Robot Learning Project Report

Chaitanya Mitash(cm1074)

March 29, 2016

This report provides details about the overall approach used for the Object classification, detection and recognition of objects in the shelf environment using learning techniques provided the training images.

1. **Object Classification:**

   Methods explored for feature extraction and training the classifier : HOG, SURF, SIFT, k-nearest-neighbor, SVM, Bag of visual Words.

   Implementation used (Best Result):-

   (a) Used the *extractHOGFeatures* method in MATLAB to extract features within the given bounding box and associated it with the given object label. Zero padded the training features to get constant size vector for each image. Tuned cellSize parameter for extracting HOG features to get the optimal results.

   (b) For the purpose of classification, used 2 separate classifiers and based on the confidence values, predicted the label. The two methods used were *fitcecoc* and *fitcknn* in MATLAB. The first one is training of an error-correcting output codes (ECOC) multiclass model using support vector machine (SVM) binary learners and the second one is a k-nearest neighbor classifier. Both individually gave 90% correct classification, however when used together the result improved to 93%.

   (c) I had also attempted to use SURF features with Bag of visual words classification which uses multiclass model using support vector machine (SVM) binary learners but the accuracy was 70%.

2. **Object Detection**

   Methods explored : Cascade Image Detection, HOG, SURF, SIFT, SVM, knn, sliding window.

   Implementation used (Best Result):-

   (a) In the first stage I used the cascade detection. The detector uses a sliding window and simple classifiers to classify parts of the image as negative(not containing object) across multiple stages. The number of stages and allowed false positive rate in each stage are significant parameters to the method.

   (b) The classifier used at this stage was HOG. I also tried with Local Binary Patterns but HOG gave better results. For the training the positive samples were the images containing object and the bounding box defining the object in the object. The negative samples were the images with that object bot present in the scene.

Figure 1: Object Recognition for Crayola

(c) At this time, I used the *step* method over the detectors to get the most probable bounding boxes. The training was such that we get quite a few false positives.

(d) Now, I iterate over these bounding boxes and extract the SIFT features from each of these.

(e) Separately I have extracted features from the training data and stored feature vectors for each object.

(f) I match the two set of features and based on threshold value of matching predict if the object is detected.

(g) To tune the threshold values I used validation sets from the training data and did some manual tuning.

(h) I also implemented sliding window of a fixed size and HOG based classification, which gave me 55% accuracy. This was maybe because of the difference in the feature vectors of training and testing. I tried the same size window for training and still the result did not improve.

(i) Another attempt was to match the SIFT features of the whole scene with the features for each object and tuning the threshold. This gave me similar accuracy as the proposed solution i.e. 70%

Object Recognition:

(a) The method used for recognition is the same as for detection. First we obtain the most probable bounding boxes using cascade detection with HOG features.

(b) Now, we iterate over these bounding boxes to get the bounding box with most number of SIFT features matching the feature set from training images.

(c) Also attempted the sliding window and HOG classifier to get the best match but the result was not good because of the same factor as in detection.

(d) The accuracy of the recognition was off by quite a few pixels. While debugging I found it giving decent results for crayola as you can see in the image above. I am guessing, learning the threshold was off for a few objects. The reason could be the difference between the exact bounding boxes in validation set vs the bounding boxes here which might contain parts of the shelf and background.

To improve :-
The detections from the cascade detector seemed very reasonable.However, it had quite a few false positives which were a part of the shelf and the background. The SIFT based classifier did not work well on that. Thus, to further improve the solution I would have segmented out the objects from the scene as a pre-processing step.