GESTURE VOLUME CONTROL

Presented By:

Lokam Teja Ram – B200793CS

E Krishna Prasad – B200803CS

Anagha M V – B200762CS

P Gnana Prakash - B200840CS

Abhina Sunny – B200692CS

LITERATURE REVIEW:

In recent years, there has been a growing focus on developing hand gesture recognition systems using computer vision and machine learning techniques. These systems offer a more intuitive way for users to interact with devices like volume controllers. Numerous research studies have explored creating volume controllers based on hand gestures, employing various methods including vision-based techniques like color and motion detection, depth-sensing cameras, and machine learning algorithms.One particular study showcases the use of an Artificial Neural Network (ANN) for gesture recognition using an accelerometer, specifically the Wii remote which registers movement along its X, Y, and Z axes. To optimize system performance and memory usage, the author adopts a two-tiered approach. The first tier involves user authentication through gesture recognition based on accelerometer data. The second tier employs (Fuzzy) automata for gesture recognition, with data normalization using k-means and a fast Fourier algorithm.[1]

In a research paper the author discussed a novel method of hands gestures recognition based on the identification of certain shape-based elements. A single camera is used to record the user's gesture and feed it into the system. The development of a system that can recognise certain human gestures and use them to transfer data for device control is one of the main goals of gesture recognition. Real-time gesture recognition enables users to control computers by making certain gestures in front of a video monitor. The development of a system that can recognise certain human gestures and use them to communicate data for device control is one of the main objectives of gesture.[2]

In Another study the volume control is done using hand gestures and incorporates the use of OpenCV. By utilizing a computer's webcam, the module captures images or videos, processes them to extract relevant data, and adjusts the volume based on recognized gestures. Users can manipulate the volume without needing physical touch or input devices like mice or keyboards. Using OpenCV and Python, the module recognizes specific human gestures to enact the desired adjustments in device settings. By intercepting video input and analyzing gestures within a specified range, the module successfully alters the computer's volume.[3]

REFERENCES:

[1] H.A JALAB "Static hand Gesture recognition for human computer interaction", 1-72012.

JC.MANRESARVARONAR. MASF.

[2] Real-Time Hand Gesture Recognition for Device Control: An OpenCV-Based Approach to Shape Based Element Identification and Interaction by Nishant Kumar, Hridey Dalal, Aditya Ojha , Abhinav Verma, Dr. Mandeep Kaur

[3] Volume Control using Gestures by Martendra Pratap Singh, Arzoo Poswal, Eshu Yadav

PROBLEM DEFINITON:

Gestures play a significant role in human communication and are increasingly integral to Human-Computer Interaction (HCI) technologies. These evolving HCI systems enable users to convey instructions to machines through hand movements, facial expressions, voice commands, and touch interactions. Gesture recognition has become a focal point of research in fields such as computer vision, pattern recognition, and HCI. The primary objective is to explore the utilization of hand gestures as a vital mode of machine control. In this pursuit, OpenCV libraries are employed to address the challenges associated with gesture recognition in the context of Human Computer Interaction**.**

TENTATIVE SOLUTION:

Data Preparation:
No specific dataset preparation is required as the system operates in real-time, capturing frames from the webcam.

Feature Extraction:
Utilize OpenCV and MediaPipe Hands library to detect hand landmarks in each frame.
Extract relevant features such as the distance between specific landmarks representing fingertips.

Gesture Classification:
Define thresholds for the distances between landmarks to classify gestures (e.g., volume up, volume down, neutral).

Volume Control:
Based on the detected gestures, adjust the system volume accordingly using PyAutoGUI or other appropriate libraries.

Real-time Processing:
Continuously capture frames from the webcam.
Process each frame to detect hand gestures and adjust the volume accordingly.

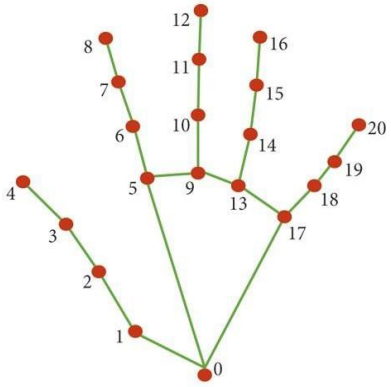
Performance Evaluation:
Validate the system's performance in real-time by interacting with the gesture volume control system.

METHODOLOGY

Mediapipe:

MediaPipe is a software framework developed by Google that helps computers see and understand the world through video input. It's like giving computers the ability to analyse and interpret images and videos in real-time. MediaPipe is often used for tasks like recognizing and tracking objects, detecting hand movements, or identifying facial expressions. It's a powerful tool for creating applications that involve computer vision and understanding the visual world.

Single-shot detector model is used for detecting and recognizing a hand or palm in real time. single shot detector model is used by the MediaPipe. First, in the hand detection module, it is first trained for a palm detection model because it is easier to train palms. Furthermore, the non-maximum suppression works significantly better on small objects such as palms or fists. A model of hand landmark consists of locating 21 joint or knuckle co-ordinates in the hand region, as shown in figure.



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

OpenCV:

OpenCV is a computer vision library which contains image-processing algorithms for object detection. OpenCV is a library of python programming language, and real-time computer vision applications can be developed by using the computer vision library. OpenCV library is used in image and video processing and also analysis such as face detection and object detection.

The various functions and conditions used in the system are explained in the flowchart of the real-time Virtual mouse system as shown in Figure.

The proposed gesture volume control system is based on the frames that have been captured by the webcam in a laptop or PC. By using the Python computer vision library **OpenCV**, the video capture object is created and the web camera will start capturing video.

Capturing the Video and Processing:

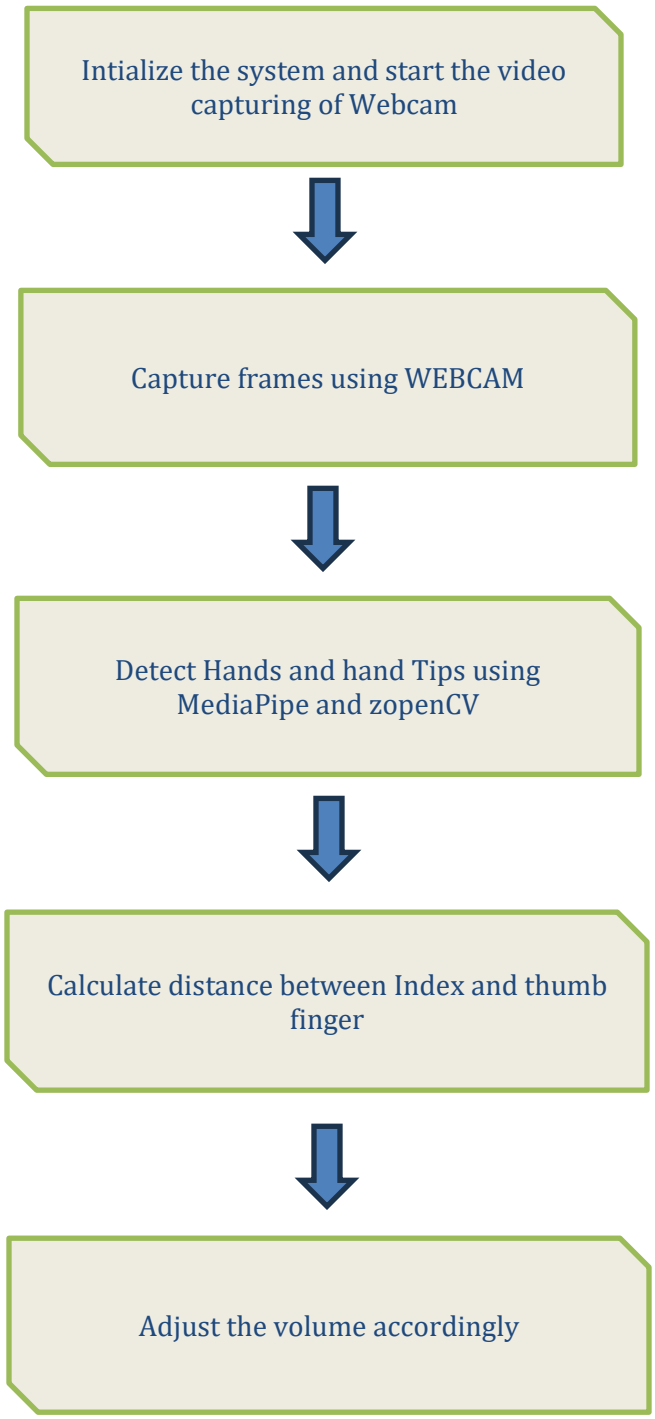
The Gesture volume control system uses the webcam where each frame is captured till the termination of the program. The video frames are processed from BGR to RGB colour space to find the hands in the video frame by frame

Once we get the 21 points of the hand, we take co-ordinates of thumb and index finger then find the distance between those two points using the formula

$$\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}$$

Then according to the distance we change the volume of the device using pyautogui .

Design:


Intialize the system and start the video capturing of Webcam
Capture frames using WEBCAM
Detect Hands and hand Tips using MediaPipe and zopenCV
Calculate distance between Index and thumb finger
Adjust the volume accordingly

TOOLS AND TECHNIQUES:

OpenCV (cv2):

OpenCV (Open Source Computer Vision Library) is a popular library used for computer vision tasks such as image and video manipulation,
object detection, and more.

Mediapipe:

Mediapipe is a framework developed by Google that provides tools for building machine learning pipelines for various perception tasks, including body pose estimation, hand tracking, facial recognition, and more.Here we utilize the mediapipe module for detecting and tracking hand landmarks in the webcam feed.

pyautogui:

PyAutoGUI is a Python library for automating keyboard and mouse interactions. In this code, it's used to simulate key presses based on hand gestures detected by the hand tracking model.here it is used to simulate pressing the volume up or volume down keys.

Image Processing:

Techniques such as converting the color space of the image from BGR to RGB and extracting hand landmarks coordinates from the detected hand landmarks are used for processing the webcam frames.

Hand Landmarks Detection:

The code detects hand landmarks using the pre-trained hand tracking model provided by Mediapipe.It then extracts the coordinates of specific landmarks (such as fingertips) to determine hand gestures.