Homework Assignment 4

Yi Liu djl174
ZhaoYang Xu chm564
Jingzheng Li bnd559

January 13, 2021

1 Codebook Generation

1.1 Data Set Selection

This report uses the data from CalTech 101 image database. In our data set, a total of 370 images are selected from 10 categories, among which 300 are classified as the training set, and the rest 70 are classified as the test set. The 10 categories are airplanes, bonsai, butterfly, carside, chandelier, kangaroo, ketch, starfish, sunflower, watch, yinyang respectively. In consideration of the computational power, we did not extend our experiment to a larger data scale.

1.2 K-means clustering algorithm

We experiment with a few different values of K, for example, $K = 500, K = 1000$. When K is small, the number of clustering will be less and cannot achieve a good classification. Relatively, larger K values will requires longer computation time, but obtain better result. To avoid over fitting problem, we did not choose larger K values.

2 Indexing

2.1 SIFT

To generate a codebook, we extract features from the training images. With the help of SIFT, we easily locate the local features(known as key points) and descriptors. The main advantage of SIFT is, they ignore the position, orientation and scaling change problem while detection for the edges or corners of images.

In details, scale invariant means one can use different size, viewpoints or even depth when you take a picture of an object. Using SIFT we can easily match them by the same features. The following are some other advantages of SIFT.

1) Locality： Since the features are local, so they are robust to occlusion and clutter.

2) Distinctiveness： The individual features can be matched to a large database of objects. Meanwhile, many features can be generated even for small objects.

3) Efficiency： Using SIFT is close to real-time performance.

2.1.1 Why gray scale image?

In OpenCV, we read the image in grey-scale format. As far as we understand, no colour does not simply mean it reduces information. In certain aspects, colour information has no help to detect important edges or some features.

One of the reasons that we do not use a colour image can be, gray scale image saves more time since less data needs to be observed than a full colour image.

Further, based on Lowe's paper, he states that the features described in his paper use only a monochrome intensity image, so further distinctiveness could be derived from including illumination-invariant colour descriptors. So that's can be the reason why SIFT operates on gray-scale images only.

2.2 Content Indexing

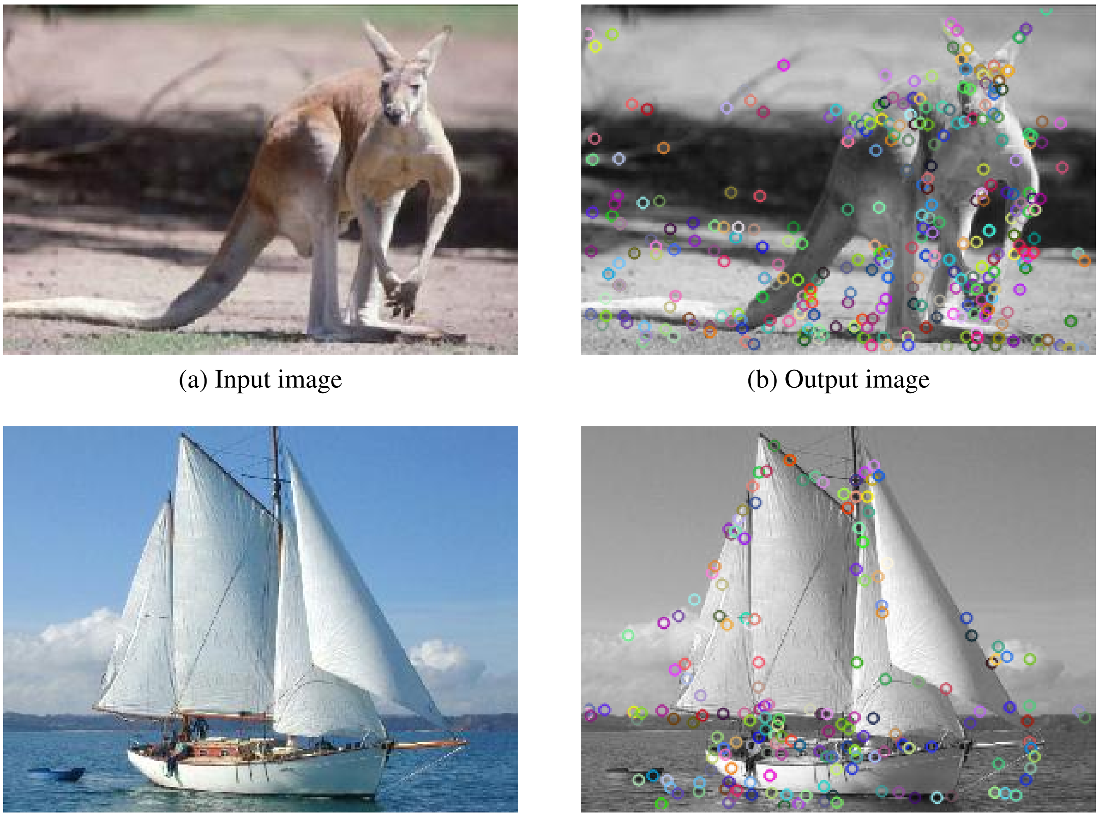
The code below used to create object of SIFT.

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

The code below used to extract the keypoints and descritpors of the input image.

```
kp, des = sift.detectAndCompute(X[i], None)
```

Then we can obtain pictures containing sift features as follows.



(a) Input image

(b) Output image

(c) Input image

(d) Output image

Figure 1: Original and SIFT pictures

In our demonstration, we selected two original images of kangaroo and ketch (left), and generated the corresponding SIFT images (right). It can be seen that SIFT has extracted a large number of feature points of the picture. We will continue the following steps based on these features.

Bow is known as bag of words. In our experiment, treat the visual image features as words. Hence Bow is a vector of occurrence counts of a vocabulary of local image features, where the features are obtained by SIFT.

Then we can obtain the histogram of cluster points of bag of words of all images and each image respectively as following

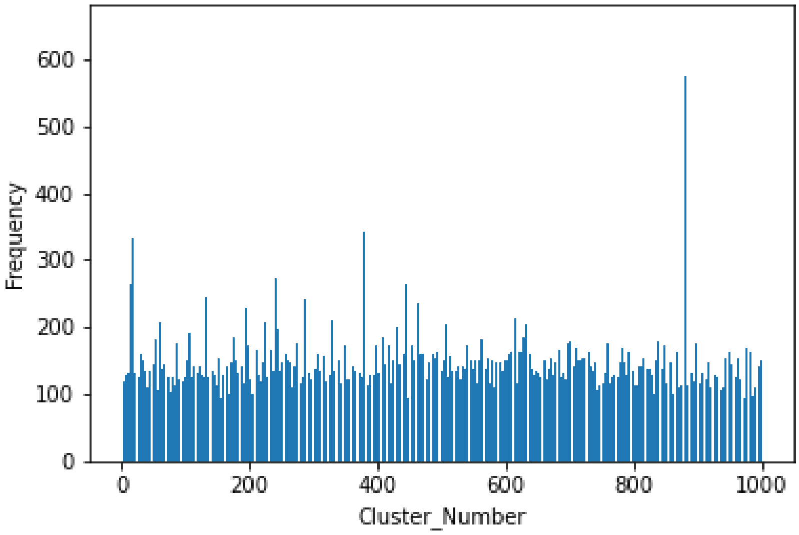
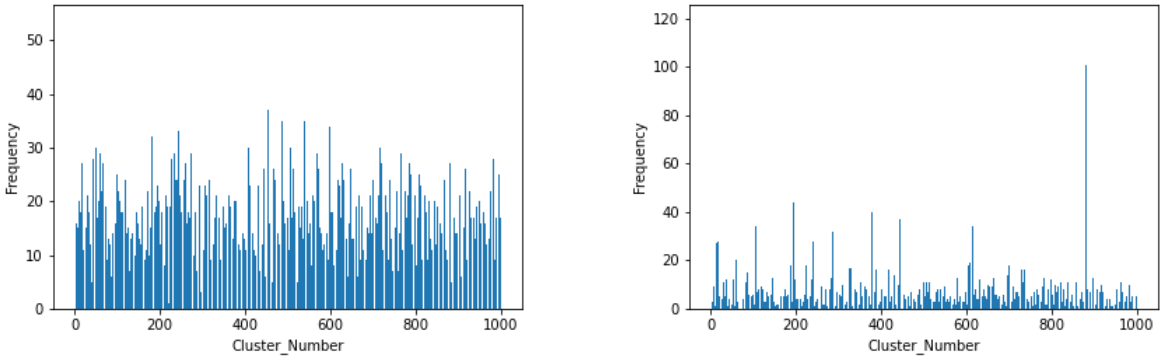


Figure 2: Histogram of cluster points of bag of words of all images (clusters = 1000)

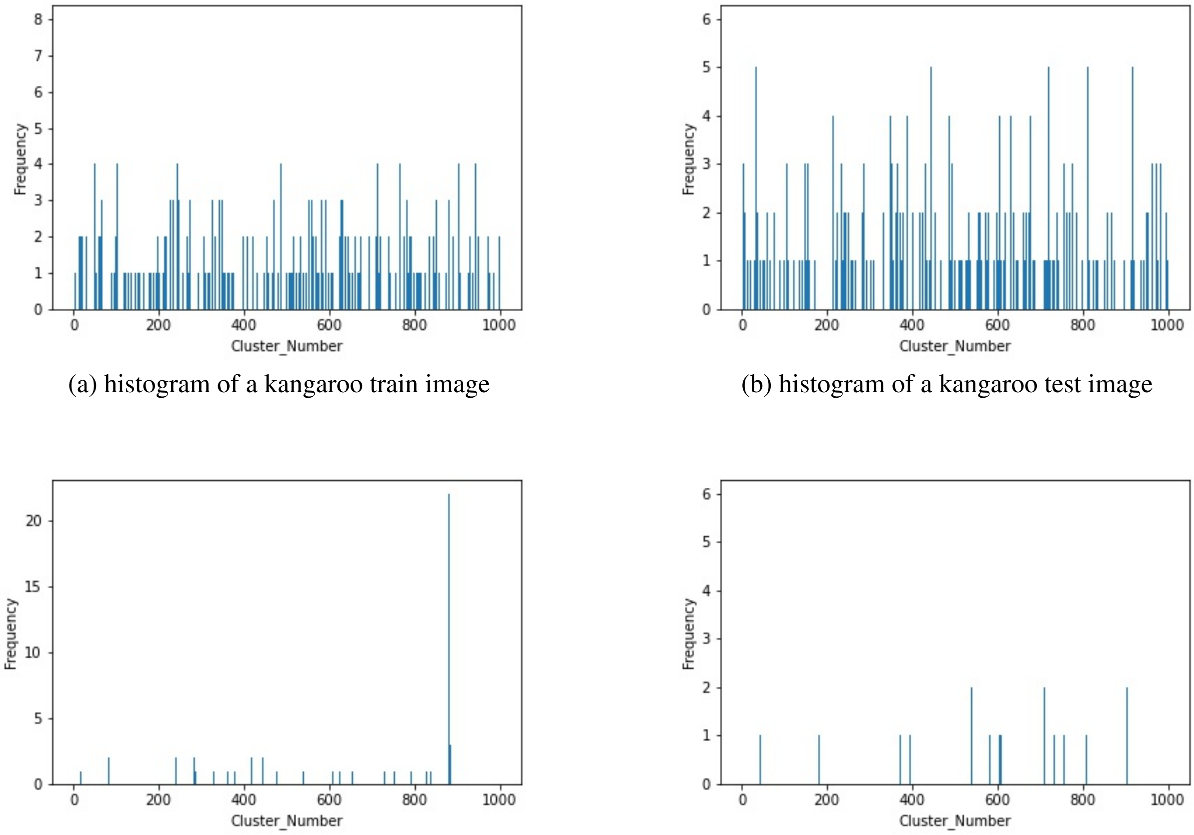
After K means clustering, the category of kangaroo has the highest average histogram among all the 10 categories, while the category of Ketch has the lowest average histogram.



(a) Histograms of kangaroo

(b) Histogram of Ketch

Figure 3: Histograms of two categories



(a) histogram of a kangaroo train image

(b) histogram of a kangaroo test image

(c) histogram of a ketch train image

(d) histogram of a ketch test image

Figure 4: Histograms of four images

The above histogram graphs of Kangaroo and Ketch is chosen by random. We can see the histogram of the ketch train image is extremely high at around K= 860. This perfectly matches the histogram frequency in the category Ketch in figure 3. So our assumption is Ketch has at least one key cluster point with very high frequency, this explains why Ketch always has a high classify accuracy in a later section.

Regards to the histogram of kangaroo, there is no extreme feature, but each cluster obtained a relatively high frequency of features. Also, the derivation of the frequency of each cluster is small. So when we queried the kangaroo category, it also shows a good result.

All the other histogram graphs are attached in folder Assignment 4.

3 Retrieving

3.1 Methods

TF-IDF: To implement the image retrieving, we use the TF-IDF similarity. TF-IDF is known as a weighting factor in searches of information retrieval. It helps reduce the less important "visual words" from the visual bag of words and compensates for the uneven occurrence of "visual words" across all images.

$$TF-IDF = TF * IDF$$

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \qquad IDF_i = \log \frac{|D|}{1 + |j : t_i \in d_j|}$$

where $n_{i,j}$ is the number of occurrences of the word in the document d_j, and the denominator is the sum of the number of occurrences of all words in the d_j. For IDF, $|D|$ is the total number of documents. $|j : t_i \in d_j|$ is the number of documents which contains t_i.

Bhattacharyya Distance for Normalized Histograms:

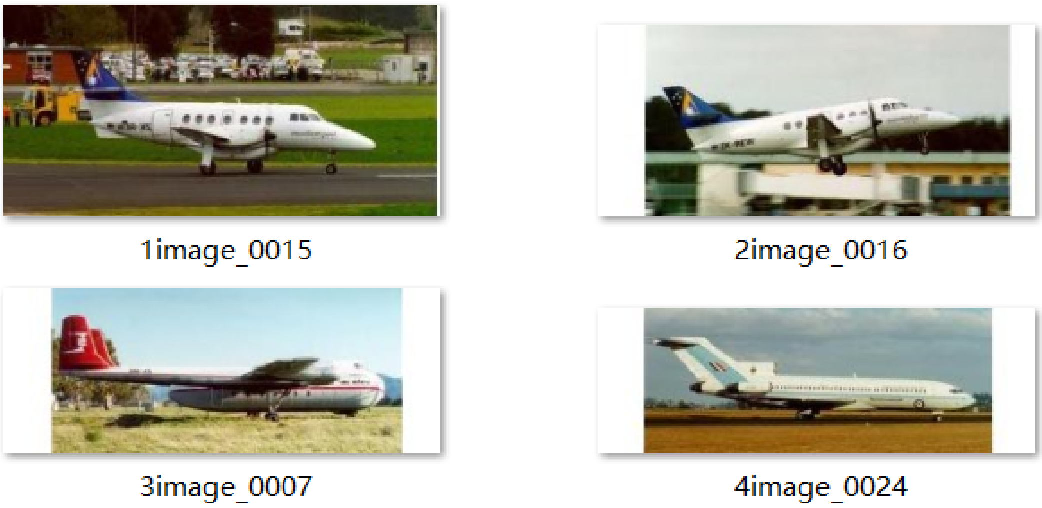
$$d(v_1, v_2) = \sum_{i=1}^{K} \sqrt{v_{1i} v_{2i}}$$

KNN Classifier: The k-nearest neighbours (KNN) classifier, the most common classification among the k nearest neighbours determines the class assigned to the object.

3.2 Retrieving

For retrieving image, we used the features from the querying image, and mapped the features to the visual words, using the K-mean model (which saved in the previous step). Then, the querying image was compared with the 300 images by Euclidean Distance. The shorter the distance, the more similar the images.

If we input features of airplane, first we can retrieve the following 4 pictures and they are all images of airplanes.



1image_0015

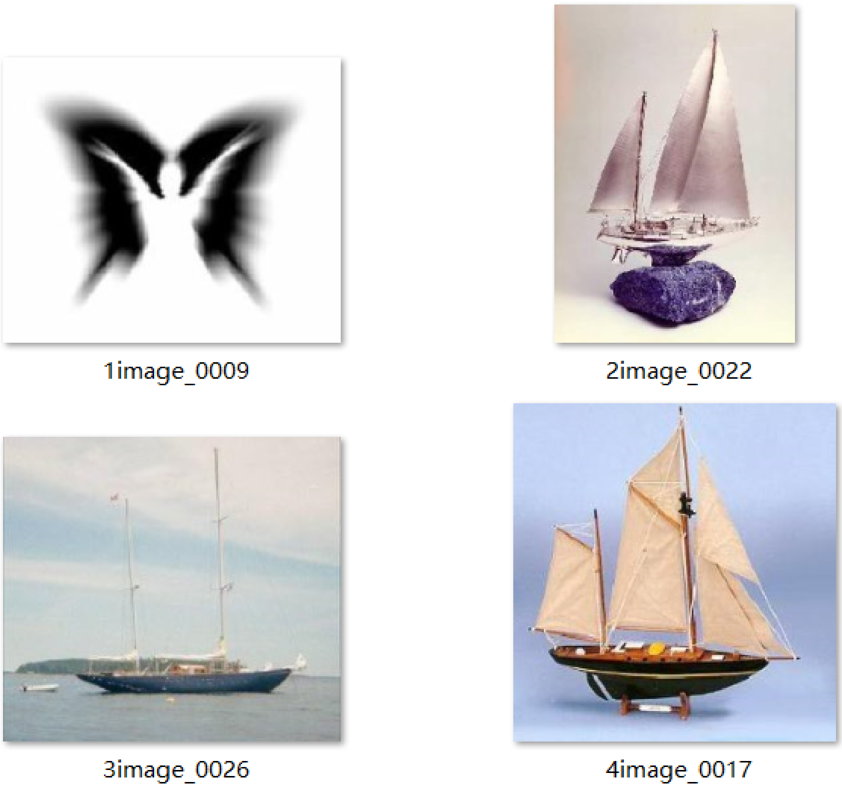
2image_0016

3image_0007

4image_0024

Figure 5: Retrieving of airplane category

If we input features of ketch, we can first retrieve the following 4 pictures.



1image_0009

2image_0022

3image_0026

4image_0017

Figure 6: Retrieving of ketch category

As we can see, when retrieving based on the features of airplane, the top four pictures belong to the correct category. It shows that the features of airplane are obviously different from the other categories. However, when retrieving based on ketch's features, the best match picture is a butterfly, then the rest three are real ketch pictures. If we look closely, we can find that the wings of the butterfly are very similar to ketch's sails. Therefore, this picture is classified into the ketch category according to feature matching.
Based on the MRR formula:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}$$

where Q is the query set, $|Q|$ is the number of queries in Q. In our model $|Q| = 10$. $rank_i$ is the rank of the first correct answer in the ith query. After queried 10 categories, we get

$$\frac{1}{10}(1+1+1+1+1+1+\frac{1}{2}+1+\frac{1}{14}+1+\frac{1}{2} = 0.907$$

We also tried to use Bhattacharyya Distance as similarity measures, but the result is not good enough. It queried nothing for many categories.

$$\frac{1}{10}(1+\frac{1}{3}+\frac{1}{6}+\frac{1}{6}) = 0.167$$

3.3 Classifying

```
clf = neighbors.KNeighborsClassifier(n_neighbors=5,
      algorithm='ball_tree',metric=bhattacharyya)
```

For classifying image, we use the KNN method. Since the training set is not large, we performed cross-validation at first. 20% of the data set is taken as the test set each time. Then we extracted the features from a test set to classify.

In our report, we choose 5 neighbours and used Bhattacharyya Distance to calculate the distance between the object and neighbours. The best category is the butterfly, the probability of correct classification is 57%. The second and third category have the same correct rate 28.6%, which are chandelier and ketch respectively.

Also, we tried the Euclidean Distance to classify the test set. This time, the top1 category turn to be sunflower, which has 71.4% correct probability, the next one is watch (57%). And there are two categories tied for the third place, kangaroo and ketch, which is correct at 42.9%.