Eye-Movement Based Electronic Wheelchair System

Date: May 2020

Institution: ANITS (Anil Neerukonda Institute of Technology and Sciences)

Platform: Raspberry Pi 3B+

Language: Python 3.x, OpenCV 3.3.0

1. Project Overview

This project was developed as a "boon" for individuals suffering from **Quadriplegia**, a high level of paralysis where the person can only move their eyes. Unlike previous systems that required a bulky and costly laptop running MATLAB, this system utilizes a **Raspberry Pi** to provide a portable, cost-effective, and low-latency solution.

2. Methodology & Algorithms

The system processes live video from a head-mounted IR camera to track iris movement and convert it into wheelchair commands.

A. Iris Detection (Centroid Algorithm)

- The system detects the eye using the traditional **Viola-Jones Algorithm**.
- To reduce latency, the eye region is cropped and converted to grayscale, then to a binary image.
- It traverses the image along the X and Y axes, averaging the coordinates of all black pixels (zeros) to find the **Centroid Point** of the pupil.

B. Directional Mapping (Threshold Algorithm)

The eye image is divided into three sections. The concentration of black pixels determines the movement:

Eye Position	Action	Logic
Up	**Straight**	Pupil centered in the top region.
Right	**Right**	Pixels concentrated in the right section.
Left	**Left**	Pixels concentrated in the left section.
Down	**Stop**	Pupil centered in the lower region.

3. Hardware Configuration

- **Microcontroller:** Raspberry Pi 3 Model B+.
- **Camera:** IR Web Camera (mounted ~12-15cm from the eye).
- **Motor Driver:** L293D or L298N IC.
- **Actuators:** DC Motors for wheelchair propulsion.

GPIO Pin Mapping (Physical/Board Pins):

- **Motor 1:** Pins 16, 18 (Input) and Pin 22 (Enable).
- **Motor 2:** Pins 23, 21 (Input) and Pin 19 (Enable).

4. Safety Features: Blink Detection

To give the user total control, a **Blink Detection** logic acts as an ON/OFF toggle.

- If no black pixels are detected (eyes closed) for five consecutive frames, the system initiates a stop.
- After four consecutive blinks, the system toggles the motor state between ON and OFF to prevent accidental movement.

5. Troubleshooting Guide

If the system is running but the wheelchair is not behaving as expected, check the following:

Issue	Potential Cause	Solution
Motors "Hum" but don't turn	Insufficient Current	Ensure DC motors are powered by a separate 12V battery, not directly from the Pi's 5V pin.
Wrong Direction	Reversed Polarity	Swap the wires on the Motor Driver output terminals for the affected wheel.
No Eye Detection	IR Lighting / Distance	Ensure the IR LEDs are active (for low light) and the camera is exactly 12-15cm from the eye.
High Latency	Resolution Overload	Check if the Haar Cascade is processing the full frame instead of the cropped ROI.
GPIO Error	Improper Shutdown	Always run GPIO.cleanup() at the end of the script to reset the pins.