

Project #1: Spatial and Frequency Filtering

Instructor: V. Paúl Pauca

Yijie Yang

3 Spatial Filtering

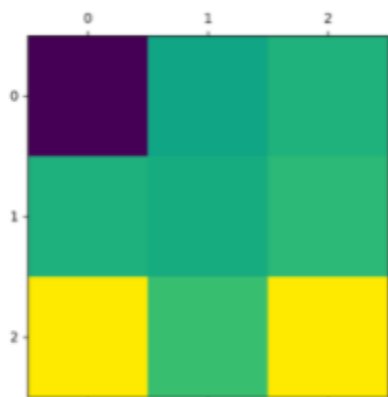
- 3.1 Implementation



Original image



Filtered image



Box filter with size $k=3$

- 3.2 Smoothing, Denoising, and Edge Detection Filters

- Smoothing and denoising*



Gaussian filter of size 3x3



Gaussian filter of size 9x9



Gaussian filter of size 27x27



Median filter of size 3x3



Median filter of size 9x9



Median filter of size 27x27

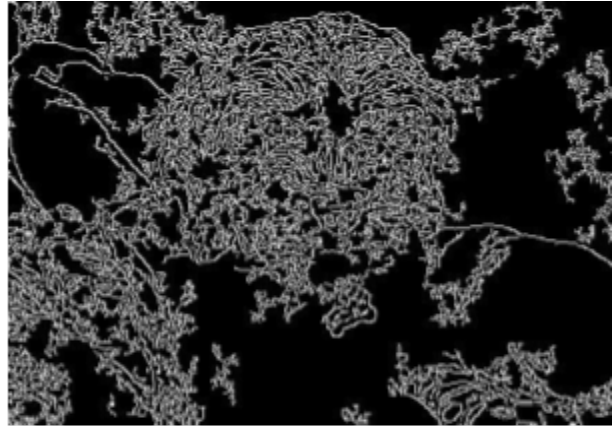
Based on the images above, we notice that as the size of filter increases, the original noisy image is denoised, but the resolution is correspondingly lowered. The Gaussian filter applies a linear operation on the noisy image so the image is smoothed and can be hard to identify the edges. The median filter is a non-linear operation. Since we replace the neighboring pixels with the median

values, the filtered image will appear to preserve the edge but lose some details of the original image.

- *Edge detection*



Canny edge detector on
original image

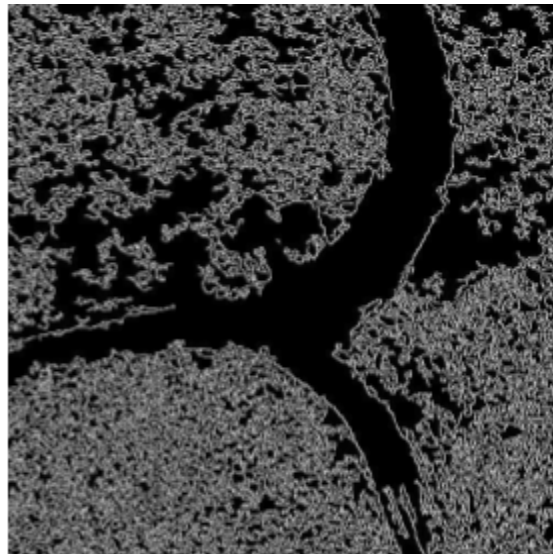


Canny edge detector on
noisy image

By comparing the original image and the noisy one, we can see the noisy image has more random variations in color that make it unsmoothed. When they are filtered using Canny edge detector, the original one identifies a clearer sketch for the puppy's shape. Whereas the noisy image has more random edges that makes us harder to distinguish the shape of the puppy.



Original field image



Canny edge detector image

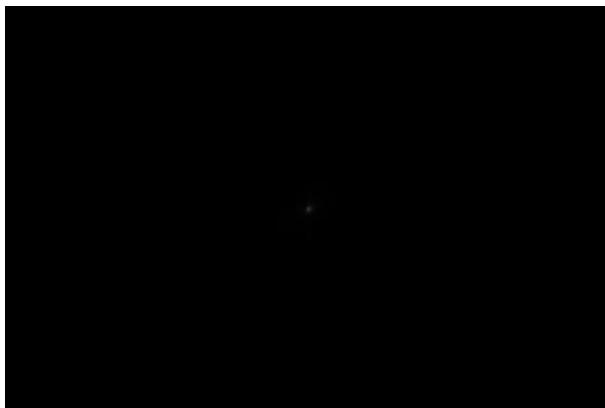
The field image has the trails of bright colors, thus the edges between the trail and forest are easily detected. However, since the forest has shadows and variations in the color of trees, Canny detects an excess of edges in the forest which do not exist. There can be some better ways to extract edges from the noisy data. For example, we can blur the noisy image a little bit first. This way some details of noise will be smoothed and not appear as potential edges. Then we can increase the brightness of the image to the point that causes the color of trails exceeding maximum. Thus the trail will be replaced with an identifiable dark color and the shadows in the forest will be not so obvious to be detected. Therefore, we get an image of better edge for the noisy data.

4 Frequency Analysis

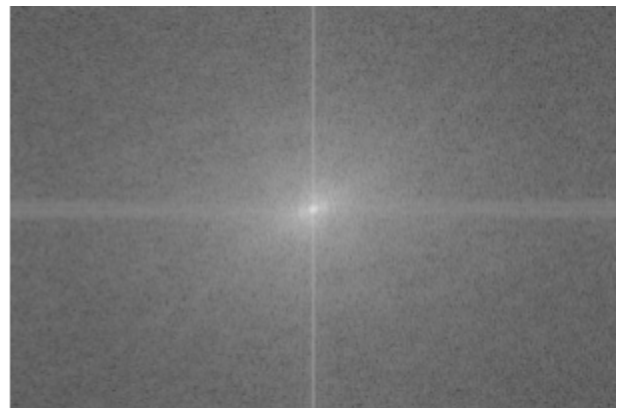
- 4.1 Implementation



Gray puppy image

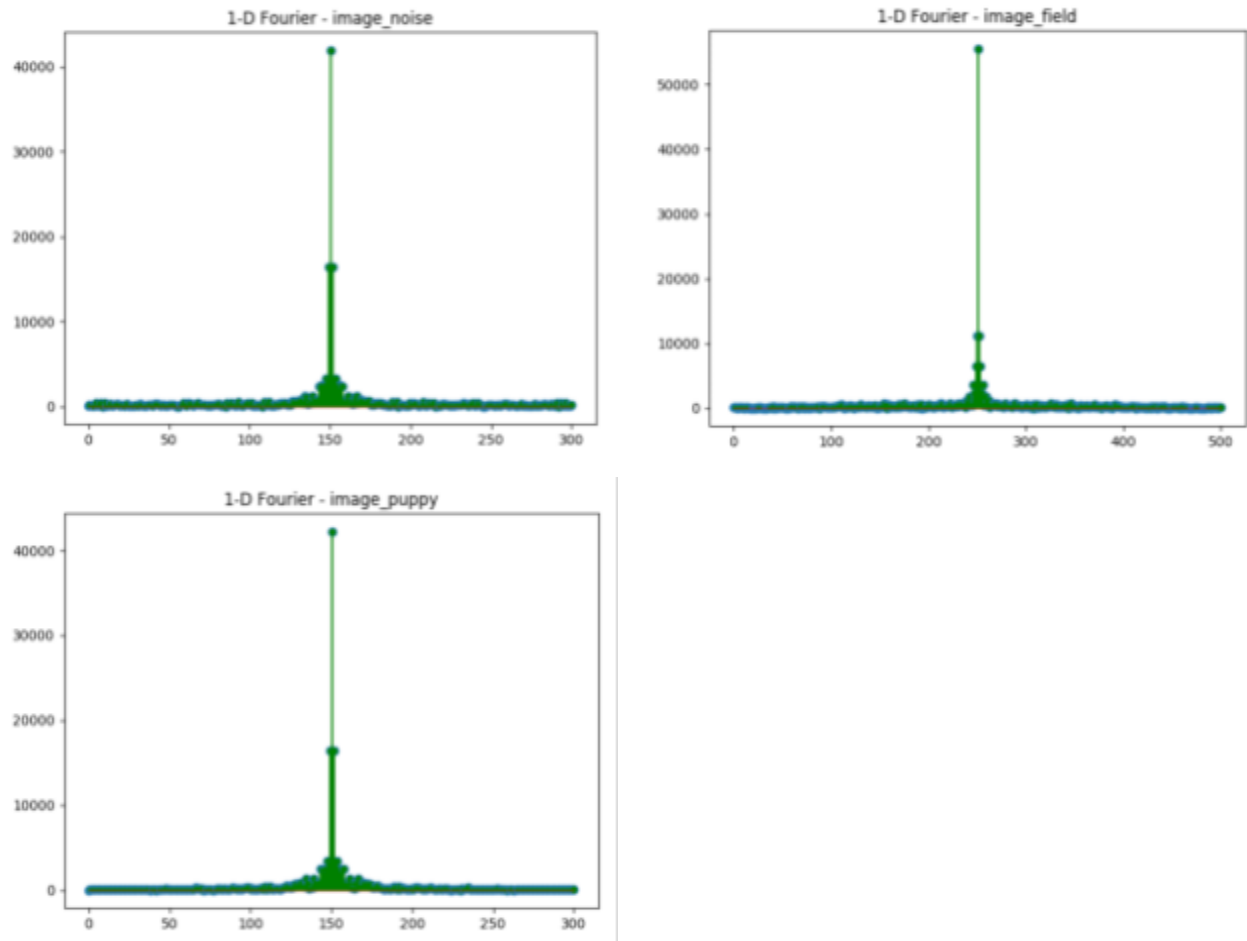


2D-DFT magnitude image



2D-DFT log(magnitude + 1) image

- 4.2 Frequency Analysis



Based on the plots above, there is some correlation between the image content and the decay of the magnitude of the Fourier coefficients. The original puppy image is more similar to its noisy one than to the one with field image in Fourier coefficients. The decays from the middle zero both drop to about 17000 in frequency, whereas for field image, it drops to about 10000 and has almost a flat in frequency of other signals. The effect of noise makes the Fourier coefficients more unsmoothed as well. There is some oscillation on the decay of magnitude. The reason log plots of the magnitude of the Fourier coefficients show higher values around the axis is that the low frequency of signals appear more often in the image since the values along the axis are closer to the center and lower in frequency compared to signals in other places. It makes sense a smoothed image tends to have more low frequencies rather than noisy high frequencies. When we zero-out high frequencies of Fourier coefficients away from the center, we only allow the low frequencies to pass, so this is an ideal low-pass filter. The result is shown in 5.1, which gives us a blurred image with lots of details taken out from original image. The Fourier coefficients in the center zero-out leaves only the high frequencies, which is an ideal high-pass filter. In 5.2, we observe that only details like hair are revealed in the filtered image.

5 Frequency Filtering

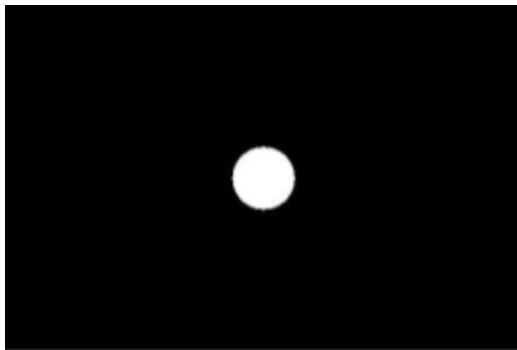
- 5.1 Implementation



Gray puppy image



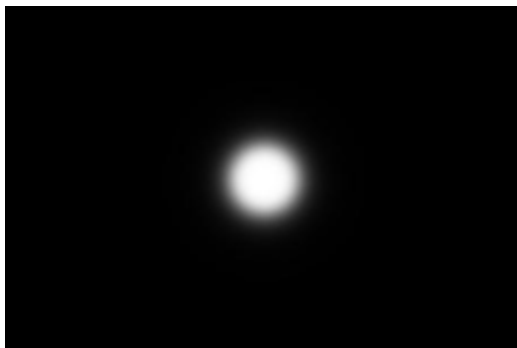
2D-DFT image



Ideal low pass magnitude



Ideal low pass filtered

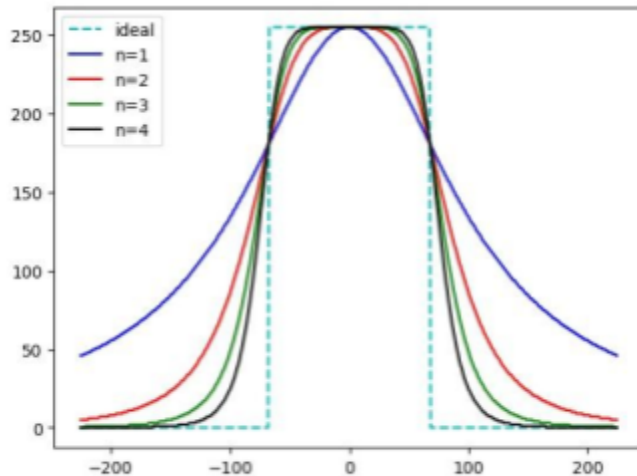


Butterworth-scaled magnitude

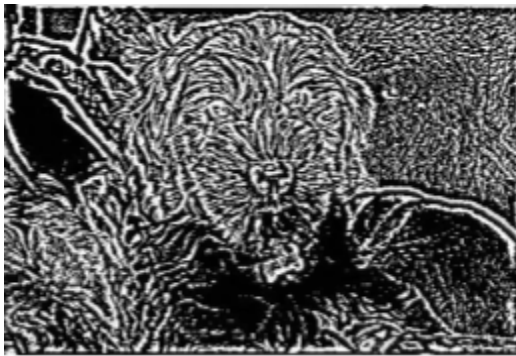


Butterworth-scaled image

- 5.2 Low-pass and High-pass Filtering



Butterworth filtering



Butterworth-scaled
High-pass magnitude



Ideal high-pass filtered

Based on the Butterworth filtering plot, we can see that low magnitude of Butterworth filtering is very similar to Gaussian which is symmetric and has a almost normal distribution. As the magnitudes increase, the curves become steeper, which are closely resembling the cylinder-shaped ideal low pass filter. Comparing the Butterworth-scaled and ideal high-pass filters, we see there are more low frequencies pass through the Butterworth filter and results a detail of puppy's hair (high frequencies) and different parts of puppy's body (low frequencies) in contrast to the ideal one which mainly captures the hair on the face of puppy.