The software system of WAACEYBOARD is divided into two parts. The low-level programs to be run on the Raspberries Pi mainly for eye-tracking and sensors data integration, and the high-level part to be run on the Google Glasses, mainly for the user interface and text-to-speech.

Raspberries Pi

Processing the images at the rate of 30~40 fps, constantly updating the positional data of the eye.

The positional data is communicated serially to Google Glass via Bluetooth. To the Glass OS, the Pi appears effectively as a peripheral BT device.

The positional data is in the format (x, y). (NaN, NaN) indicates no pupil detected in the current frame.

Also, whenever the IMU sensor is triggered. Ad hoc signal is relayed to the Glass.

Data type: (x1, y1)>(x2,y2)>(x3,y3)…(NaN, NaN)…(NaN, NaN) > (xn, yn)

(Emergency)

1. Filter out colour channels and keep the gray scale image only
2. Use an edge detection algorithm to leave only the hollow geometrical shape highlighting the boarder of high colour contrast
3. Locate the region of interest by detecting the round/oval feature
   1. If no region of interest detected, output (NaN, NaN)
4. Estimate the pupil centre
   1. Method 1: Approximating the pupil as an oval and find the centre where the line sections meet
   2. Method 2: Taking the average between the top and bottom, left and right pixel distance
   3. The raw pupil centre is a normalised ranging from 0 to 100 for both x and y axis
5. Apply the Kalman filter to the time series data, Kalman filter is a statistical method to filter out system’s statistical noise, smoothening the trajectory.
   1. When a new value is input to the filter, it will weigh the relative “importance” of this new piece of information based on the statistical variance, before updating the value
   2. Say, if there is a sudden external shock, causing the pupil location to displace erratically, the model will judge the new value less “important” based on its haphazardness and rely more on the past trajectory
   3. It is essentially a statistical model with memory of the past
   4. (NaN, NaN) input will be neglected, and use model to extrapolate for the missing pupil position during blinks

Google Glass

Thee keyboard interface is built in as a software using Google Glass’s API (Android Oreo 8.1). It will update the interface at the same rate as the BT serial data update.

1. Calibration will be done whenever the WAACEYBOARD is being worn
   1. Have around 2 seconds of sampling period for each corner
   2. During the sampling period, the coordinate data is stored as an array/list
   3. Take the averaged value of the coordinate and this will be pinned as the reference for linear interpolation.
2. When serial data is sent to the Glass, refer to the calibrated value to convert the raw coordinates to the corresponding pixel coordinate in the interface
3. For each element (button, arc, etc.) in the interface, define its centre and boundary.
4. At every moment, find the nearest centre to the current eye gaze coordinate and infer whether user is looking at the corresponding region (Nearest neighbourhood, graph search, calculating Euclidean distance and cosine similarity, other algorithms.)
5. Start the timer when gaze stays at the same region continuously
6. Refresh the interface and light up the corresponding region gradually
7. When the duration reaches the threshold, activate the corresponding command (Hover select)