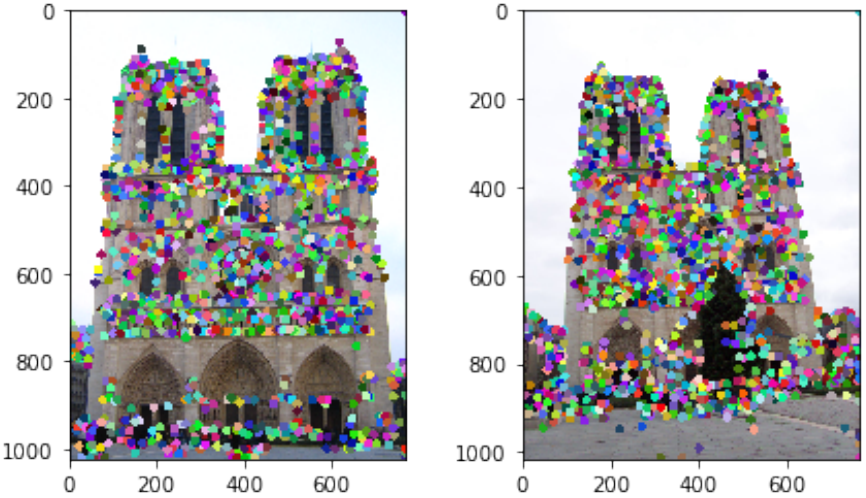
CS 4476 PS3

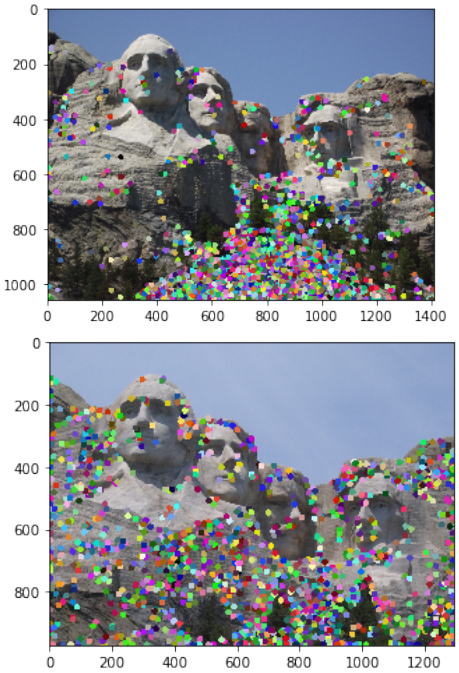
<Jinghong Peng>
<jpeng78@gatech.edu>
<jpeng78>
<903568613>

1.1: Harris Corner Detector

<insert visualization of Notre Dame interest points from proj3.ipynb here>

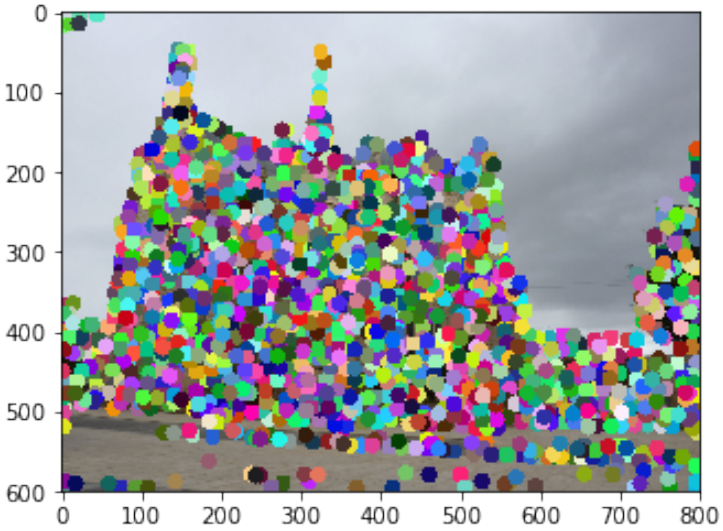


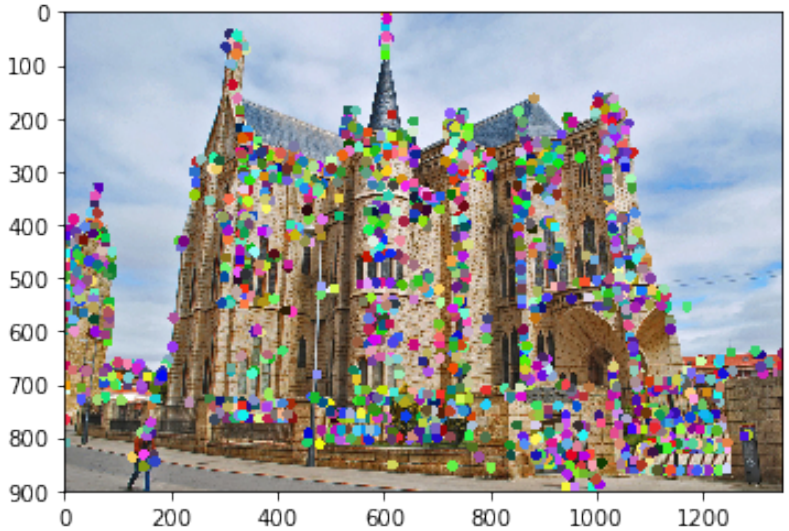
< insert visualization of Rushmore interest points from proj3.ipynb here >



1.1: Harris Corner Detector

< insert visualization of Gaudi interest points
from proj3.ipynb here >





1.1: Harris Corner Detector

- Briefly describe how the Harris corner detector works.

We take the SSD(Sum squared difference) of pixel values before and after shifting the window in all 8 directions, and then find the the largest SSD value. We find the eigenvalues of M, which is $\lambda 1$ and $\lambda 2$. R is calculated by detM – alpha(traceM)^2. When R is large, it means $\lambda 1$ and $\lambda 2$ are both large, and thus the region is a corner.

- What does the second_moments() helper function do?

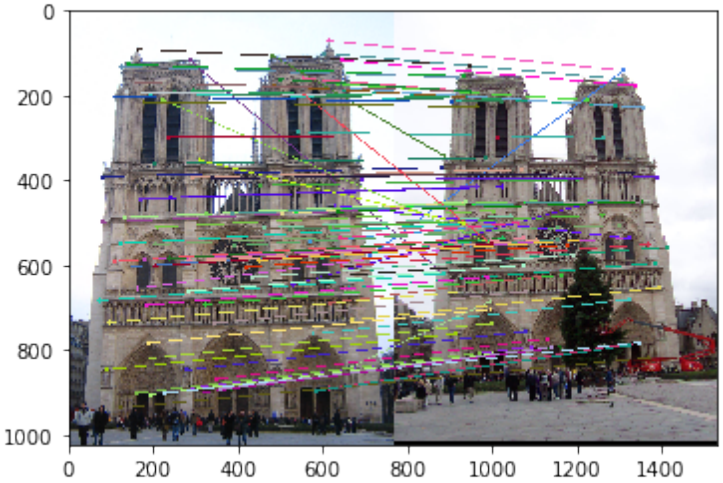
We convolve with a gaussian kernel to get sx2, sy2, sxsy, essential parts of the matrix. These values are used by corner_response()

- What does the corner_response() helper function do?

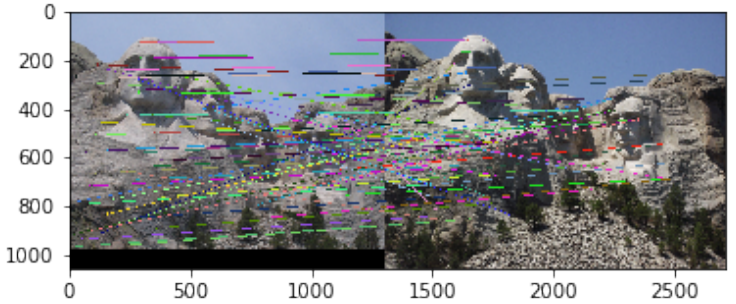
Corner_response uses values calculated from second_moments to calculated the R score. R is used to determine if it is a corner or not.

1.3: Feature Matching

<insert feature matching visualization of Notre Dame from proj3.ipynb>

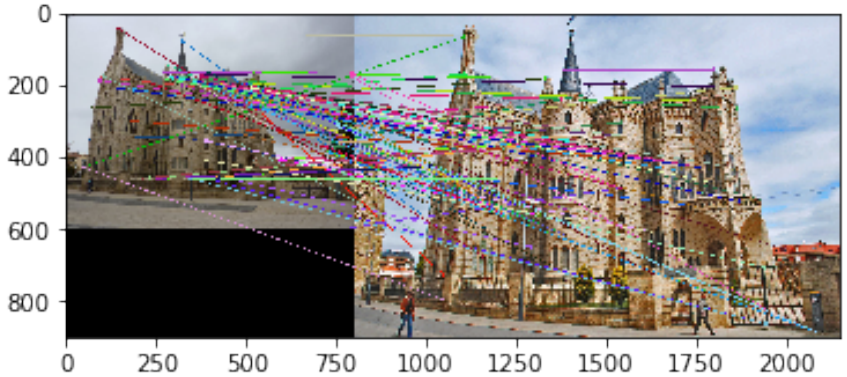


<insert feature matching visualization of Rushmore from proj3.ipynb >



1.3: Feature Matching

<insert feature matching visualization of Gaudi from proj3.ipynb >

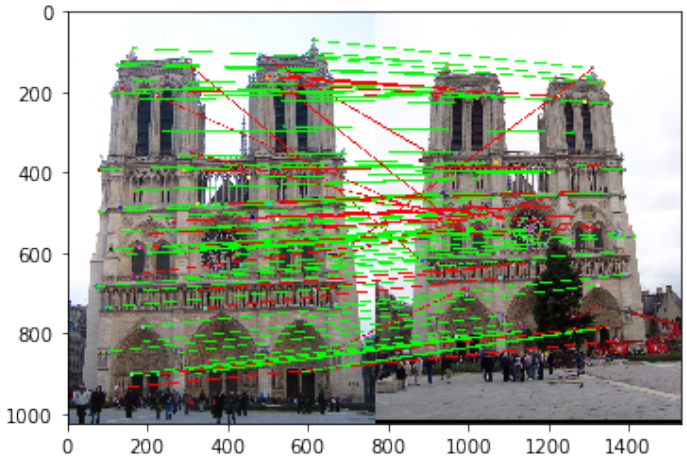


<Describe your implementation of feature matching.>

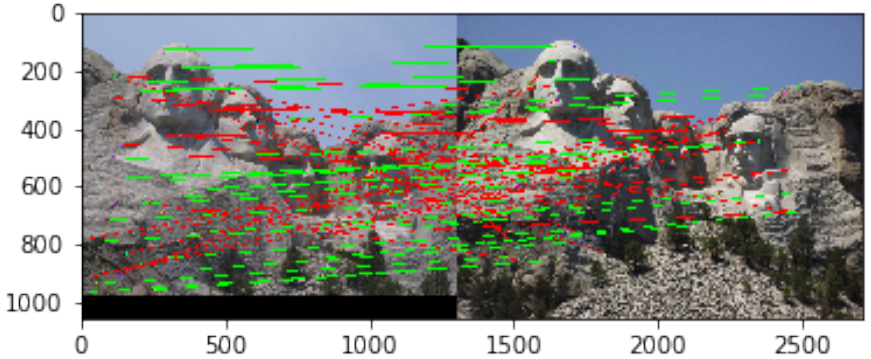
I first use computer_feature_distance to get the distances from each feature in features1 to each feature in features2. Then I iterate through each feature, and get the ratio of closest distance/2nd closest distance. I set the threshold to 0.95. And build an array of result using the filtered matches.

Results: Ground Truth Comparison

<Insert visualization of ground truth comparison with Notre Dame from proj3.ipynb here>

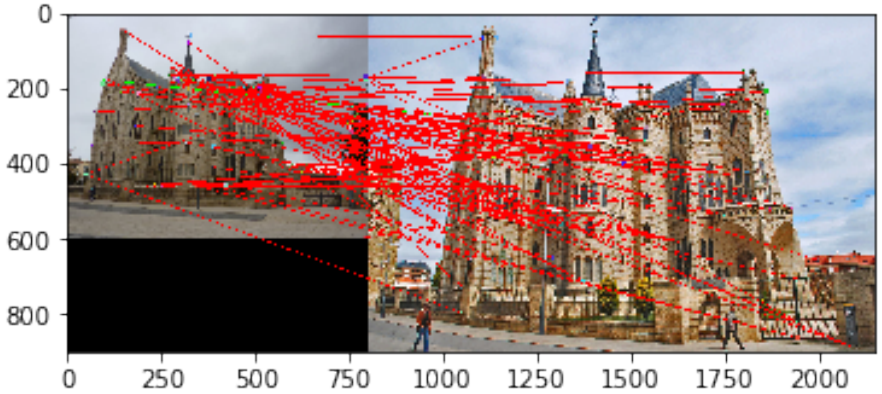


<Insert visualization of ground truth comparison with Rushmore from proj3.ipynb here>



Results: Ground Truth Comparison

<Insert visualization of ground truth comparison with Gaudi from proj3.ipynb here>



<Insert numerical performances on each image pair here. Also discuss what happens when you change the 4x4 subgrid to 2x2, 5x5, 7x7, 15x15 etc?>

Image1: You found 98/100 required matches
Accuracy = 0.670000

Image2: You found 92/100 required matches
Accuracy = 0.350000

Image3: You found 100/100 required matches
Accuracy = 0.010000

1.4(a): Hyperparameter Tuning part 1 [Extra credit]

<Insert images of the ground truth correspondence and their corresponding accuracies for varying sigma in the second moments [3, 6, 10, 30] >

When changing the values for large sigma (>20), why are the accuracies generally the same?

1.4(a): Hyperparameter Tuning part 2 [Extra credit]

<Insert images of the ground truth correspondence and their corresponding accuracies for varying feature width in the SIFT [8, 16, 24, 32] >

What is the significance of changing the feature width in SIFT?

1.4(c): Accelerated Matching [Extra credit]

<Insert Runtime/Accuracy of your faster matching implementation. What did you try and why is it faster?>