

Student Guide:
Bag-of-Words based Image Classification

UAB

Universitat Autònoma
de Barcelona

Master in Computer Vision and
Artificial Intelligence



I. Introduction

In this practicum we try to answer the question: “What objects are present in the image ? “. Or “What scene category are we looking at ?”. We will use the bag-of-words (BOW) approach to answer these questions. In the *Fig. 1* below an overview of the system is given.

We have already implemented all the steps. However, in none of the steps an optimal choice was made, and there are many opportunities to improve the system. The goal of this assignment is to improve the system with the knowledge of the previous two modules, and the new lectures. The aim of this assignment is that, you will learn:

- how to represent images robust to changes of cameras, object orientation, and illuminant color.
- what photometric invariance theory is and how to apply it to a real-world problem.
- how the SIFT descriptor describes images independent of intensity and contrast variations. And how to use it for object recognition.
- how to discretize features and construct vocabularies for colors, shapes, and textures.
- what the strong and weak points of BOW representations are.
- how to evaluate retrieval and classification results with ROC-curves, confusion matrices, and EER's and AP's.

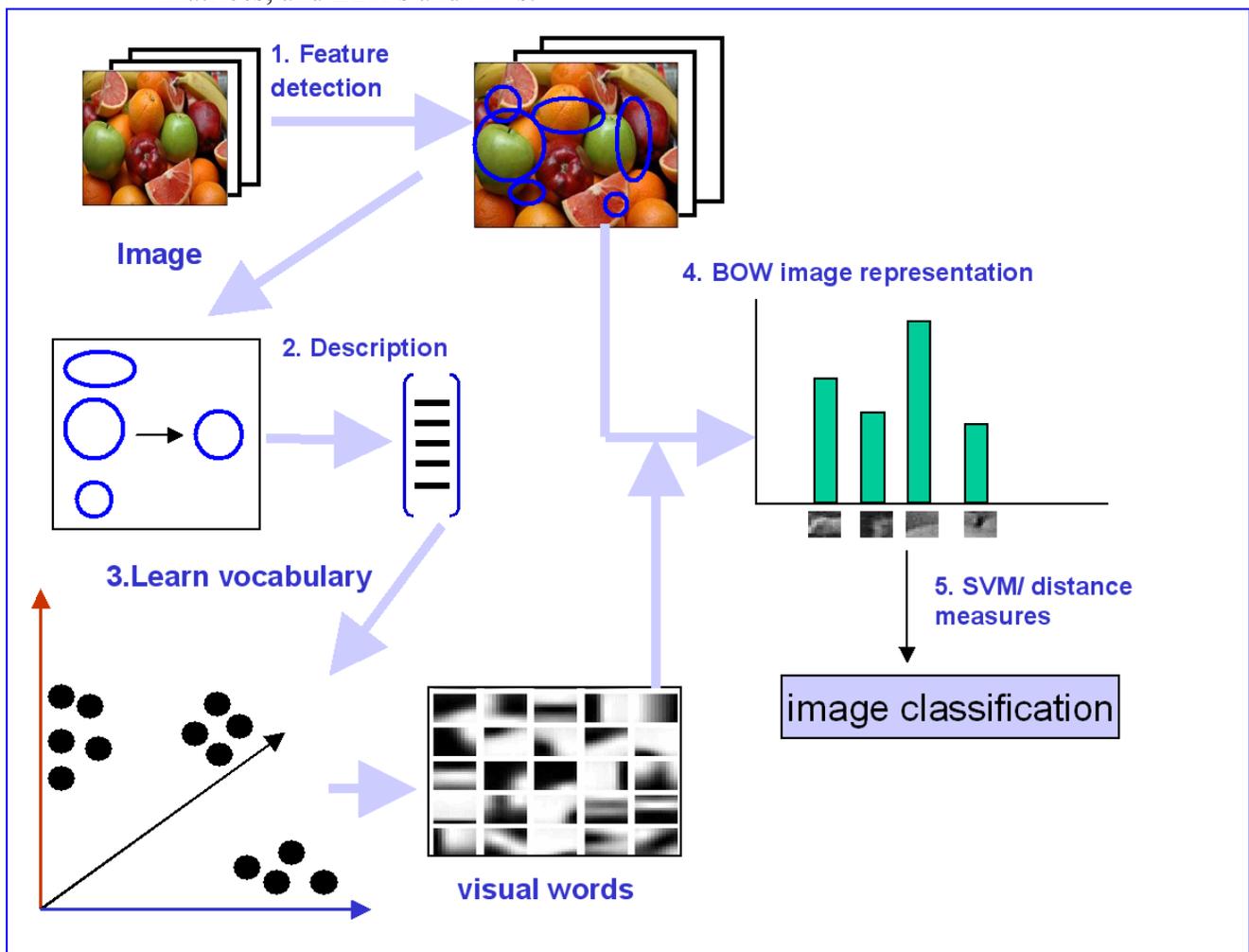


Fig 1. Overview of the subsequent steps in a BOW image classification.

II. Preparation

Before we can start, download the software and unzip the folders onto your computer (<http://cat.cvc.uab.es/~joost/master.html>). There will be three folders:

- events : containing the images.
- BOW_code : containing the matlab files.
- labels : containing the labels and the train-and test set split of the data.

The first thing to do is to change the paths in the EVENTinit.m file. You only need to change the first three paths. The data-path will be used to store the data during this assignment. In the file EVENTinit.m all the information about the data set is initialized and is stored in the structure **eventopts**. It contains information on where to find the images, where the labels are, what the class-names are, etc. First run:

➤ EVENTinit

then have a look at **eventopts**. Next run the function:

➤ make_directory_structure(eventopts)

This function will make two more directories in **eventopts.datapath**, called **local** and **global**. In global the information which is relevant to all data is saved, in local the detector and descriptor information for each of the images is saved in a separate directory.

In this practicum we will use the *libsvm* implementation of the SVM. You can find it here:

<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

Download the first zip-file in the table of “Interfaces to LIBSVM”, called A simple MATLAB interface LIBSVM from the authors at National Taiwan University. And add the path with the **addpath** command.

The data set we are working with contains seven classes: basketball, castells, marathon, polo, rowing, sailing, and snowboarding. Class 3-7 were collected by Li-Jia Li and Li Fei-Fei (ICCV 2007). For each class we have 50 images, divided in 30 training and 20 test images.

Now you can run the BOW-program which will compute the BOW pipeline, and see some of the results:

➤ BOW_script

This script performs the detector, descriptor, vocabulary, assignment and classification procedures on a row. It can take a while to compute, especially the normalization of the patch sizes takes a lot of time. As you can see although we made relatively naive choices for all of the procedures (except the SVM) we get a classification score of almost 50% which is not that bad for 7 classes (random would be 14 %). Try running the BOW_script again. It will not recompute

the already computed detectors and descriptors, and only recompute the vocabulary and image representations. Only when you change some of the settings of the detector or descriptor, will they be recomputed.

III. Improving the BOW approach

We will next discuss the various stage of the BOW approach.

1. Feature Detection:

implemented detector: The feature detector which is implemented takes randomly points out of the image at a random scale. You can change the minimum and maximum scale and the total number of points for each image. The results can be viewed with:

```
➤ display_features(eventopts, detect_opts.name, 9)    % results of detector on image 9
```

An overview of the data structures used in this assignment is given in **Table 1**.

mandatory assignment I: Try at least one other feature detector and compare the results with the random feature detector. The lecture of Ernest Valveny includes several suggestions.

Compare your detector to the random detector (does it do better than a random detector with many points ?). Provide some results in your report.

2. Feature Description:

implemented descriptor: a very simple RGB descriptor is implemented. It computed the average RGB-values over the normalized patch.

mandatory assignment II: use the SIFT detector and descriptor. There are several implementation on the web. Probably, easiest to use is the UCLA implementation, which can be downloaded from <http://vision.ucla.edu/~vedaldi/code/sift/sift.html> . This function is both a detector and descriptor in one. In our framework it would be best to call it a descriptor, which does not take a detector as input. The output of the function can be split in a detector part and a descriptor part. This will allow you to use the SIFT detector also for other descriptors. Another implementation (LINUX) can be found on <http://lear.inrialpes.fr/people/dorko/downloads.html> . The SIFT will be discussed by Marcal Russinyol.

mandatory assignment III: In the lectures of Javier Vazquez the problems of color representation are discussed. Implement yourself (i.e. not from the internet) a photometric invariant color feature. It might be convenient to use **normalize_features.m** function which extracts and normalizes the patches.

Again feel free to test with any descriptors you can think of or can find on the web.

3. Vocabulary Creation and BOW representation:

implemented vocabulary method and assignment approach: we randomly select a fixed number of points from the image to be the cluster centers, and hence our visual words. Next, we use a nearest neighbor assignment to assign the feature descriptors to the visual words.

You should already know at least one method to improve this.

4. Classification:

implemented classification method and assignment approach: we apply a linear SVM and we forget to apply cross validation to optimize its parameters.

Again you already know several methods to improve this. But also feel free to try other classification methods.

5. Object Detection

optional assignment: use one or more of the part-based object detectors of Pedro Felzenszwalb. Can these be used to improve image classification results? The Part-based detector will be explained in the lecture of Ernest Valveny.

6. Evaluation:

mandatory assignment IV: the evaluation in the report should be based on the average precision (AP) of all the classes. Furthermore, an analysis of the ROC curves or precision-recall curves for some classes and settings should be provided.

Mandatory assignment V: Every group will get an extra assignment. This techniques will be briefly discussed and must be used by the group to which it has been appointed.

data structure	abbreviation	matrix size	description parameters
Detector	DET	$p \times 3$	p is number of feature points by n $[x \ y \ s]$, position and scale
Descriptor	DES	$p \times k$	k is descriptor length
Vocabulary	VOC	$m \times k$	m is number of visual words
BOW - representation	BOW	$n \times m$	n is number of images

Table 1. overview of data structures used.

IV. Practical information

Groups: the practicum has to be made in groups of 3/4 students.

Important Dates:

22 Jan-19 Feb.: The project will last three weeks.

29 Jan. : Extra assignment will be handed out. **Submission of first results in AP.**

5 Feb. : Discussion meeting + **submission second results in AP.**

11 Feb. : Publication of final test set.

12 Feb. : Discussion meeting with groups separately.

15 Feb. : **Final submission** of classification results in AP for all classes.

19 Feb. : Presentation of the project.

22 Feb. : Final submission date for report.

Supervision:

For any questions during the three weeks of the project email (joost@cvc.uab.es) or come to office O/119 in the CVC. Use "PROJECT I" as subject of your emails, which makes it easier to manage. There will be project meetings on Tuesdays to discuss progress. The first Tuesday every group will get its own *extra assignment*. The techniques will be briefly discussed and has to be used by the group to which it has been appointed.

Notes:

The final note will be based on participation (15%), presentation (25%), report (50%) , and competition (10%).

Final Report

The final report has to be submitted on 22th of February. The report should contain the following chapters.

1. **Introduction** (max 1 page)
2. **Feature Detection** (max 2 pages).
3. **Feature Description** (max 3 pages).
4. **Visual Vocabulary and BOW representation** (max 2 pages)
5. **Classification** (max 2 pages)
6. **Part-based detection** (optional : max 2 pages)
7. **Results** (max 2 pages).
8. **Conclusions** (max 1 page)

Do not use more pages than the maximum indicated above ! You get one extra page in the chapter to which the *extra assignment* belongs. The five mandatory assignments should be discussed in the report.

Chapter 2-6 can already include some results to clarify the choices you made. Discuss the theoretical properties you expect the detectors, descriptors, vocabulary, classification and the

detection methods to have. Results can also be negative, such as “We expected non-linear SVM to outperform linear SVM, however, from the results we found that This might be caused by....., etc”. It is important that you analyze why you obtain the results you get (do not get frustrated when you cannot find a good explanation, this happens sometimes).

In Chapter 7, give results in Average Precision for all classes (and analyze some ROC curves or precision-recall curves.). The results should be given for:

1. Using only the color descriptor from mandatory assignment III (without shape and texture descriptors). You can use whatever detector, vocabulary or classification method you like.
2. Using your best performing system (training only on the training data of course !).

Chapter 8 should summarize the most important conclusions from the previous chapters.

Bugs:

For sure there will be some bugs in the code. If you find one mail (joost@cvc.uab.es) and we will email all groups. Thanks !

Final Remark:

The first project for module 3 is organized by Josep Lladós, Ramon Baldrich, Ernest Valveny, Maria Vanrell, Robert Benavente, Javeir Vazquez, Marçal Russinyol, and Joost van de Weijer. We hope you learn a lot and enjoy the coming month.

Good luck!