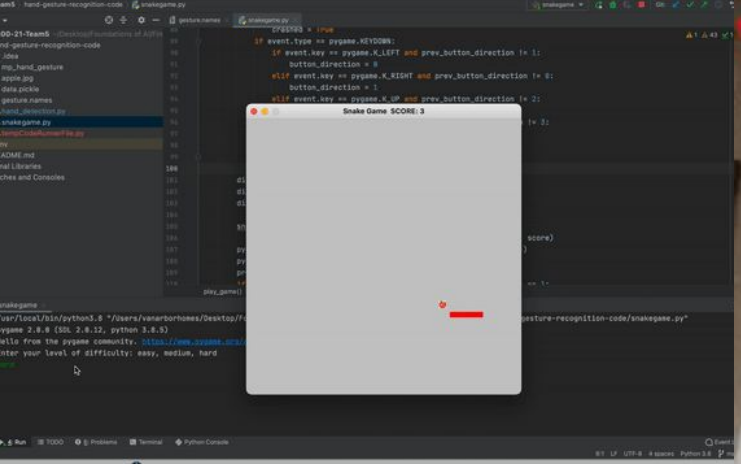
Adversarial SNAKE GAME

Team 5:
Marjan Moshfegh Gohari, Artik Vinubhai Bharoliya, Monideep Chakraborti, Zhengtian Zhang, Katie Lowen



CONCEPT

Turn the traditional snake game into an adversarial game, where the human player and the AI player compete.

1. Human Plays
2. Score Calculated
3. AI Plays
4. Highest points wins!

https://github.com/TheAILearner/Snake-Game-with-Pygame

Our Adjustments

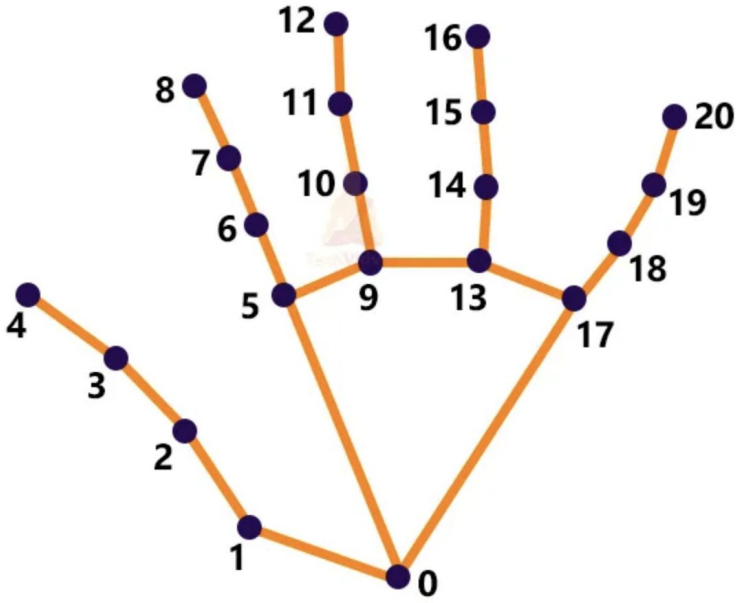
The model recognizes the hand-gestures based on 21 key points in the hand (4 for each finger and 1 for the palm).

We chose 4 hand gestures based on these key points that would mitigate the potential for an incorrect detection.

We refactored the game so that we can incorporate the hand recognition with the game.

Exposed the position of apple and snake so that agents know what is the initial state and what is the goal state.

Human Interaction: Hand Detection



Model is pre- trained on 10 hand gestures:

- **“Ok”**
- **“Peace”**
- **“Thumbs up”**
- **“Thumbs down”**
- “Call me”
- “Stop”
- “Rock”
- “Fist”
- “Live long”
- “Smile”

https://techvidvan.com/tutorials/hand-gesture-recognition-tensorflow-opencv/

Hand Detection Tools

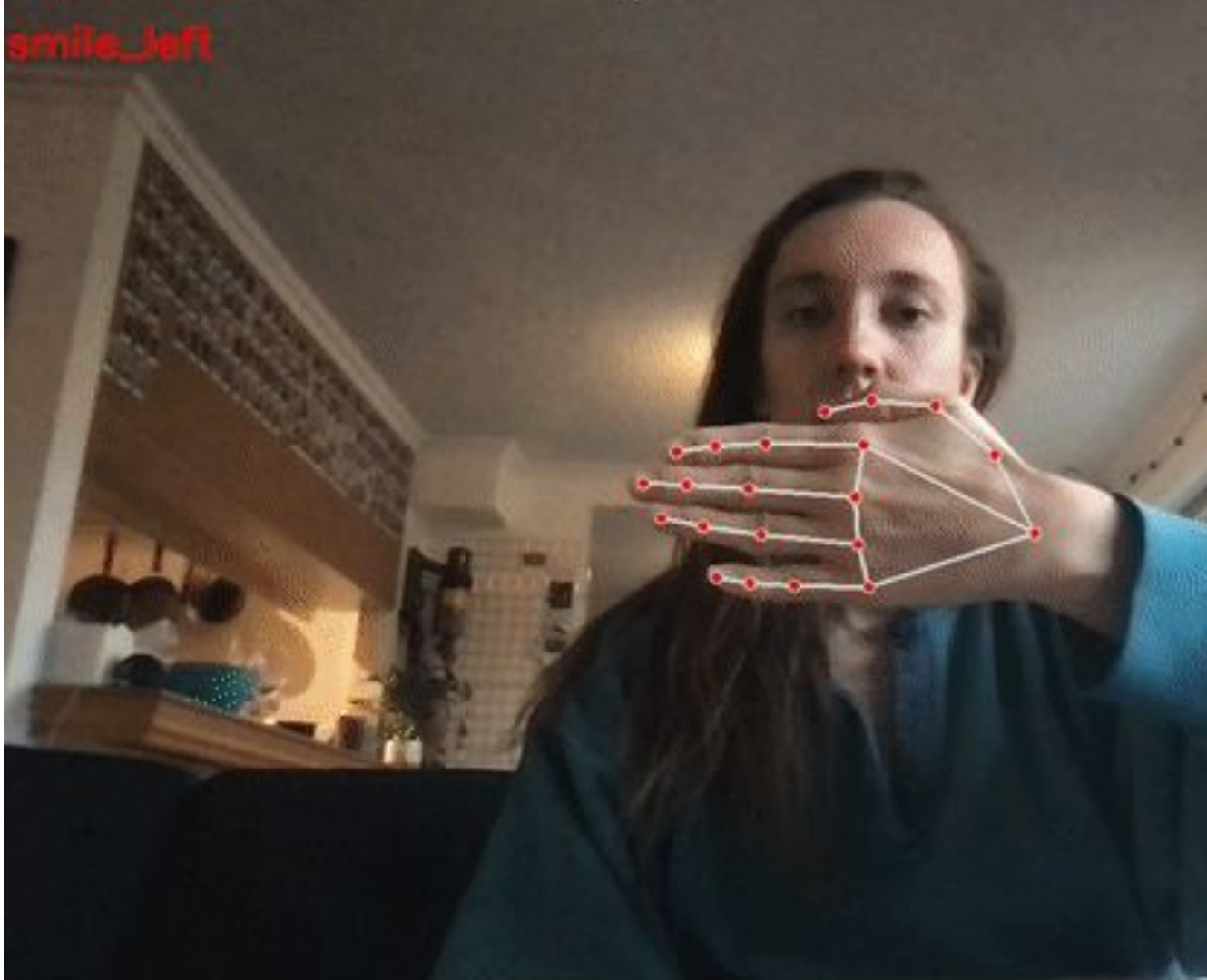
cv2

numpy

Mediapipe

tensorflow

from tensorflow.keras.models import load_model



Incorporating Artificial Intelligence

LEVEL ONE

(easy, snake speed = 10)

Depth First Search

Slowest search algorithm out of the three

Data structure: **Stack**

Explores the deepest connection for every node

In the snake game, the agent (snake) majority of time **moves up and down** till it reaches the goal (apple)

LEVEL TWO

(medium, snake speed = 30)

Breadth First Search

Neither too fast or too slow search algorithm

Data structure: **Queue**

Explores the connection level wise

In the snake game, the agent (snake) majority of time **moves in straight lines** till it reaches the goal (apple)

LEVEL THREE

(hard, snake speed = 60)

A * Search

Data structure: **Priority Queue**

Explores the connection based on **step cost** and **heuristic function**

Step cost: **euclidean cost**

Heuristic function: **manhattan distance**

In the snake game, the agent (snake) majority of time **moves in straight lines and diagonals** till it reaches the goal (apple)

https://medium.com/@nicholas.w.swift/easy-a-star-pathfinding-7e6689c7f7b2

GAME DEMO

Thank you!

Sources

https://theailearner.com/tag/snake-game-using-hand-gestures/

https://techvidvan.com/tutorials/hand-gesture-recognition-tensorflow-opencv/

https://github.com/TheAILearner/Snake-Game-with-Pygame

https://medium.com/@nicholas.w.swift/easy-a-star-pathfinding-7e6689c7f7b2

http://ai.berkeley.edu/search.html