Swaran Ponshunmugam
CSC 391G: Computer Vision
Dr. Pauca
Project One

PART ONE: SPATIAL FILTERING

Python Script reads in image and displays it using CV2. The script also asks user for a size 'k' that will be applied towards the sizing of the blur. CV2's built-in blur function makes this possible. Though this user-defined value 'k' can be used in the following applications, k is discarded and specific size values are used instead for Gauss filtering, Median filtering and other 2D transformations.

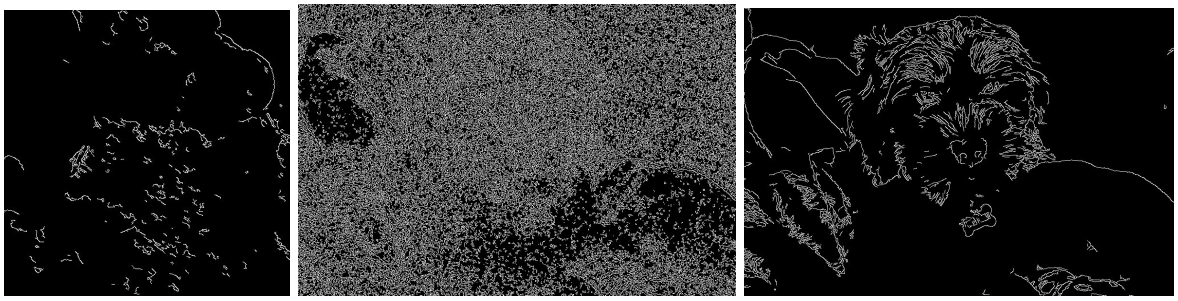
An affine transformation from the Python CV2 text is applied to the puppy image. This distorts the size of the head of the puppy and results in a widening of the top portion near the puppy's eyes and a condensing of the puppy's body.

A Gauss and median filter are applied to the puppy picture. The filter size makes it so that with an increase in size, there is an increase in smoothness (denoising), but a noticeable reduction in resolution. The image is not very noisy (subjective) in its original form. Both filters simply work to reduce the resolution, especially at the 27x27 size. The median filter does add an artistic effect because the blurring makes it so that sharp color

differences (the brown and white hairs in the puppy) are well defined but small differences are negated and clustered as the same (the eyes of the puppy is hard to see with the median filter).

Gaussian Filter, left. Median Filter, Right

Canny Filtering is used for the puppy image (original), noised puppy image and for the nature image (all in database). The result is that with both original pictures (nature and puppy) one can make out what is being represented. The edges are clear and few. In the noisy puppy picture, there are too many edges detected and a clear image is not visible.



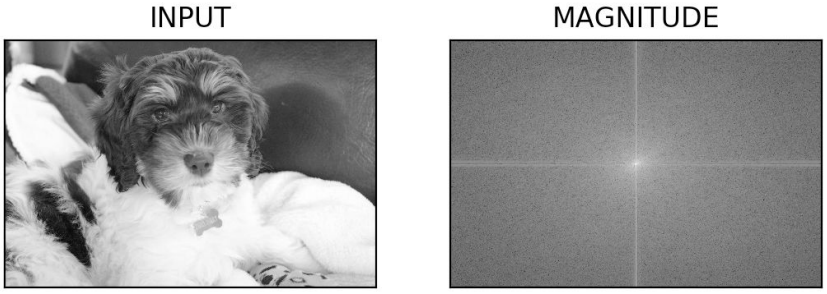
Canny Nature, left. Canny Noisy puppy, middle. Canny Puppy, right.

PART TWO: FREQUENCY ANALYSIS

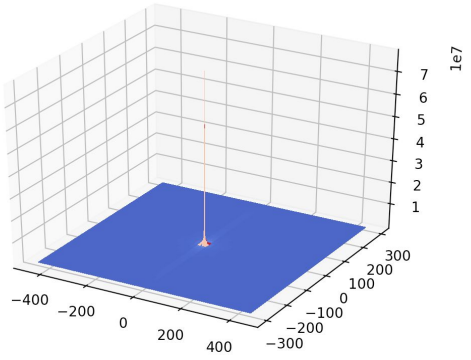
The frequency analysis suggests that images with greater differences in the image with greater detail have higher tendency to deviate from the center of the plots. So with the image content, the less the decay in visibility and fineness, the more likely the plots will be centered.

Noisier photos have will have higher frequencies and therefore the noisier picture of the puppy has coefficients that are greater than that of the original puppy picture.

Deviation from the center of the axis because there are sharper differences and higher frequencies that are recognised, as seen in the differences in a original puppy image and a noised puppy image. Manipulation of the coefficients for the fourier transformation help in edge detection and smoothening alike. The "zeroing" out of low frequencies increases the distance from the center of the axis and vice versa for high frequencies. This can be useful because manipulating the frequencies can help in how pixels appear side to side (smoothening) and how you can bring out the higher frequencies (to find edges).

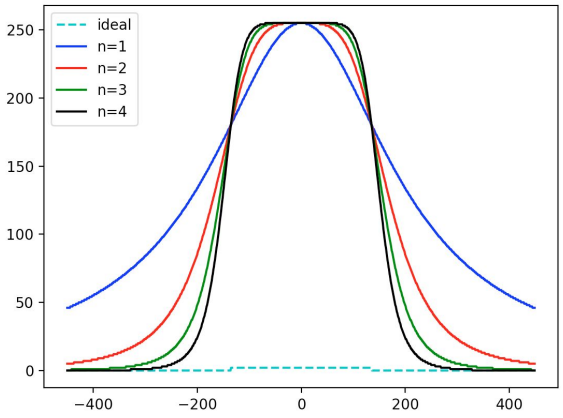


For noisier images, the distinction of the lines in the Fourier representation is easier



PART THREE: FREQUENCY FILTERING

This part explores differences in ideal low pass and Butterworth filters. Importantly, the Butterworth filter smoothes the blurring of photos better than a low pass. Instead of zero-ing out the coefficients, the Butterworth filter slowly decreases the coefficients as distance from the filter increases. The lowpass filter proves useful when dealing with different frequencies in photos; by moderating the high frequencies and sustaining the low frequencies, there is a visible noise reduction, whereas Butterworth was effective in sharpening the image.



BUTTERWORTH FILTERING (n=1,2,3,4,ideal)

PICTURE	MAGNITUDE (n=1,2,3,4)