Master 2 Image Mining Course
Lab 1: Bag-of-Features representation for Content Based Image Retrieval

November 2023

The goal of this practical work is to experiment image representations based on aggregation models of handcrafted local descriptors, i.e. Bag-of-Features representations, in the context of Content-Based Image Retrieval (CBIR), which follows the following steps (see also [Aru 2020]):

- SUPPORT: Is it better to reduce the representation support to salient points (sparse representation) using a *detector* or to sample evenly the images (dense representation)?

- DESCRIPTION: What information should be attached to every point of the support, for what expected properties (invariance,...)?

- QUANTIFICATION: How to reduce the set of descriptors to a significant *codebook*?

- AGGREGATION: How to pool a collection of "words" extracted from an image to form a relevant *global* descriptor?

- METRICS: What similarity or distance measure should be used to compare two global descriptors?

- SEARCH: How to browse the set of global descriptors to find the most ressemblant images from a given query image?

You are expected to provide a report, that will include the answers to the questions, a justification of the chosen methods and, as much as possible, illustrations with your own experiments.

1 Software preliminaries

A template Jupyter notebook is provided, that works with the standard opencv-python package (i.e. without opencv-contrib) and Python3:
https://perso.ensta-paris.fr/~manzaner/Cours/Masters_ParisSaclay/Image_Mining/TP_CBIR.ipynb
A (small) example data set is also provided, coming from [Lab 2006]:
https://perso.ensta-paris.fr/~manzaner/Cours/Masters_ParisSaclay/Image_Mining/CBIR_15-Scene.tgz

The following packages / modules should first be installed:

- jupyter
- scikit-learn
- matplotlib
- random
- python-opencv
- numpy
- os

OpenCV and Scikit-Learn are Python libraries that are extremely popular in image analysis and machine learning. Many online guides and tutorials can be found (pay attention to the versions used in the provided examples). We will only mention here the official websites for the reference guide of functions:
https://docs.opencv.org/4.1.0/
https://scikit-learn.org/stable/user_guide.html
and the following tutorials:
https://opencv24-python-tutorials.readthedocs.io/en/stable/index.html
https://scikit-learn.org/stable/tutorial/index.html

2 Data preparation

DOWNLOAD the dataset and uncompress it to a working directory. Adapt the Section I of the notebook to your path and data. You will adjust here the sizes of the learning and testing datasets according to your computing power and development version.

3 Detectors and descriptors

COMPARE the two possible methods "Kaze" and "ORB" for detecting and describing keypoints, and see how they behave on the different image classes. Briefly recall the principles of the detector and descriptor for each method.

4 Codebook construction

BUILD the codebook by clustering the descriptors extracted from the learning set, by using the (default) K-Means method or another method you prefer. What is the rule of thumb you use to determine the size of the codebook, considering (a) the number of classes, (b) the average number of keypoints per image?

5 Indexing the Test dataset

INDEX the testing dataset by computing, for each image, a global descriptor relative to the distribution of words from the learned codebook present inside the image. Describe the method you use and display the representation of your global descriptor for some images of different classes.

6 Image retrieval

EXPERIMENT the usability of your method for CBIR, by randomly picking an image from the testing set, and displaying the closest images with respect to some distance in the (global) descriptor space. Explain the metrics you use, and discuss your results qualitatively. How could you quantititavely evaluate the performance of your method?

References

[Aru 2020] K.S. Arun, V.K. Govindan, and S.D.M Kumar, "Enhanced bag of visual words representations for content based image retrieval: a comparative study", Artificial Intelligence Review 53, 1615-1653

[Lab 2006] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories", CVPR, 2006.