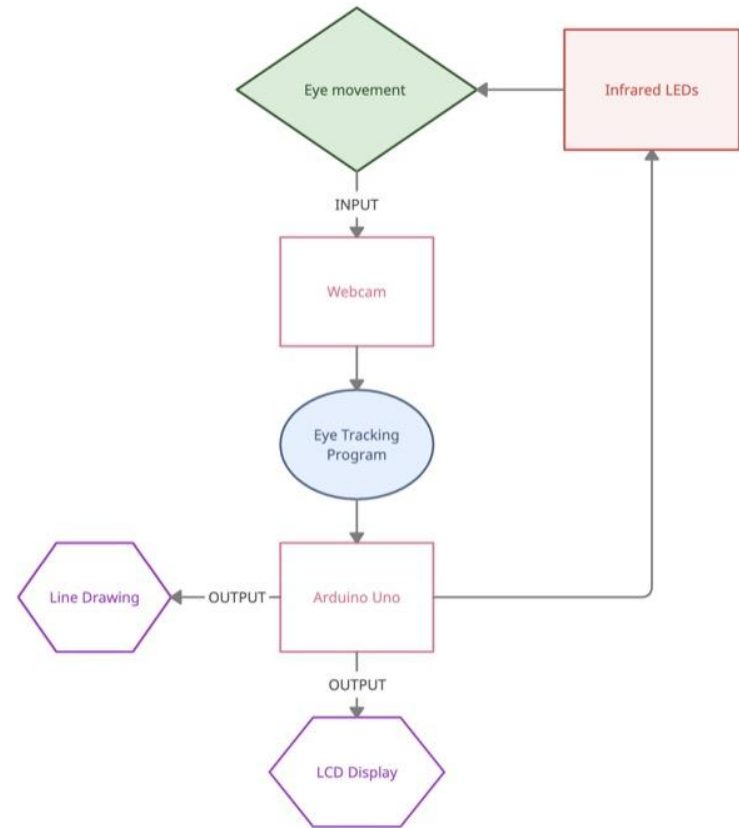


Eye-Art: Eye-tracking Art-making with Infrared LEDs and Webcam

Maitreya Rose and Zora Che



Eye-Art: Eye-tracking Art-making with Infrared LED and Webcam

Zora Che, Maitreya Rose

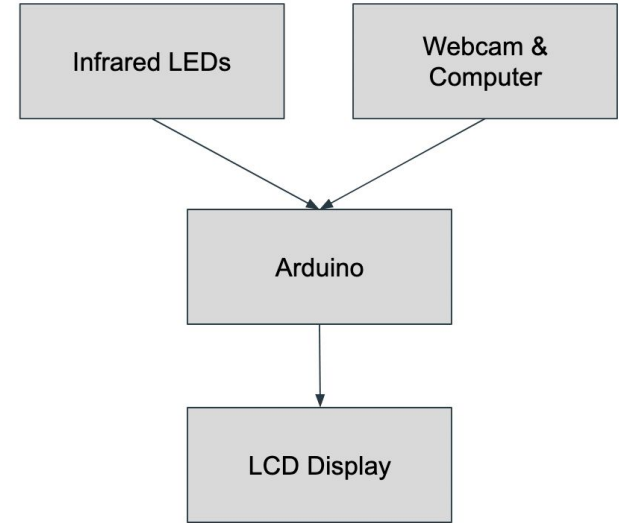
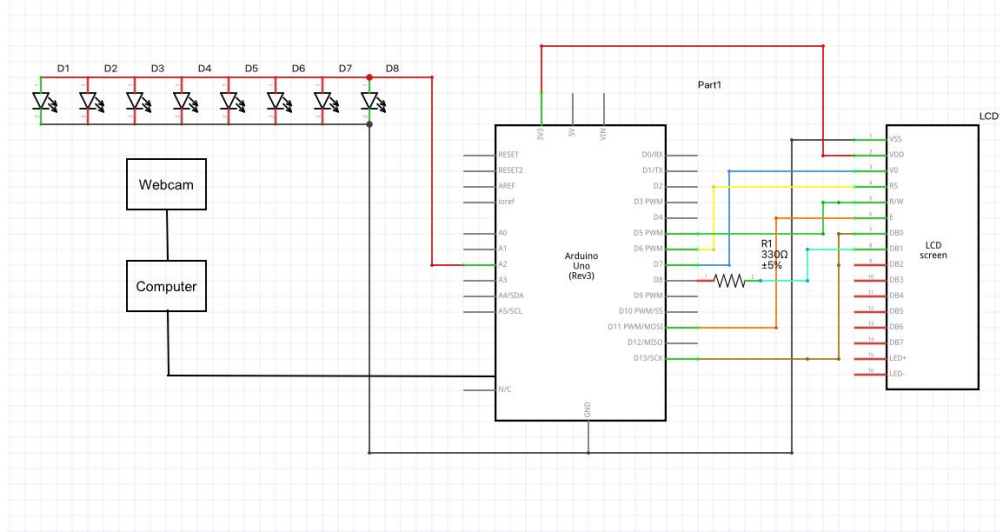
Abstract

Utilizing the reflective property of the iris to infrared light can produce accurate eye-tracking with a low budget. In addition to aiding people with disabilities, eye-tracking provides a novel tool towards interactive art. This project would offer insights into art-making using eyes, as well as art-making for people with disabilities.

To eye-track, we use a web camera with its infrared blocker removed in order to pick up on the infrared reflection of the irises when facing an arrangement of infrared LEDs. The Arduino processes the signal of the infrared LEDs and then maps the eye movements as lines on an 84cmx48 cm LCD display. A real-time eye tracking script in Arduino-compatible MATLAB is adapted to track coordinates of eye movement.

For stability of the program, participants are asked to place their chin on a chin rest while making eye-art.

Schematic and Block Diagram



Parts List

- Webcam (that may have infrared filter removed, such as Logitech c615) (arrived today)
- Infrared LED x 8 (available for pickup)
- 22 AWG wire (~ 2 meters) (available for pickup)
- Chin rest (A simple version would be cushion with tape marking) (making)
- Display
 - Graphic LCD 84x48 - Nokia 5110 [<https://www.sparkfun.com/products/10168>] (ordering)
 - Header for LCD to avoid soldering
 - Computer screen (obtained)
 - For a more complex configuration of current project to incorporate more detailed rendering

Current Focus

- Camera disassembly process
 - Removing the IR filter to track eye motion
- LED display spec sheet
- MATLAB coding to map the coordinates to the LCD display
 - Using coding to map eye movements
 - Adapting output coordinate as coordinates instead of as a final image for input into our LCD screen
 - Finally we need to adapt these coordinated to line code for the LCD screen