Semi-supervised learning for object recognition in RGBz images

Andrew Duchi
Stanford University
aduchi@stanford.edu

1. Introduction

In this project I am focused on the problem of using depth information in the absence of labeled depth training data for the task of object detection in RGBz images. This task is important for computers to fully utilize both the breadth of image training data that does not include depth sensors while allowing depth-enabled systems to utilize the valuable added information provided by a depth sensor.

The vision of this project is to enable an RGBz system to use currently existing data sets that have only visual images for initial training and refine the object classifiers augmented with depth information that is gathered as the machine operates in a testing environment. The general principle is that a a classifier can initially be learned over the RGB images, then we can use detections in the test data (based only on RGB) to create a classification system that includes depth features by extracting the depth features from high certainty detections.

2. Problem Statement

The specific problem I am tackling is to improve object detection scores in general RGBz images given only labeled RGB images as training data. The datasets I am using are the RGB-D Objects and RGB-D Scenes Dataets from the University of Washington[3]. The Objects dataset will be used for training as it has segmented images of 300 objects from multiple angles and includes depth data so that I can compare my method to performance of a method using depth data in training. The Scenes dataset will be used to evaluate final performance and object detection scores as it contains 8 natural scenes with multiple non-localized objects.

3. Technical Approach

My system for object detection is a four step process:

1. Learn RGB Object Detector
2. Detect Objects using RGB Detector in Test Data
3. Create Depth-Based Object Detection Model
4. Detect Objects using RGB-Z Detection in Test Data

The specific process using for learning of the RGB detector and detection using our RGB model are not central to this project, as the addition of depth features should work with any type of RGB model. For this project, I used Histogram of Oriented Gradients (HOG) for my RGB features and performed classification of bounding boxes using a Support Vector Machine. These were selected because HOG with SVM is a frequently deployed approach to object detection and is the backbone of many detection systems, is relatively easy to implement, does not require the intensive feature matching of SIFT based systems, and has been demonstrated to work well in practice.[1]

For augmentation with depth data, I am initially using HOG features computed from the depth images and plan to move to the state-of-the-art Spin-Images depth information representation. HOG features should function reasonably well as HOG normalizes the image such that different object depths will not be an issue and focuses on finding transition areas in the image such as strong edges, which is some of the most important information that low-noise depth information can give us. I plan to transition to Spin images because they are specifically designed for depth-based similarity matching and have been shown to achieve good results on this data.[2]

I am pursuing multiple approaches to relearning of the object detection system. The simplest method I am using is to simply retrain our SVM model with the detected objects and use their depth masks; however, this should not improve performance significantly as the labeling given by our SVM must be separable using our Kernel, so we should not expect classification to change. A potential improvement that I am working on is to identify high confidence classifications and use those to retrain the SVM (omitting "difficult" examples that may have been mislabeled) with depth features included.

Identification of "high confidence" classifications is somewhat difficult as margin distance of our examples does not translate to a true probability of correct classification in an SVM as it does in statistical classification methods such as logistic regression. That said, distance to the margin can be used as a first heuristic. I plan to augment this by using clustering in the depth space to identify outliers that may have been incorrectly classified based on the RGB data alone.

4. Intermediate/Preliminary Results

As a preliminary analysis, I focused only on the classification task, ignoring the problem of scanning images for detection, to reduce computational load for my early results. In this task, I use only training data (for training and evaluation, I've not yet moved into the testing phase) with HOG features extracted from the RGB image and the corresponding depth image (128 each for 256 total features). In terms of data, to reduce the need for large amounts of space, I focus on classification of only one object (the "coffee mug") and randomly sampled from all other objects to obtain negative examples. I have approximately 4000 positive and 4000 negative examples.

4.1. Baselines

As a first test, I looked at the classification accuracies of the model after fully supervised training with only RGB data and then also with RGBD data. The results can be seen here:

data	RGB	RGB-Z
Accuracy	95%	.9881
F1-Score	.9592	.9899
Precision	.9323	.9822
Recall	.9877	.997

The results show that on the training data, RGB classification performs quite well, and adding depth information can give some increase in performance. However, it is important to note that adding depth information doubles the size of the feature vector, so there is the potential that this increase is the result of overfitting, not actual improvement given that we are only evaluating on training data.

4.2. Semi-supervised Depth Model Learning

The next preliminary experiment performed was to learn the RGB model, then relearn an SVM in which we used the labels given by classifying with the RGB model and use both

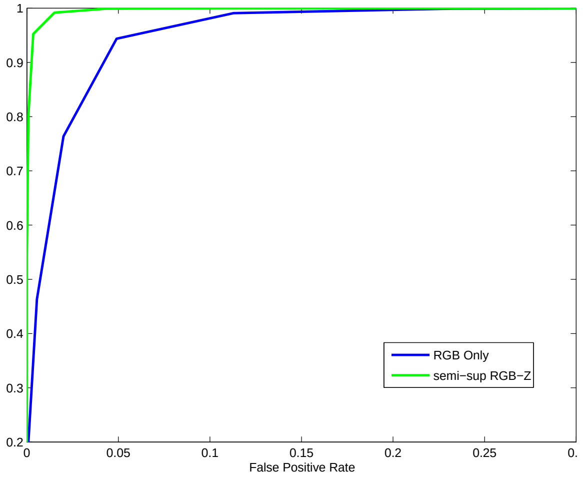

Figure 1. ROC

RGB and depth features. This approach gave the following results:

data	RGB	Semi-sup RGB-Z
Accuracy	95%	.9881
F1-Score	.9592	.965
Precision	.9323	.9376
Recall	.9877	.9940

We can see that the accuracy and F1 score are better than the RGB model and the ROC curve also shows that the performance of the model over RGB and depth data that is learned in a semi-supervised fashion is superior to an RGB model. This result surprises me somewhat, as I would expect that SVM to perform the exact same classification; however, I guess the additional features in a radial kernel space have changed the separability of our data points and clustered the true positives more distinctly from the negatives.

4.3. Improvement with Noise

As a final preliminary test, I note that our initial RGB model has quite high accuracy and seems to be an easy task, so I consider the difficulty associated with augmenting the model with depth data when we have varying levels of incorrect detections. You can see the change in accuracy (positive for improvement) from training an RGB-only model as the false positive rate ranges from 0 to 90% on the training data (corrupting our positive examples). Note that the table only shows up to 40% as both rgb and rgbz images were overwhelmed by the inaccuracy of training at that point and failed to build successful models.

False Positive Rate	RGB F1	RGB-Z F1
0%	.959	.9899
10%	.9511	.9852
20%	.9377	.9785
30%	.9156	.9610
40%	.8471	.9188

References

[1] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection, 2005.

[2] A. Johnson and M. Herbert. Using spin images for efficient object recognition in cluttered 3d scenes, 1999.

[3] K. Lai, L. Bo, X. Ren, and D. Fox. A large scale hierarchical multi-view rgb-d object dataset, 2011.