Scope and Design

Computer Vision & Robotic Arm Project

Robert Schneider - February 3, 2015

Intro

Because this project is split up into parts, I will introduce these features based on the part they are present on. The four stages are as follows:

Stage 1: learning computer vision and creating a Skeletal tracking algorithm, that can detect a human body, and trace out a basic representation of the bodies skeleton.

Stage 2: Using computer vision to analyze the orientation of a known object.

Stage 3: Implement a virtual robotic arm simulator that is capable of mimicking the human arm movement generated from stage 1.

Stage 4: Building a physical Robotic arm and controller and implement the previous software.

Design Features

Stage 1 explores the discipline of computer vision by learning image acquisition and analysis. Once an image is acquired, certain methods will be used in order to recognize edges, objects, groupings of color, and other features that I have not come across yet. I am walking through a text by Richard Szeliski on Computer vision, hoping that it will help me understand some of these methods and how to achieve my desired outcome of creating a skeletal representation of a human body, and the other stages I wish to achieve. Some of features that I have considered are multiple cameras for triangulating distance and helping with orientation. Matlab has proven to be very helpful in taking care of some of the lower level functions, which will make my programing easier and give me more time to focus on other implementations.

The goal for stage 2 is to generate a 3D model of an object from an image, and compare it to an already known 3D representation of the object in order to determine the objects orientation. This stage is imperative to the real world application of this project. If we can analyze the Orientation of an object, then we will be able to determine how we need to move in order to be consistent with where we pick up the object by.

Stage 3 involves working with a virtual robotic arm simulator. There are some open source simulators that exist, so some time will go into learning the in’s and out’s of these simulators. What will be a big part of this stage is understanding the Robotic arm workspace, that is the sphere that to robotic arm can move to and exist in. The key is figuring out how to handle objects that are present within this work space, so the arm can manipulate them, but also avoid colliding with them.

Stage 4 will be a physical manifestation of stage 3. The difficulty will be working with hardware, and working with the integration of the robotic arm controller and the software.

As I get further and further in the process, I will discover more design features. A lot of it is still unknown to me, but I will learn as I proceed.

Functionality

A key part of the functionality is going to be the handing of information from one stage to another. The software that analyzes the image, and generates a 3D representation of an object, will pass that information to some form of AI that will know what to do with that data, and how to translate that into actions. The goal of this project is to have as much as possible be autonomous. Once a process is initialized and calibrated, then the system will run continuously. That is, once the software has an adequate representation/grasp of the object in sight, then it will know how to act. On an assembly line, it would continuously repeat actions, and possibly improve its performance based on developing a better 3D representation of the object as it encounters the object more.

User Experience and Interaction

The user will have an interface window to control the robot manually. It will also have the ability to be programed to repeat a set list of movements. An analogy would be controlling the robotic arm with a remote control(computer keys/commands), then having the arm able to loop/repeat the movements when specified. I would also like to implement the robotic arm to mimic my arm using vision, so that when I bend my arm, it bend its arm in the same way and same degree. It could also use this way of control(and loop movement) instead of using keyboard commands. Then, with a camera system equipped to the robotic arm, it will have the ability to learn its workspace, and with some user direction, learn a 3D representation of an object (User direction will include specifying where the object you want to learn is and moving the arm to point to the object, then rotate/flip the object so the camera can get another perspective of the object, and increase it's understanding of the model it knows). Then, once the software has a good understanding of the model, it will be comfortable determining how it is oriented, and pick it up the right way (how to pick it up will also probably need to be specified from the user). I am not entirely sure how much information I am missing regarding the processes presented in this paper. My knowledge of the scope of project will grow as I work through the stages and achieve a better understanding of what will be required. Stay tuned for updated located on my website.