CENG 483 - Introduction to Computer Vision Spring 2017-2018 Take Home Exam 1 Content Based Image Retrieval

Duygu Dogan
e1941962@ceng.metu.edu.tr

***Abstract*—In this document, I have explained the methodology that I used to implement a CBIR system.**

***Index Terms*—Content Based Image Retrieval, Interest Points, Local Feature Descriptors, SIFT, Dense-SIFT, Clustering, Bag of Features Representation, Mean Average Precision**

I. Introduction

The main goal of the assignment is to implement a CBIR system based on Bag of Features (BoF) representation of SIFT and Dense-SIFT local descriptors.

A CBIR system is used to retrieve similar images from a large image database. It is performed by comparing the representation of images with the representation of query image.

In order to achieve this, below steps were followed:

- A visual word dictionary for Bag of Features representation were constructed by using k-means clustering.
- The BoF representation of each image was extracted to a histogram.
- Images were ranked by their similarity by using simple Euclidean distance algorithm.

II. Used Technologies

- Python 3.6
- Anaconda
- Jupyter Notebook

III. Implementation

A. Dictionary Construction

In order to achieve our goal, I first adapted the bag of features representation to computer vision. Key points and descriptors of the images were detected by using SIFT method with 800 features. All descriptors were gathered in an array of descriptors. After extracting the local descriptors of the images, a visual dictionary was constructed using k-means algorithm. By running the clustering algorithm with K = $\{32,70\}$, descriptors that are close to its cluster center gathered together and mapped to its cluster id.

K-means clustering algorithm works as follows:

1) Pick K random cluster numbers.

2) Assign each descriptor to its closest cluster by calculating its distance to the center with Euclidean distance algorithm (Equation 1.).

3) Find the new cluster center by taking the average of all points.

4) Repeat the above steps until there will be no change.

A sifted image can be seen in figure 1.

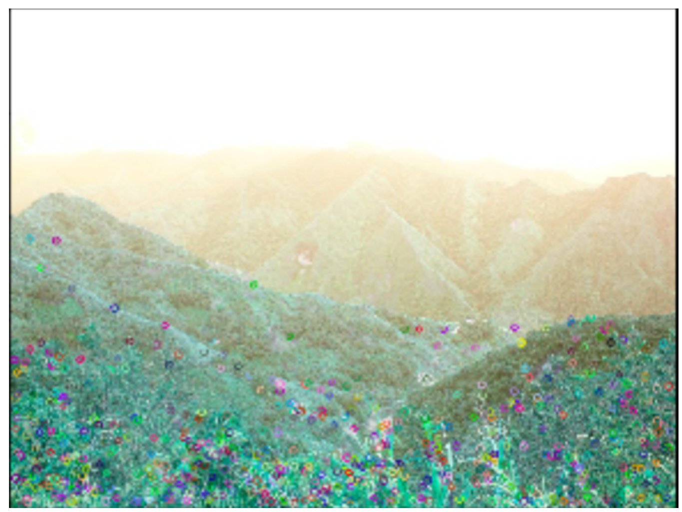


Fig. 1. Sifted Image

B. Bag of Features Representation

After mapping the descriptors to their cluster id, histograms of each images were computed. All of them has K clusters.

C. Ranking Images by Similarity

After the histograms of the images were constructed, the images were compared by their similarity using simple Euclidean distance algorithm (Equation 1.).

$$d_{euclidian}(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1}$$

IV. Interpretation

This interpretation will be mainly based on the images showed in figure 2 and 3.

Fig. 2. Sample Image named wSfsVRubus.jpg

Fig. 3. Sample Image named ptgUzEjfeb.jpg

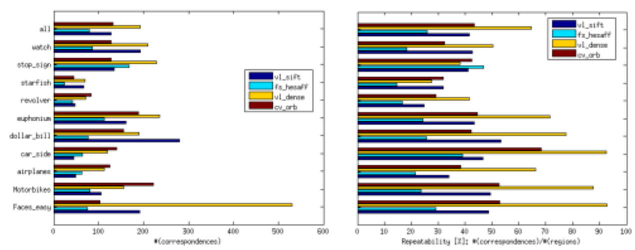
As can be seen from below figures, histograms of two almost same images are also very similar to each other. By having a look at the output file, we can also clearly say that the difference between these two images are one of the lowest among all.

A. SIFT and Dense-SIFT

Though I did not implement a Dense-SIFT method, I can easily say that SIFT method is a quite slow approach.

"The interest points are described by 2D ellipses and when a transformed ellipse overlaps with an ellipse in the second image a correct correspondence is recorded. The number and rate of correspondences for each detector is of interest. A detector performs well if the total number is large and has high precision if the ratio of correct matches is high." [1]

As can be seen from the figure 4 that I borrowed from an article published in Journal Neurocomputing, dense SIFT features are clearly better than SIFT features. [1]



(a) (b)

Detector	*Avg # of corr.*	*Avg. rep. rate*
vl_sift	127.5	41.6%
fs_hessaff	79.3	26.0%
cv_orb	132.0	43.5%
vl_dense	192.3	64.6%

(c)

Fig. 4. Detector evaluation in object class matching. Meta-parameters were set to return on average 300 regions. (a) average number of corresponding regions, (b) repeatability rates, and (c) the overall results table.

B. Choose of Parameters

Regions for SIFT method was chosen as 800 since n = 500 did not give an accurate result.

K for clustering algorithm was chosen as 70 since K = 32 also did not give an accurate result while 16 cluster was running much faster.

Since sifting and clustering took a lot of time, I encountered some technical difficulties caused by CPU and could not get a chance to try more cluster numbers. However, I observed that running K-means clustering algorithm with wrong values of K may be resulted in getting deceptive results.

V. Conclusion

As it was explained above sections, the task was implementing a CBIR system with bag of features representation and I clearly observed the effects of choosing the value of cluster number and the interest points for SIFT method and the approaches between SIFT and dense SIFT methods.

REFERENCES

[1] Antti Hietanen, Jukka Lankinen, Joni-Kristian Kmrinen, Anders Glent Buch, and Norbert Krger. 2016. A comparison of feature detectors and descriptors for object class matching. Neurocomput. 184, C (April 2016), 3-12. DOI: https://doi.org/10.1016/j.neucom.2015.08.106