Project #2 Final Demo: $1, Eye Gaze Gestures

CIS4930 Human-Centered Input Recognition Algorithms
Instructor: Dr. Lisa Anthony, Spring 2023

April 25, 2023

Project #2 Overview

- Algorithm: $1 Recognition

- Language: Python for data collection, C++ for data analysis

- Existing dataset: $1 unistroke gestures, browser interaction gestures[*]

- New dataset: User eye movements for $1 Unistroke gestures [6 people]

- Analysis: user-dependent recog. accuracy and GHOST heatmaps

Online / Live Demo

- Calibration
- Collecting Data
- Recognition
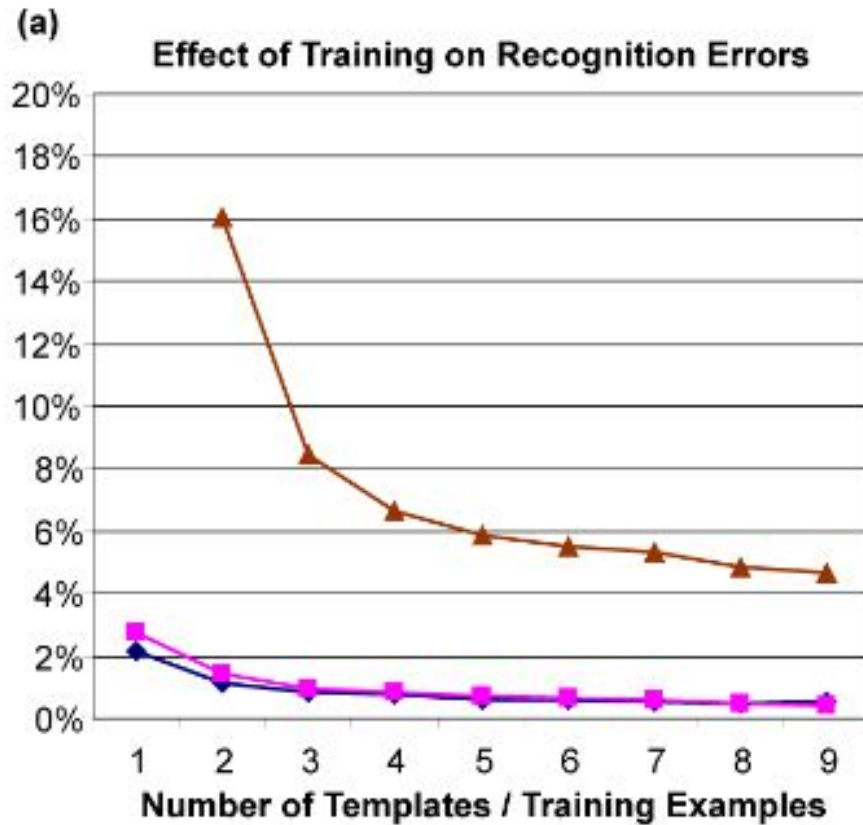
Collecting Data
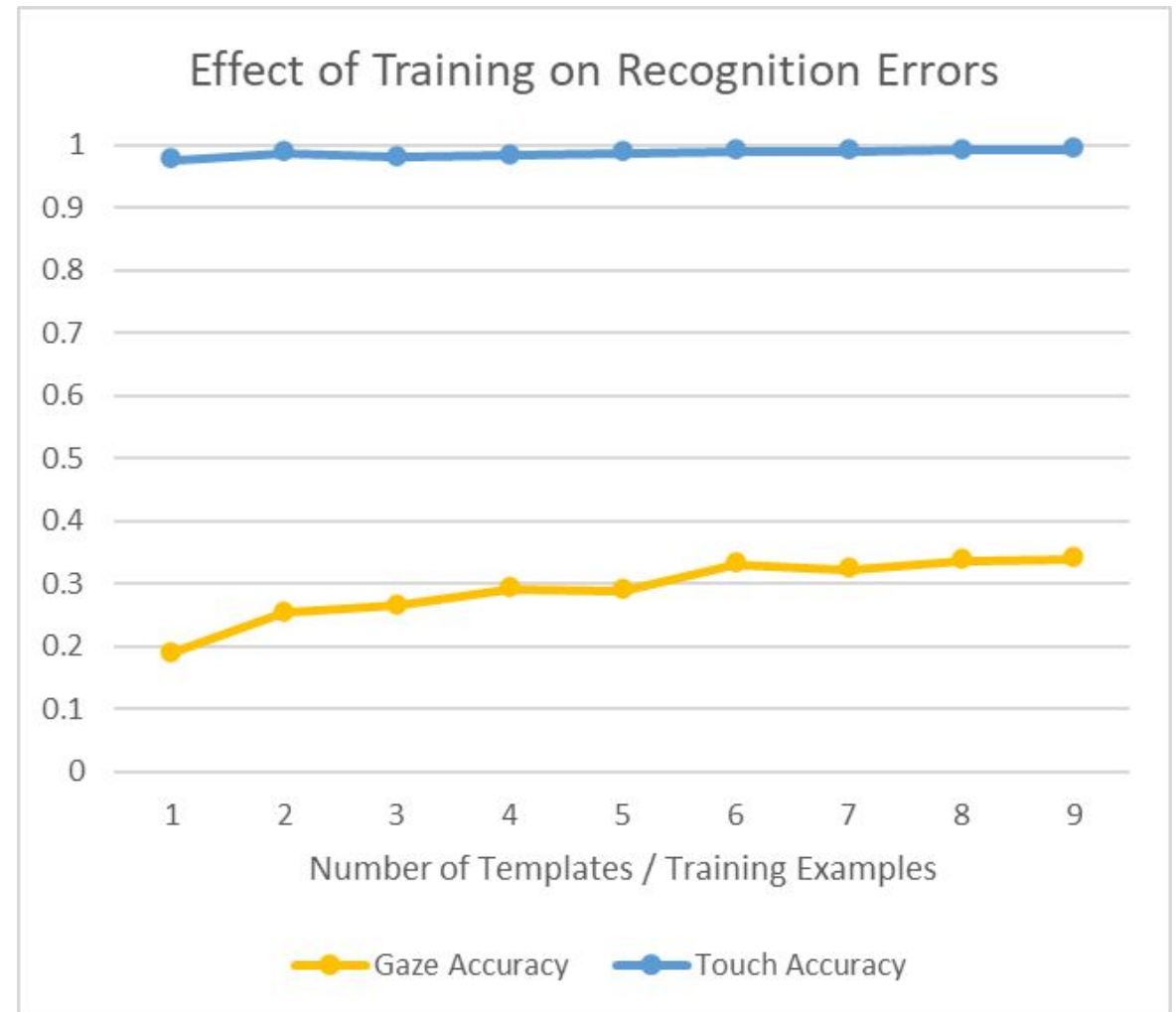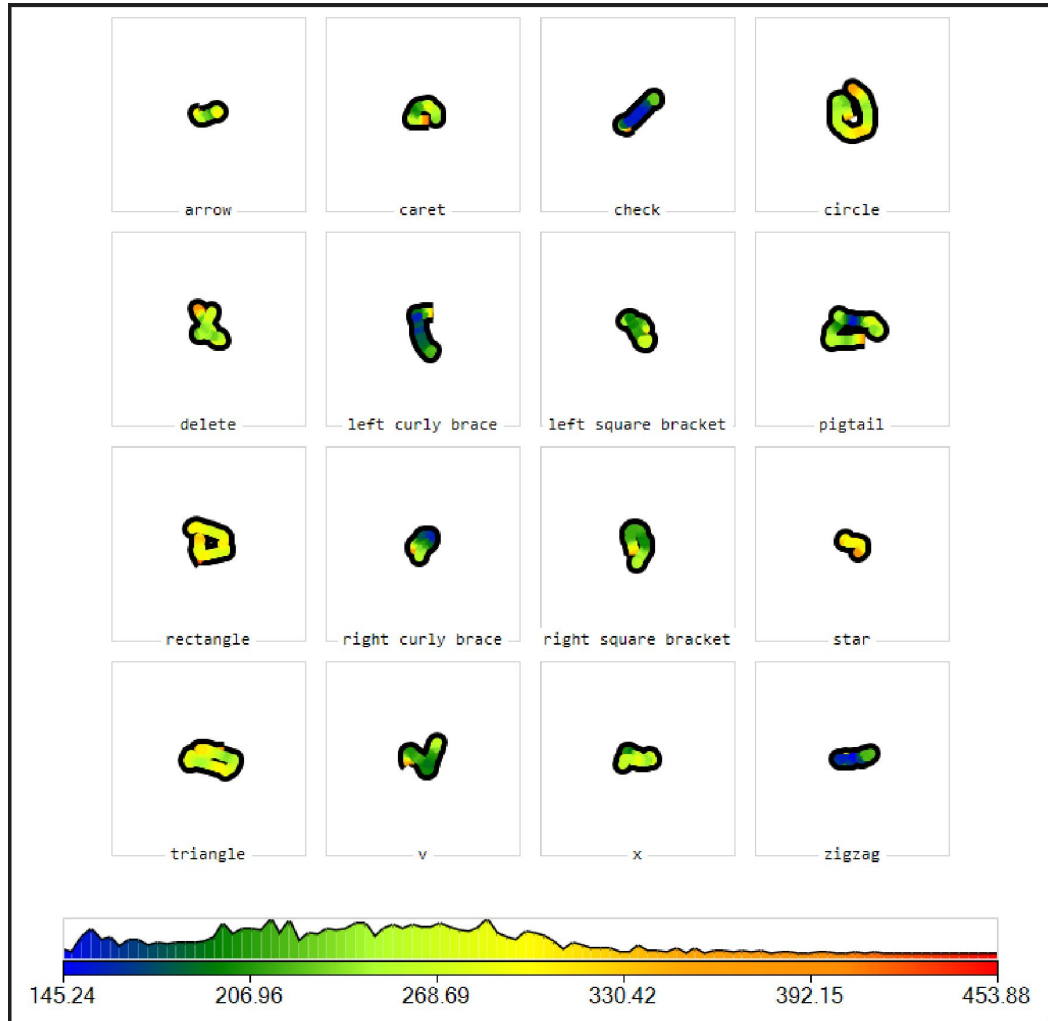
Offline Recognition Tests

Figure 9(a) from $1 paper
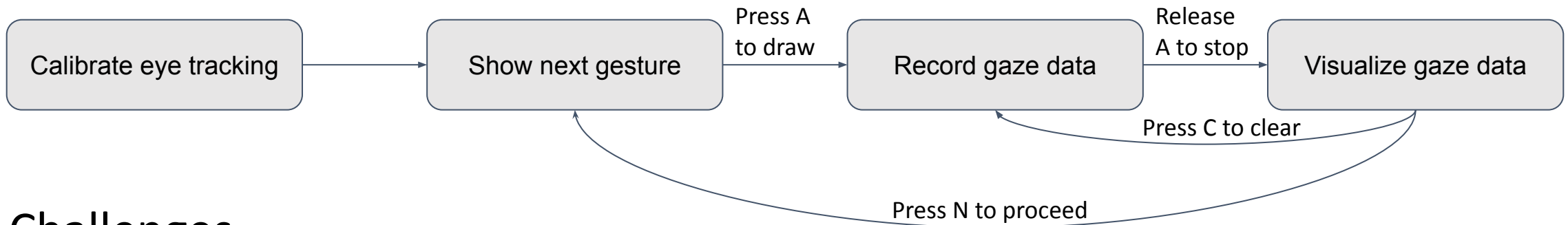

Gaze accuracy versus touch accuracy

Analyses

- Users tend to start at a point, then re-adjust their gaze to start drawing.
  - Notice the tail on the circle, v, rectangle, and other gestures.
- Users were generally more consistent with gestures that start and end at the same point.
  - See the circle, triangle, and rectangle.

Implementation and Challenges

Implementation
- Python with OpenCV for computer vision

| Calibrate eye tracking | → | Show next gesture | Press A to draw → | Record gaze data | Release A to stop → | Visualize gaze data |

Press C to clear

Press N to proceed

Challenges
- Detecting Eyes - Struggles with consistently detecting gaze using a computer webcam
  - Solution: Configuring the program to take input from phone camera
- Tracking Gaze Accurately - Point of gaze was often inaccurate after calibration
  - Solution: Ensure that eyes were detected during each calibration step
- Resampling Down - Gestures often had less than 64 points.

Predicted and Actual Outcome

- Expected lower recognition accuracy for eye gaze gestures than for mouse-based gestures
- Actual outcome was lower recognition accuracy. An average recognition accuracy of 29.02% is not reliable, but is much better than chance (6.25%), showing some promise.

The End!