CMPT 412
Assignment 4 Report

Maoshun Shang

301280479

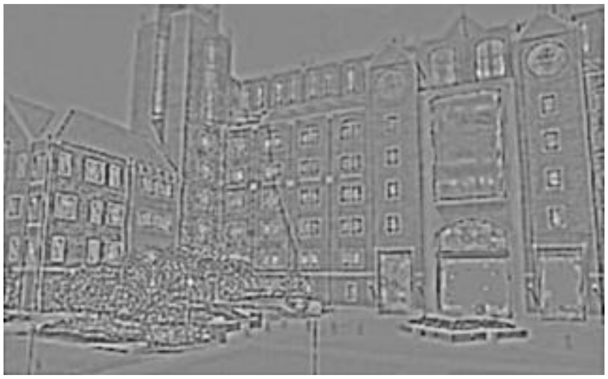
Introduction

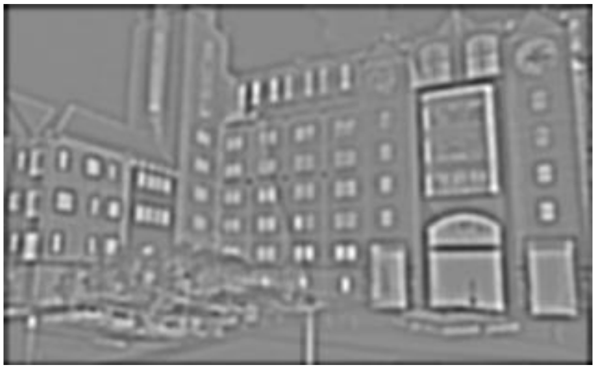
This assignment mainly focuses on the technique studied in lectures, including basics about image classification, k-means clustering, dictionary, and kNN etc. The programming language used is MATLAB, which helps with reducing the complexity of heavy computation on matrix and vector calculations.

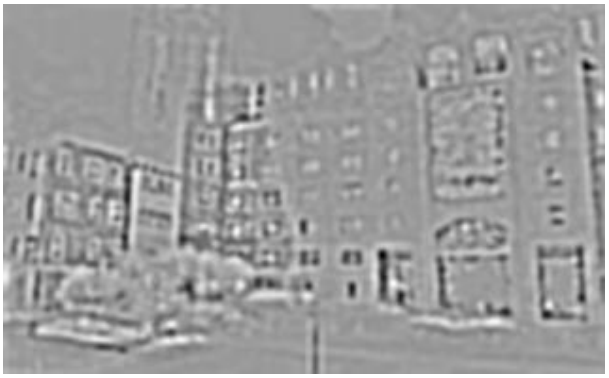
Part 1: Build Visual Words Dictionary

1.1: Extract Feature Responses

original Image


filter response 18th layer


filter response 19th layer


filter response 20th layer

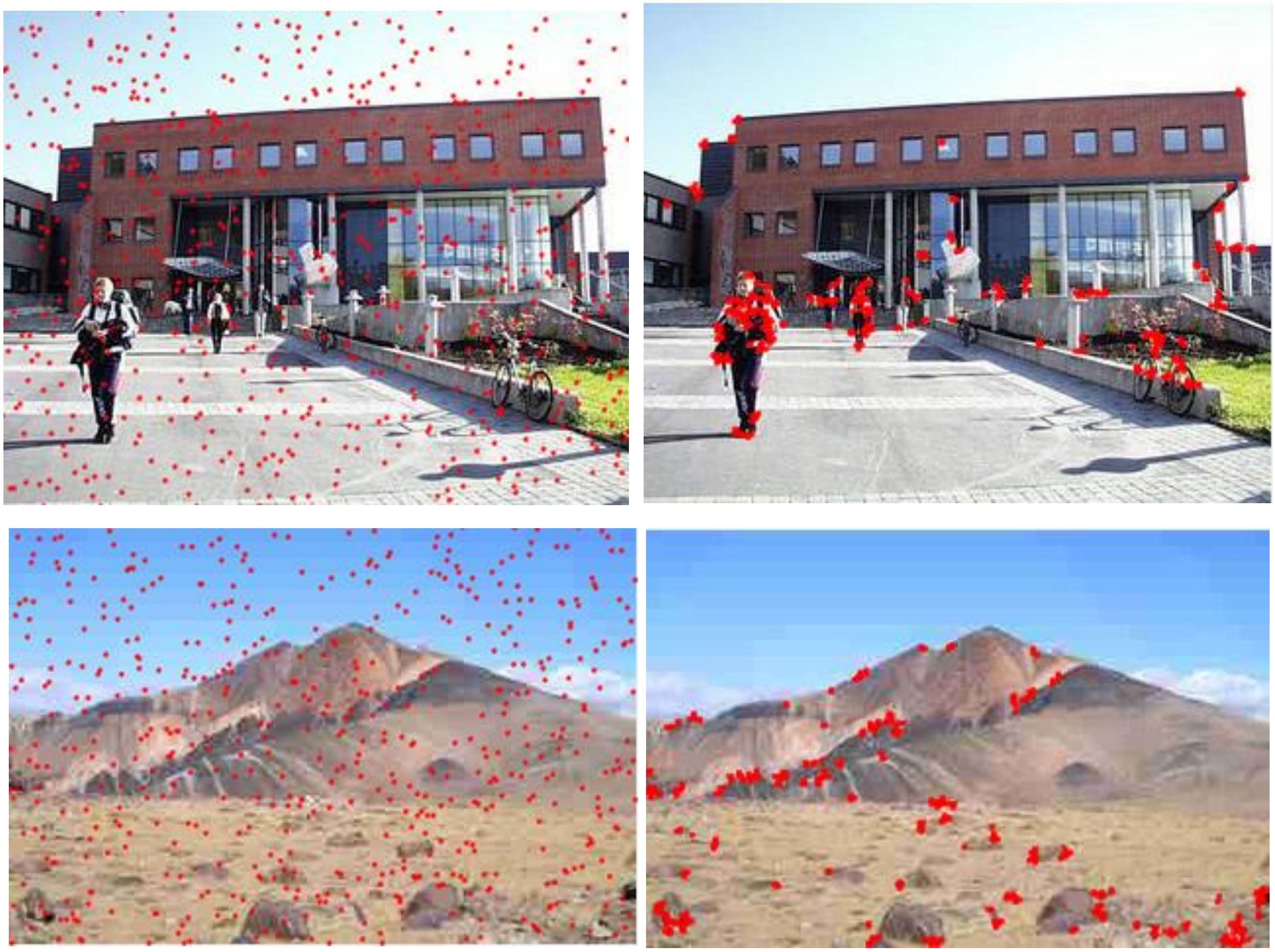
While the 3 chosen filter responses are from the result of a same filter, the 18th layer is from the 'L' colour component, the 19th layer is from the 'a' colour component, and the 20th layer is from 'b' colour components.

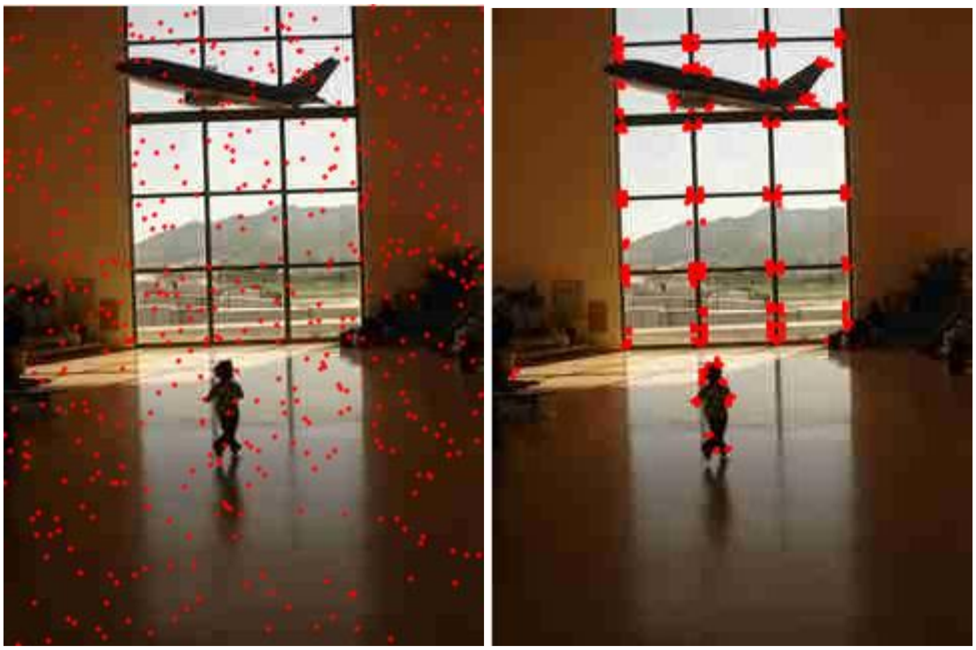
From the images above, we notice some of the details from the original image get ignored or amplified in the filtered image. We can see the clouds disappeared in the 18th and 19th layer but exist in the 20th layer. However, in the 20th layer, most of the details of the building is lost. In addition, the last filter result is more blurred comparing to the other two.

CIELAB colour space is a colour space defined by three variables, where L defines luminance, a defines green-red components, and b defines blue-yellow components. LAB colour space aims to describe colours in a more human vision way, so that we use LAB colour space to approximate the accurate result of image classification tasks as human beings. In contrast, the RGB colour space is more appropriate for device colour outputs.

1.2 Collect sample of points from image

Take 500 points (alpha = 500), the left images are random sampled points, and the right images show harris points. For the harris points, the alpha value is set to be 0.04.

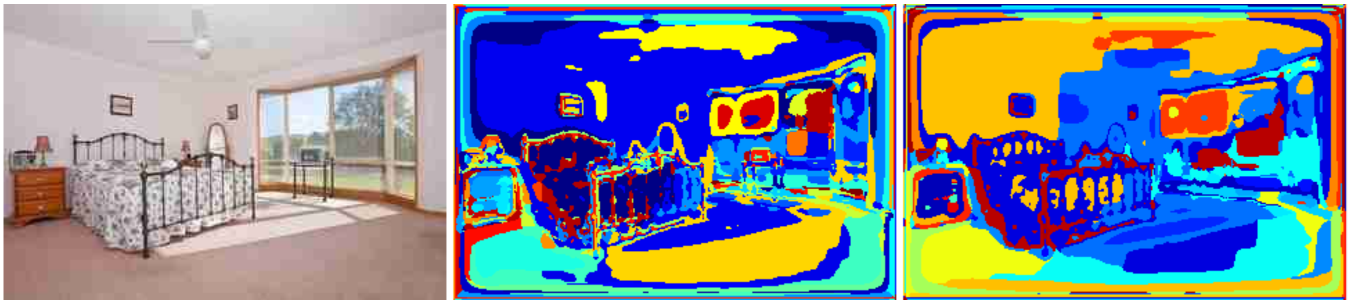




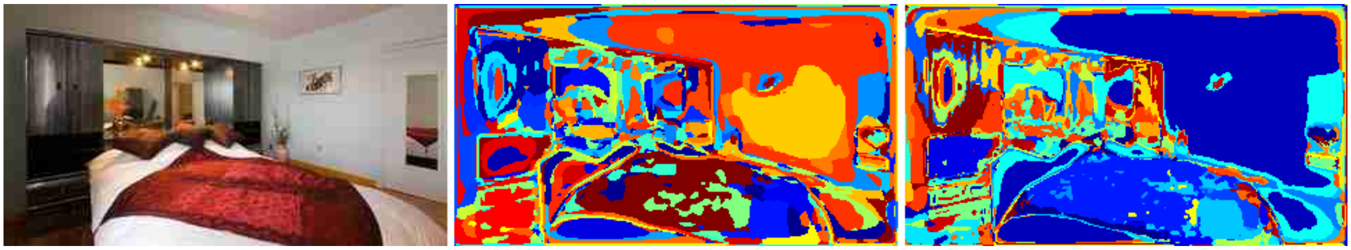
Part 2: Build Visual Scene Recognition System

2.1: Convert image to word map

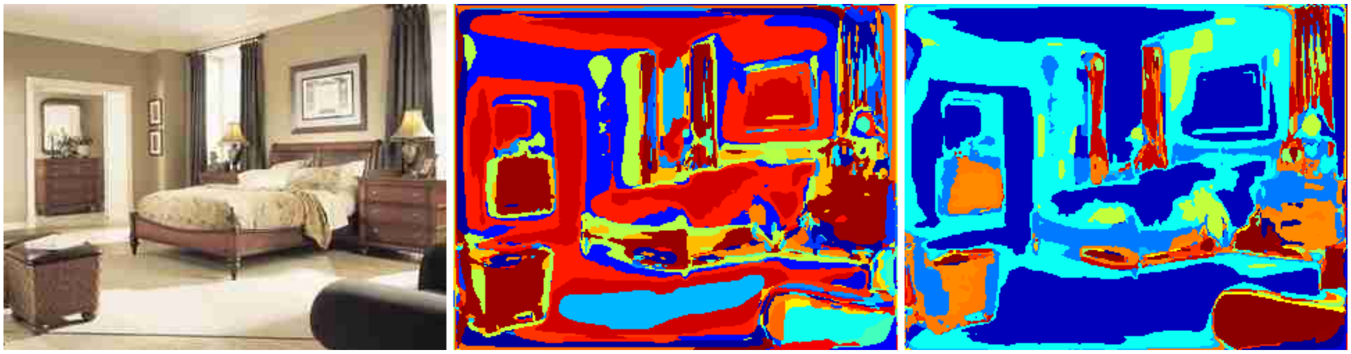
The 6 original images are chosen from 2 sets, bedroom and auditorium. Each row below contains 3 images, they are the original image, the visual words from random dictionary, and the visual words from harris dictionary.



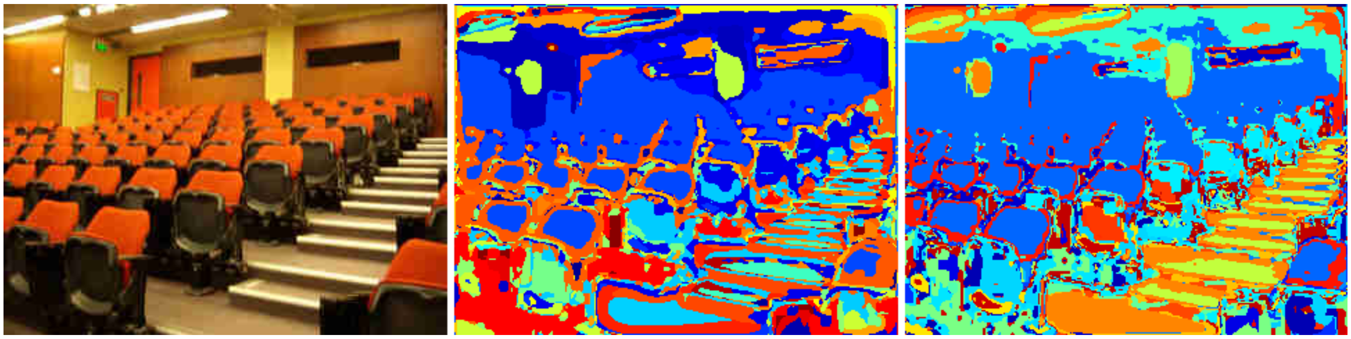
word map (bedroom 1)



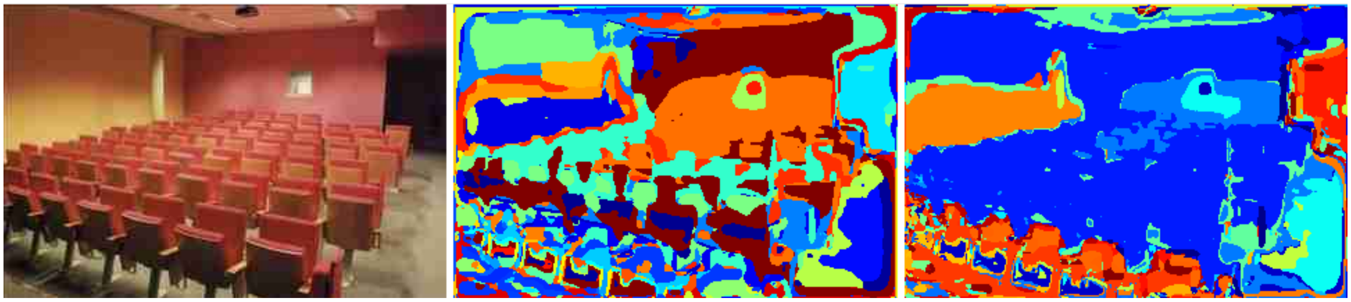
word map (bedroom 2)



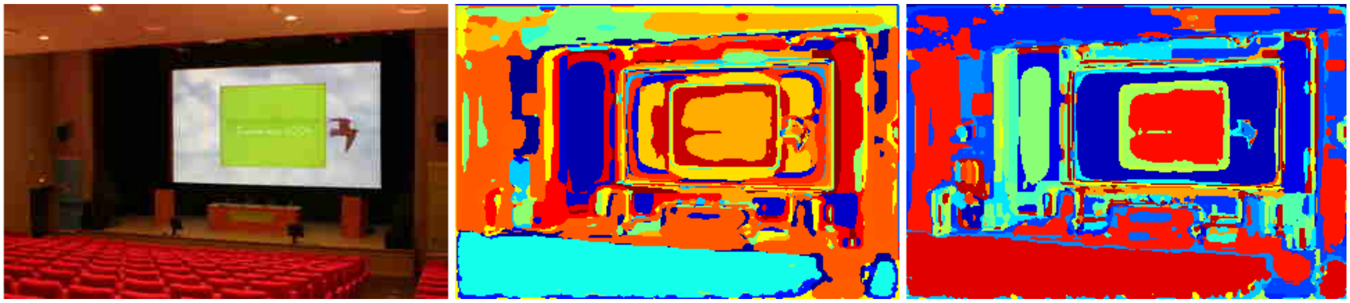
word map (bedroom 3)



word map (auditorium 1)



word map (auditorium 2)



word map (auditorium 3)

Both of the dictionaries do capture most of the semantic meanings to some extent, but they did not capture the semantic meanings perfectly.

They do capture some of the semantic meanings because, for example, the beds in the first sets of images are recognizable from both random dictionary image and the harris dictionary image. In addition, most of the areas of the bed is represented by blue.

However, take the bedroom 1 (first set of images) as an example, the visual words image from random dictionary use the same colour (blue) to represent both the wall and the carpet. For visual words image from harris dictionary, the carpet as an example, is represented by 8 different colours. Those are some minor imperfections of these two dictionaries.

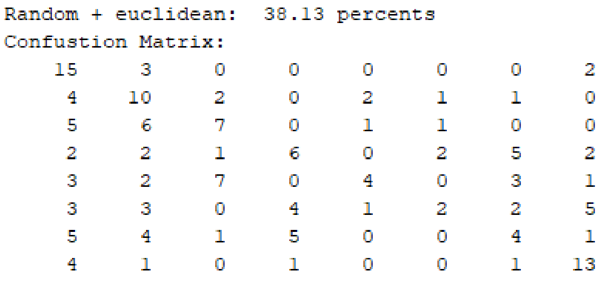
The harris dictionary is better comparing to the result of random dictionary. Because on average, harris dictionary can represent one single object by fewer colours. For example, the blanket in the second set of bedroom images is being mapped to 6 colours by random dictionary, while it is only mapped to 4 colours by the harris dictionary.

In addition, the reason that harris dictionary is better could be explained from the algorithm's perspective. Harris dictionary involves computing the image x and y direction gradients. Thus, harris dictionary separates the boundaries of different object better than random dictionary who samples point randomly.

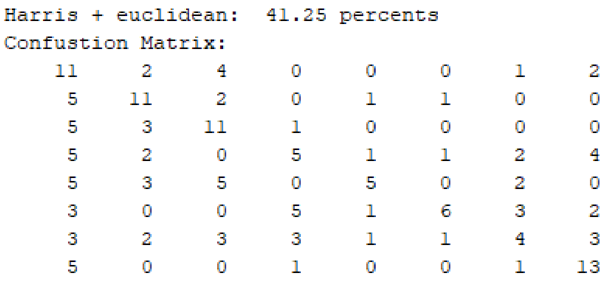
Part 3: Evaluate Visual Scene Recognition System

Q3.2 Evaluate Recognition System

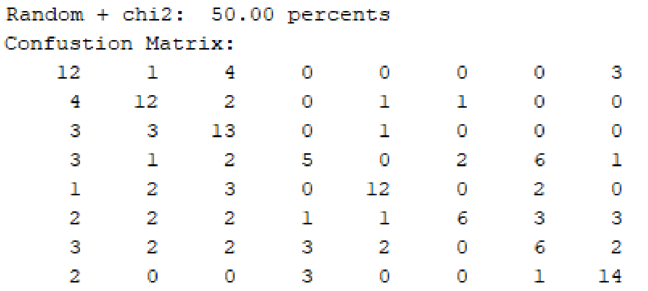
NN



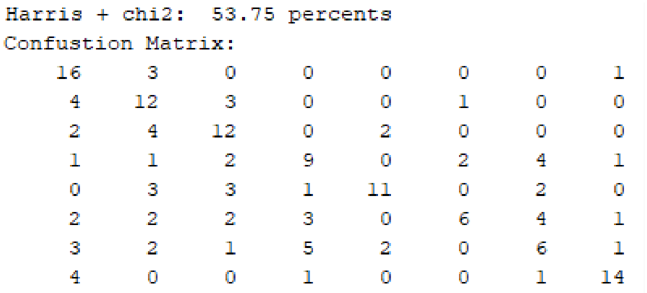
```
Random + euclidean:  38.13 percents
Confustion Matrix:
    15     3     0     0     0     0     0     2
     4    10     2     0     2     1     1     0
     5     6     7     0     1     1     0     0
     2     2     1     6     0     2     5     2
     3     2     7     0     4     0     3     1
     3     3     0     4     1     2     2     5
     5     4     1     5     0     0     4     1
     4     1     0     1     0     0     1    13
```



```
Harris + euclidean:  41.25 percents
Confustion Matrix:
    11     2     4     0     0     0     1     2
     5    11     2     0     1     1     0     0
     5     3    11     1     0     0     0     0
     5     2     0     5     1     1     2     4
     5     3     5     0     5     0     2     0
     3     0     0     5     1     6     3     2
     3     2     3     3     1     1     4     3
     5     0     0     1     0     0     1    13
```



```
Random + chi2:  50.00 percents
Confustion Matrix:
    12     1     4     0     0     0     0     3
     4    12     2     0     1     1     0     0
     3     3    13     0     1     0     0     0
     3     1     2     5     0     2     6     1
     1     2     3     0    12     0     2     0
     2     2     2     1     1     6     3     3
     3     2     2     3     2     0     6     2
     2     0     0     3     0     0     1    14
```



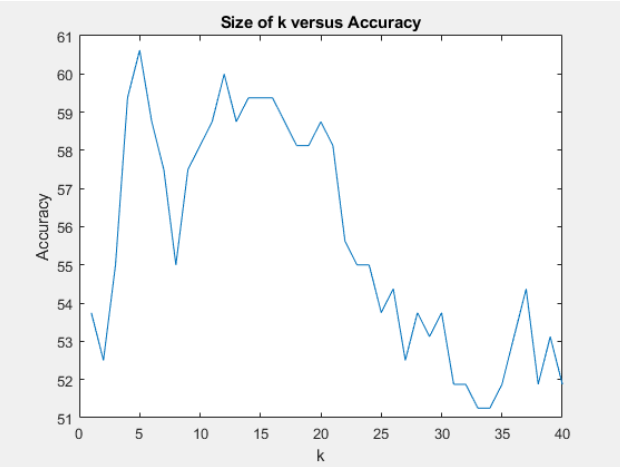
```
Harris + chi2:  53.75 percents
Confustion Matrix:
    16     3     0     0     0     0     0     1
     4    12     3     0     0     1     0     0
     2     4    12     0     2     0     0     0
     1     1     2     9     0     2     4     1
     0     3     3     1    11     0     2     0
     2     2     2     3     0     6     4     1
     3     2     1     5     2     0     6     1
     4     0     0     1     0     0     1    14
```

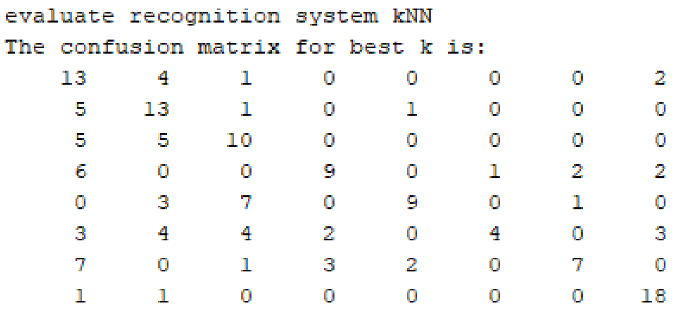
Comparing the accuracy of the two dictionaries, the result is within expectation that the harries dictionary has higher accuracy when using both metrics. Because when sampling harris points, we only focus on the points that sitting on the edges, meaning the sampled points are the boundary of two objects.

For the two distance metrics, the chi2 metric always has better performance over the Euclidean metric. This is due to chi2 metric is better for calculating the distance between histograms. (recall that in the script ‘getImageDistance.m’, we pass two histograms as parts of the function input. In addition, in the script ‘getImageFeatures’, we compute the image features as a histogram.)

KNN

The measures below is done by using harris sampling and chi2 distance metrics.





```
evaluate recognition system kNN
The confusion matrix for best k is:
    13     4     1     0     0     0     0     2
     5    13     1     0     1     0     0     0
     5     5    10     0     0     0     0     0
     6     0     0     9     0     1     2     2
     0     3     7     0     9     0     1     0
     3     4     4     2     0     4     0     3
     7     0     1     3     2     0     7     0
     1     1     0     0     0     0     0    18
```

The best value of k appears around 6. According to the trend of the graph above, larger k is not always better. Because for small k, the neighbor size is too small that the result is too “local”. That is, the kNN method lose its feature because we only consider a small number of neighbors. In contrast, for the large k, the classification result could be affected by outliers because we need to consider a huge number of neighbors. In this case, since we already computed the best performing k, so we choose that k value for the k nearest neighbor classification. In addition, choosing k = 6 might cause the tie of even votes from different neighbors. To break the tie, we can choose k = 5 or k = 7 to avoid such situation. Because odd k avoids even number of votes from neighbors (so for odd number of neighbors, one of the class will stand out), also maintains (approximately) the peak performance of k nearest neighbor classification algorithm.