

Problem Statement

Connor Campbell, Chase McWhirt, Jiawei Mo

CS 461, Fall 2018

Abstract

One problem facing the field of robotics is grasping, as the mechanics of grabbing an object are complicated. The project defined in this paper is only a piece of fully achieving robotic grasping. Since the robotic hand has no sense of touch, the problem we'll be facing is purely optical. When a robotic hand approaches an object, the optical problem is one of occlusion. The robotic hand can obscure complete vision of the object making the current system lose track of where an object is. Another problem is expecting the object to have uniform color at all times. If lighting isn't consistent, or if the robot hand casts a shadow over the object, the color won't be uniform. Through this project, we will have to create an understanding of where the robotic arm is and abstract it out of the scene to solve occlusion. Secondly, we'll narrow the scope of color sets that are looked for in certain areas of an object. This will allow us to expect shadows and, therefore, color variance. Doing both simultaneously will greatly expand the range of use cases for robotic grasping.

1 Background

Before we can talk about the problem, it's important to understand the background of the project. We'll be working on only a piece of a larger project. The larger project is to allow a robotic hand, with only the context of two camera feeds, to be able to grasp an object. We're considering different aspects when thinking about our problems. For example, using different colored backgrounds could affect the success rate of the project significantly. It's important to understand that our video streams will be pixelated view of reality. Any given pixel will contain a wide variety of rays that can be considered real world 2D rays. Thus, making the robotic arm and object stand out are essential. It's also important to understand how the robot is "seeing". Two cameras will be providing video feed to our robotic hand. These cameras will be placed at adjacent corners of the space we are using to view the scene. The pixels from the feed can be projected creating a 3D projection. With two camera feeds, where those 3D projections allow us to determine an object's approximate location in 3D space. Essentially, each rays intersection allows us to determine depth. The scene will also likely take place within a very confined space. Ideally, this confined space would be a box that can itself change locations. When considering these metrics, we can appreciate the challenge that comes with solving the following problems.

2 Problem

Unlike humans, robots do not 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; 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 the arm and the object, and analyzing their feedback to determine the object's position and type.

The current setup has a few problems. To begin with, lighting changes can make it difficult to identify an object. When the hand is covering the object, the shadow changes the light 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 in different levels of light, without that issue.

Another problem is when the robotic hand gets in the way of the cameras. Since the cameras are at fixed positions, the robotic arm will block parts of the object when it moves in to pick it up. As a result, the apparent shape of the object on the screen will change, causing the computer to fail at identifying it and cannot accurately determine its position and shape.

3 Solution

For the lighting and shading issue, we will be attempting a machine learning approach. We will begin by recording the robot arm moving the objects in a variety of ways in a variety of lighting conditions. We will take key frames from these recordings and, using Photoshop or a similar software, label the pixels as either part of the object, part of the arm, or part of the background. We will feed this data into a learning algorithm, then test to see how well the algorithm learned to work around the shading issues.

The occlusion issue is a bit more complex. Any given frame from the camera feed after the arm begins to grasp the object won't necessarily have all of the information the computer is expecting or needs for its calculations. As such, the computer will have to fill in the blanks somehow. One method is by remembering previous information. Before the occlusion occurs, the computer can store a more complete understanding of the object in its memory and fit that to the currently available information. It can also use knowledge of the hand's position to figure out the object's position and orientation.

4 Performance Metrics

We will be focusing on two metrics for the identification. First, the computer should be able to label the pixels in an image as either part of the robot arm or not, with at least 90% accuracy. Second, the success rate at recognizing pixels belonging to an object under any given lighting condition should be as good or better than the current method.