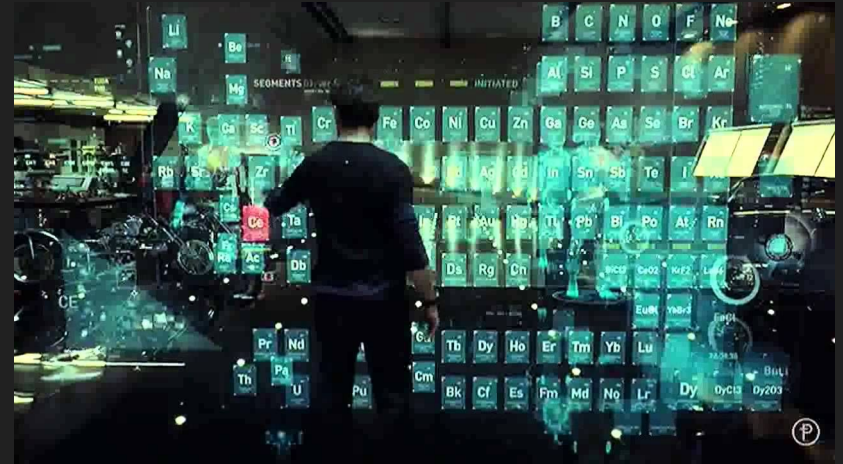


Virtual Whiteboard

As a user of Virtual Whiteboard, I would like to open a web browser and navigate to several different pages in an office/classroom environment

- Supported actions
- Complexity of frequent gestures
- Acceptable gesture error rate
- Supported use environment
- Software, Signals



Quantitative Requirements

- Cursor precision
 - Acceptable error?
 - Edge cases?
 - Question from last presentation
 - Smallest standard button (30px x 40px)
 - Screen size (1920px x 1080px)
 - User correction
- Gesture recognition error
 - Gesture frequency
 - Move cursor
 - Scroll
 - Left Click
 - Right Click
 - Gestures rate
 - Weighted error -> Sample weighing
 - User story
 - ~10 gestures total
 - < 10% total error

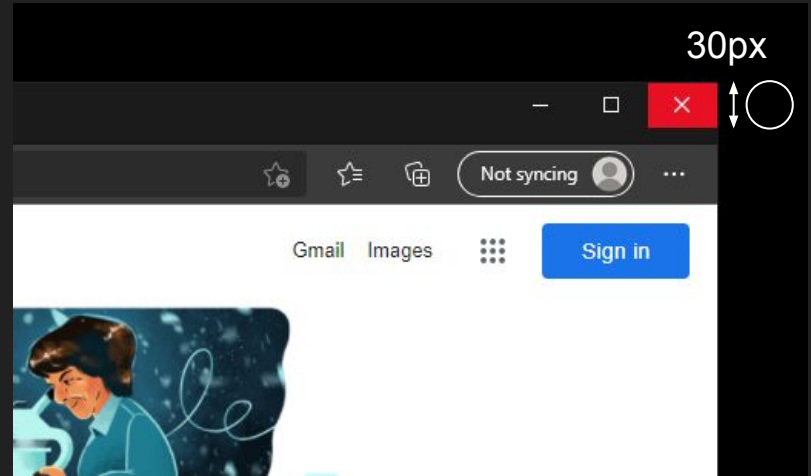
Quantitative Requirements

Cursor Precision

- Radius of 15px on 1920x1080 screen

Gesture Recognition Error

- < 10% training error
- Stretch goal: lower for common gestures



Solution Approach Comparison

IMU

- Low accuracy compounds over time¹

Infrared

- Resolution: $0.2 \text{ mm}^2 - 5.3 \text{ mm}^2$
- Only gets location of sensors, need more complex algorithms to determine gestures

Ultrasonic

- Resolution: $\sim 1 \text{ cm}^2$
- No sensors needed, complex calculations
- 3D

Computer vision

- Resolution scales with camera & distance
- Accuracy approaches $>95\%$
- Speed dependant on model complexity
- Lots of previous work done, intersects most with our experience as a group

Solution Approach

Computer vision (Pose estimation)

- Compared to other options, best solution for hand detection
- Can feed pose data to gesture recognition

Gesture Recognition with Deep Learning

- Complicated function of joint locations to gesture classification
- Availability of datasets

OS Interfacing (mouse library)

- Needed to connect CV and gesture recognition with computer cursor
- mouse library easy to use and incorporates features of other Windows API libraries
- Produces neat and readable code with fitting attribute names

System Specification - Computer Vision

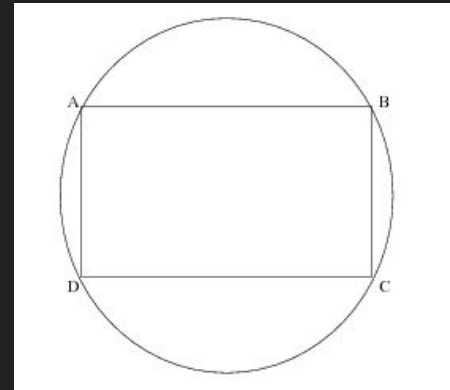
Computer vision/Image recognition

- Input: raw camera data
- Output: hand pose estimation
 - Relative depth
 - Hand landmark coordinates
 - Smoothing/filtering
- MediaPipe library



Data Transformation

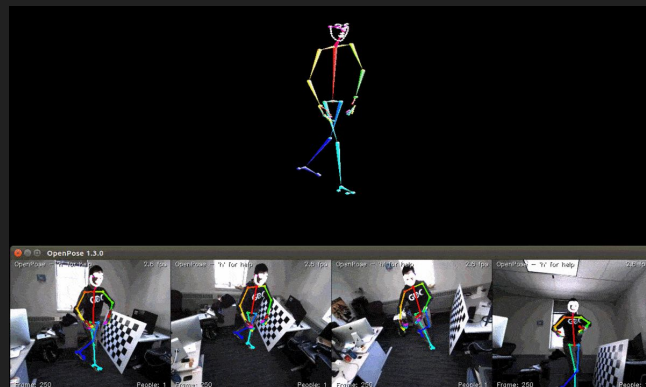
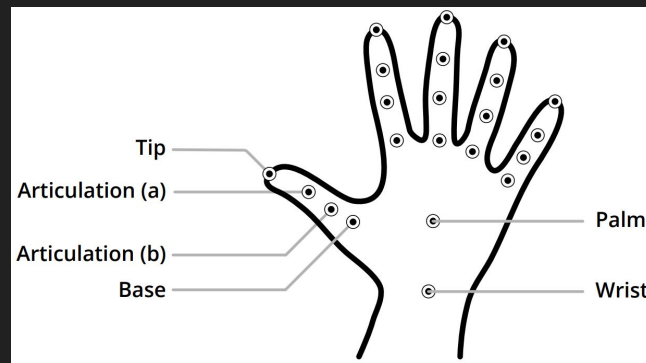
- Collect user range of motion
- Project hand location to screen (Python ctypes)
 - Fit circle to ROM w/ least squares
 - Screen inscribed within ROM



System Specification - Gesture Recognition

Gesture Recognition of (N total gestures)

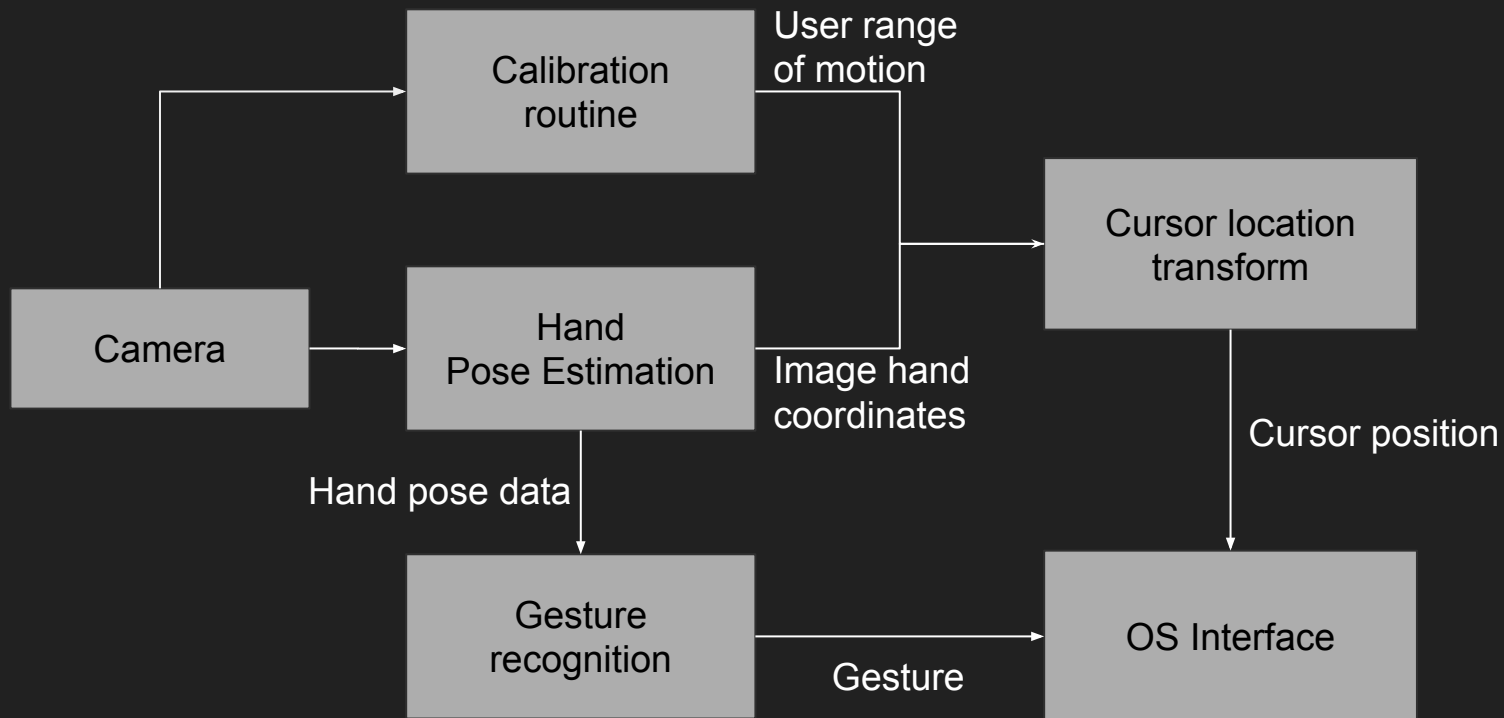
- Deep Neural Net w/ Pytorch
- Pre-trained Model¹
- Custom Model
 - Transfer Learning (thanks Prof. Tamal)
 - Freeze first 2 layers
 - Train pre-trained with dataset
- Dataset - <https://data.mendeley.com/>
 - 12 single hand gestures (L & R)
 - 3 two-handed gestures
 - Apply pose recognition



System Specification - OS Interfacing

- Inputs
 - Transformed cursor position
 - Gesture
 - Record time sequenced data (t, x, y, g)
 - Filter/smooth data
 - Outputs
 - OS operations
- OS Operations
 - Move cursor
 - Right-click
 - Left-click
 - Middle click
 - Scroll wheel movement
 - Open on-screen keyboard

Block Diagram



Implementation Plan

Camera - purchasing (~\$60)

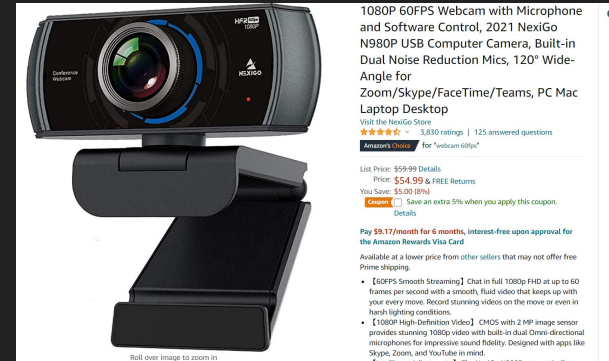
CV - integration done by us, using pre-existing API

AWS Credit for Model Training - (up to \$150 of credits)

Pre-trained gesture recognition model - Open source from github

Custom gesture recognition model - done by us

OS Interfacing - done by us using existing Python library



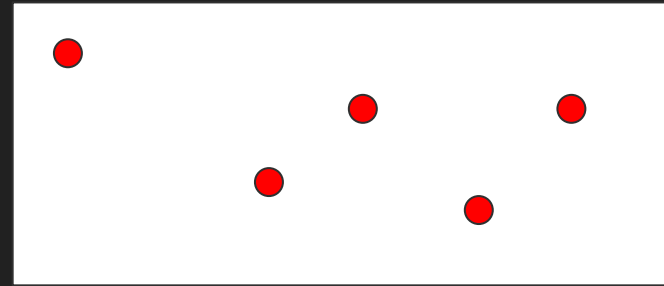
Metrics and Validation

Metrics of 'goodness'

- Cursor accuracy
- Gesture accuracy

Validation Plan

- Custom cursor accuracy app
 - Timed
 - Random red targets (30px)
 - Baseline with mouse
- Walk through user story
 - Start at desktop -> Click on chrome -> Click on cmu.edu bookmark -> ... -> record errors
 - # Missed gestures
 - # Incorrect cursor locations



- User satisfaction survey (1-10)
 - “This product would be useful in an office presentation.”
 - “This product would be useful in a classroom environment.”
 - “The cursor went where I intended.”
 - “The product clicked when I wanted.”
 - “I could use this product to do anything I could with a mouse and keyboard.”
 - “The product felt intuitive to use.”

