School of Computer Science and Engineering

J Component report

Programme	**:**	MTech CSE with BA
Course Title	**:**	Artificial Intelligence and Knowledge based systems
Course code	**:**	CSE3088
Slot	**:**	A1 + TA1
Title	**:**	Snake Game using Hand Gestures
Team Members	**:**	Noel Benny 20MIA1020
		S P Mithun 20MIA1038
Faculty	**:**	Dr. R Reena Roy

Table of Contents

Abstract

The major goal of this system is to demonstrate how a computer game may be played using human gestures. The secondary goal of this system is to develop a system that enables a player to play a game without the use of a physical controller. The goal of this system is to create a gesture recognition application. The system's camera or webcam may be used to identify human hand gestures. The programme includes a set of instructions for recognising human hand motions. The movements must be performed on the palms of the hands. The system will be divided into three modules: user interface, gesture recognition, and analysis. The user interface module provides the user with all of the necessary graphical user interface to record the arm positions that will be utilised to conduct gestures. To identify gestures, utilise the gestures recognition module. Finally, depending on the estimated analysis, the analysis module will evaluate the human hand gesture and conduct game controls. A totally reliable hand gesture recognition system is still being researched and developed. This system effort might be viewed as a tiny step toward laying the groundwork for future development.

Keywords: MediaPipe, TensorFlow, cv2, Pygame, PyAutogui

Introduction

Over the last decade, computer technology has advanced dramatically and has become an essential element of daily life. The keyboard is the most important computer accessory for Human Computer Interaction. In some real-world scenarios, such as Human Robot Interaction, the keyboard is unsuitable for Human Computer Interaction. The use of hand gestures is the most natural and intuitive approach for Human Computer Interaction that is a feasible alternative for the computer keyboard. This system is thus intended to investigate and build a Computer Control system based on hand gestures. To fully utilise the capabilities of a camera, it may be utilised for vision-based computer control, completely eliminating the need for a computer keyboard. Motion controllers are the way of the future in gaming. Hand gestures are particularly intuitive and effective for one-on-one engagement with computers, and they create a Natural User Interface.

Objective and Goal

The goal of this project is to play a snake game using the MediaPipe object detection API and the computer's camera. Instead of using the computer, it will control the snake with hand gestures. We can anticipate over 90% of hand motions using a webcam. This project will also be a revival of the classic snake game that we used to play using keypads.

Literature Review

A research article by Yashaswi Raj and Rajat Sharma describes the engineering and execution details of a hand's movement as well as a motion recognition framework that was developed at the Meerut Institute of Engineering and Technology. The system works in real time (on a DEC Alpha, 25 edges per second), uses live 15-piece coloured video from a fixed camera, and handles foundation muck nicely. The 2D deformable Active Shape Models (clever snakes) are used to do the following, and a genetic computation is also used to carry out an underlying global image search. The Point Distribution Model is a nonexclusive and extendable model that can be used to track any 2D deformable object, and it was utilised for both the snake calculations and the hereditary computations.

Features

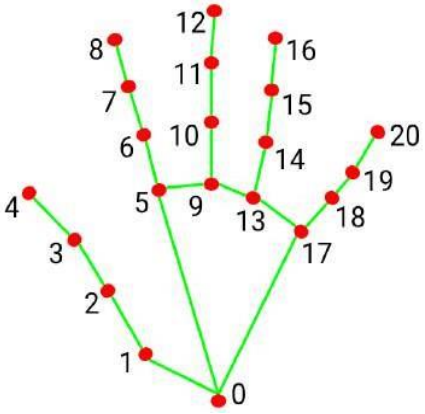
Hand Gesture Recognition

The capacity to recognise the motion and shape of hands can play a key role in enhancing user experience across a range of technology platforms and areas. For instance, it can provide as the foundation for hand gesture control and sign language comprehension. It can also make it possible for digital information and material to be superimposed on top of the actual environment in augmented reality. Despite being instinctive for humans, strong real-time hand perception is a difficult computer vision problem because hands frequently occlude themselves or one another (for example, finger/palm occlusions and hand shaking) and lack high contrast patterns. A high-fidelity hand and finger tracking system is MediaPipe Hands. It uses machine learning (ML) to extrapolate 21 3D hand landmarks from a single shot. Our solution delivers real-time performance on a cell phone, and even scalable to several hands, unlike existing state-of-the-art systems, which mostly rely on powerful desktop environments for inference. We anticipate that making this hand sensing capabilities available to a larger research and development audience will lead to the creation of innovative use cases, igniting new research directions.

Mediapipe Hand

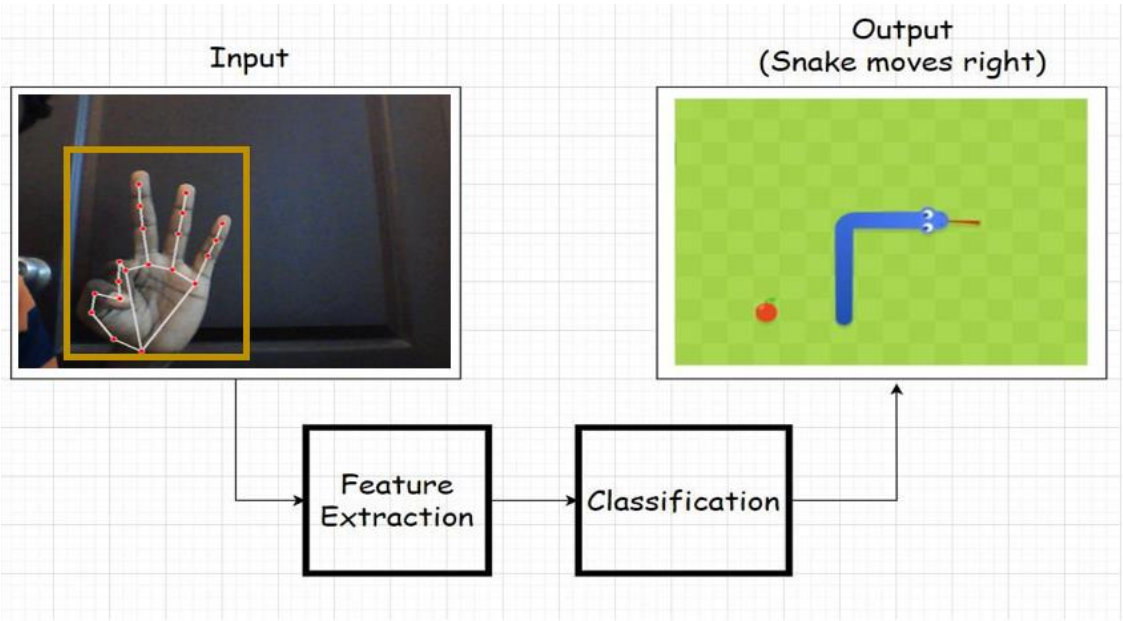
A high-fidelity hand and finger tracking system is Mediapipe hand. It employs machine learning to identify 21 3D hand landmarks. For the following, many models collaborate. An orientated hand bounding box will be returned by a palm detection model from the entire frame, and the hand landmark model, which operates on cropped pictures, returns the 21 landmarks. The palm identification model training algorithm uses a non-maximum suppression approach. A frame in this technique may have numerous bounding boxes in it. This method produces the best bounding box that meets the requirements we have specified. Once the palm has been recognised. On to hand landmark models we go. This model was trained using more than 30,000 photos of hand models. It was also possible to employ a synthetic hand model over different backdrops, which was subsequently translated into equivalent 3D points or landmarks.

Following are the 21 3D landmarks,



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

Design



Module Description

Hand Gesture Recognition

For hand identification, we utilise the Python-based machine learning and AI libraries Tensorflow and Mediapipe, as well as the cv2 package for webcam video capture. A number of AI-based recognition models, including hand and face recognition, are already included into Mediapipe. We utilise mediapipe hand or the built-in hand gesture

recognition model for our project. The gesture names are loaded after the gesture recognition model. The webcam is then set up, and we begin taking pictures. The hand landmarks in the frame are recognised by Mediapipe's hand recognition model. We make the gesture prediction using the discovered landmarks. The AI selects the suitable landmark based on comparisons between the predefined landmarks in the gestures and those that have been discovered. Using the Python package pyautogui, the required key is hit after the gesture has been located. It is a library that can mimic the movement of the pointer and keystrokes. Clicking "q" ends the application.

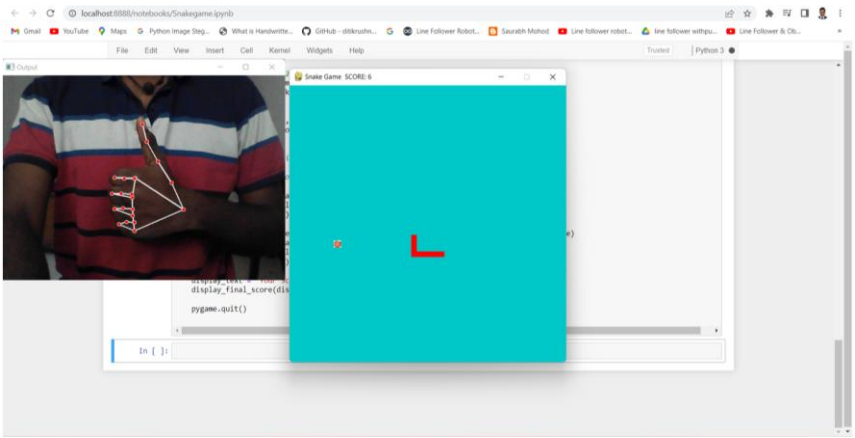
Snake Game

Python's pygame package, which is generally used for creating games, was used to create the snake game. An easy example is the snake game. When a key is pushed, the snake advances in that direction. The snake travels in the direction indicated by the key pushed in the code above using PyAutogui. Eat the apple is the objective. If the snake bumps into itself or a wall, it will die. When the snake expires, the ultimate score will be shown along with the current score, which is shown on top of the game window.

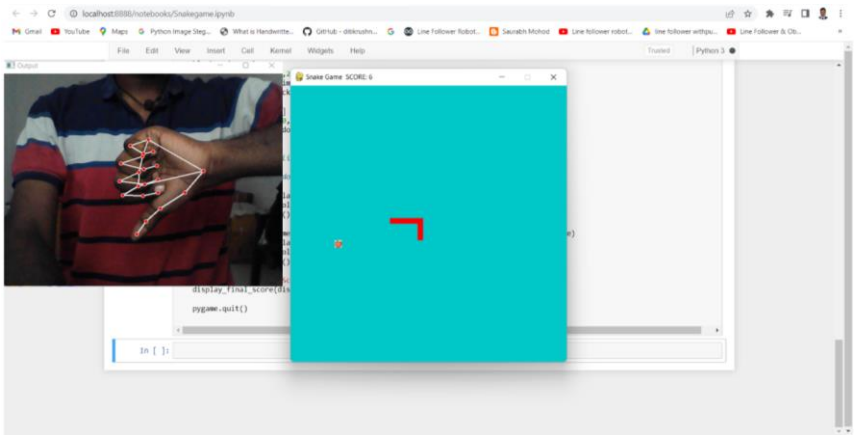
Conclusion and Results

The MediaPipe module of the Python programming language is used to create the machine vision-based keyboard cursor control system. When a person is playing a game, the system may manage the keyboard cursor's movement by monitoring their hand. Different hand movements will be used to control the keyboard cursor. The system has the potential to replace the computer keyboard in a useful way, but given the obstacles it faces, it is not able to do so entirely. If the template matching hand gesture identification approach is combined with a machine learning classifier, the accuracy of hand gesture recognition can be increased. The system's primary restriction is that it needs to be used in a well-lit area. The system's inability to totally replace the computer keyboard stems from the fact that it is highly frequent for people to use computers in outdoor settings with insufficient illumination.

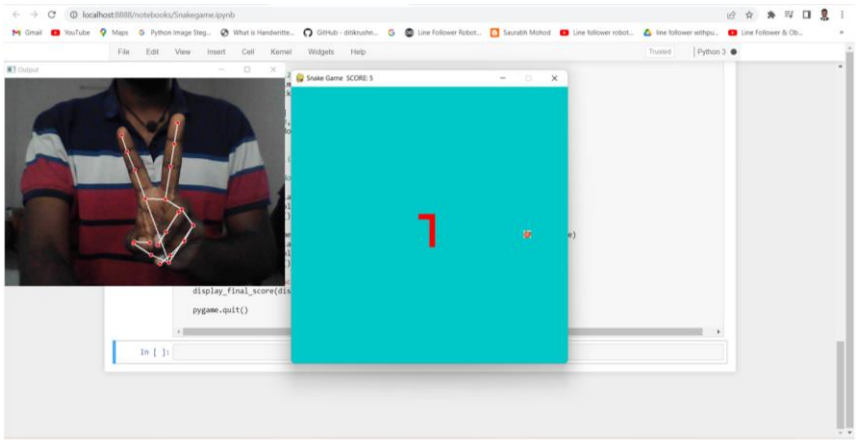
Hand Gesture to move up is ‘Thumbs Up’



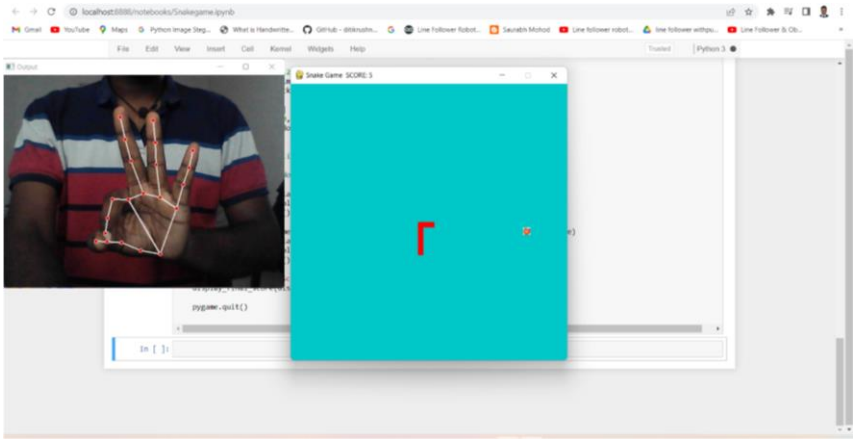
Hand Gesture to move down is ‘Thumbs Down’



Hand Gesture to move left is ‘Peace’



Hand Gesture to move right is ‘Okay’



References

https://google.github.io/mediapipe/solutions/hands.html

https://www.geeksforgeeks.org/snake-game-in-python-using-pygame-module/

HOWSE, JOSEPH. OPENCV: COMPUTER VISION PROJECTS WITH PYTHON. PACKT PUBLISHING LIMITED, 2016

B. Dorner. Hand shape identication and following for gesture based communication understanding. Taking a gander at People Workshop, Chambery, France, 1993

F. Cootes and C.J. Taylor. Dynamic shape models - \'Savvy Snakes\'. In Proceedings of the British Machine Vision Conference, pages 266{275, Leeds, UK, 1992. SpringerVerlag