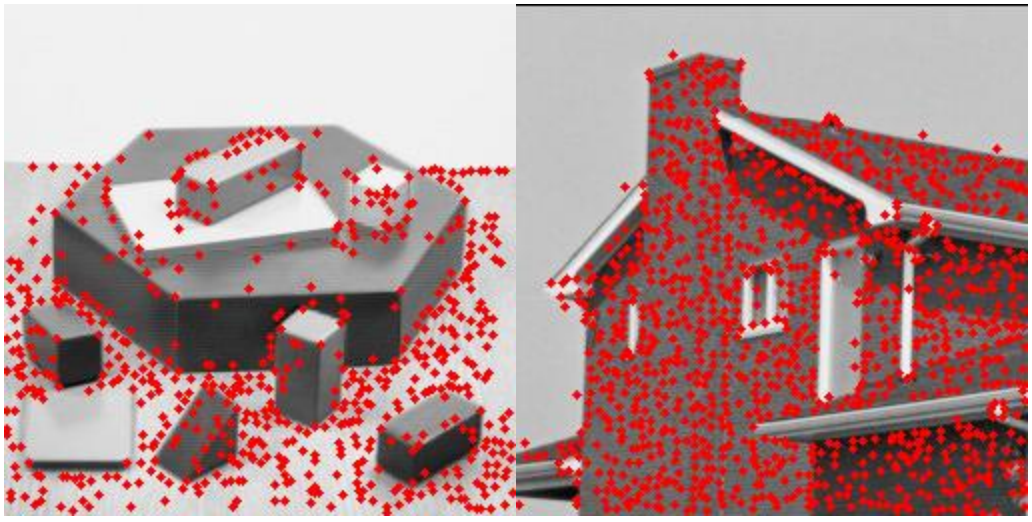
CV Lab 2 Report – Omer Alkaya

Implementation details for extract_harris.py:

As requested in the TODOs, I have convolved my image with the Sobel kernel to obtain the image derivatives for them to be used in the determinant and trace calculation for the matrix M. After obtaining the Harris response, I have simply done thresholding as requested and done non-maxima suppression using a simple loop within which each pixel is compared to the maximum filter. Finally the results are stacked and returned as “corners”.



Implementation details for extract_descriptors.py:

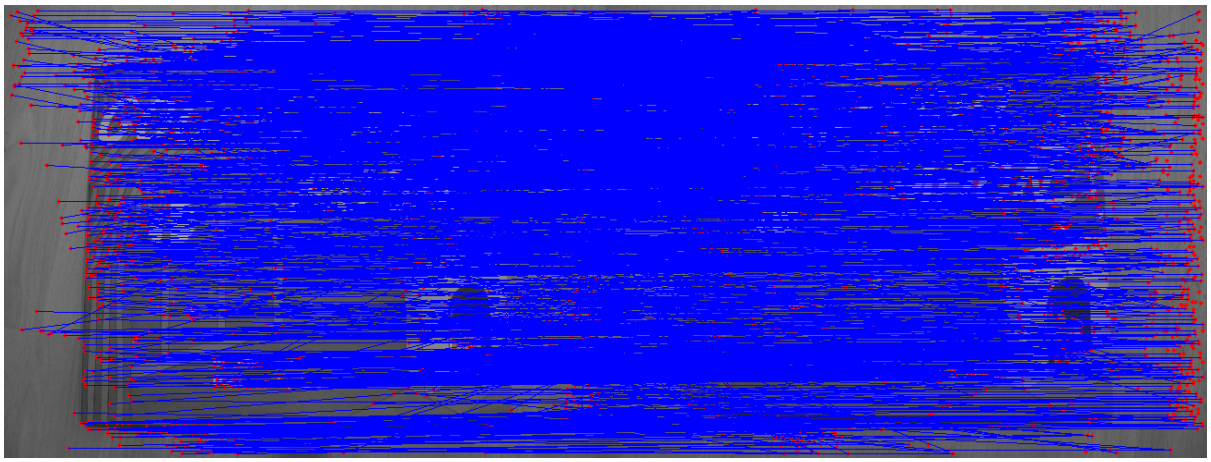
To filter the keypoints, I have used a simple mask derived from chaining comparisons between the rows and the columns of the keypoints against the image borders. Finally, this mask is applied to the keypoints to obtain the list of safe keypoints to use down the line.

Implementation details for match _descriptors.py:

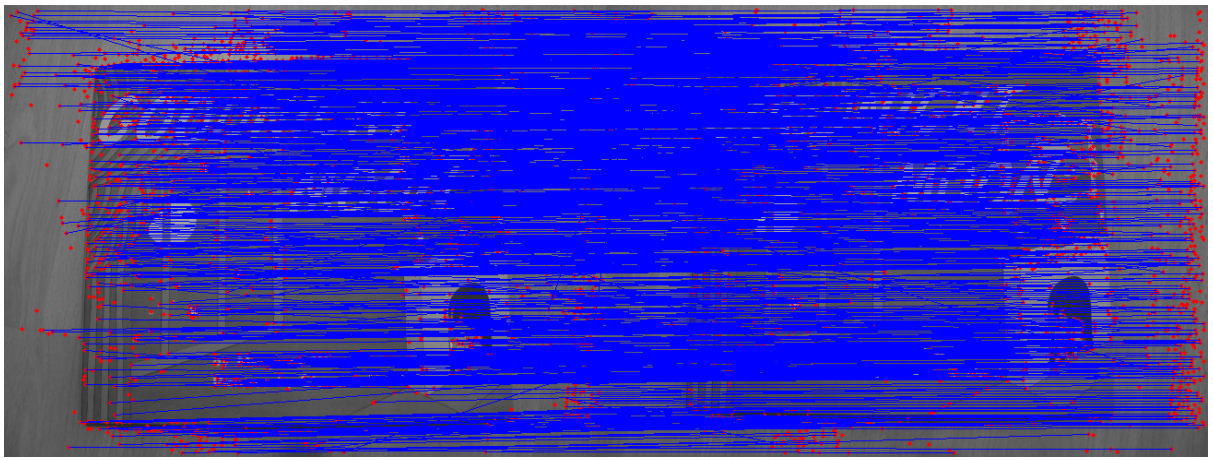
To calculate the squared distances with numpy I have resorted to broadcasting followed with a basic numpy sum statement. For the implementation of the different descriptor matching strategies, I have mostly used numpy array operations such as argmin, vstack, where and partition to index within

arrays. I have annotated my code for further descriptions of each step. The resulting matchings are as follows:

One-Way:



Mutual:



Ratio: