CS 537: Assignment 2:

Behnam Saeedi
(Saeedib@oregonstate.edu)
Due Feb 15th 12:00pm

1 Types of Key-Points Detected

To ensure that the results that we are receiving are based on correct Assignment 1 solutions, we used the solution provided by the TA. for part 1. This means we are detecting SIFT based keypoint detection for all images. However, one important thing to note is that this model was originally trained on corner detection for monuments and buildings and this might cause some unwanted consequences.

```
sift = cv2.xfeatures2d.SIFT_create(30)
```

Each keypoint is processed through the weights of the CNN for computing the description of each point. These points will be used for one to one and many to many matching per image. this means we have to create 2 repositories. One for quarries containing 35 images with 30 points each and the other is images containing 140 images with 30 points each. Each one of these points have 128 features.

2 CNN Architecture and Hyper-Parameters

The CNN selected is also from homework one's solution. We are using the descriptor that was designed in solution key to ensure that any mistakes made in Assignment 1 will not affect the outcome of the Assignment 2. The hyper parameters selected were:

- Optimizer:
  - Momentum: 0.9 was the selected value for momentum
  - Dampening: 0.9
- 7 layers of convolution
- each of the 7 layers have batch normalization
- all activation functions are ReLu
- No bias, no pooling

3 Tables and Plots

The following table and graphs illustrates the outcome of the recall and precision computation for each stage. There are two sets of graphs provided. The first set shows the two computed values as functions of the other, and the second set of graphs shows both of those computed accuracy as a function of k:

K	Recall	Precision
1	0.10714285714285714	0.42857142857142855
2	0.2642857142857143	0.5285714285714286
3	0.4928571428571429	0.6571428571428571
4	0.75	0.75

Table 1
One to One matching

K	Recall	Precision
1	0.04285714285714286	0.17142857142857143
2	0.10714285714285714	0.21428571428571427
3	0.20714285714285716	0.27619047619047615
4	0.2714285714285714	0.2714285714285714

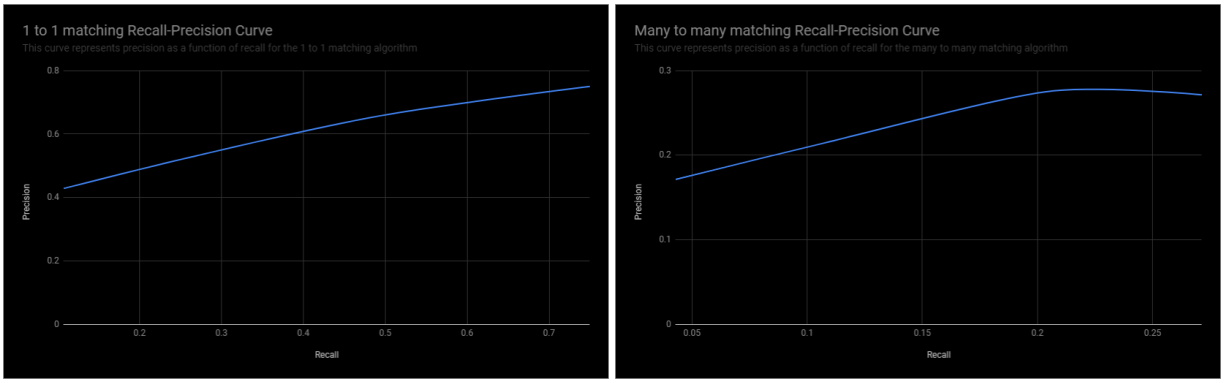

Figure 1. Precision as a function of recall

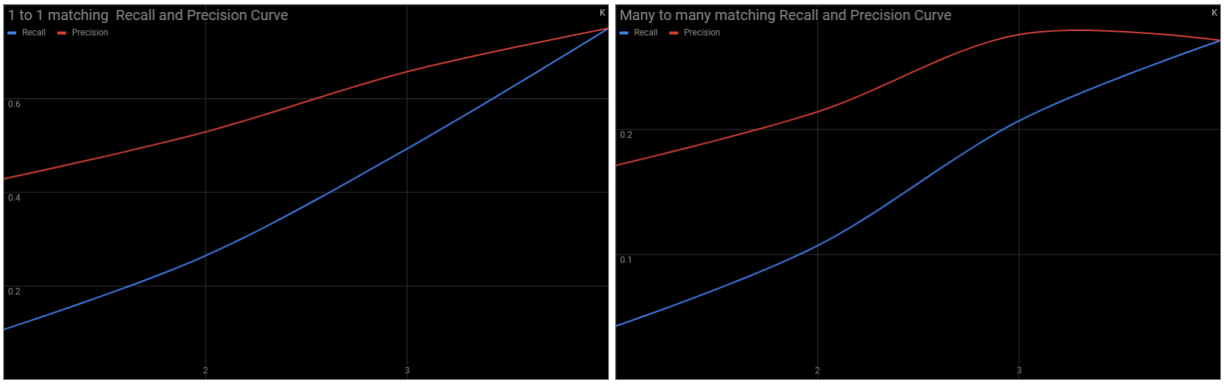

Figure 2. Precision and recall as a function of k