Computer Vision Project
Winter 2017/18
Franz Köferl, Vincent Christlein, André Aichert,
Tobias Geimer, Franziska Schirrmacher

Sheet 4, starting from December 11th, 2017, due February 2nd, 2018, 14h

Introduction and General Comments:

- We provide the 'horse' subset of the PASCAL VOC 2007 dataset and some background images plus labels for training and validation. Please download them from:

  `https://faubox.rrze.uni-erlangen.de/dl/fiWCLxzTJ9hWmL7EXp8S2xqT/data.zip`.

- We also provide in StudOn a code skeleton ('exercise4.m'). However, you are not obliged to use it.
- For this exercise we recommend to use the package 'vlfeat'. It provides MATLAB interfaces to SIFT and its dense version PHOW, as well as other tools, e.g., k-means, etc. See `http://www.vlfeat.org/install-matlab.html` how to setup 'vlfeat' for matlab. Note: you can also use 'C' or 'C++' or even 'Python' (in Python it might however be more difficult to get/write vlfeat-wrappers).
- Test your algorithms with different parameters and try to get a deeper understanding regarding their behaviors. Show your implementation and your results to one of the advisors.
- Further notes:
  - `parfor` instead of `for` brings a huge speed improvement.
  - All commands provided by 'vlfeat' assume column-major order of the matrices.

Exercise 4.1: Visual Image Recognition / Image Retrieval

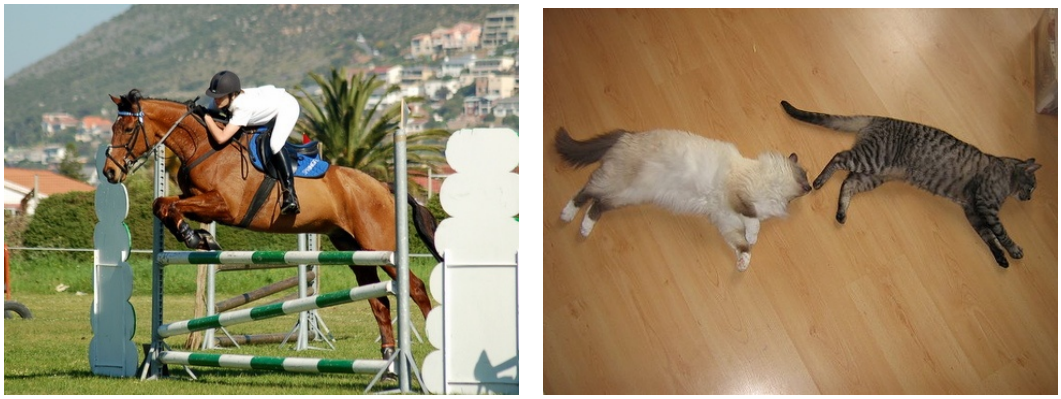

Fig. 1: Image classification: 'horse' or 'no horse'?

In this exercise you will create a very basic framework for visual image recognition which will later be extended in Exercise 4.2. The system will be tested on a subset of the PASCAL VOC 2007 benchmark dataset. We will focus on the 'horse' dataset, where we will classify each image in {'horse' / 'no-horse'}, see Fig. 1 (actually we compute the retrieval performance, i. e. given a horse image, how likely is the chance to retrieve a horse image from the dataset). Therefore, the *bag of (visual) words* model for image recognition is going to be used, where each image is represented by a global vector which is

formed by local feature descriptors. The global descriptor is eventually used for the classification / retrieval.

a) Codebook generation

To generate a codebook, a representative number of descriptors needs to be selected. Therefore, use a random selection of about 150000 descriptors from the *training* images (a subset of 1000 images is enough). Note: a helpful function might be `vl_colsubset`. Compute a codebook (aka vocabulary / background model) with the help of k-means ($k = 1000$). Details:

- Descriptor: Use the SIFT descriptor (`vl_sift`). Make sure that you convert the image to grayscale and float (`im2single`).
- k-means: you can use either MATLAB's k-means, or probably faster: vlfeat's version (`vl_kmeans` with the options: method=ann, initialization=plusplus).

b) Encoding

Encode all (no subset) image descriptors using the vector quantization (VQ) representation, i. e., a histogram of visual words is used (see lecture slides 40–47). Normalize the histogram by the l_2 norm. To get the nearest cluster centers for each descriptor you may use `vl_kdtreebuild` and `vl_kdtreequery`.

c) Classification

The goal is to classify the encodings of the test set using a linear SVM. Use the encodings related to the images of `horse_train.txt` as positive training data (label = 1) and the ones of `background_train.txt` as negatives (label = -1). You can use `vl_svmtrain` (options: solver=sdca, BiasMultiplier=1), the scores are then given by $w^T \cdot x + b$. The SVM regularization parameter C should be cross-validated (see bonus exersice), for now you can just set $C = 1$. Note: for `vl_svmtrain` you need to compute λ dependent on C, i.e. $\lambda = 1/(C \cdot \mathrm{N})$, where N denotes the number of images.

d) Evaluation of the Retrieval performance

Due to the high data imbalance you actually won't get a good *classification* result (= accuracy), in contrast to the retrieval performance. Thus, compute the average precision (what does it compute?) for your encodings of the test set. Note: `vl_pr` might be helpful. How many true positives do you get among the first 50 images with the highest schores?

Note: I get a retrieval score of 0.32 for the basic setting – this might however be different for you.

What happens if you change the fraction of training images to 0.5 (0.1)?

What could you do to improve the actual classification rate (hores/no horse)?

Exercise 4.2: Pipeline Vartiations

a) Exchange the SIFT descriptor with PHOW

Use the PHOW descriptor (`vl_phow`) (options: step=4, floatdescriptors=true). PHOW is a dense version of the SIFT descriptor applied with multiple bin sizes to achieve some scale invariance (rotational invariance is lost). What is your AP after this step?

b) Encoding with VLAD

Encode each image using the VLAD representation (see lecture slide 48) with l_2 normalization (don't use `vl_vlad`!). To get the nearest cluster centers for each descriptor you may use `vl_kdtreebuild` and

`vl_kdtreequery`. For this representation a smaller number of clusters for the k-means clustering can be used (e.g., $k = 100$).

c) Normalization of the VLAD descriptor

Implement the power normalization and the intra-normalization (see lecture slide 48). Compare the average precision values: which one works better?

Exercise 4.3: Bonus

a) RootSIFT

Normalize the descriptors with the Hellinger kernel [1]. SIFT vectors can be compared by a Hellinger kernel using a simple algebraic manipulation in two steps: (i) l_1 normalize the SIFT vector (originally it has unit l_2 norm); (ii) square root each element. Which average precision do you achieve? What might be the reason for the better average precision?

b) Implement a spatial pyramid

In order to bring back some locality, compute a higher dimensional encoding using a spatial pyramid with two levels, i. e, a maximum level of 1 (see lecture slide 63). What is your AP after this step?

c) Proper parameter cross-validation Different parameters should be validated in advance using solely the training data. Try different C parameters, $C = 10^r$ for $\{r \in \mathbb{Z} | -3 \leq r \leq 3\}$ and see which one works best, evaluate this using an inner 5-fold cross-validation.

References

[1] Relja Arandjelović and Andrew Zisserman. Three things everyone should know to improve object retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2911–2918. IEEE, 2012.