CS-E4850 Computer Vision, Answers to Exercise Round 7

Yangzhe Kong, Student number: 765756

October 31, 2019

Exercise 1. Comparing bags-of-words with tf-idf weighting.

a) Inverse Document Frequency:
1'cat': 4.3219,
'dog': 2.3219,
'mammals': 5.6439,
'mouse': 3.3219,
'pet': 0.7370

b) Term Frequency:
Query
'cat': 0.25, 'mouse': 0.25, 'mammals': 0.25, 'pet': 0.25
Document 1
'is': 0.1333, 'and': 0.0667, 'too': 0.0667, 'may': 0.0667, 'dog': 0.0667,
'pet': 0.2000, 'a': 0.2000 , 'mouse': 0.0667, 'be': 0.0667, 'cat': 0.0667
Document 2
'and': 0.1429, 'dog': 0.1429, 'mouse': 0.1429, 'all': 0.1429, 'mammals':
0.1429, 'are': 0.1429, 'cat': 0.1429
Document 3
'and': 0.0833, 'well': 0.0833, 'may': 0.0833, 'dog': 0.0833, 'mouse':
0.0833, 'a': 0.0833, 'eat': 0.0833, 'get': 0.0833, 'but': 0.0833, 'along':
0.0833, 'cat': 0.1667

c) The tf-idf weighted word occurrence histograms for the query and documents are shown in Figure 1, 2, 3 and 4.

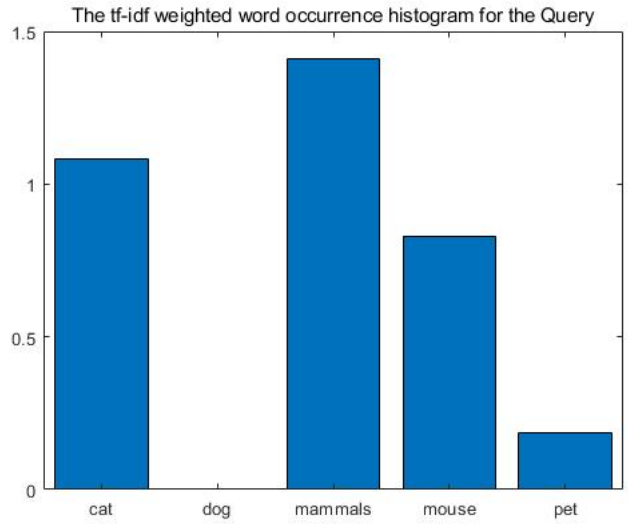

Figure 1: The tf-idf weighted word occurrence histogram for the Query

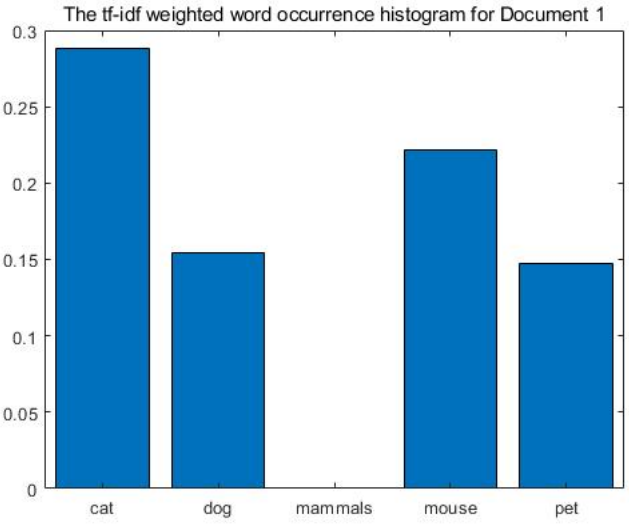

Figure 2: The tf-idf weighted word occurrence histogram for Document 1

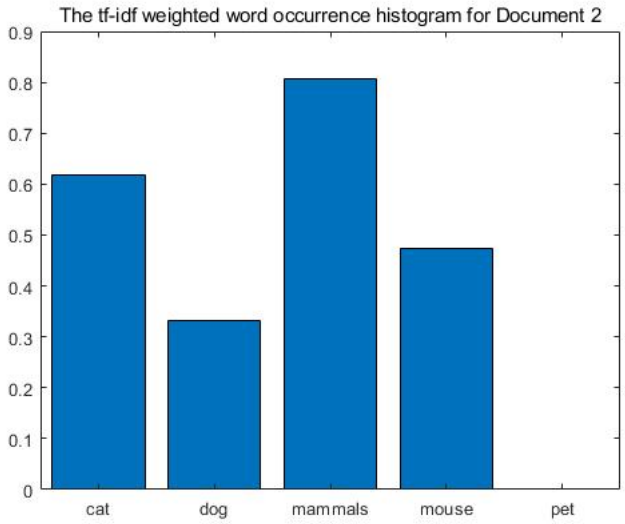

Figure 3: The tf-idf weighted word occurrence histogram for Document 2

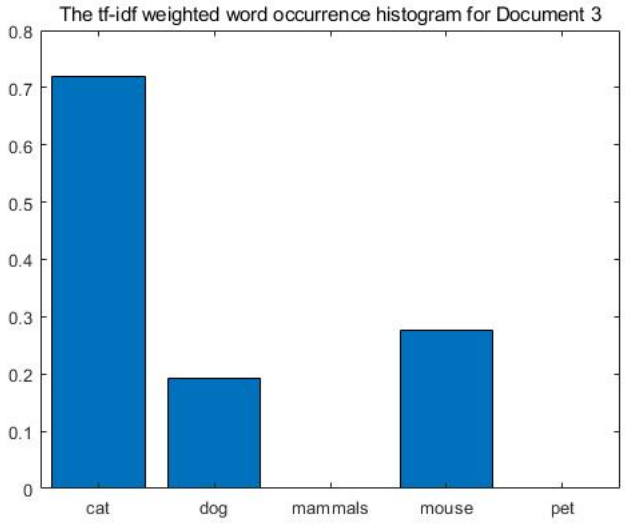

Figure 4: The tf-idf weighted word occurrence histogram for Document 3

d) According to the formula

$$sim(d_j, q) = \frac{\langle d_j, q \rangle}{\|d_j\| \, \|q\|} = \frac{\sum_{i=1}^{V} d_j(i) \times q(i)}{\sqrt{\sum_{i=1}^{V} d_j(i)^2} \sqrt{\sum_{i=1}^{V} q(i)^2}}$$

We can get that:
Similarity between the Query and Document 1 is 0.6289
Similarity between the Query and Document 2 is 0.9547
Similarity between the Query and Document 3 is 0.6432

e) According to the similarity measures in e), Document 2 is the most similar and Document 1 is the least similar.

Exercise 2

a) precision answers the questions of "how many selected items are relevant", which is defined as

$$\frac{\#number\ of\ relevant\ items}{\#total\ number\ of\ returned\ items}$$

precision answers the questions of "how many relevant items are selected", which is defined as

$$\frac{\#number\ of\ relevant\ items}{\#total\ number\ of\ relevant\ items}$$

So

$$precision = \frac{number\ of\ correct\ images}{number\ of\ images\ returned} = \frac{300}{350} = 0.857$$

and

$$recall = \frac{\#number\ of\ correct\ images}{\#number\ of\ total\ relevant\ images\ in\ the\ database} = \frac{300}{500} = 0.600$$

Exercise 3

Part I Sparse features for matching object instances

Stage I A: SIFT features detector

Question:
Note the change in density of detections across the image. Why does it change? Will it be a problem for matching? How could it be avoided?

Answer:
There are plenty of detections in the part of the architecture, but few is found in the part of grass in the front of the architecture. This is mainly because SIFT feature detector finds the local extreme points inside the images, which

are concentrated in the edges and corners. Edges and corners are common for architectures but rare for grass. It will be a problem if the image contains very few edges or corners, which means that the SIFT feature detection couldn't find too much features. A potential solution to this problem can be decreasing the *peakThreshold* value to allow the SIFT feature detector to detect more features.

Question:
Occasionally, a feature is detected multiple times, with different orientations. This may happen when the orientation assignment is ambiguous. Which kind of image structure would result in ambiguous orientation assignment? **Answer**: Image structures where dense edges and corners appear will cause the problem of ambiguous orientation assignment because a feature may be detected multiple times but in different scales so their orientation calculation may be different.

Stage I B: SIFT features descriptors and matching between images

Question:
Note the descriptors are computed over a much larger region (shown in blue) than the detection (shown in green). Why?

Answer:
Because we would like to enhance the robustness of the matching and make it less sensitive to illumination change. By choosing a larger region for computing the feature vector we can gain more invariance in our feature detector. We first divide the patch into 4x4 sub-patches and compute histogram of gradient orientations (8 reference angles) inside each sub-patch, then the resulting descriptor has 4x4x8 = 128 dimensions

Question:
Notice that there are many mismatches. Examine some of the mismatches to understand why the mistakes are being made. For example, is the change in lighting a problem? What additional constraints can be applied to remove the mismatches?

Answer:
Yes, changes in lighting can be a problem. Also we have too many detections so the uneven distribution will surely lead to mismatches, for example, mismatches between features on the grass part and features on the architectures. Also reoccurring pattern would also cause mismatches. Changing the value of *peakThreshold* or avoiding the change of lighting can be a good idea to remove the mismatches.

Stage I C: Improving SIFT matching using Lowe's second nearest neighbour test

Question: Examine some of the remaining mismatches to understand why they have occurred. How could they be removed?

Answer:
RANSAC algorithm can be used to further remove the remaining mismatches.

Part II : Affine co-variant detectors

Question:
The transformation between the images induced by the plane is a planar homography. The detections are only affine co-variant (not as general as a planar homography). So how can descriptors computed on these detections possibly match?
Answer:
When the viewpoint is not so extreme, we can still approximate the planar homography transformation by an affine transformation and use SIFT features to estimate the transformation. But when the viewpoint is becoming more extreme, less matches will be found. Also we can see that SIFT+affine adaptation performs better than SIFT in the sense that it better approximate the planar homography transformation.

Part III : Towards large scale retrieval

Question:
The size of the vocabulary (the number of clusters) is an important parameter in visual word algorithms. How does the size affect the number of inliers and the difficulty of computing the transformation?
Answer:
The bigger the vocabulary, the more inliers we would find. And also it would be slower to compute the transformation.
Question:
In the above procedure the time required to convert the descriptors into visual words was not accounted for. Why?
Answer:
Converting the descriptors into visual words won't take too much time compared to finding inliers.
Question:
What is the speedup in searching a large, fixed database of 10, 100, 1000 images?
Answer:
More images we have in our database, the larger the speedup we can obtain compared to searching without implementing visual words in each case.
Question:
Why does the top image have a score of 1?
Answer:
Because the query image have the same scene and very similar environmental settings (like lighting and viewpoint angle) with the top image so surely after extracting feature vectors and converting the features into visual words, they will have the same visual words for most cases. So then considering the weighted histogram similarities they will have a similarity value very close to 1.
Question:
Why is the top score much larger than 1 now?
Answer:
Because we rescore the top 16 images based on the number of inlier matches, and the final score will be the maximum between the old score and the new score. Since for

the top image, it has lots of inlier matches, its new score will be much larger than 1, which causes the final top score mych larger than 1.

Question:

Are the retrieval results improved after geometric verification?

Answer:

Yes. The results improve a little bit because without geometric verification some images that have the same scene with the query image but with different environmental settings (like lighting and viewpoint angle), which should be ranked higher than mismatches, are actually ranked lower than the mismatches. But after the geometric verification, those images are ranked higher than before, which justifies the improved performance.