395T Visual Recognition, Fall 2012
Assignment 2: Recognizing object categories with bag-of-words models

Out: Friday Sept 21
Due: Friday Oct 5, 11:59 pm

Goal

For this assignment, the goal is to implement a bag-of-words image recognition system, and then evaluate it on natural images containing four object categories. Feel free to use existing software for any of the steps below (when available); cite any components you have borrowed in your writeup.

What to implement

Below are the main steps you will need to include. Aside from following this basic pipeline, the details for the implementation are up to you.

1. **Detection**: Determine where in the image to extract local feature descriptors. You could use a single interest operator (e.g., DoG, MSER, Harris), or a combination of interest operators, or else forgo interest point selection and densely sample feature points.

2. **Description**: Choose a local image descriptor to extract at each of the selected points (e.g., SIFT, SURF, pixel intensities in the patch).

3. **Visual vocabulary**: Sample some descriptors from the training images, and apply k-means (or some other approach) to them to quantize the feature space into "visual words".

4. **Bag of words formation**: Map the set of features from an image into its bag-of-words histogram, using the visual vocabulary constructed above. Do this for all training images and all test images.

5. **Training**: Build a multi-class classifier of your choice (e.g., SVM, Naïve Bayes, k-nearest neighbors,…). Use the training partition provided (as given by the subdirectories).

6. **Testing**: Apply the learned classifier to categorize each test image into one of the four categories.

Provided data

Download the image data from the course webpage. The classes are airplanes, motorcycles, faces, and cars. The file has train and test subdirectories per category, with 50 test images per class and up to 500 training images per class.

What to include in your submission

Submit a tar file containing your commented source code, and a pdf report that includes the following:

- A description of precisely how your approach was implemented, listing the steps and details of any choices you made.
- A visualization of a couple visual words using examples from the data.
- Report the classification accuracy – the percentage of images in the test set that are correctly classified by your program.
- Compute and display a confusion matrix, and explain the result.
- Discuss the overall performance of your system.
- Test two variants of your system, and design comparative experiments to show how they differ. Carefully *explain* the impact on accuracy. These two variants could be based on adjusting choices you made in 1:6 above, or some other feature that you optionally add to the system (e.g., using segmentation, context, feature selection, spatial pyramid match kernels, vocabulary-trees). Illustrating the impact on accuracy could go beyond the overall accuracy number, such as impact on particular classes, types of examples, etc. *Note: please avoid simply reporting the impact of tuning parameters, unless you have a good conceptual explanation for the impact*.

Grades will be determined based on implementation completeness as well as the insights demonstrated by the report discussion. Use figures or images to make your point clear when appropriate.

Submit your tar+pdf files via email to Austin, (austin@cs <you know the rest>). Also **submit a hardcopy** of the pdf report to Kristen.

Acknowledgements

The dataset is provided courtesy of the Caltech Vision Group, and collated by Svetlana Lazebnik.