Problem Statement

Connor Campbell, Chase McWhirt, Jiawei Mo

CS 461, Fall 2018

Abstract

The project defined in this paper is only a small part of a larger project that involves teaching robots how to grasp objects and interact with the material world. Since the robotic hand in this project has no sense of touch, the problem we will be facing is purely optical. Specifically, the fact that the computer can't expect an object to have uniform color at all times. If lighting is not consistent, or if a shadow is cast over the object, the color may change from what is expected. Through this project, we will attempt to find a better means of recognizing objects that accounts for shadows and color variance. We will explore three methods to solve this problem: k-means clustering, neural networks, and support vector machines

1   Problem

Unlike humans, robots do not generally have a sense of touch and rely on cameras for vision. As such, computer vision is an important part of designing robots to interact with the world on their own. Our task is part of an ongoing project attempting to improve a computer's understanding of a three dimensional environment through machine learning and computer vision and developing computer algorithms to help robotic arms determine the appropriate path to grasp an object. At the moment, the computer is using two stationary cameras placed at different angles facing a robotic arm and an object on a table, and analyzes their feedback to determine the arm and object's locations. Our project will focus on the computer's vision. Specifically, its ability to identify objects in its camera feed.

The current setup has a few problems. To begin with, lighting diffusion can make it difficult to identify an object. When the hand is covering the object, the shadow changes the color captured by cameras. Also, if the environment's light intensity changes, it will result in a color change to captured streams. The computer is expecting pixels representing the object to be in a certain range in the RGB spectrum, and such changes will potentially move the apparent colors out of the expected range. As a result, the computer fails to recognize the presence of the object it's looking for. We could expand the expected color range, but that could cause the computer to recognize parts of the image that are not the object, such as the background, as part of the object. Our goal is for the computer to recognize the object without that issue.

2   Solution

We will be given images of the robot arm in various poses. We will then label the pixels as either part of the object, part of the arm, or part of the background using an image editing software. These images will act as our training and testing data for our algorithms.

We will be attempting a few different methods, including supervised machine learning approaches. At first, we will focus on getting the algorithms to recognize the robotic arm in an image, and move on to identifying various objects if there is time remaining. We will begin with a "straw man" approach that will act as a baseline to compare the other methods to. This straw man will involve running a k-means clustering algorithm over all of the data to determine a color range to accept as part of the arm.

From there, the data will be separated into groups based on the arm's pose determined by the angles that each joint is attempting to be at. One method will involve running a k-means clustering similar to the straw man on each individual group. The second method will involve neural networks. Each data

group will have a neural network trained to scan over the image labelling each pixel. The third method will utilize support vector machines in a similar manner to how the second methods utilizes neural networks.

3   Performance Metrics

About 10% of our data will be set aside for testing our methods. The method that proves most accurate at labelling the pixels will be accepted if it beats the straw man method. Ideally, the accepted method should have at least 80-90% accuracy.