Bag of Word: Place Recognition

Background

We will use the bag-of-words (also known as bag-of-features, bag-of-visterms) model for object recognition. First of all, check the bag-of-words tutorial at http://people.csail.mit.edu/torralba/shortCourseRLOC/index.html

Load the KTH database images in the same way as in the SIFT lab:

Include the sift package in Matlab's path: addpath('../sift');
- Have a look at the documentation

- In the remaining of the exercise, we will use images from the KTH-IDOL2 database, containing a sequence of images recorded by a robot driving through an office building. Get sequence information sequenceInfo = getIDOLSequenceInfo('min_cloudy1');

Calculate Features

We have pre-calculated all the SIFT interest points and detectors for you. There is a pointer to the file containing the SIFT descriptors in sequenceInfo(i).siftFileName. The sift file contains an interest point in every row and the first columns are xpos ypos scale and orientation followed by the 128 columns of sift descriptor.
- Then store each descriptors, and image label sequenceInfo(i).placeID so that you can correspond the sift vectors and their associated label.

Cluster Features

After you have all the descriptors, use k-mean clustering to cluster our descriptors.
To do this, put all the descriptors (excluding the xy pos and scale and orientation) from all the images into a big matrix with each descriptor as a row vector.
Then use MATLAB's kmeans function to cluster descriptors using different k values. Typically, a large number of clusters (e.g., 500, 1000, 2000) has been used in the literature. You need to experiment with the number of clusters with respect to the number of local features obtained in the training data.

Map a raw SIFT descriptor to its visual word:
Each raw descriptor is assigned to the word vector (cluster center) it is nearest to in terms of Euclidean distance. [see provided `dist2.m` code for fast distance computations; note that `[minvals, mininds] = min(D, [], 2);` returns a vector containing the minimum value per row of the matrix D along with the column indices where each of the mins are found.]

Map an image's features into its bag-of-words histogram

The histogram which we will call the signature for image I_j is a k-dimensional vector: $F(I_j) = [freq_{1,j}, freq_{2,j}, ..., freq_{k,j}]$, where each entry $freq_{i,j}$ counts the number of occurrences of the i-th visual word in that image, and k is the number of total words in the vocabulary. In other words, a single image's list of n SIFT descriptors yields a k-dimensional bag of words histogram. [Matlab's `histc` is a useful function] Finally label every histogram with the label sequenceInfo (i).placeID so that you know which room the signature corresponds to.

Testing

Now go back and load the features for the other sequence getIDOLSequenceInfo('min_cloudy2'); This new set has not been trained on so it can be used for testing the technique for similarity. Now step through each image and using the same set of visual words computed on the previous sequence compute a histogram signatures for an image and measure that histograms distance to all the signatures computed from the previous step. Finding the minimum distance signature and its associated placeID gives you your classification for this new image. Compare that and the actual placeID from the structure to check for successful classification.