Computer Vision Homework 1

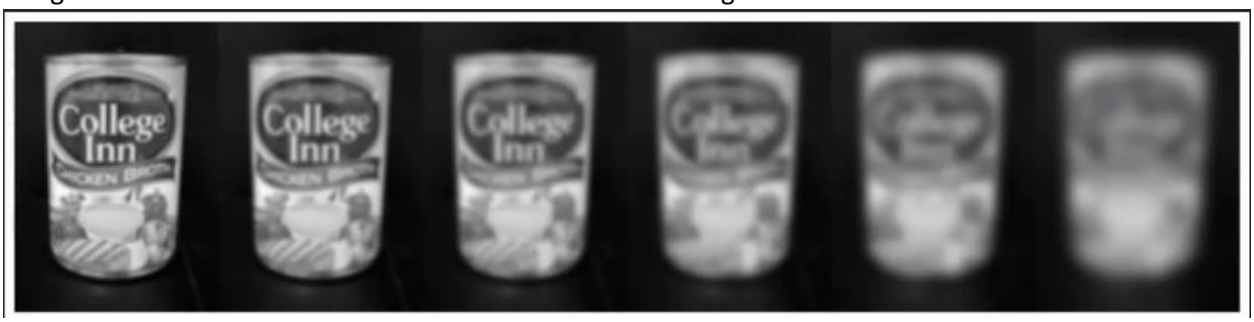
Yuval Goshen 205810179

Adi Levy 203695002

Part 1

1.2

The shape of the gaussian pyramid is $(levels, image\_height, image\_width)$.
We get a 3D tensor because there are $levels$ number of images.



1.3

We have tested our DOG pyramid on the given gaussian pyramid:

1.4

We used the formula

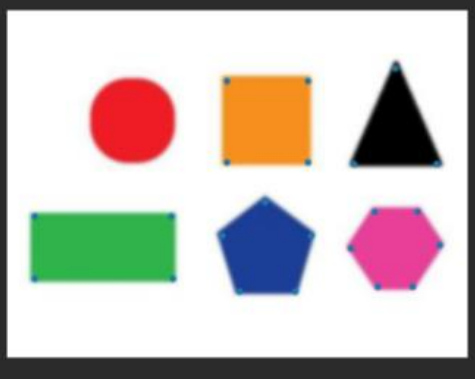
$$R = \frac{(D_{XX} + D_{YY})^2}{D_{XX}D_{YY} - D_{XY}^2 + \epsilon}$$

Instead of building the hessian matrix.

1.6

We go perfect result on the sanity check:

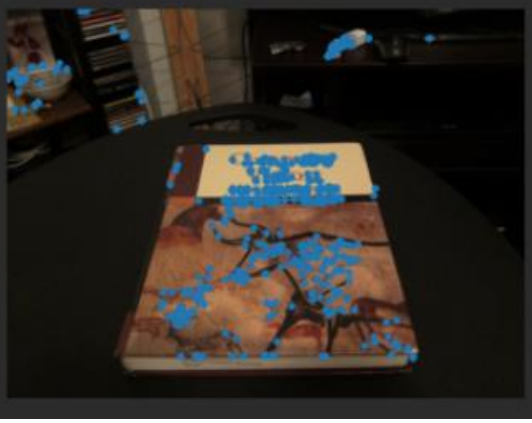
```
36 plot_corners('data/sanitycheck.jpg')
```



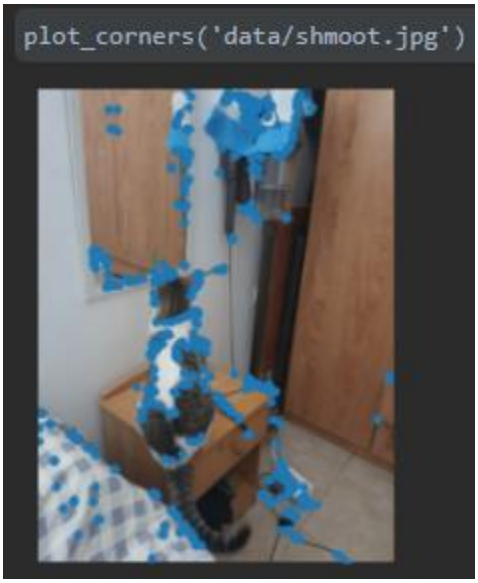
And we tried some more images:

```
plot_corners('data/model_chickenbroth.jpg')
```


```
plot_corners('data/pf_desk.jpg')
```



And on a picture we took:



The results seems reasonable- it detected key points such as the closet door handles, some corners, and some points on the cat pattern.

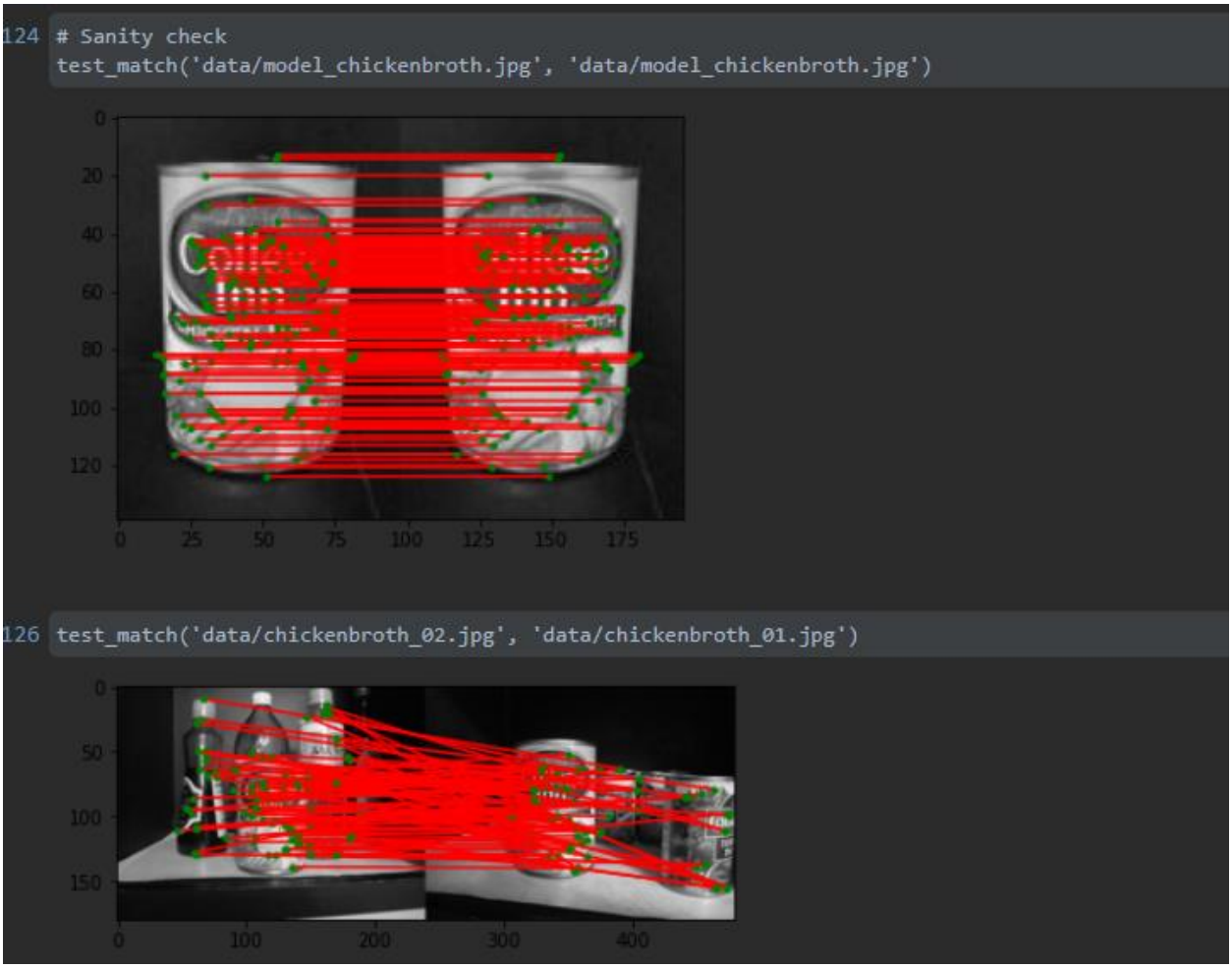
May be there are too many features, while some of them are not good corners. That might be improved by changing to higher thresholds.

Part 2

2.4

We performed sanity check for an image with itself and same edges were matched.

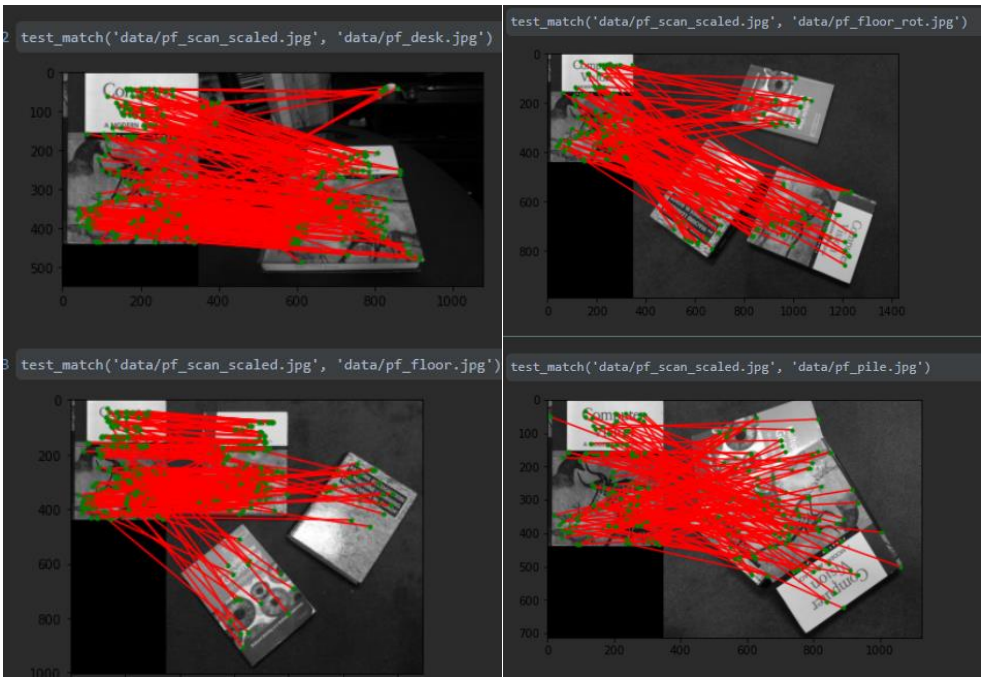
In addition, we have chosen a chicken broth image with itself, some of the corners were matched well, but some corners that did not exist in one picture matched to other corners in the second one.
The ratio in the call to brief Match might be too low and it creates fake matches.

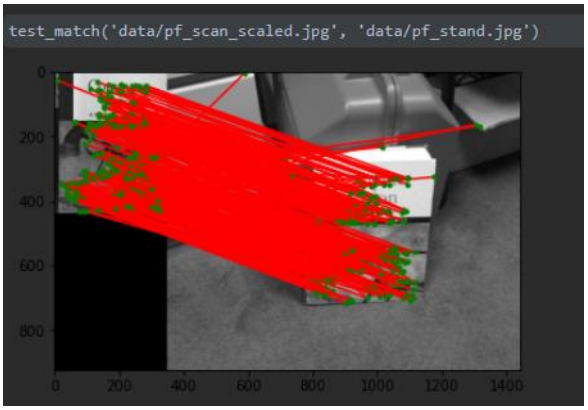


We also tried to match incline images:

But there are too many descriptors matching to see. That might be because there are many corners in an image of a city, and corners of different buildings might be similar.

Book matches:



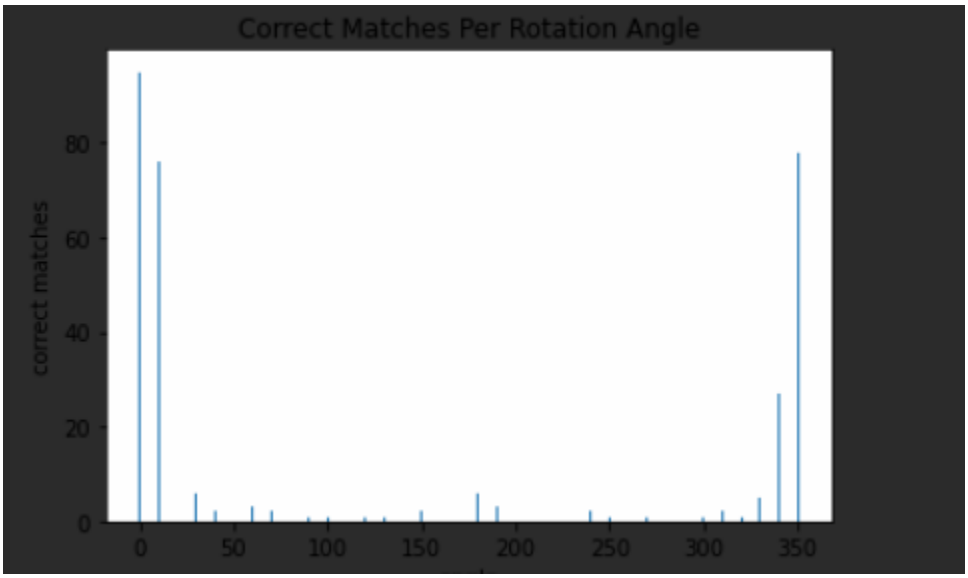


1. We can see that when the book is rotated in the image, we lose some descriptors (because we did not rotate the patches). But in images where the book is in a similar orientation, we get some good matches
2. We can see that when other books are added to the image, the same thing happens like in the chicken broth case, fake descriptors matches are created.

2.5

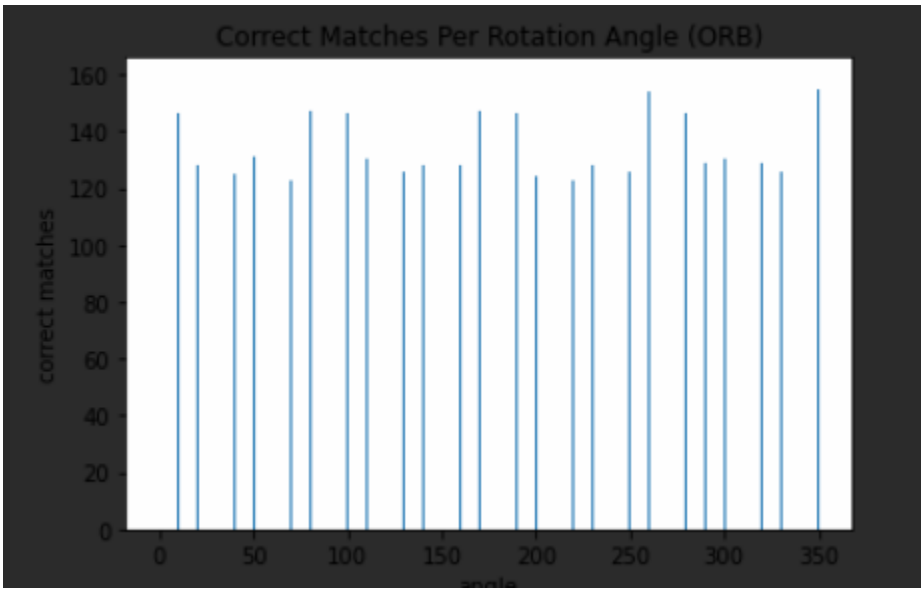
Regarding illumination: Brief is quite invariant to illuminations since it looks for points with gradients but doesn't consider the size of the gradient, only direction. In addition, we transform the image to grayscale, therefore light in different color doesn't matter.

If we want our brief descriptor to be more invariant to scale, we should use the down sampling portion of the SIFT algorithm, therefore including the same data when downscaling.



We can see, that as we hypothesized earlier, rotating the image significantly harms the performance of our brief descriptors matching. This is probably because the transform we use from a patch to a test pattern vector, changes the vector significantly when rotating the patch. Therefore the descriptors are different.

2.6



Running with orb, we got almost the same result for all angles, which means that it is rotation invariant.

The difference from brief is that before transforming the patch to a test vector, ORB rotates the patch so that the highest gradient in the patch will be oriented in a certain direction (which is always the same). Therefore creating rotation invariance when detecting key points.