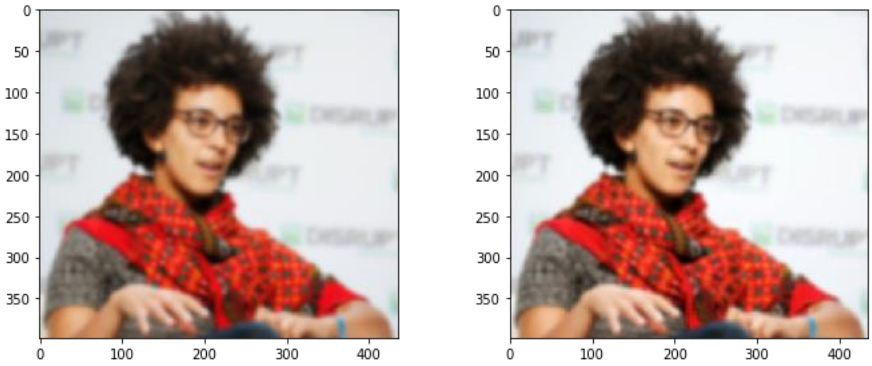
CSCI 631 Homework 1

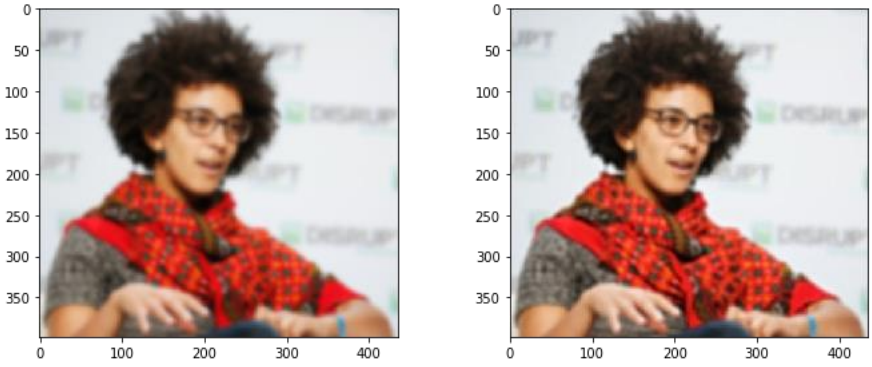
Problem 1

Following is the influence of various values of window sizes and sigma.

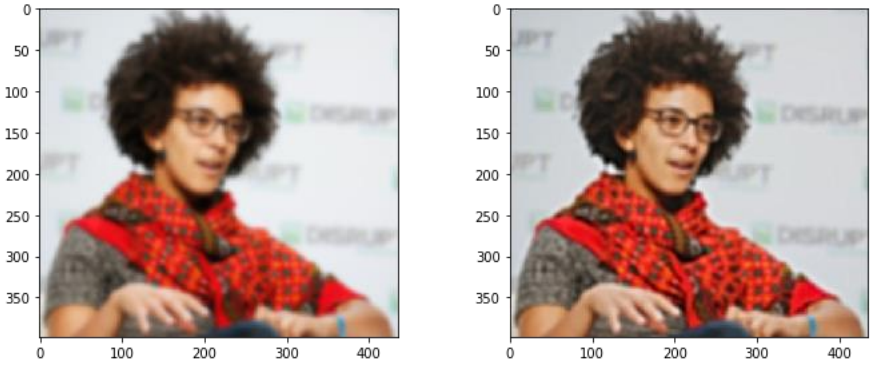
1. Window size = 30, sigma = 1



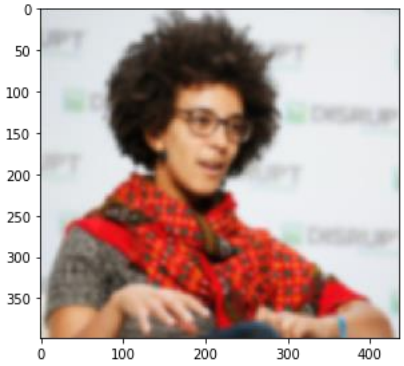
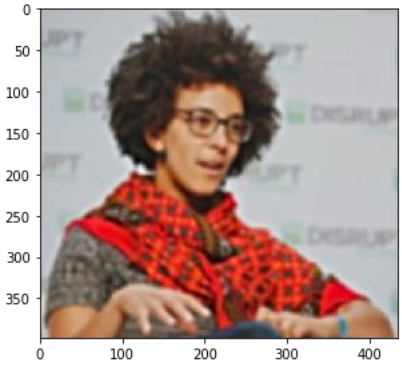
2. Window size = 35, sigma = 2



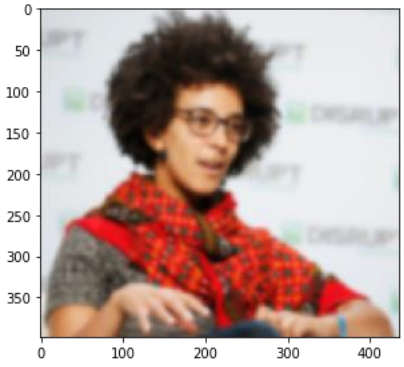
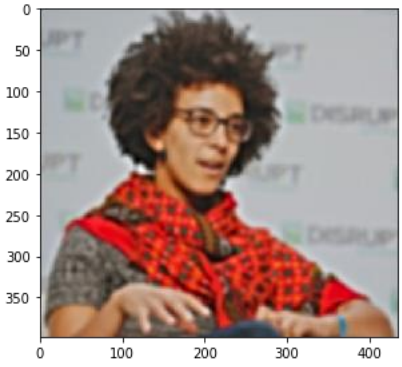
3. Window size = 40, sigma = 3



4. Window size = 45, sigma = 4

5. Window size = 50, sigma = 5

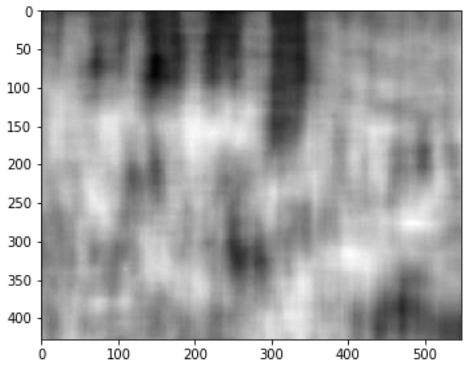
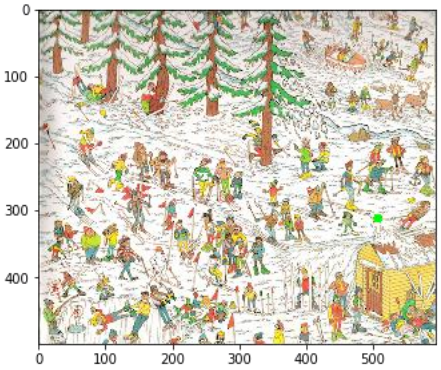



From the above results we can conclude that, out of all the parameters, a window size of 35 and a sigma value of 2 give a relatively better result.

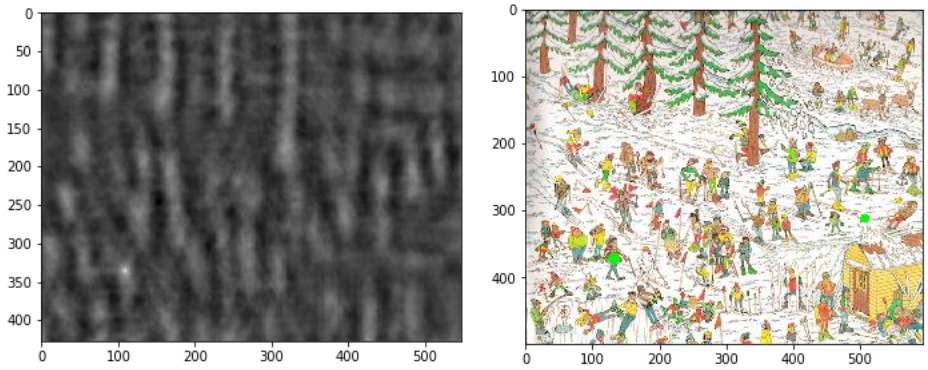
Problem 2

The first sub problem of cross correlation does not identify the correct position of Waldo. Whereas, the normalized cross correlation and SSD function correctly identify the position of Waldo.

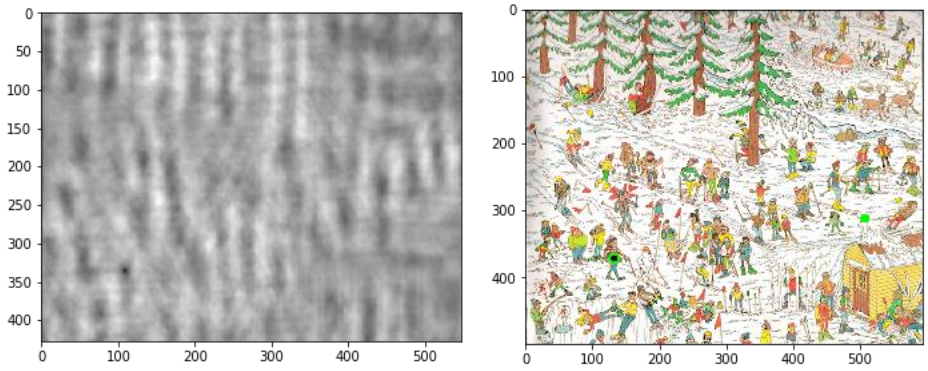
Output of crosscorr (Img, f):

Output of normcrosscorr(Img, f):



Output of SSD(Img, f):



Problem 3

yes, I read the medium article on canny edge detection. It explains in detail that the canny edge detection process involves following steps:

1. Noise reduction:

Noise from an image is removed by smoothening or blurring it. This is done by convolving image with a Gaussian kernel (3x3, 5x5, 7x7).

After applying the Gaussian blur, we get the following result:

Original image (left) — Blurred image with a Gaussian filter (sigma=1.4 and kernel size of 5x5)

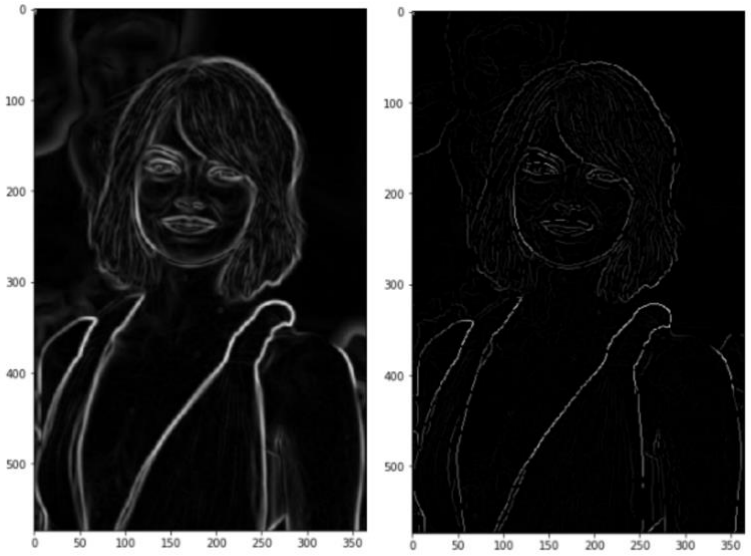
2. Gradient calculation

The Gradient calculation step detects the edge intensity and direction by calculating the gradient of the image using edge detection operators. Edges correspond to a change of pixels' intensity. To detect it, the easiest way is to apply filters that highlight this intensity change in both directions: horizontal (x) and vertical (y).

Blurred image (left) — Gradient intensity (right)

3. Non-maximum suppression:

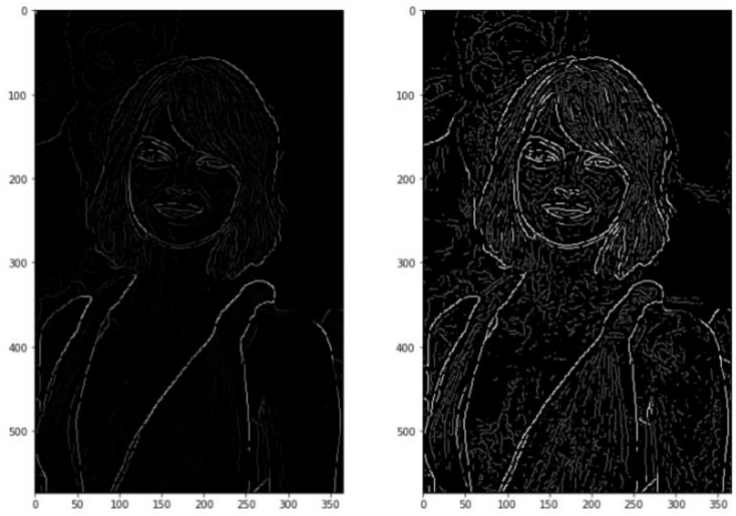
The edges detected by previous step vary in their width. Ideally, the final image should have thin edges. Therefore, we perform Non-maximum suppression. In this, we check the direction of the gradients of pixels. The pixels along this direction with maximum intensity are selected and the rest are discarded. Thus, we get thin line or edge.



Result of the non-max suppression.

4. Double threshold:

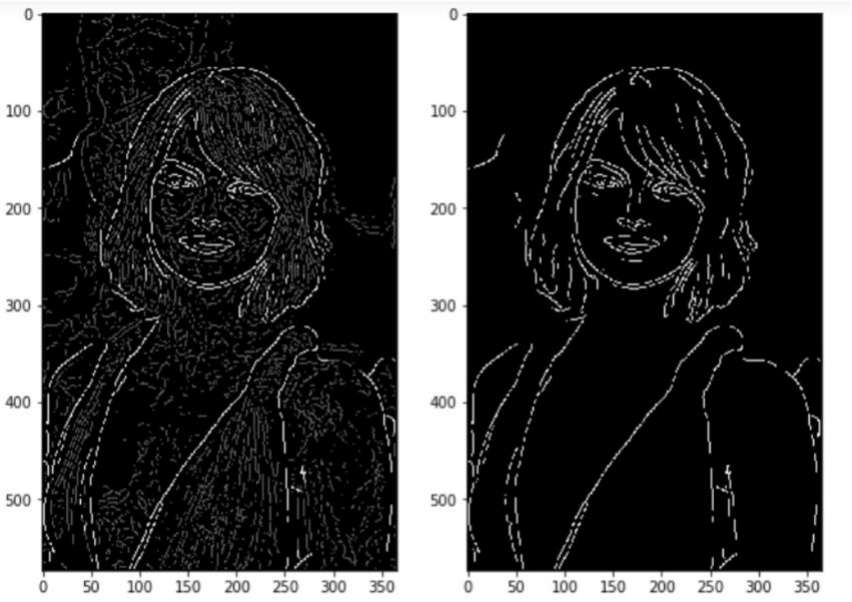
High threshold is used to identify the strong pixels (intensity higher than the high threshold). Low threshold is used to identify the non-relevant pixels (intensity lower than the low threshold). All pixels having intensity between both thresholds are flagged as weak and the Hysteresis mechanism (next step) will help us identify the ones that could be considered as strong and the ones that are considered as non-relevant.



Non-Max Suppression image (left) — Threshold result (right): weak pixels in gray and strong ones in white.

5. Edge Tracking by Hysteresis

Based on the threshold results, the hysteresis consists of transforming weak pixels into strong ones, if and only if at least one of the pixels around the one being processed is a strong one



Results of hysteresis process