Programming Assignment 3

CS 485

Shubham Gogna

Due: April 14, 2015

Dr. Mircea Nicolescu

Part One: Harris Corner Detector

To implement the Harris corner detector, we can apply a couple of tricks to simplify the process. This starts with convolving the image in the X and Y direction with the first derivative of the Gaussian to build I_x and I_y respectively. This gives us the gradient image in the X and Y direction. If we convolve the components I_x^2 , $I_y^2, and\ I_xI_y$ with a Gaussian, we can build the autocorrelation matrix for each pixel.

$$A_w = Gauss * \begin{bmatrix} f_x^2 & f_xf_y \\ f_xf_y & f_y^2 \end{bmatrix}$$

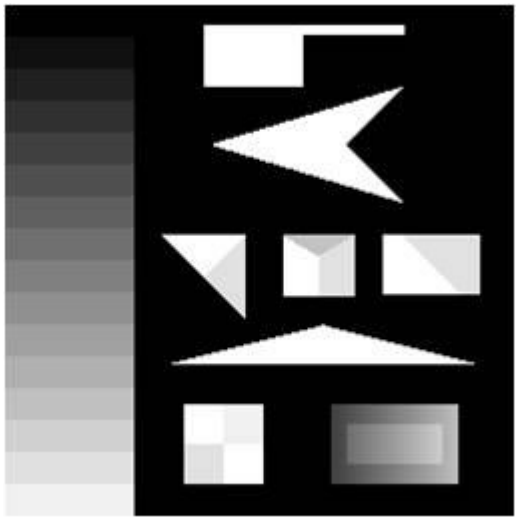
Using the autocorrelation matrix, we can compute a value for each pixel that will classify it as a flat region, edge, or a corner.

$$R(A_w) = \det(A_w) + \alpha * trace^2(A_w)\ , Corner\ when\ R(A_w) > threshold$$

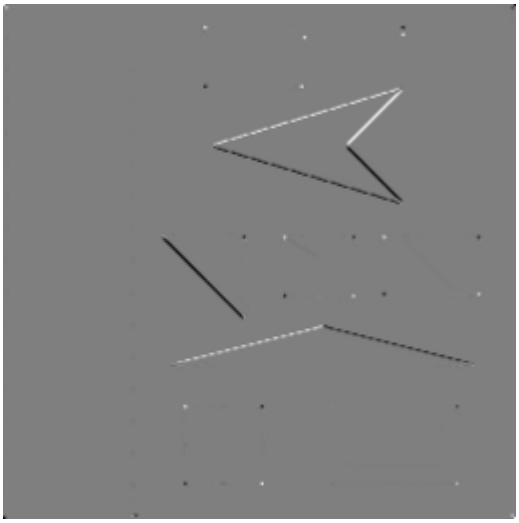
In order to better visualize corners, we can use a 3 by 3 neighborhood to compute the local maxima of $R(A_w)$. The following are the results of using the parameters:

$$\sigma_{Integration} = 1.5 \quad , \quad \sigma_{Differentiation} = 0.7 * \sigma_{Integration} \ ,$$

$$R(A_w) > threshold = 1\%\ of \max R(A_w)$$

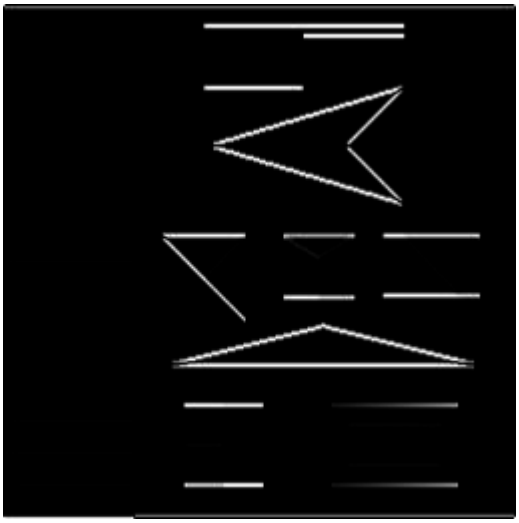


Test 1 (Original)

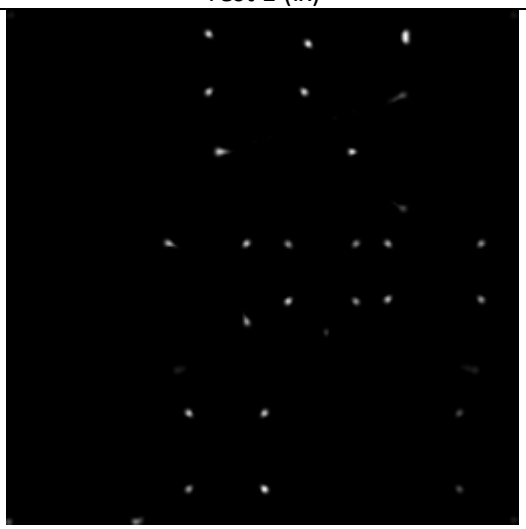


Test 1 (IxIy)

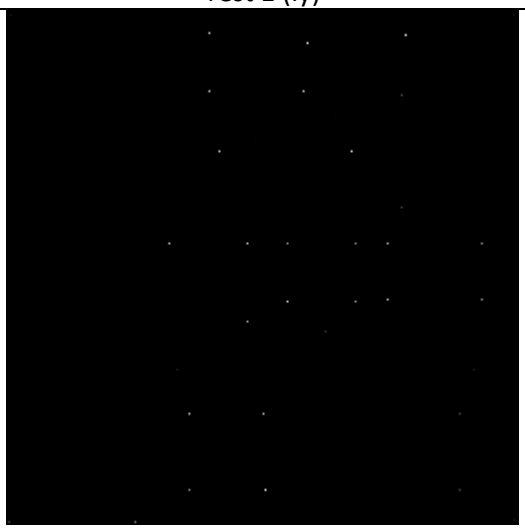
Test 1 (Ix)



Test 1 (Iy)

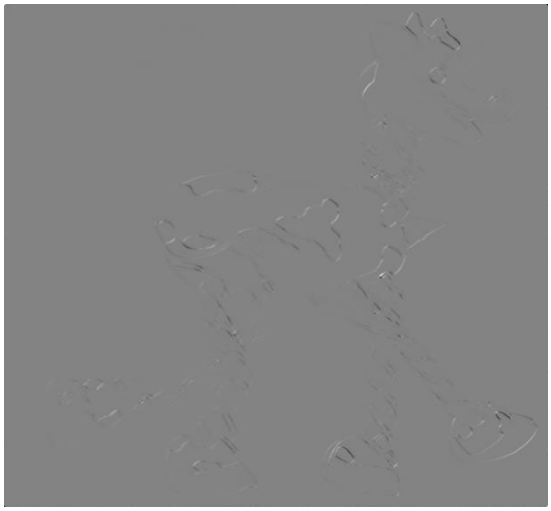


Test 1 R(Aw)

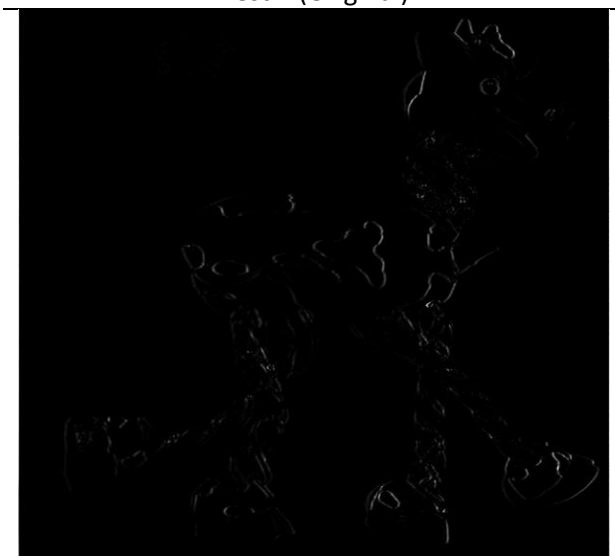


Test 1 Non-Maxima Suppressed R(Aw)

Test 2 (Original)



Test 2 (IxIy)



Test 2 (Ix)

Test 2 (Iy)

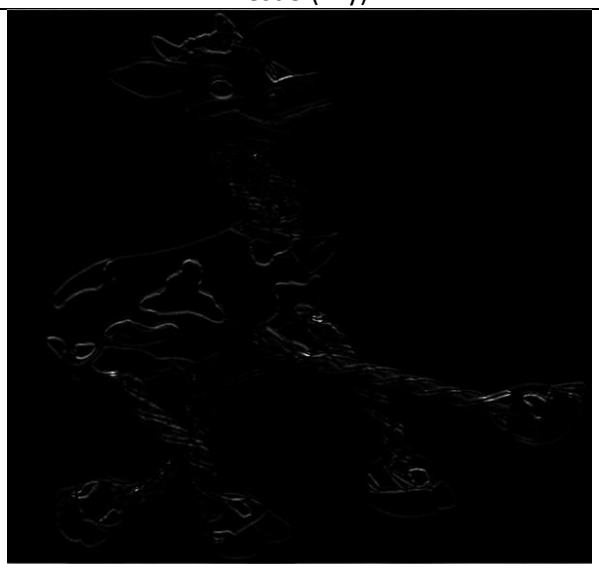
Test 2 R(Aw)

Test 2 Non-Maxima Suppressed R(Aw)

Test 3 (Original)

Test 3 (IxIy)

Test 3 (Ix)



Test 3 (Iy)

Test 3 R(Aw)

Test 3 Non-Maxima Suppressed R(Aw)

Part Two: Harris-Laplace Interest Point Detector

To implement the Harris-Laplace interest point detector, we start by generating the Gaussian scale space and the difference of Gaussian scale space. Then we apply the Harris corner detector using an integration sigma equivalent to each level on the Gaussian scale space. After the corners have been detected, we can apply non-maxima suppression to find the strongest corners. Finally, we perform a non-maxima suppression using the difference of Gaussian scale space by observing the closest 26 neighbors of the corners to reject unstable corners.

$$\sigma_0 = 1.5 \quad , \quad \sigma_{Integration} = 1.2^{level} * \sigma_0 \quad , \quad \sigma_{Differentiation} = 0.7 * \sigma_{Integration}$$

$$R(A_w) > threshold = 1500 \quad , \quad DoG\left(x, y, \sigma_{Integration}\right) > 10$$

Boat 1

R(Aw), sigma = 2.16, M = 2, Interest Points = 90,737

Non-Maxima Suppression, sigma = 2.16. M = 2, Interest Points = 1381

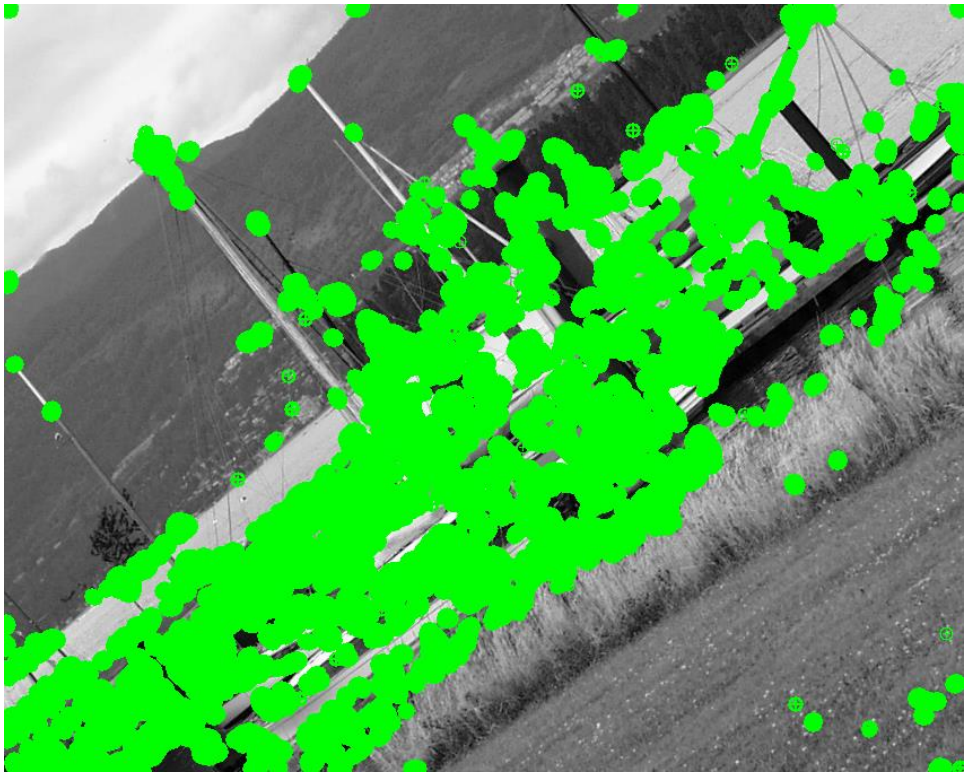
Scale Non-Maxima Suppression, sigma = 2.16, M = 2, Interest Points = 35

R(Aw), sigma = 4.4789, M = 6, Interest Points = 175,982

Non-Maxima Suppression, sigma = 4.4789, M = 6, Interest Points = 274

Scale Non-Maxima Suppression, sigma = 4.4789, M = 6, Interest Points = 5

Boat 3



R(Aw), sigma = 2.952, M = 3, Interest Points = 86,648

Non-Maxima Suppression, sigma = 2.952, M = 3, Interest Points = 863

Scale Non-Maxima Suppression, sigma = 2.952, M = 3, Interest Points = 3

R(Aw), sigma = 4.47898, M = 6, Interest Points = 124,446

Non-Maxima Suppression, sigma = 4.47898, M = 6, Interest Points = 249

Scale Non-Maxima Suppression, sigma = 4.47898, M = 6, Interest Points = 5

Boat 5

R(Aw), sigma = 2.16, M = 2, Interest Points = 45,358

Non-Maxima Suppression, sigma = 2.16, M = 2, Interest Points = 627

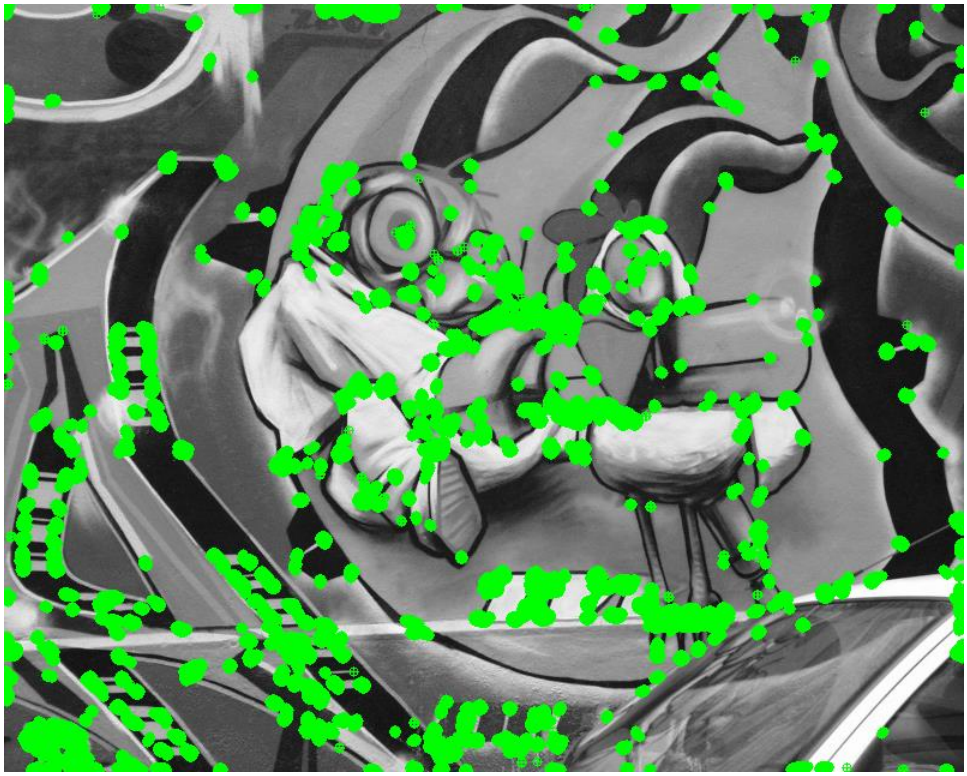
Scale Non-Maxima Suppression, sigma = 2.16, M = 2, Interest Points = 15

R(Aw), sigma = 3.73248, M = 5, Interest Points = 73,936

Non-Maxima Suppression, sigma = 3.73248, M = 5, Interest Points = 229

Scale Non-Maxima Suppression, sigma = 3.73248, M = 5, Interest Points = 11

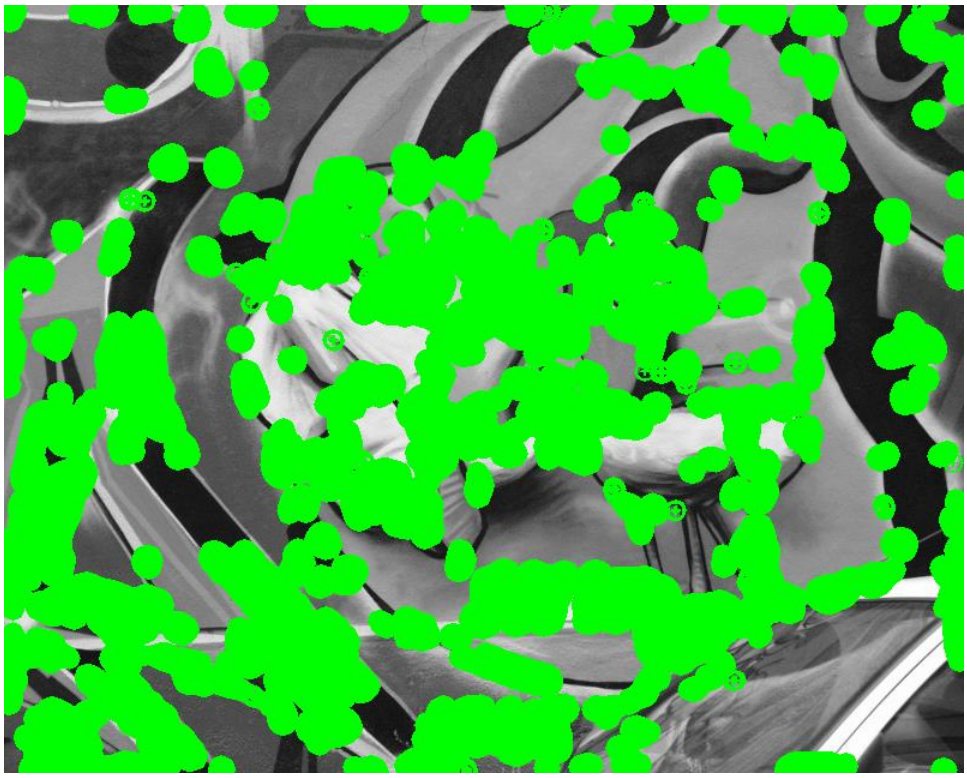
Graf 1



R(Aw), sigma = 1.8, M = 1, Interest Points = 22,128

Non-Maxima Suppression, sigma = 1.8, M = 1, Interest Points = 706

Scale Non-Maxima Suppression, sigma = 1.8, M = 1, Interest Points = 4

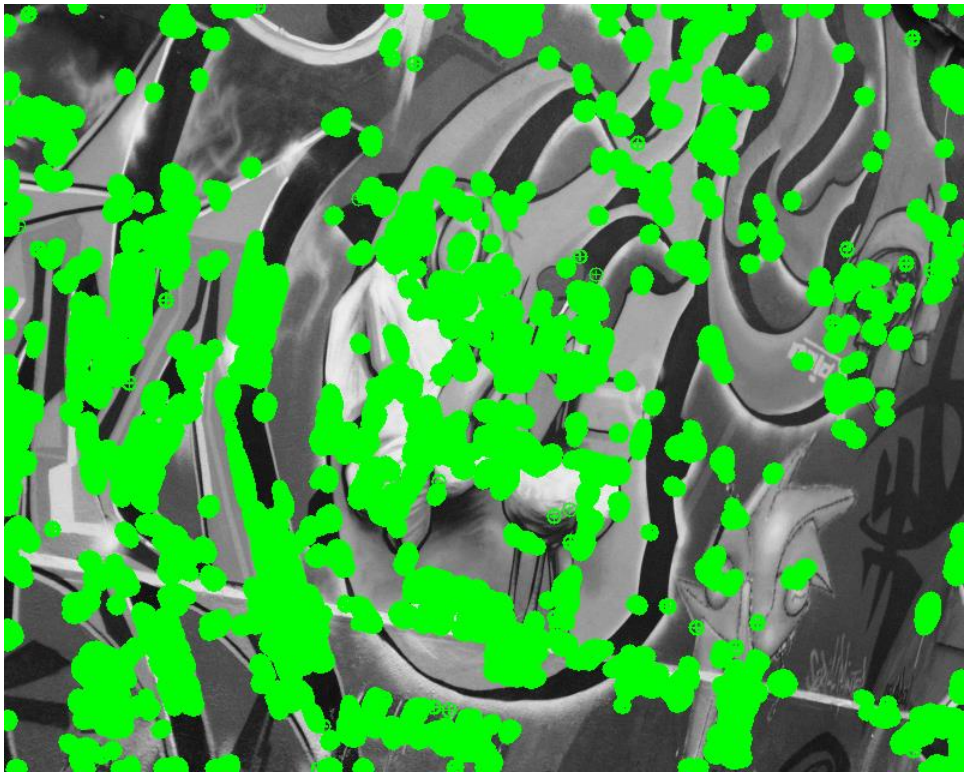


R(Aw), sigma = 3.73248, M = 5, Interest Points = 96,162

Non-Maxima Suppression, sigma = 3.73248, M = 5, Interest Points = 329

Scale Non-Maxima Suppression, sigma = 3.73248, M = 5, Interest Points = 11

Graf 3



R(Aw), sigma = 2.592, M = 3, Interest Points = 62,221

Non-Maxima Suppression, sigma = 2.592, M = 3, Interest Points = 773

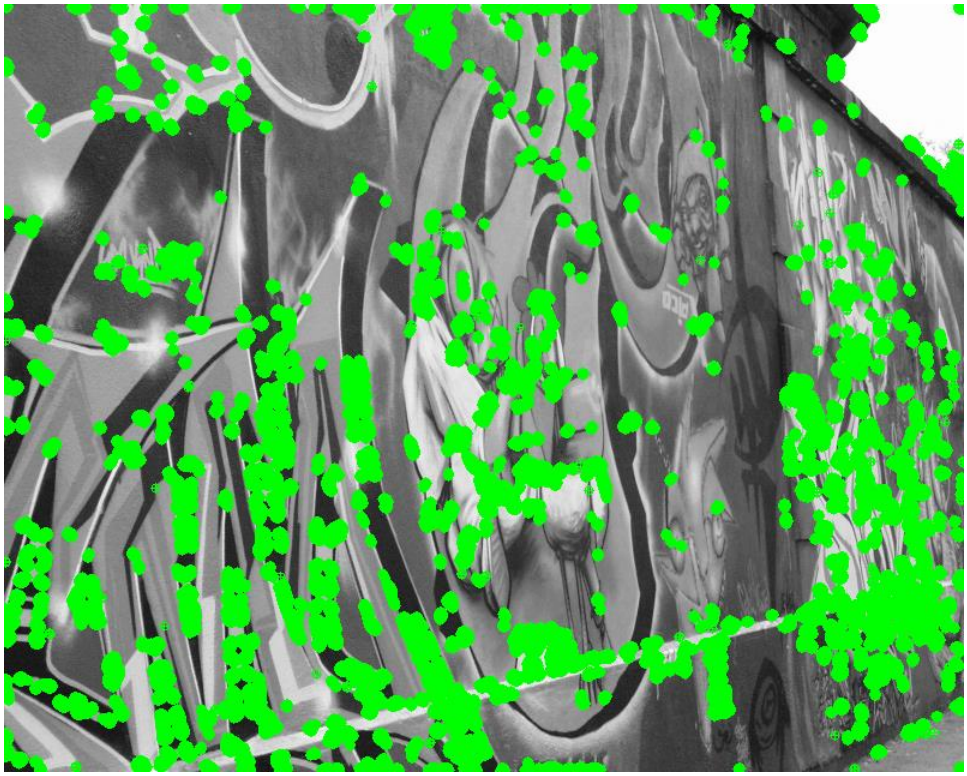
Scale Non-Maxima Suppression, sigma = 2.592, M = 3, Interest Points = 1

R(Aw), sigma = 6.44973, M = 8, Interest Points = 268,756

Non-Maxima Suppression, sigma = 6.44973, M = 8, Interest Points = 10

Scale Non-Maxima Suppression, sigma = 6.44973, M = 8, Interest Points = 8

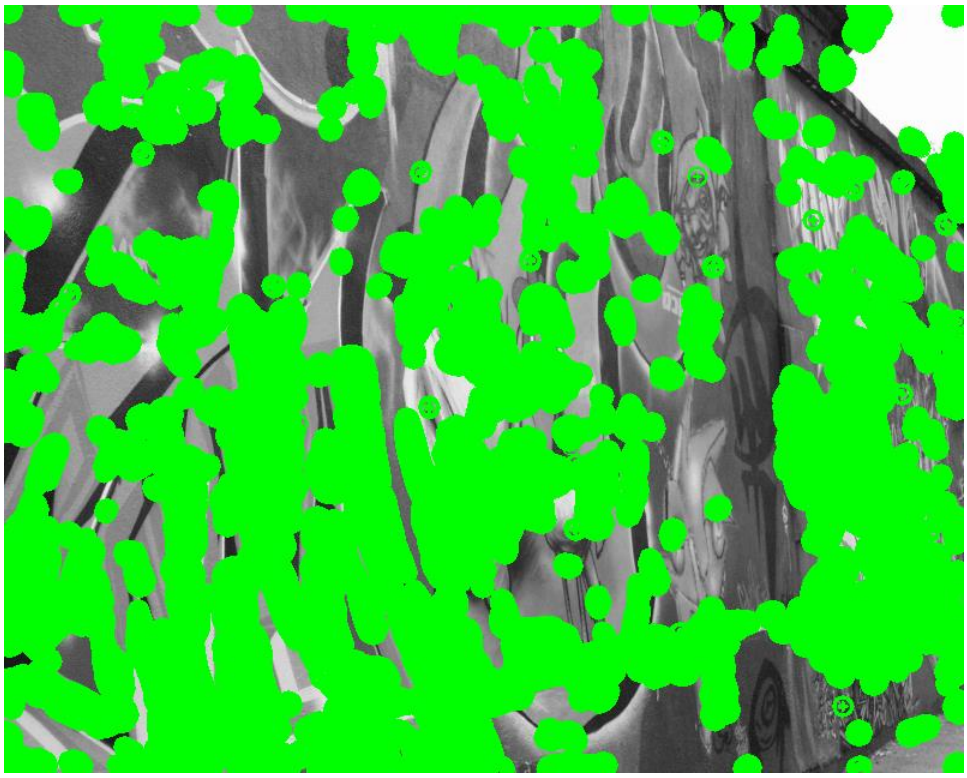
Graf 5



R(Aw), sigma = 1.8, M = 1, Interest Points = 37,142

Non-Maxima Suppression, sigma = 1.8, M = 1, Interest Points = 1132

Scale Non-Maxima Suppression, sigma = 1.8, M = 1, Interest Points = 11

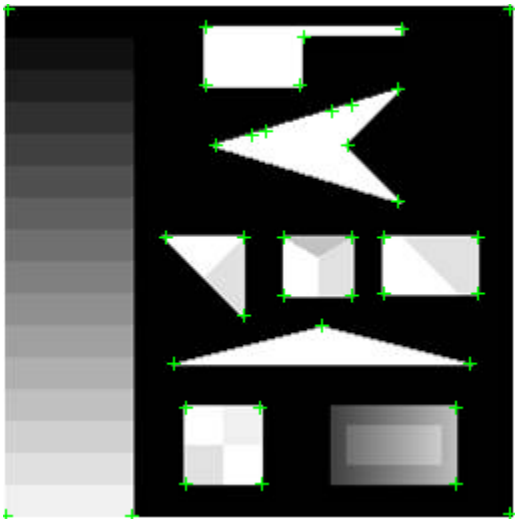


R(Aw), sigma = 3.73248, M = 5, Interest Points = 136,373

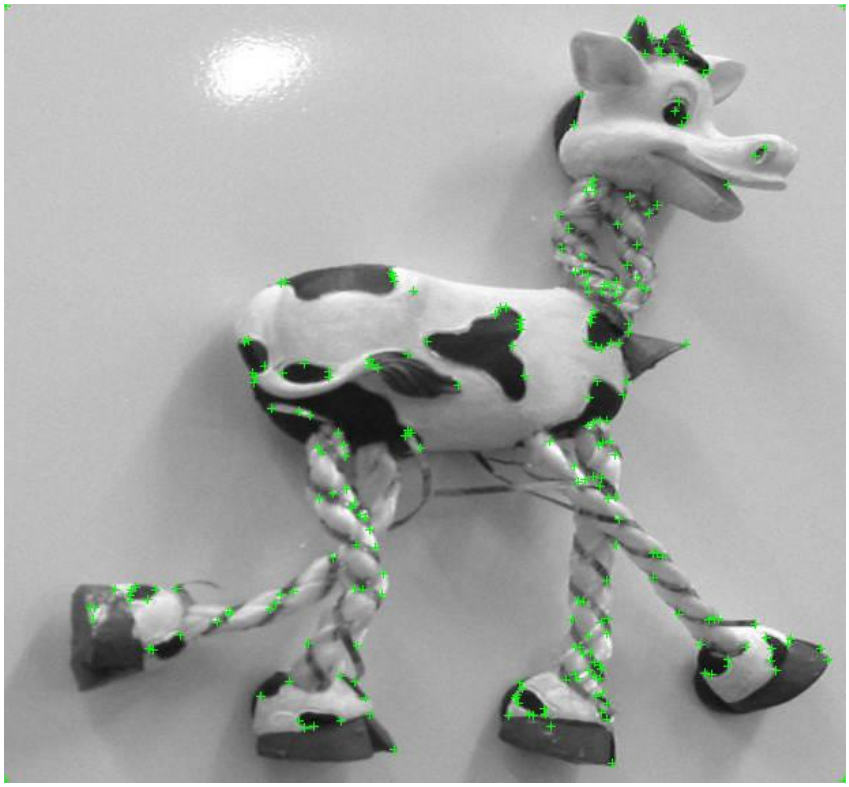
Non-Maxima Suppression, sigma = 3.73248, M = 5, Interest Points = 377

Scale Non-Maxima Suppression, sigma = 3.73248, M = 5, Interest Points = 11

Part One Overlays:



Test 1 Overlay



Test 2 Overlay

Test 3 Overlay

Discussion of Results

For the first part of the project, the corner detector behaved as expected compared to the images on the lecture slides. The gradient images matched up as well as the final images with the overlay. For the second part of the project, the interest point detector did a good job with finding potential interest points and using the DoG scale to filter for the good ones.

The instructions for 2(e) stated that the interest points from the previous question (corners that were applied non-maxima suppression in the same scale) should be iterated over along with successive DoG layers to find stable interest points. When I tried to use those corners, very few (often zero) would match up with points in the DoG layers which resulted in little to no interest points being detected. To resolve this issue, I used the interest points from before the non-maxima suppression. These corners gave much better results (visually) than the previous version. It is possible that this was the incorrect method, but it was the only one that I found to give reasonable results.