CS 674: Robot Learning

Project Report

by

Krishna Anantha Padmanabhan
ka478
167007637

28th March 2016

1 Introduction

The aim of this project was to use algorithms that can be used towards robotic grasping and manipulation. This project was divided into three parts, namely **Classification, Detection, and Recognition**. All these parts have been tackled and a few algorithms were taken into consideration for solving these, but I finally ended up choosing the algorithms which I thought would best suit the problem at hand. These algorithms and their descriptions have been described in the following sections. Also, I have addressed the issue of how I adapted the algorithm to the given three scenarios along with results and further discussions.

2 Methods Used

There were many approaches to be chosen from. I have used Scale-invariant Feature Transform (SIFT), K-means clustering algorithm, and Support Vector Machine in my project. In the following paragraphs I will give a brief description about all of them.

2.1 Scale-Invariant Feature Transform (SIFT)

SIFT is an algorithm to detect and describe local features in images. The algorithm was published by David Lowe in 1999[1]. We find interesting points on the object that can be extracted to provide a feature description of the object. This description is used to identify the object when attempting to locate the object in a test image containing other objects. There following key stages are involved in the algorithm,

- Scale-space extrema detection: We detect points of interest termed as keypoints. We use Guassian filters to scale images and then apply a few mathematical functions in order to select a candidate keypoint.

- Keypoint localization: Scale-space extrema detection produces many keypoint candidates some of which may be unstable. Hence, the next step is to perform a detailed fit to the nearby data for accurate location, scale, and ratio of principal curvatures. This allows only good points to stay further for the next step. That is, points that have low contrast, etc are rejected.

- Orientation assignment: Here, each keypoint is assigned one or more orientations based on local image gradient directions.

- Keypoint descriptor: Previous steps found keypoint locations at particular scales and assigned orientations to them. Now we compute a descriptor vector for each keypoint such that the descriptor is highly distinctive and partially invariant to the remaining variations. This step is performed on the image closest in scale to the keypoint's scale.

2.2 K-Means Clustering Algorithm

k-means clustering aims to partition many observations into k number of clusters in which each observation belongs to the cluster with the nearest mean. The standard algorithm was first proposed by Stuart Lloyd in 1957 as a technique for pulse-code modulation[2] and a more efficient version was proposed and published in Fortran by Hartigan and Wong in 1975/1979 [3] [4]. The algorithm is as follows,

- Place K initial centre points in the space.
- Measure the distance between each object and the centres and assign it to the nearest centre.
- Once all of the objects have been assigned a group, recalculate the positions of the K centres.
- Repeat the above steps until the centres are no longer moving. This results in a partitioning of the data space into Voronoi cells.

2.3 Support Vector Machine (SVM)

SVM's are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. SVM has been used for the classification part of this project. The original SVM algorithm was invented by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963 but the latest version of it was proposed by Corinna Cortes and Vapnik in 1993 and published in 1995[5].

I have used SVM to construct a set of hyperplanes to classify data based on the position of the point in the space and to maximize the margin and achieve efficient classification.

3 Adaptation

The algorithm was adapted in order to fit the three parts as follows

3.1 Classification

We have 12 types of objects in our problem. I took the images and created bounding boxes in order to classify them accordingly and used them for training SIFT. The next step was to extract keypoints in the objects. Once this was done, I used k-means clustering algorithm to create clusters that would calculate vectors in each image to train the Support Vector Machine.

3.2 Detection

In order to do detection, we need a good sample of training images from the classification part so that we can have a good accuracy of detection. For this, we will use feature matching and homography to find the objects in our testing images. In feature matching, I used the SIFT features in order to detect objects and to do this I used the Flann matcher. A threshold is set to identify when an object has been detected.

- FLANN Matcher: FLANN stands for Fast Library for Approximate Nearest Neighbors. It is a library for performing fast approximate nearest neighbor searches in high dimensional spaces[6] [7] [8].
- Homography: A homography is an isomorphism of projective spaces. It is a bijection that maps lines to lines [9].

3.3 Recognition

Recognition is a step in which we recognize the space in which an object is present. In other words, we predict the bounding box coordinates of where we think the object is in a test image.

4 Results

The results presented here may not be accurate since the processing time was very high and I could not be sure of the results unless I re-ran the code a lot of times. This is physically impossible since time is limited. I had to keep increasing my clusters since there was a low accuracy at lower number of clusters. I finally settled at 15 clusters after which I could not do any more computations without working on the other two parts of the project.

In Detection and Recognition, the main difference came in to play according to how I selected the images from the training set. On selecting better images with clear orientations, there was an obvious increase in accuracy.

The average scores that I obtained according to the metrics specified in Kaggle were,

- classification 0.6462

- Detection 0.6234
- Recognition 138.03

5 Discussion

The main take away from this project was that I had a chance to test out the algorithms that I had always wanted to do so. In the process, I gained a very valuable insight into how a machine learning algorithm could be used in robotic grasping and manipulation. If I had more time then I would definitely run the classification algorithm more number of times in order to find out the number of clusters at which I get the maximum accuracy. Also, I would have definitely tried to implement more interesting approaches such as Speeded Up Robust Features (SURF) or any genetic algorithms.

References

[1] David G Lowe, *Object recognition from local scale-invariant features*, Computer vision, 1999. The proceedings of the seventh IEEE international conference on Vol. 2. IEEE, 1999.

[2] Stuart P. Lloyd, *Least squares quantization in PCM*, Information Theory, IEEE Transactions on 28.2, 1982: 129-137.

[3] J.A. Hartigan, *Clustering Algorithms*, John Wiley & Sons, Inc, 1975.

[4] J. A Hartigan, M. A. Wong, *Algorithm AS 136: A K-Means Clustering Algorithm*, Journal of the Royal Statistical Society, Series C 28 (1): 100108. JSTOR 2346830, 1979.

[5] Corinna Cortes, V. Vapnik, *Support-vector networks*. Machine Learning 20 (3): 273. doi:10.1007/BF00994018, 1995.

[6] Marius Muja, David G. Lowe, *Scalable Nearest Neighbor Algorithms for High Dimensional Data*, Pattern Analysis and Machine Intelligence (PAMI), Vol. 36, 2014.

[7] Marius Muja, David G. Lowe, *Fast Matching of Binary Features*, Conference on Computer and Robot Vision (CRV) 2012.

[8] Marius Muja, David G. Lowe, *Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration*, International Conference on Computer Vision Theory and Applications (VISAPP'09), 2009.

[9] Bruce E. Meserve, *Fundamental Concepts of Geometry*, Dover, ISBN 0-486-63415-9, 1983.