

# Big Brother

In-Depth Interview Assessor

# Meet the team:



Sazan Khalid



Chris Bond



Caitlyn  
Dovel



Cameron  
Worden



Jimmy  
Angelos



Ty Swanson

## AGENDA

- Purpose, Goals, and Steps to Completion
- Methods and Live Testing
- Practical Application On Full Interviews
- Explaining the Output

# Why Our Project Matters

- Provides an unbiased evaluation of the interviewee's confidence, engagement, and nervousness, ensuring a fair assessment of each candidate without human biases, as every candidate is assessed under the same criteria
- Can easily scale to handle large volumes of interviews, making them ideal for large-scale recruitment without compromising the quality of assessment
- Companies can make better-informed decisions about which candidates are likely to perform well under pressure
- Interviewers can review the emotional analysis both live and post-interview to better understand how their questions or the interview setting affected their responses

# Goals and Steps to Competition

- Receive live and recorded input of the interview
- Curate models that analyze various visual mannerisms including:
  - Emotion
  - Hand movement
  - Head movement
- Utilize bucketing to analyze nervousness and confidence levels
  - Blinking
  - Eye movement
  - Yapping
- Output two-pronged feedback in the form of a summary and in-depth review

# Head Movement

1. Uses Haar cascades to detect the presence of eyes, mouth, smiles, and overall faces. Initialized at the start and applied to each frame of the video.
2. Calculates the center and deviation from a predefined frame center. It also detects specific features within the face region (ROI), including eyes, smiles, glasses, and mouth.
3. Displays the detected features on the video feed, drawing bounding boxes around each detected facial component
4. Deviations from the center of the frame are tracked over time and collected into arrays, along with other feature metrics like the position and size of detected features
5. Deviations of the head's center from the preset frame center are plotted against time

# Hand Movement

Real-time hand recognition and movement analysis system using OpenCV, MediaPipe, and matplotlib for visualization

1. Initializes video capture and sets the frame dimensions. Uses a while loop to continuously capture video frames from the webcam.
2. Each frame is processed using MediaPipe to detect hands. If hands are detected, this is recorded as 1, otherwise 0 for out of frame.

3. Bounding boxes are drawn around detected hands in the video feed for visual feedback.
4. Detection results along with timestamps are stored in a Pandas DataFrame.
5. Plots the % of time hands were detected in the frame vs. out of the frame. Plots the presence of hands over time against timestamps. Interactive features using matplotlib sliders help adjust the view window of the graph.

# Live Testing the Hand Model



# Emotion Detection

Real-time face recognition and emotion analysis system using OpenCV, DeepFace, and matplotlib for visualization

1. Captures video from webcam, where each frame is converted to grayscale, and faces are detected using a Haar cascade classifier.
2. DeepFace analyzes the dominant emotion (e.g., happy, sad, angry) from the facial expression.
3. Emotions detected and their timestamps are collected into a list and converted into a Pandas DataFrame.
4. Detected faces are highlighted by rectangles and emotions are displayed in real time.
5. Using matplotlib, the distribution of detected emotions over the duration of the video capture is plotted and another plot shows how emotions change over time.

# Live Testing the Emotion Model

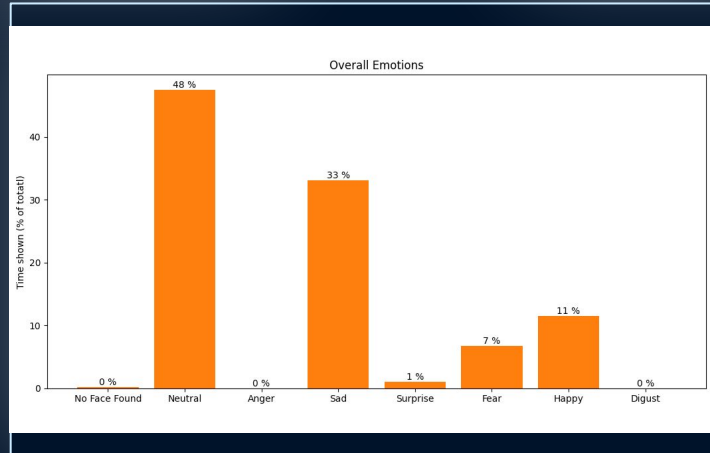
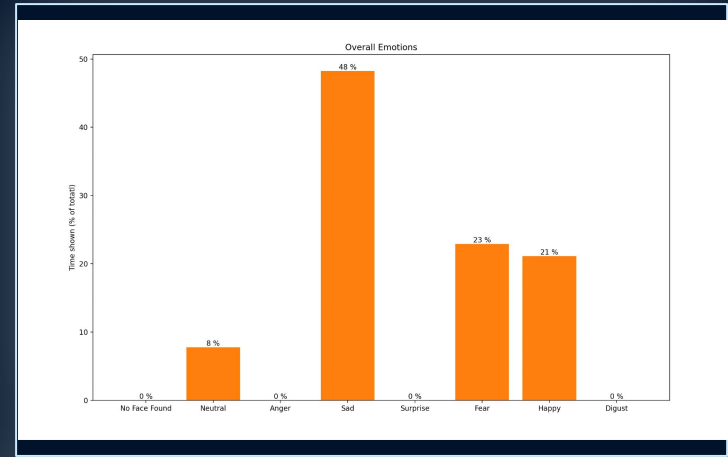
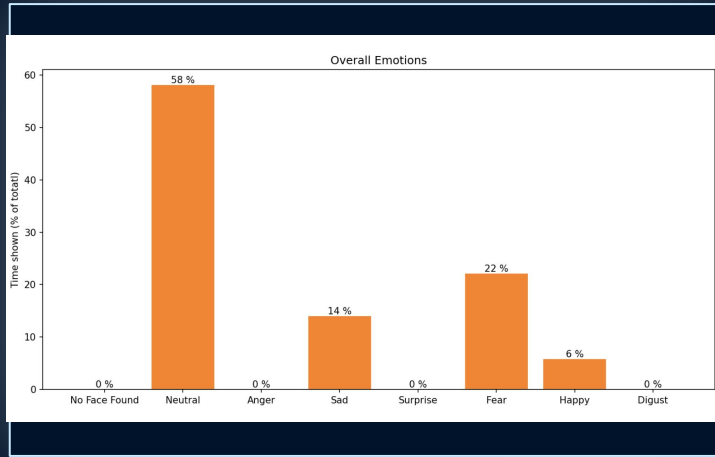
# Speaking, Blinking, and Eye Movement

1. Uses MediaPipe to analyze eye movement and mouth activity. Ex: tracks the positions of the irises and corners of the mouth.
2. Calculates the positions of the irises within the eye sockets to determine the direction of gaze and eye movement. Measures the distances between certain landmarks around the eyes and compares these to detect changes in the eye aperture for blink detection.
3. Assesses whether the mouth is open or closed by comparing the vertical distance between the upper and lower lips to the distance between the eye.
4. Calculates the coefficient of variation for these metrics over specified intervals to assess the variability and stability of each measure.
5. Uses matplotlib to show the standard deviation of eye movement, blinking, and mouth activity metrics over time.

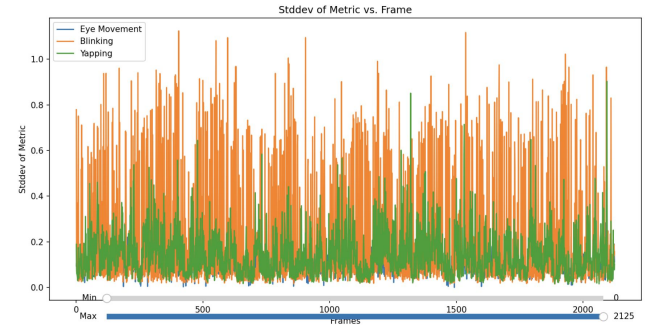
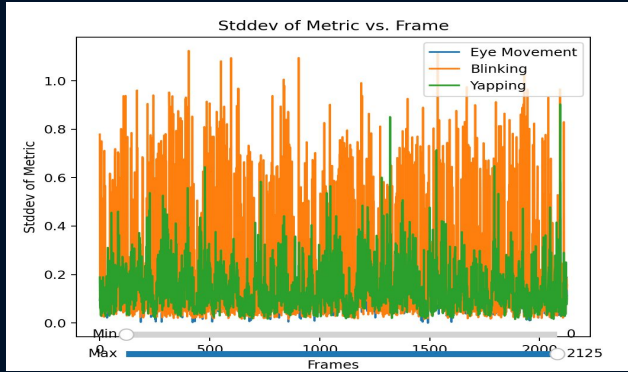
# Live Testing the Speaking, Blinking, and Eye Movement Model

# Testing the Models on Full Mock Interviews

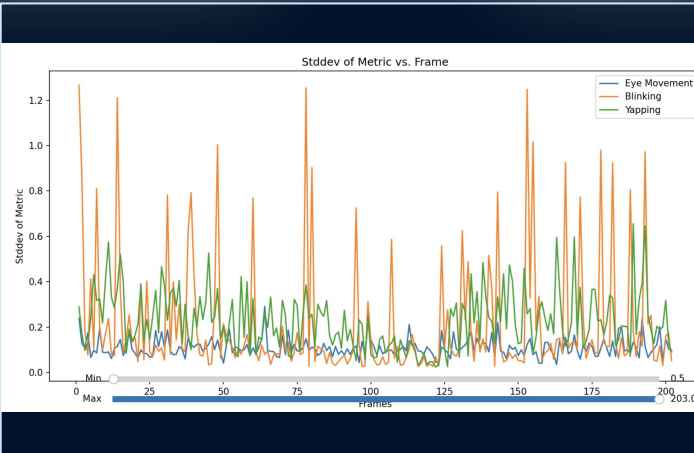
# Explaining the Output: Emotion Detection



# Explaining the Output: Nervousness



W



Slider changes the zoom of the graph

K

# Application and Conclusion

## Strengths:

- Ability to detect emotion and movement of various facial landmarks
- Use of this detection for feedback on nervousness and engagement

## Weaknesses:

- Model's and detections using pre-trained models
- Rigidity within analysis of emotion detection and speak/blink/pupil models

## Conclusions:

- Improves fairness of candidate evaluations with real-time feedback
- Provides insights that can transform how organizations interact with and select future employees
- Serves as a benchmark for future developments in human resources technology



THANK YOU!

QUESTIONS?

Connect with Hoyalytics!



@hoyalytics



@hoyalytics



@hoyalytics

