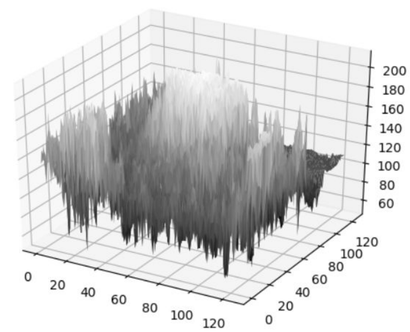
2D images of magnitude and log(magnitude)+1 of the Fourier coefficients

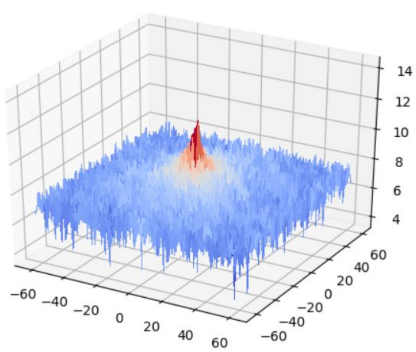


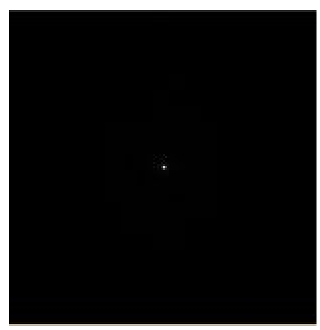
window-04-04



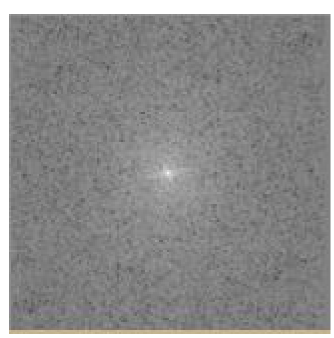




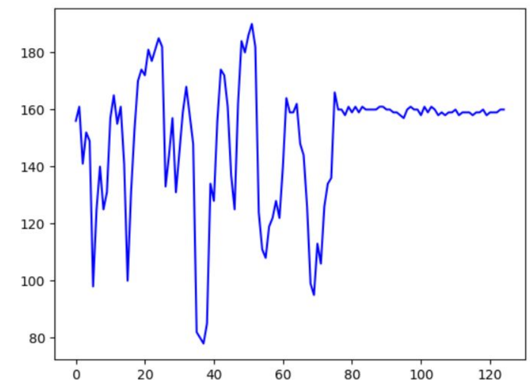


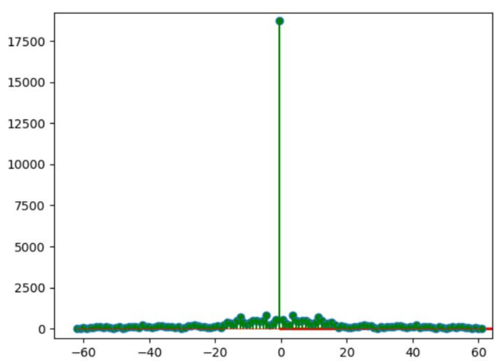


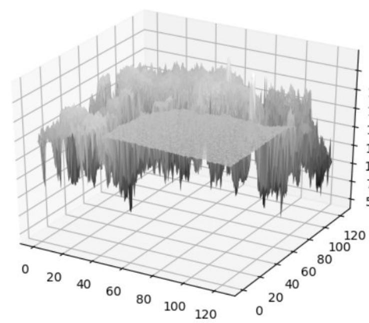
2D images of magnitude and log(magnitude) + 1 of Fourier coefficients

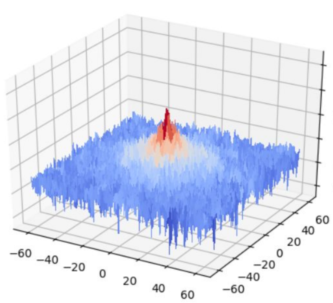


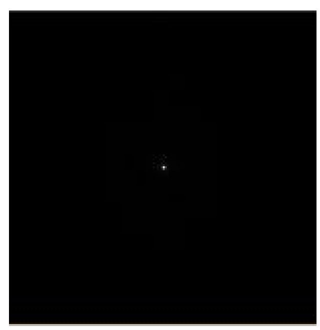
window-06-06



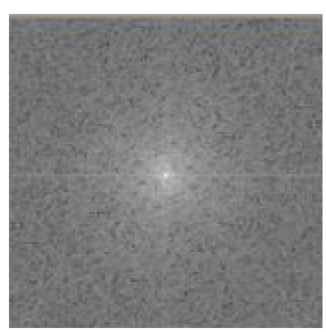








2D images of the magnitude and log(magnitude)+1 of Fourier coefficients



A. The image content appears to affect the decay of the magnitude of the Fourier coefficients, based on the contrast of the contents of the image. In the images of the puppy, the values mostly go from lower magnitude to higher magnitude, and have a more gradual normal curve shape on the 2D DFT. In the image with a mixture of greenery, sandy, and water content, there is more change in the values of the magnitudes. In the image of mostly sand and murkier water that is a consistent color, the magnitudes are sporadic maybe because of the edge created, and then it levels out around 160, probably because of the color consistency.

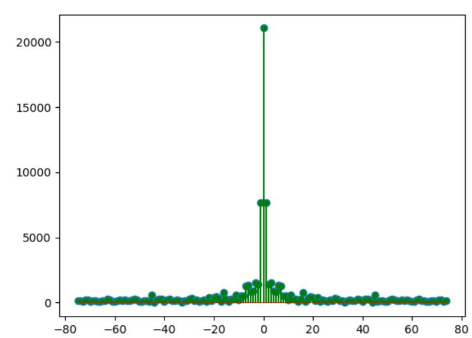
B. For the noisy and not noisy images of the puppy, the noisy one has a lot more sporadic back and forth and creates a more jagged curve. The original image of the puppy is a much smoother plot, and it's likely because of the lack of noise. However, the majority of the two plots mirrors the same magnitudes and trends.

C. Some of the log plots of the magnitude have higher values around the outside, such as the two puppy images. This happens because of noise in the images because it is a prevalence of the higher frequencies.

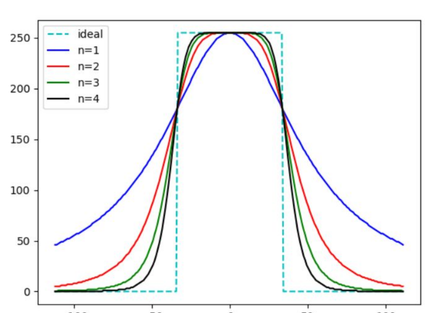
D. If you zero the values of the lower frequencies, it would be useful for edge detection because you are not including the most frequently occurring low valued frequencies. However, when you don't include the high frequencies further away from the center, it smooths the image out because it closely matches the magnitudes coefficients while removing the noise.

5 Frequency Filtering

DSC_9259-0.50



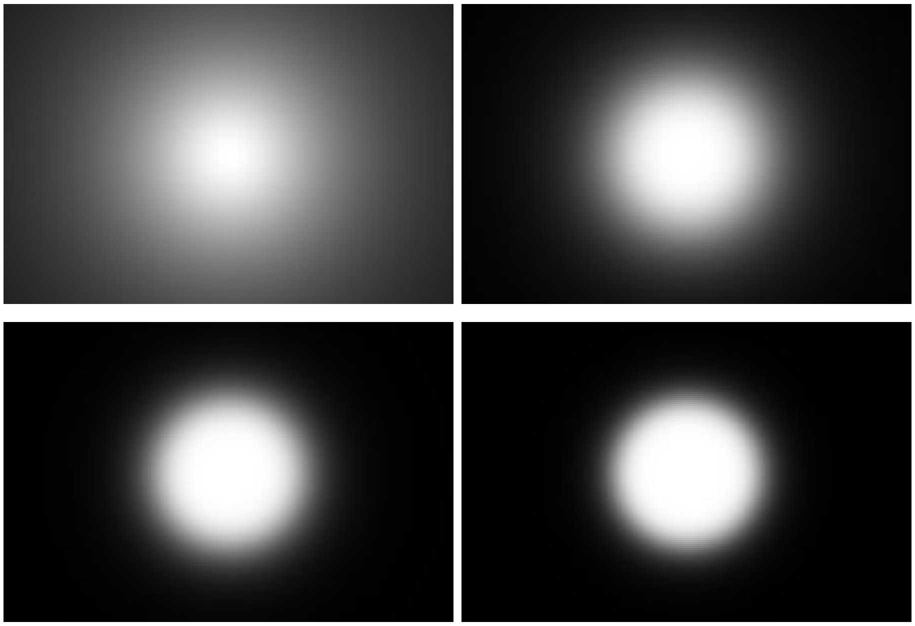
2D DFT



Butterworth scaled Magnitudes (low pass)

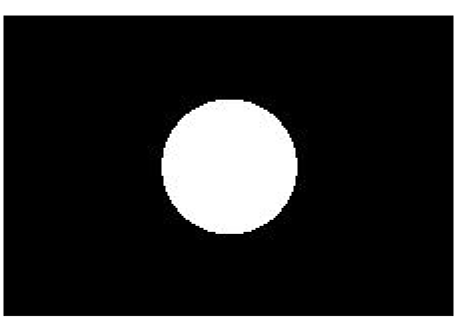
Butterworth Low Pass

Butterworth filters (low pass) n=1, 2, 3, 4

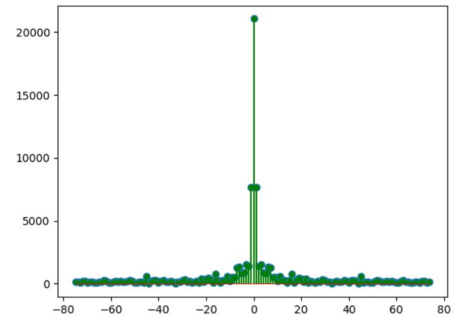


Butterworth (low pass) image gray n=1, 2, 3, 4

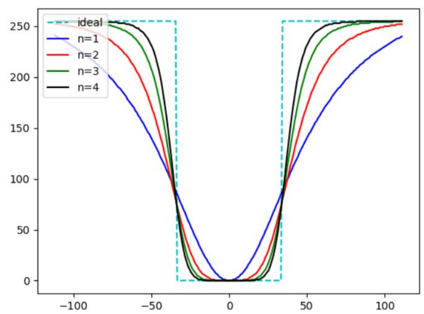
Ideal Low pass gray filter



Butterworth High Pass

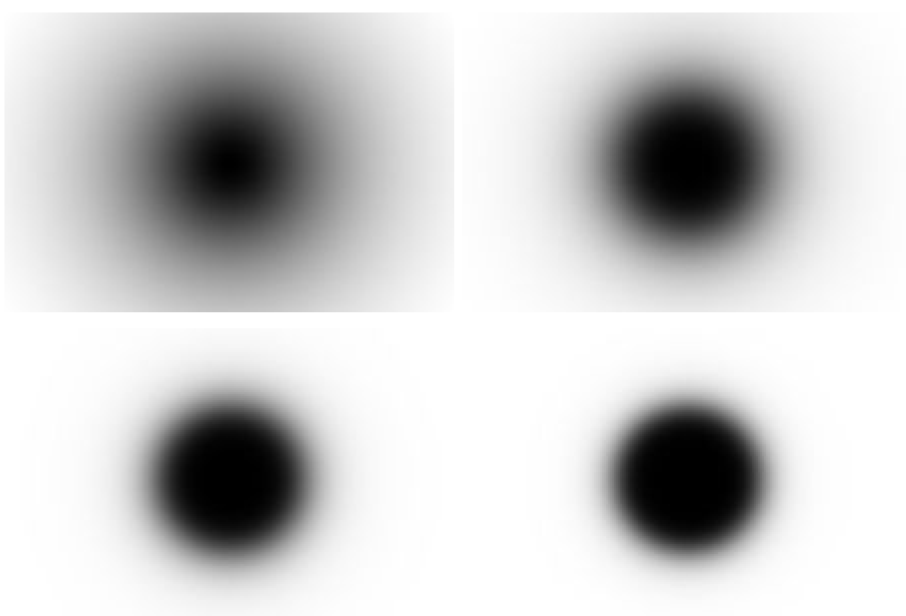


2D DFT

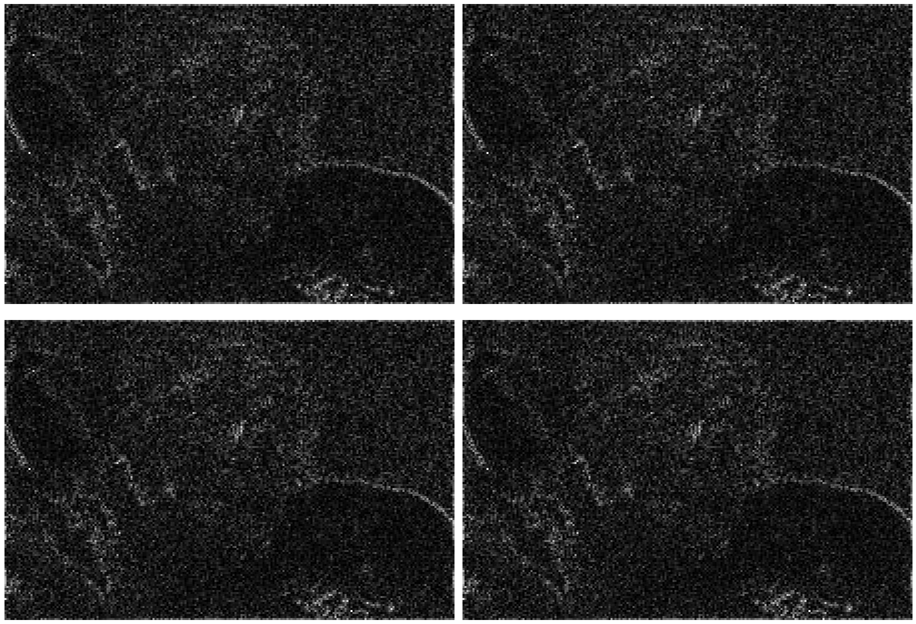


Butterworth Scaled Magnitude (high pass)

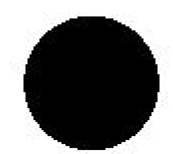
Butterworth (high pass) filter n = 1, 2, 3, 4



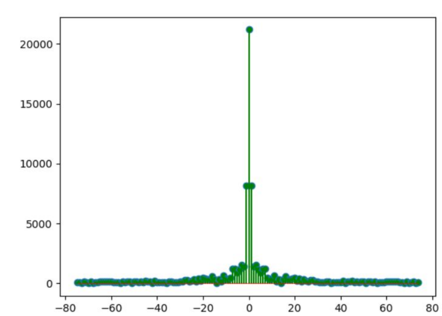
Butterworth (high pass) image grey n = 1, 2, 3, 4



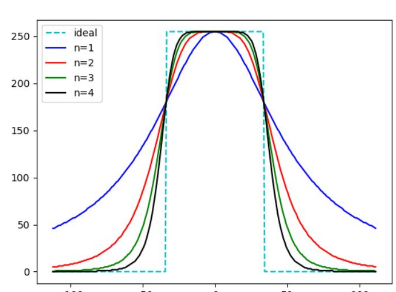
Ideal High Pass gray Filter



Butterworth Low Pass

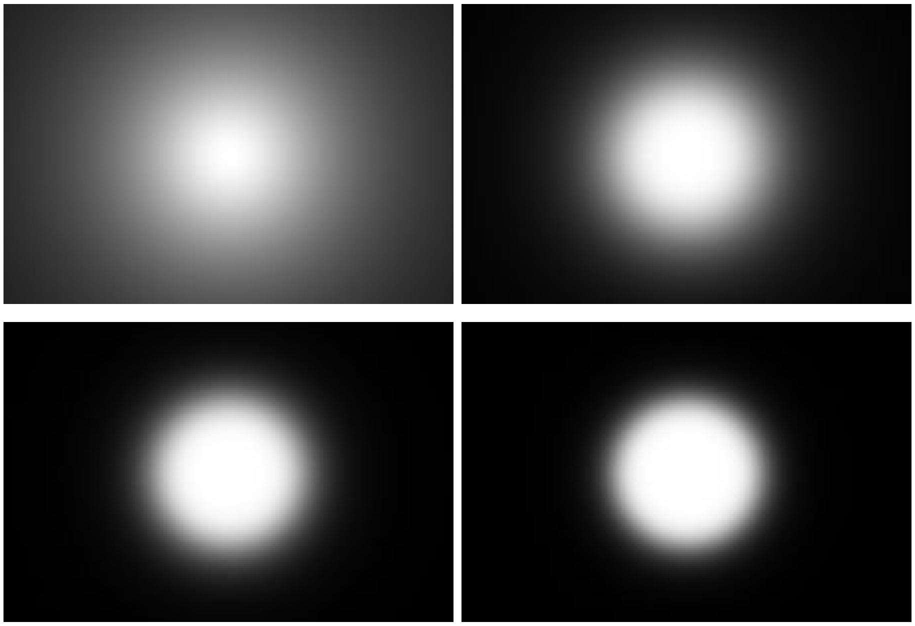


2D DFT



Butterworth scaled magnitudes (low pass)

Butterworth low filter pass n = 1, 2, 3, 4

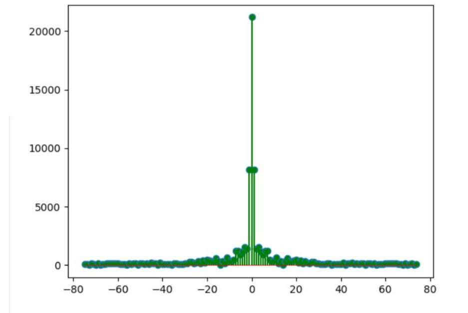


Butterworth low pass image gray n = 1, 2, 3, 4

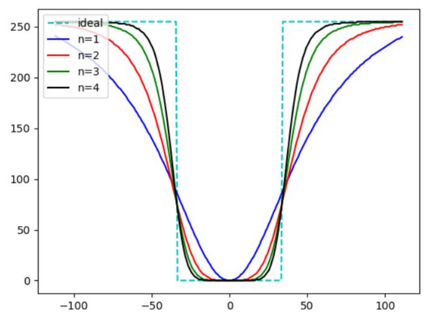
Ideal Low Pass

Butterworth High Pass

DSC_9259

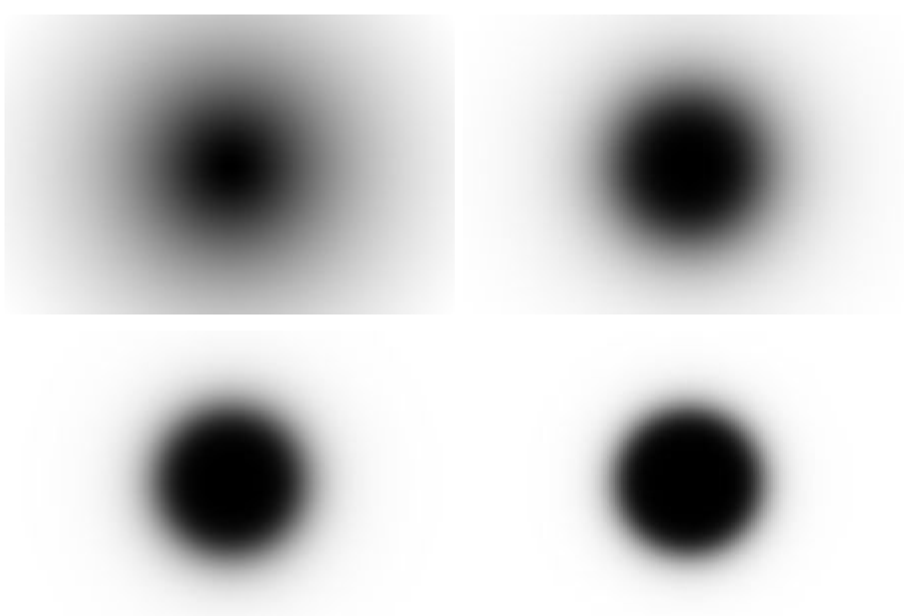


2D DFT

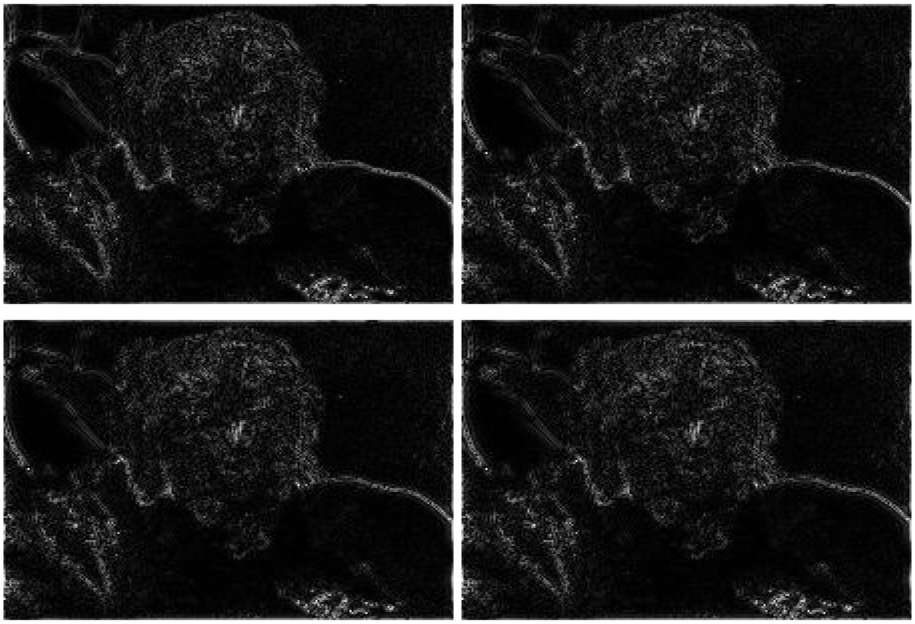


Butterworth Scaled Magnitude (high pass)

Butterworth Filter High pass n = 1, 2, 3, 4



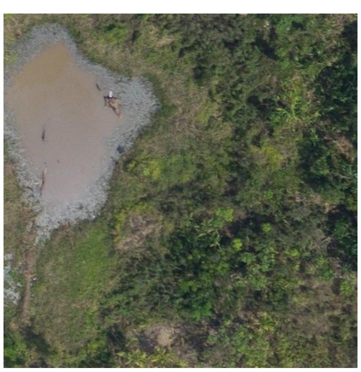
'Butterworth (high pass) gray image n = 1, 2, 3, 4

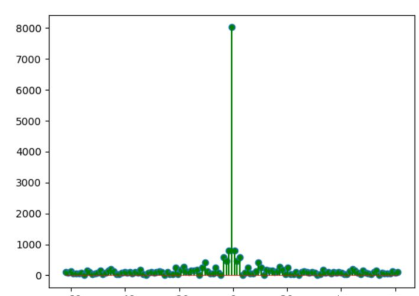


Ideal High Pass

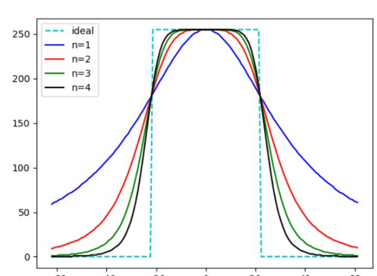
Butterworth Low Pass

Window-05-10



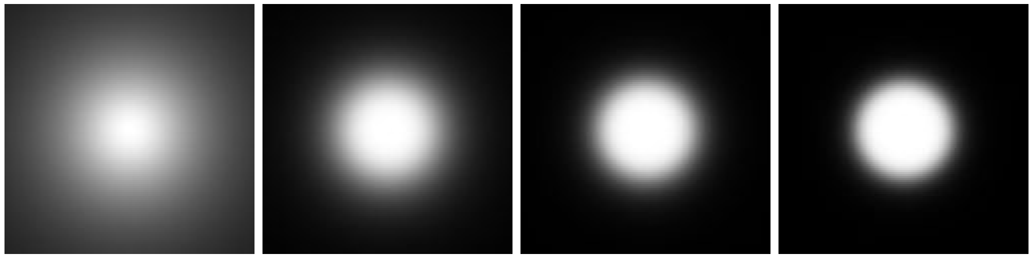


2D DFT

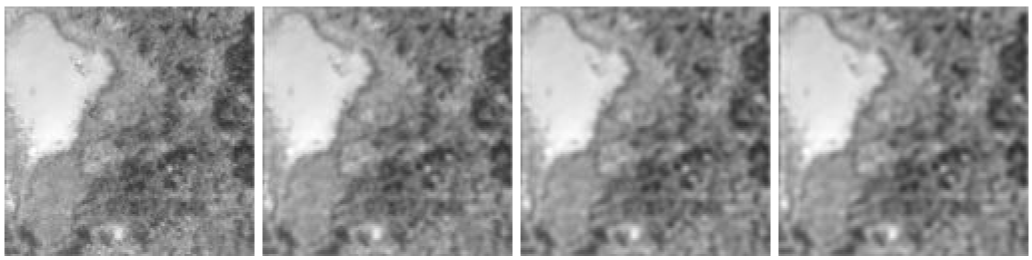


Butterworth scaled magnitude (low pass)

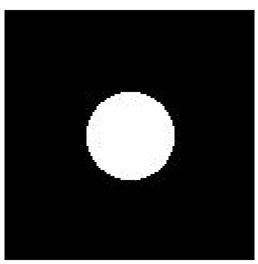
Butterworth Filter low pass n = 1, 2, 3, 4

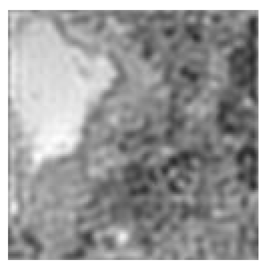


Butterworth low pass image grey n = 1, 2, 3, 4



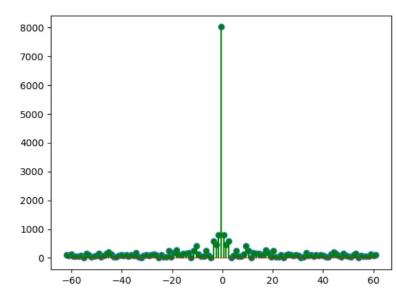
Ideal Low Pass





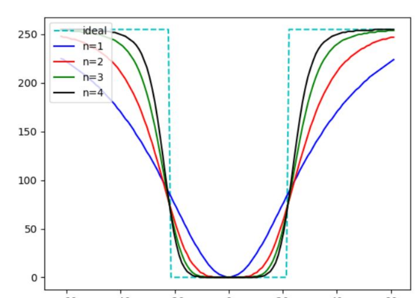
Butterworth High Pass

window-05-10

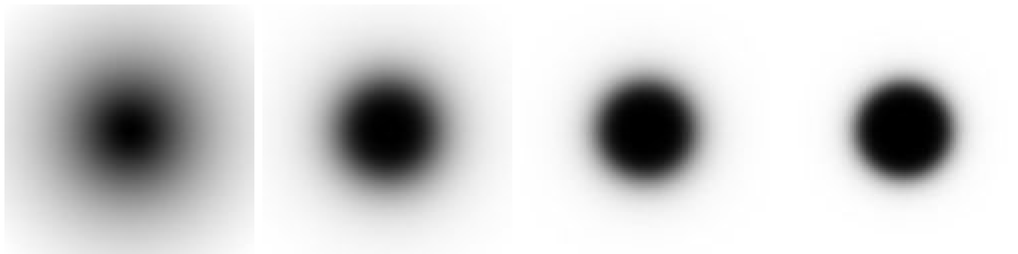


2D DFT

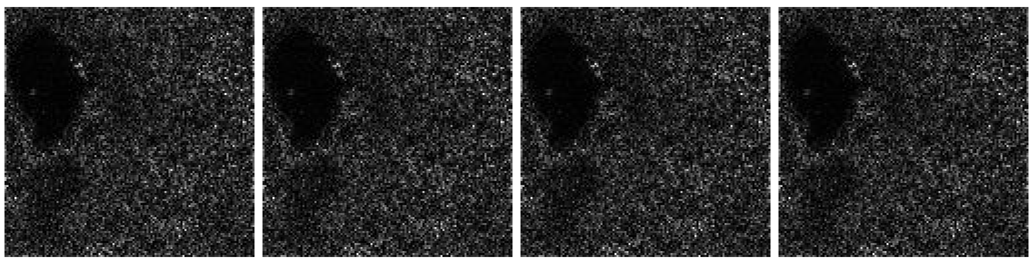
Butterworth scaled magnitude (high pass)



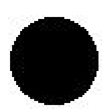
Butterworth filter high pass n = 1, 2, 3, 4



Butterworth high pass image grey n = 1, 2, 3, 4



Ideal High Pass



The ideal low pass filter is a white circle against a black background versus the ideal high pass filter which is a black circle against the white background. These ideal pass filters are different than the Butterworth filters, which blur out the edges of the circle depending on the value of n used. For n = 1, the filter is very blurred out and there is a lot of gray faded out from the white circle in the middle. When n = 4, It only soften the edges of the circle a little bit and the background is still mostly black. This means the butterworth filter does more smoothing to the images. The image after the Butterworth filter for n = 1 is a lot more clear than the image resulting from n = 4 Butterworth filter. However, on the noisy image of the puppy, it leaves a lot more of the noise in the image. The ideal low pass filter makes the image have more ripples than the Butterworth filter does. It looks like it filters the image less smoothly and leaves it blurrier and rippled on each of the ideal low pass images.

For the high pass filtered images, the high pass filter seems to be an edge detector. This is because the high pass filter is zeroing out the low frequencies of the image, which are the important ones. The high pass filters were created by taking (1 - lowPassFilter) for both the ideal high pass filter and the high pass Butterworth filters. The n = 1 Butterworth high pass filter seems to make the image less noisy than the n = 4 Butterworth high pass filter.