Proof of Concept

CATtitude

Ian Kinstlinger and Gisele Calderon

11/13/2014

To date, we show proof of concept that we can control the opening and closing of a solenoid valve by facial recognition of pre-recorded video. In order to reach this point, we broke the task into smaller steps reaching pivotal proofs of concept along the way. For the next step, we want to pair the solenoid valve control by facial recognition of live streaming video rather than prerecorded video.

Before dealing with sending voltages of a solenoid valve, we wanted to see if we can signal an obvious output from pre-recorded video that can distinguish between faces and no faces. We tested with two videos: ceiling and Rich. We looked for the appropriate placement of an output file .txt within the analyze-video.py Python code. Once the placement was found, we were successfully able to output text files that were able to make facial recognition in a face.txt or no\_fact.txt output.

Subsequently, we aimed to communicate with Arduino by using the Serial.port feature to blink an LED when facial features are recognized by pre-recorded video within the analyze-video.py. The blink recipe was set up to look for Rasp Pi communication only when faces appear.

Because communication between the Arduino and Rasp Pi was established, we could move on to understanding how to translate an output to a voltage sent to the valve instead of a blinking light. Some key additional hardware was added to our Arduino in order to ease this portion of our project. Conveniently, we had access to a RepRap Arduino Mega Pololu Shield (RAMPS) with a good power supply that can send 12V to the valve. The additional feature proved clutch to the progress of our project.

We wanted to understand how we can “speak” to our RAMPS with the familiarity of g-code firmware. We used Pronterface as an intermediate step necessary to understand whether we can send g-code to order opening and closing valve commands. Pronterface is a 3D printing software that is a g-code sender. Then, we used Marlin3D printer firmware that essentially maps out the RAMPS board for proper communication and manipulation with python. We were able to find the fan port which we connected to the solenoid valve.

Back to our analyze-video python code, we directly mapped out g-code to where we send opening and closing voltages to our valve from the recognition of faces in pre-recorded video. We have reached this stage for successful proof of concept!