BITS F464 (ML) Class Project Report

ASHISH KUMAR, 2017B1A70854P

23 February 2021

1 Introduction

1.1 Matching strategies

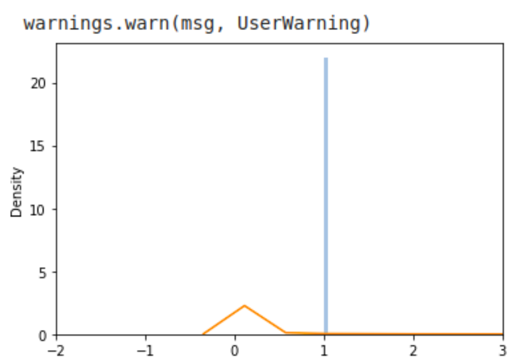
- KNN algorithm was used to match the descriptors extracted by SIFT algorithm
- The K parameter was kept at 2, as keeping the K parameter higher was observed to be leading to intermixing of the scores of imposters and matches

2 Results and Analysis

2.1 Histogram

2.1.1 Match is exact same image match

- When the match is defined as the keypoint knnMatch for same images, we obtain the genuine-imposter histogram.



We can see that the genuine match = blue is separated from the imposter match = orange. Also, as this definition of match is not quite accurate we use the next definition of match for further analysis.

The score in this case was defined as.

$$score = \frac{len(good)}{number_keypoints}$$

where good is a list containing all manhattan distances which are atleast 30% closer than the next neighboring keypoint. $number_keypoints$ is the minimum of the number of keypoints in image1 and image2.

- When the match is defined as the keypoint knnMatch for same person, we obtain the following genuine fig. 1 imposter fig. 2 histogram. The match score here contains all the images beginning with the same subject id as follows.
  - $1_P 1_S 1_1.jpg - 1_P 1_S 1_1.jpg$
  - $1_P 1_S 1_1.jpg - 1_P 1_S 1_2.jpg$

- ...
- $1_P 1_S 1_1.jpg - 1_P 1_S 2_5.jpg$
- ...
- $1_P 2_S 1_4.jpg - 1_P 2_S 2_1.jpg$
- ...
- $1_P 1_S 1_1.jpg - 1_P 1_S 1_1.jpg$

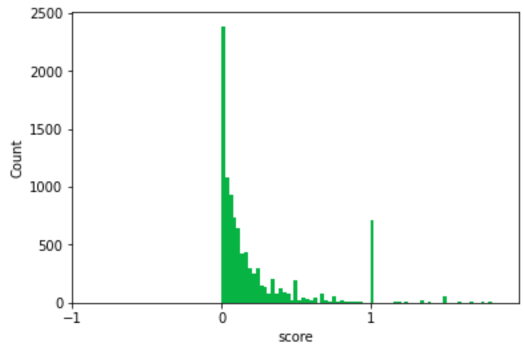


Figure 1: Genuine match histogram

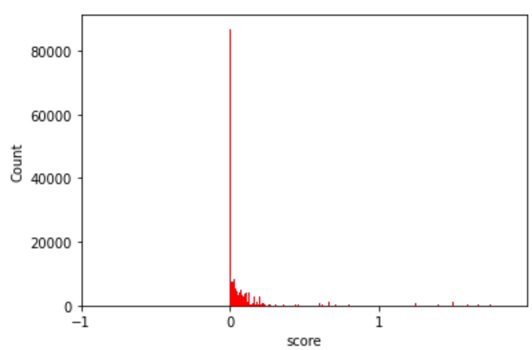


Figure 2: Imposter match histogram

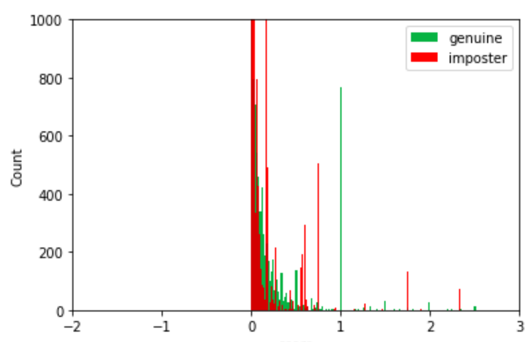


Figure 3: Genuine-Imposter histogram overlap

- We can observe that due to a larger number of imposters match compared to genuine matches. The absolute bar heights for imposter matches seem to dwarf the bar heights of genuine matches.

  But,when we draw the density histograms for genuine and imposter matches, we observe genuine density histogram fig. 4 and imposter density histogram fig. 5.

  In this we can observe that the density of genuine matches is higher towards the upper spectrum of score compared to imposter matches.

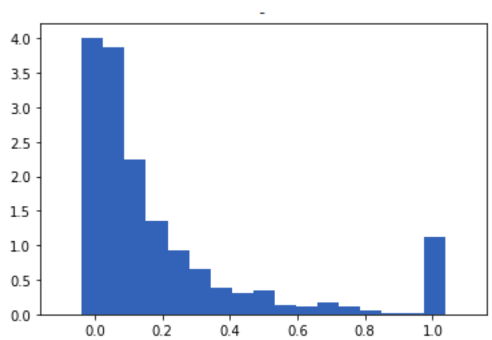


Figure 4: Genuine density histogram

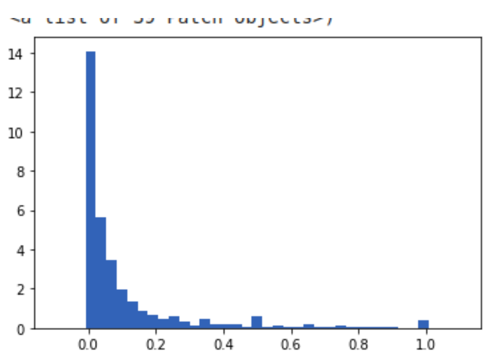


Figure 5: Imposter density histogram

2.2 SIFT algorithm working

The SIFT algorithm stands for Scale Invariant Feature Transform.

This algorithm in short creates features which are independent of rotation and the scale of the image i.e. matching two images of the same object but of different scale/size will give us a correct match.

The algorithm in itself consists of the following 5 steps -

2.2.1 Scale-space Extrema Detection

Laplace of Gaussian (LoG) is a popular edge detection algorithm.

The edges in the image are detected using this method by first blurring out the image, then applying LoG to detect the edges which are a great keypoint. We can also calculate the Difference of Gaussian (DoG)fig. 6 between the gaussian of the images to generate the keypoints consisting of edges and corners.

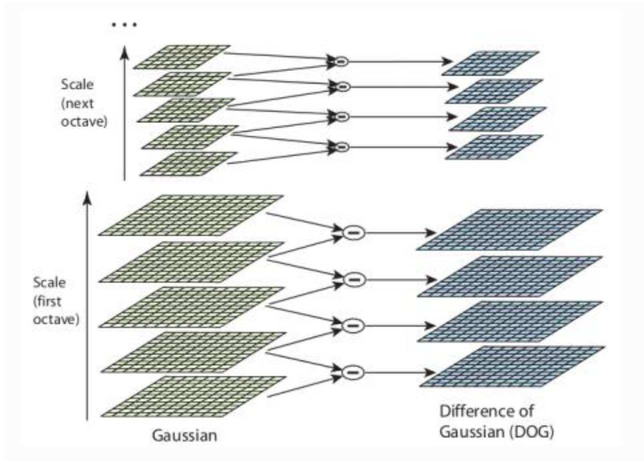


Figure 6: Difference of Gaussian for 2 images ate different scales

2.2.2 Keypoint Localization

In this part of the algorithm, the various extrema are localized and mapped.

The various potential keypoints are also normalized w.r.t their intensity as DoG algorithm has a higher response for edge keypoints compared to LoG.

2.2.3 Orientation Assignment

An orientation is assigned to the image during this part of the algorithm to assign invariance to the image against rotation by creating histograms

2.2.4 Keypoint descriptors

Now keypoint descriptor is created. A 16x16 neighbourhood around the keypoint is taken. It is devided into 16 sub-blocks of 4x4 size. For each sub-block, 8 bin orientation histogram is created. So a total of 128 bin values are available. It is represented as a vector to form keypoint descriptor.

2.2.5 Keypoint Matching

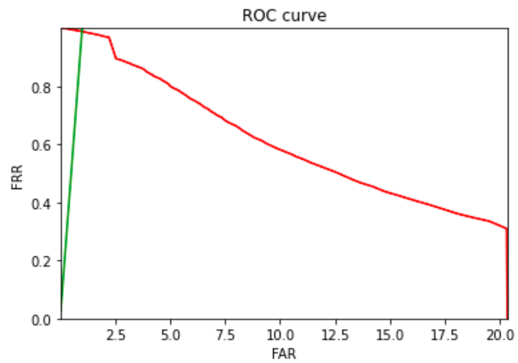
Now keypoints of an image are matched to the keypoints of the other image and scores are assigned.

2.3 ROC curve

An ROC curve is obtained by plotting FRR vs FAR, by varying the decision threshold. The area under the curve gives the error rate of the system. A system having less area under the curve is better at classification.

To plot the ROC, I generated a list of 100000 points b/w the minimum and maximum of the FRR or FAR values, whichever covers the range of both. This was done to incorporate maximum number of FAR and FRRpoints in the graph to reduce extrapolation.

Then I calculated the FRR and FAR at each of these thresholds and plotted using matplotlib. The resultant curve was obtained. fig. 7



Intersection coordinates: (0.9934115625493604, 0.9934115625493604)

Figure 7: ROC in red, $x = y$ in green

We can observe that our model has a high error (area under the curve). I think that this may be due to -

- the images being taken in different light environments in different sessions. Ex -
  - 1_P1_S2_5.jpg and 1_P2_S1_1.jpg.
  - 6_P1_S2_2.jpg and 6_P2_S1_1.jpg

  The solution might be to use artificial lights to maintain a uniform environment arount commercial system and remove any shadows which might give an impression of folds in the skin which might be classified as edges by DoG algorithm.
- hair covering the ROI. Ex -
  - 3_P2_S1_1.jpg
  - 8_P1_S1_3.jpg

  The solution might be to hold hair above forehead in an automated commercial system.
- No wrinkles. Ex- 9_P1_S1_3.jpg. The solution might be to raise the brows to make wrinkles by voluntar muscle movement.

  All these small problems with the images, amount to large errors as our sample size if quite small.

2.4 Equal Error Rate(EER)

In fig. 7, the green line is for the line $x = y$, The intersection of the ROC with this line gives us the FAR and FRR values which are equal.

fig. 7 lists the intersection point below the graph as ***FAR= 0.993, FRR=0.993***

Therefore, the threshold score over which we should classify the images as genuine match is ***0.993***

2.5 Correct Recognition Rate(CRR)

Correct Recognition Rate (CRR): It is defined as the number of actual matches that are obtained at rank one recognition

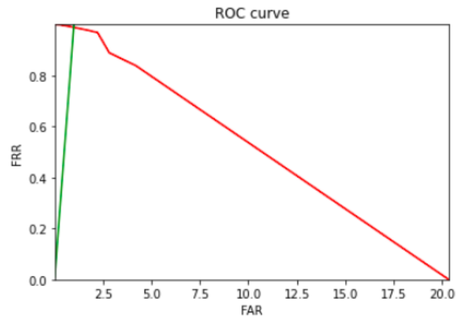
$$CRR = \frac{Number of matches correctly recognized}{Total number of matches}$$

Taking the recognition rate to be rank-1 recognition we obtain ***CRR = 0.000192***

Taking the recognition rate to be the EER we obtain ***CRR = 0.1304***

2.6 Ablation studies

I modified the K parameter to be 15 instead of 1 and the result obtained were similar, $CRR = 0.98$ fig. 8.




```
Intersection coordinates: (0.9868561814826874, 0.9868561814826874)
```

Figure 8: ROC for KNN with k=15

Also, using unsymmetrical scores in the match, gave similar results to the symmetrical ones.

2.6.1 How good is the dataset

To study this, I used the pandas describe to describe the score column of match and imposter dataframes. The precentiles for the genuine match fig. 9 was higher than imposter match fig. 10. This shows that our knnMatch model has performed in the good direction, even though the absolute accuracy of the model from CRR is low.

```
count    10375.000000
mean         1.390772
std         11.902253
min          0.000000
25%          0.030303
50%          0.105263
75%          0.309859
max        364.000000
Name: score, dtype: float64
```

Figure 9: Match score statistics

```
count    296350.000000
mean          1.588666
std          17.284204
min           0.000000
25%           0.000000
50%           0.000000
75%           0.055556
max         592.000000
Name: score, dtype: float64
```

Figure 10: Imposter score statistics

2.7 Accuracy

Accuracy: is maximum value of $(100 - (FRR + FAR)/2)$ across all thresholds. The accuracy obtained for our model was ***48.61%***