

Camgaze.js: In-browser Eye Tracking and Gaze Prediction

Purpose: To enable a modern web-browser to determine where the user is looking on the screen, solely using HTML5, a commodity camera, and whilst preserving user privacy

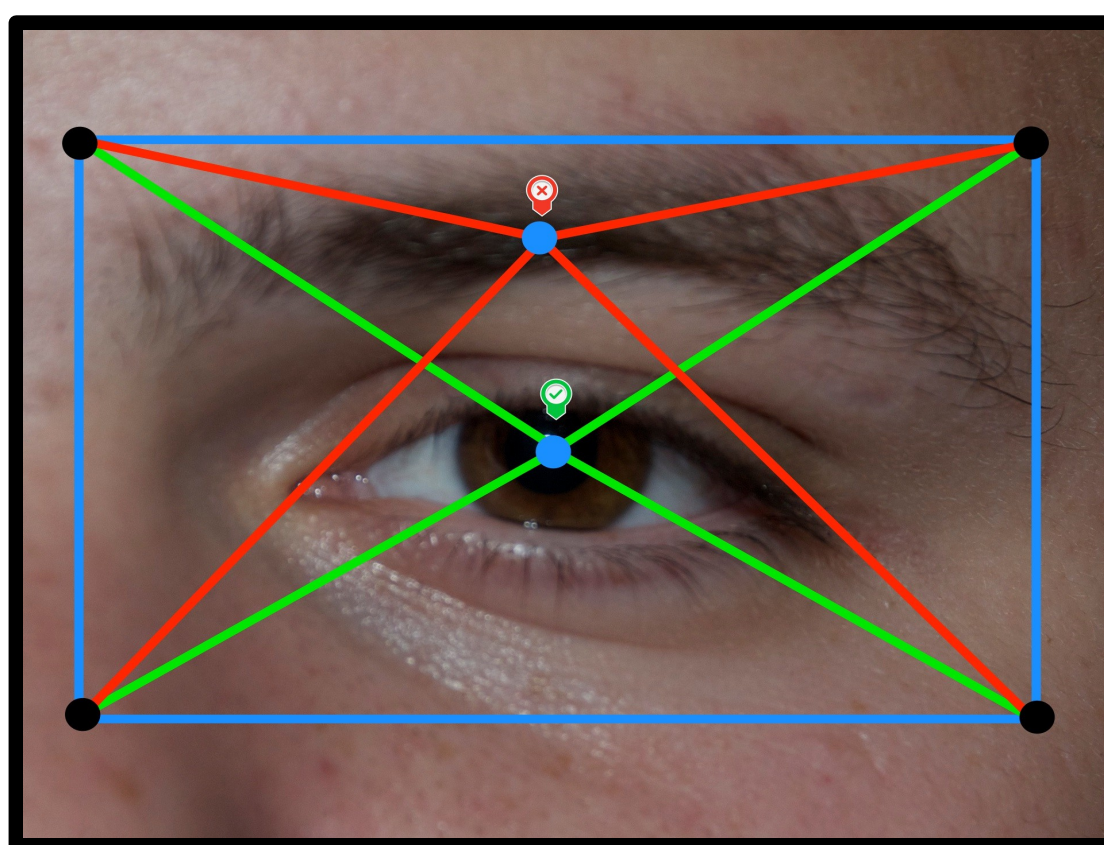
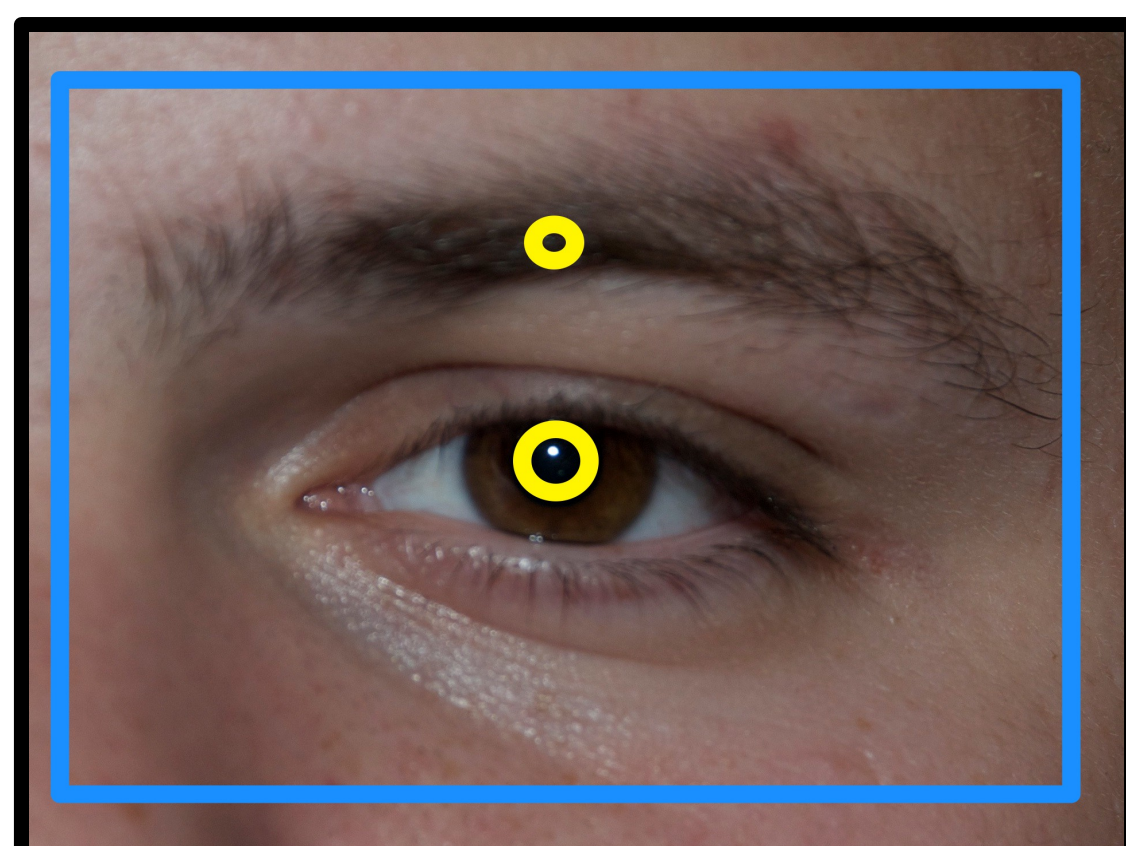
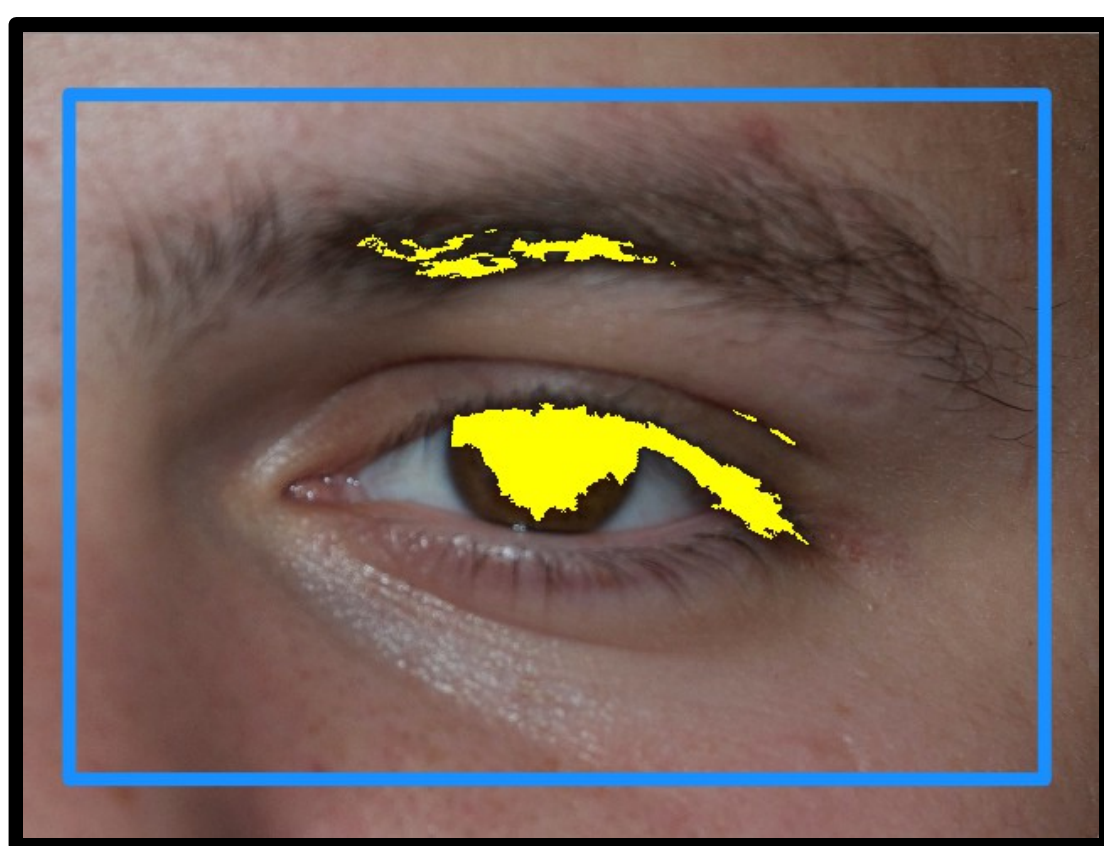
Motivation: Using a web-based eye tracker, we can determine what the user finds important on the screen whilst minimizing user effort and maintaining user privacy

Overview

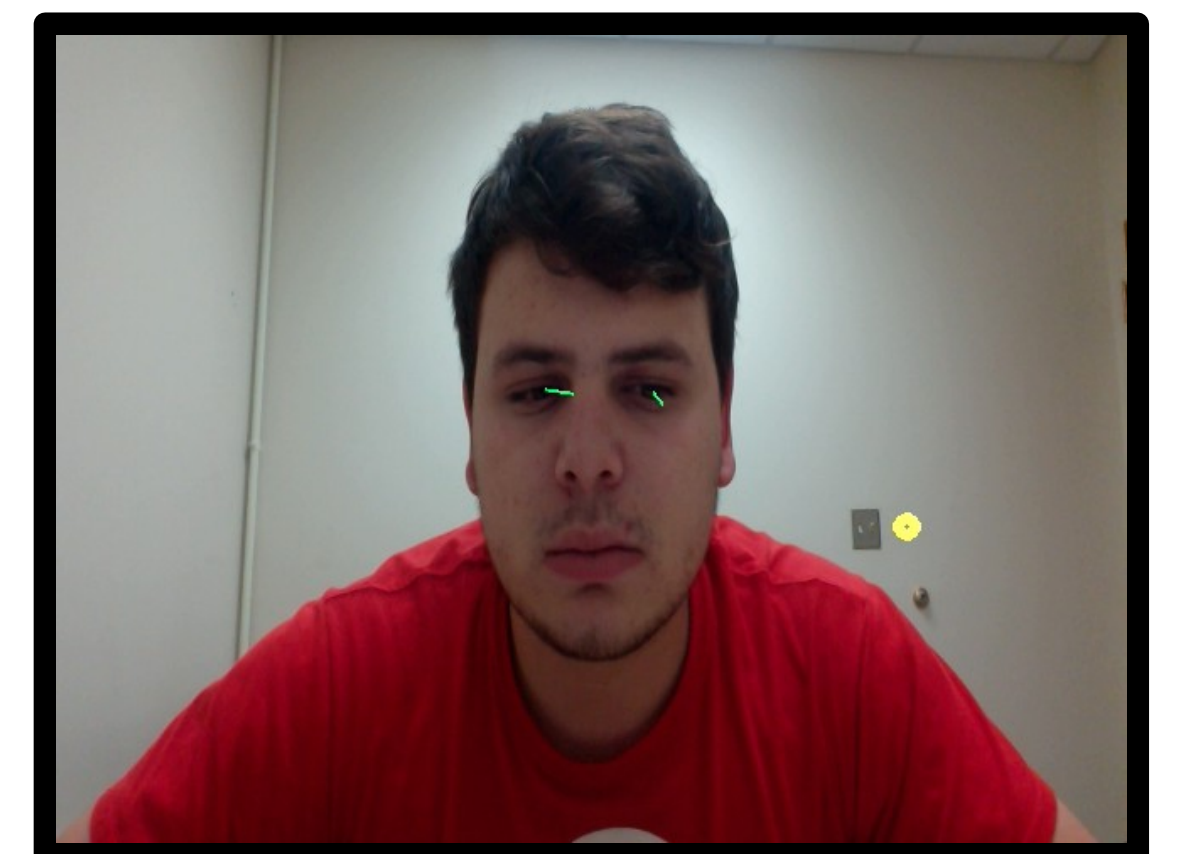
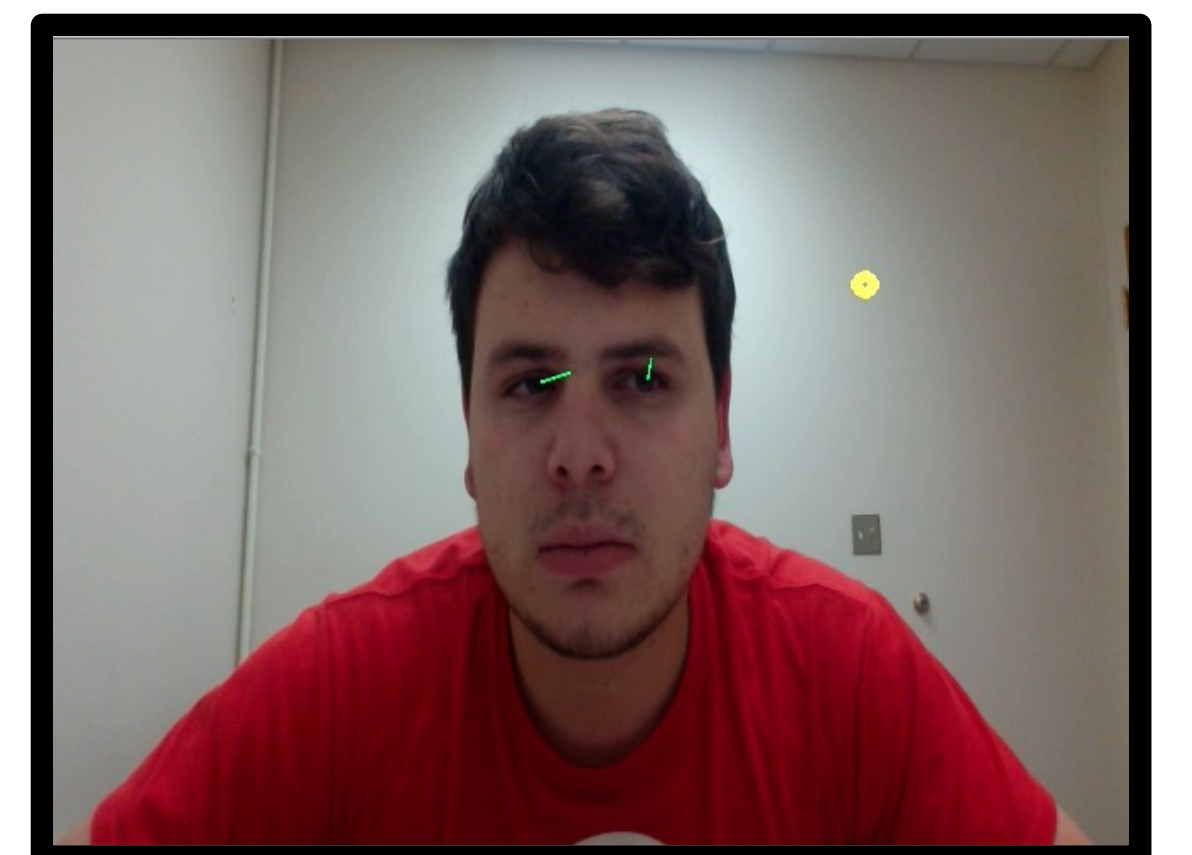
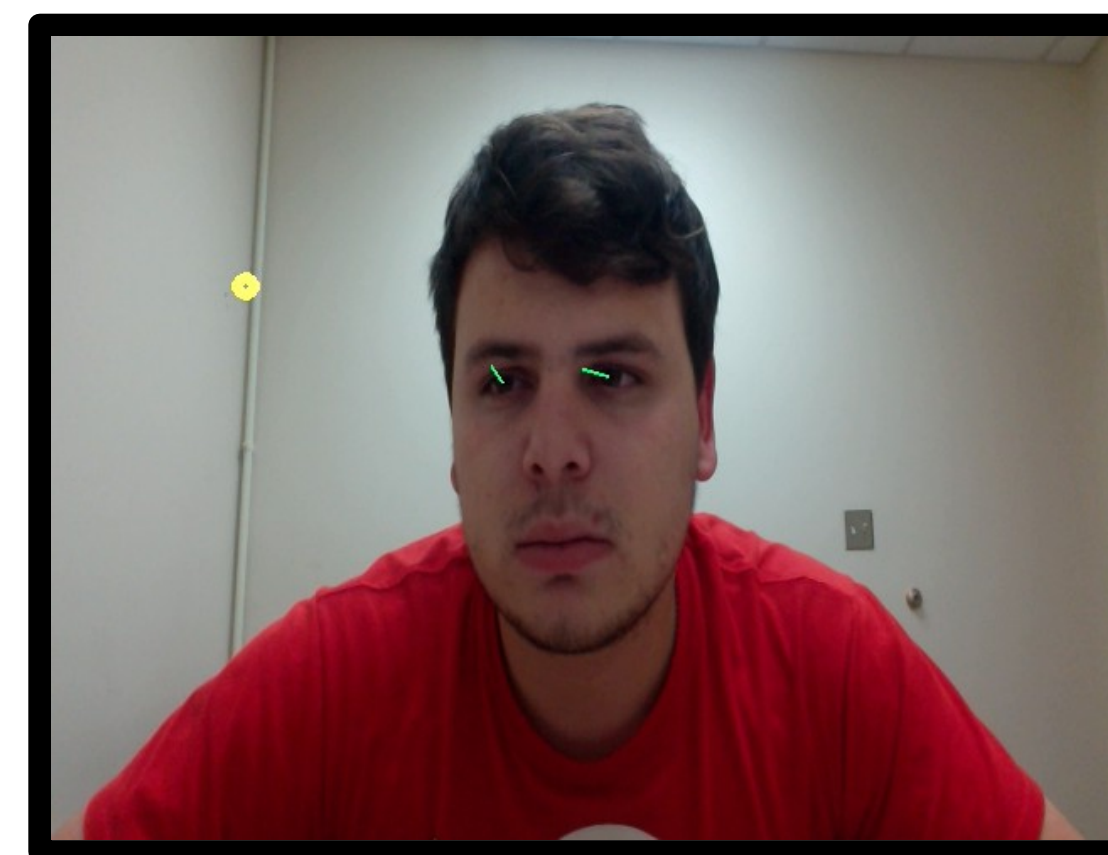
- Obtain video via Web RTC
- Determine pupil centroids
- Use reference point to determine gaze
- Add face position and head orientation vectors to the gaze metric to determine the resultant vector
- Use calibrated mapping to project the vector onto the screen

Determining the Gaze Metric

1. Detect the eyes using a Haar Cascade Classifier
2. Convert the ROI image to grayscale
3. Threshold the image and conduct blob detection on the binary image.
4. Assign the blob with the lowest error weight to be the detected pupil
5. Determine the blob centroid and retrieve the resultant vectors from the boundary rectangle to predict gaze direction

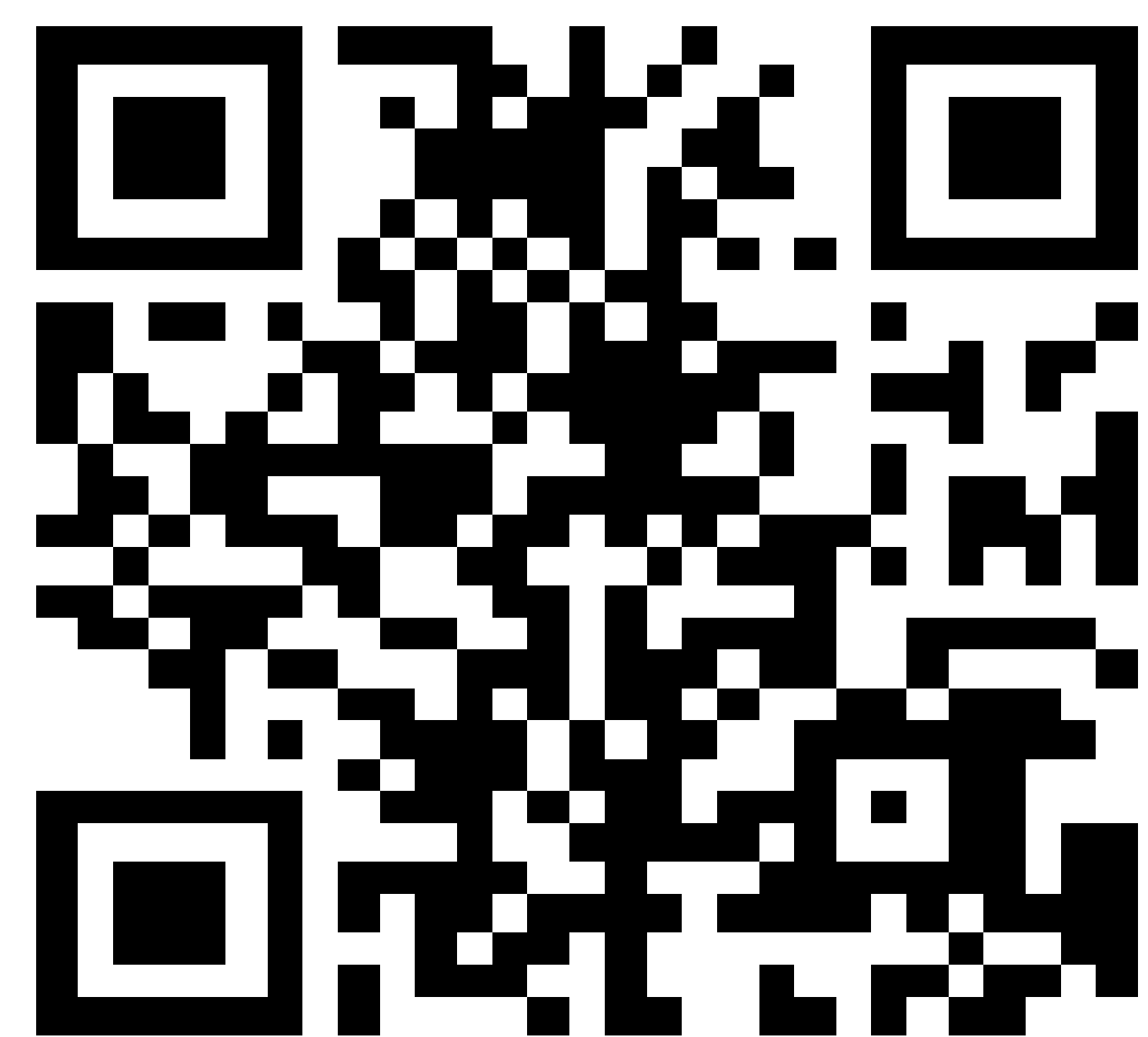


Calibration



Results & Future Work

- Able to determine gaze within 2.1 inches
- Plan to use AAM for reference point detection
- Use neural networks for calibration
- Undergo large scale, crowd sourced user testing

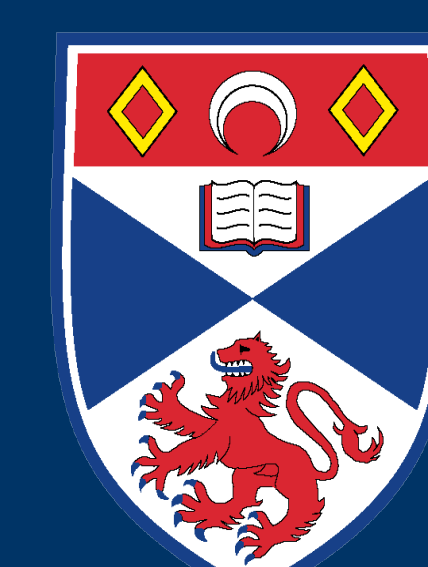


This material is based upon work supported by the National Science Foundation under Grant No. CNS-1062743

Chirstian Poellabauer

Patrick Flynn

Aleksejs Sazonovs



University of
St Andrews

Alex
Wallar

