Final Project

William Harrington, Sheetal Konnur
ECE478

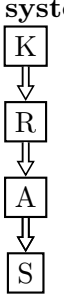
Introduction

This report contains a detailed explanation of the Final project for the Marie Curie group.

Goals

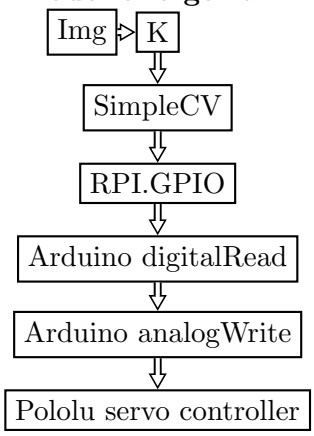
1. Use an arduino to drive the servo controller
2. Use a raspberry pi 2 to do image processing with a kinect
3. Interface raspberry pi 2 and arduino
4. Implement some sort of behavior in the robot based on
5. Add mechanical support for shoulders

High level system model



- K = kinect
- R = raspberry pi 2
- A = Arduino
- S = Servo controller

Model of algorithm



- Img = Image
- K = kinect
- SimpleCV = Python wrappers for OpenCV
- RPI.GPIO = Python wrappers for controlling Raspberry Pi GPIO pins
- Arduino digitalRead = Arduino function to read if pin is high or low
- Arduino analogWrite = Arduino function for Pulse Width Modulation on capable pin
- Pololu servo controller = Device responsible for driving servo motors that move robot

Raspberry Pi 2 Model B

- Justification (Why use a raspberry pi 2 for this?)
  - *Mobility*
    We want it to be small so that it can fit inside the robot. Having to use a full size laptop with a robot limits its mobility significantly and we'd like to avoid that.
  - *More computing power than the previous version of raspberry pi*
    A color image consists of three two-dimensional matrices for red, green, and blue. To process this information OpenCV makes heavy use of matrix operations. It is important that whatever we do with OpenCV that the raspberry pi 2 can handle it.
  - *GPIO for interfacing*
    Our high level system design shows that the raspberry pi 2 must interface with an arduino. The arduino is then responsible for making the robot move. The 40-pin GPIO header on the raspberry pi 2 can be used for that very purpose. Furthermore, its even possible that the arduino can be taken out of the picture altogether.
  - *Easy to use*
    Raspberry pi's are very easy to set up. They come with an already optimized operating system that is capable of using all the software needed to achieve what is outlined in this report.
  - *Can be set up for remote access with USB wireless module*
    Raspberry pi's operating system already has a kernel driver for USB devices which means an off the shelf USB wireless module can be purchased and used for internet access. It is likely that additional software might need to be installed but it is also likely that the raspberry pi can handle it.
- Software Dependencies
  - Simple CV
    - Elegant python wrappers for OpenCV
    - Easy to use and can use with Kinect or regular camera (e.g. Raspberry Pi camera module)
  - libfreenect
    - Open source driver for Microsoft Kinect
    - Required for using Simple CV Kinect functionality
    - Can also be used on its own to implement Kinect functionality

  - RPi.GPIO
    * Used for controlling GPIO pins
    * Easy to use
    * Capable of simple operations such as Input, Output, and PWM
- Resources
  - RPi.GPIO Documentation
  - Simple CV installation instructions
  - libfreenect installation instructions

Figure 1: Raspberry Pi 2 Model B GPIO Mapping