Electrical and Automation Engineering

Vol: 2(1), 2023

REST Publisher; ISBN: 978-81-956353-5-1

Website: http://restpublisher.com/book-series/eae/

DOI: https://doi.org/10.46632/eae/2/1/7

Controlling Media Player Using Hand Gestures

*** K. S. Chakradhar, Rejinthala Lakshmi Prasanna, Sree Rama Brunda salla chowdary, Pola Bharathi, Petlu Bhargava**

Mohan Babu University Erstwhile Sree Vidyanikethan Engineering College, Tirupati, India.

*Corresponding Author Email: chakradharec@gmail.com

Abstract. *Computer usage is increasing rapidly day by day but the input devices are limited and to access them, we need to be near the screen. To overcome this problem and control the screen, we can use hand gestures. For every operation, we used different hand gestures. We proposed a python program to control the media player through hand gestures. In this method, we used libraries like OpenCV, Media Pipe, PyAuto GUI, and other libraries to capture the video, provide ready-to-use ML solutions and automate your GUI and programmatically control your keyboard and mouse. Hand gestures will be used as the input for providing natural interaction by reducing external hardware interaction. The whole process is divided into two steps. Firstly, gesture recognition through the camera is done by OpenCV and media Pipe helps to identify the gesture b its position, and the respective command is executed. Secondly, PyAuto GUI is used to automate the keyboard and controls the media player.*

Keywords. *OpenCV, MediaPipe, PyAutoGUI, Hand Gestures..*

1. INTRODUCTION

Gesture recognition is a technology that uses gestures to communicate with a computer. Due to its flexibility and user-friendliness, it is one of the active research areas in the field of human-computer interface. There are many ways to recognize gestures and facial expressions, finger counting uses many techniques, uses many algorithms. Based on the gestures we provide, it uses mathematical algorithms to convert these gestures into commands to execute. In some technical fields, sensors are used to recognize gestures, while in some fields, instead of using sensors, some algorithmsare used, such as the K-Nearest Neighbor (KNN) algorithm. Gestures are mainly divided as static and dynamic gestures. Static gestures refer stable shape of the hand, while dynamic gestures refers a series of hand movements by moving our hand. Gestures include various hand movementsthat can be used as commands to operate applications. By using such a complex algorithm, the time required to recognize and execute the command will be proportionally longer. To overcome this complexity and make the processmore accurate, we used the built-in MediaPipe library, which provides a ready-to-use ML solution developed by Google. Using the MediaPipe, we can recognize user-specific gestures and call application functions corresponding to those gestures. The MediaPipe Gesture Recognizer task allows you torecognize gestures in real time and provides the result of the recognized gesture as well as the landmarks of the detected hand. This method is more accurate and faster than other methods. In this article, we use MediaPipe and otherbuilt-in libraries to recognize gestures and use them as commands to control the media player without help of keyboard and mouse. This way we can access the computer and remotely control the video from a far distance. Depending on the number of fingers, the distance between the fingers, and the angle of the fingers, different actions will beperformed.

2. LITERATURE REVIEW

Abhik Swarnakar, Ashish Pal, and Sonu Kumar Soni proposed an algorithm focused on advanced gesture-based robotics research. They use OpenCV, numpy, math libraries and some algorithms to detect the extreme points of the fingers recognized by the camera. Once the gesture was recognized by the camera, they pre-processed the image which includes the conversion of the grey-scale image to a binary image, so we only had two regions of interest in the photo. That is, one will be the hand, and the other is related. Then they drew a contour line with the extreme points of their hands.Contours are useful tools for shape analysis and object detection and recognition. Contours are drawn along the boundaries of the hand images found after thresholding. After drawing the outline, the next step is to find the center of the palm and draw a circle, which is used for performing Bitwise AND operation between hand region and circle. Then the number of fingers is determined by the number of extreme points. Hussain, R. Saxena, X. Han, J. A. Khan and H. Shin proposed a technique by use of six static gestures and eight dynamic gestures to control a computer. Hand identification, tracking of the discovered hand, and data conversion into necessary commands are the three key phases. A pre-trained convolutional neural network that was first built ona large data set is used to train a CNN-based classifier for hand shape identification. They employed the pre-trained VGG16 CNN architecture, which has 13 convolutional layers and 3 fully linked layers. With an accuracy rate of 93.09%, they successfully tested it on seven different volunteers in various locations and lighting situations.

3. METHODOLOGY

The hand gesture recognition process is mainly divided by two steps. The first step we used is to recognize the hand using the OpenCV and Media Pipe libraries and the second step is to automate the input device's control according to the gesture.

OpenCV:

The term OpenCV, short for Open Computer Vision, is an open-source Python library that lets you perform image processing and computer vision tasks. It offers a wide range of features, including object detection, facial recognition,and tracking. Computer vision is the field of study that allows computers to replicate the human visual system. This allows the camera to capture our gestures in real-time and analyze them accordingly. It recognizes images of green, red, and blue, each color ranging from 0 to 255 channels. In this way, a digital image will be captured. By using this software, we can resize the image and rotate any corner of the image, and we can perform many operations on the image.

Media Pipe:

Media Pipe is an open-source library that gives ML solutions for hand detection which are ready to use. To use this feature we need to use "media pipe. solutions. hand". Media Pipe solutions are open-source pre-built examples based on specific pre-trained TensorFlow or TFLite models. Due to its simplicity of setup process and using python as programming language, the Media Pipe Python solution is the easiest for beginners. Media Pipe is mainly used for media processing.

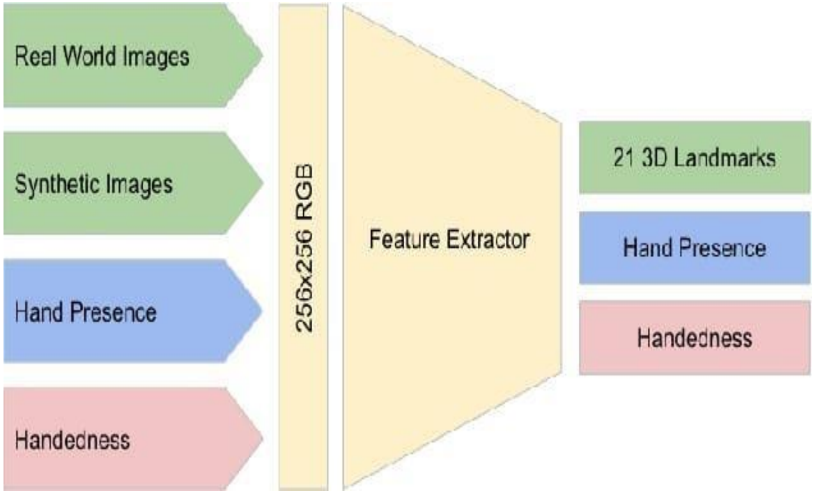


FIGURE 1. Architecture of hand landmark model

Media pipe Hands is a finger-tracking solution that uses ML to derive 21 hand landmarks from a single image. These21 landmarks cover the entire palm of our hand, and based on the movements between those points and the angle between them, based on those points, the output is determined. Media Pipe operates in a real-time work environment,it captures the movements that are happening in the Realtime by landmarks that are pointed at specified points. Theselandmarks determine the overall output based on their movements.

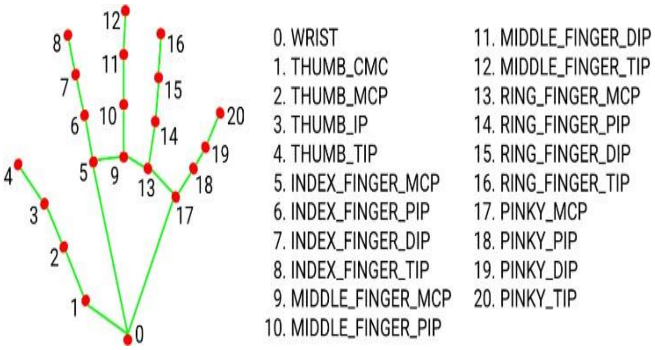


FIGURE 2. Hand Landmark detection on MediaPipe

PyAutoGUI

PyAutoGUI is an automation library in python to automate mouse and keyboard movements. It controls these input devices, so there is no need to use these input devices to control the computer. By giving appropriate instructions, the cursor will navigate to a specific position and perform actions without computer-human interaction. We need to importthe PyAutoGUI library into python using "import pyautogui" to do any automation. By writing code using PyAutoGUI, it can control other applications by sending virtual mouse clicks and keystrokes without our physical manipulation.

4. DESIGN PROCEDURE

In this design procedure, using libraries like PyAutoGUI, OpenCV, and MediaPipe, we can control the media player by hand gestures in real-time. The MediaPipe library has in-built solutions for our hand gestures. Here, 21 landmarks are pointed at the palm, while changing the gestures, based on the movements of these landmarks on our palm, the respective command is executed and the media player is controlled. In this application, mainly we are using8 gestures to control the media player. The commands that are executed by our hand gestures are to pause and play the video, volume up and volume down, forward and backward the video, and enable and disable those gesture commands. These are the commands executed by changing the gesture of our hand. For example, to pause the video, the gesture we are using is to close all the fingers and make it a fist. Here, MediaPipe calculates the angle of the fingersand the angle acquired from a fist will be lesser than 50 degrees. So, it pauses the video. Similarly, remaining gesturesare taken as input and respective commands will be executed. For gesture recognition, we are taking two conditions for the finger. They are open fingers and closed fingers. The finger is considered as open if the angle of the finger is greater than 150 and the finger is considered closed if the angleof the finger is lesser than 70 degrees. To measure the distance between two fingers, we are using the distance formulaand to measure the angle of the finger we are using the cosine theorem. To exit from this operation we have to click on the 'q' button. The whole process will be terminated. Our hand marked with 21 landmarks which covers the wholepalm to calculate the distance and angle between fingers. And this is done by two theorems. They are:

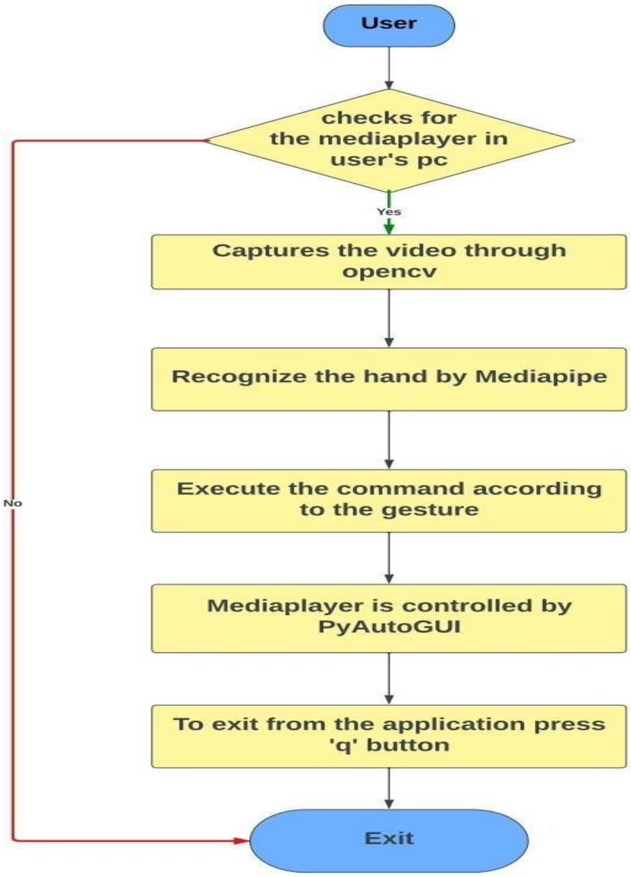

User
checks for the mediaplayer in user's pc
Yes
No
Captures the video through opencv
Recognize the hand by Mediapipe
Execute the command according to the gesture
Mediaplayer is controlled by PyAutoGUI
To exit from the application press 'q' button
Exit

FIGURE 3. Flow chart diagram of the hand gesture recognition process.

1. *Cosine Theorem*

Cosine theorem is used for calculating angles between three points by the distances among the points.

To know whether the finger is open or close, we measure the angle between the points present in each finger and if the measured angle is greater than 150 the finger is considered as opened and if it less than 70,the finger is closed. Cosine rule is derived below.

For a triangle ABC, BC=a, AB=c and AC= b.

$a2 = b2 + c2 - 2bc\,cosA;$

$b2 = a2 + c2 - 2ac\,cosB;$

$c2 = b2 + a2 – 2ab\,cosC$

$cos\ A = (b2 + c - 2a2)/2bc$

$cos\ B = (a2 + c - 2b2)/2ac$

$cos\ C = (b2 + a - 2c2)/2b$

The above equations are for cosine formula.

Distance between two points

It is used to calculate the distance between two points. Consider two points on a plane(x1,y1) and (x2,y2). The distance between two points is given by

$$D = \sqrt{}|y2 - y1|2 + |x2 - x1|2$$

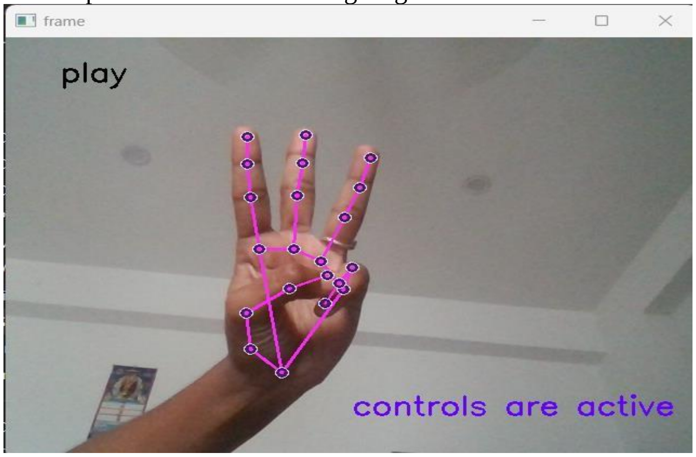
The above equation is distance formula.

To execute this process firstly we have to download the virtual environment by the command ‘pip install virtualen’. Then we have to ensure that the pip is upto date. Now we have to create the virtual environment using command ‘py -m venv venv’. Next we have to activate virtual environment by running activate script by giving command ‘.\venv\Scripts\activate’. Now we have to install the required modules or packages by the command ’pip install -r requirements.txt’. To run the source file ‘py .\public\main.py’ command is used. At last we have to select the video that has to be played.

5. RESULTS

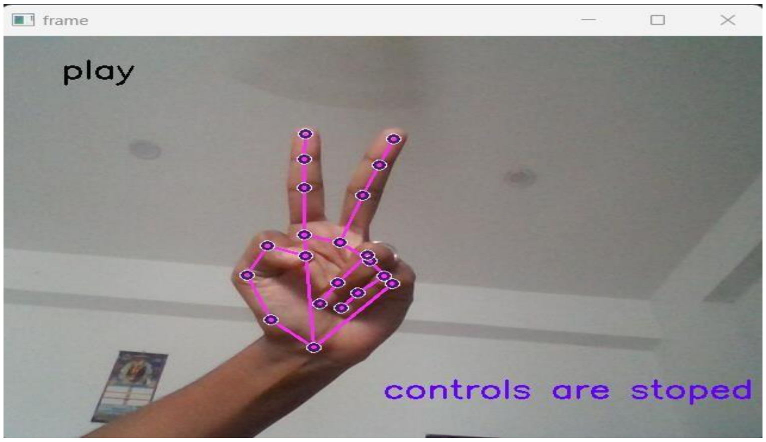
By using the inbuilt libraries, the output obtained by the system is accurate output. Totally we used 8 gestures for controlling media player. They are:

For activating commands, we used gestures in which the angle of the ring finger, middle finger, and index fingershould be greater than 150 and should be opened with the remaining fingers closed.



controls are active

FIGURE 4. Hand gesture recognition controls are activated



controls are stoped

FIGURE 5. Hand gesture recognition controls are deactivated

To deactivate the gesture commands, we used a gesture that the angle index finger and middle finger should begreater than 150, and the remaining fingers are closed.

To increase the volume of the video, the gesture we used is thumb finger open and pointed upward with the remaining fingers closed of user’s right hand.

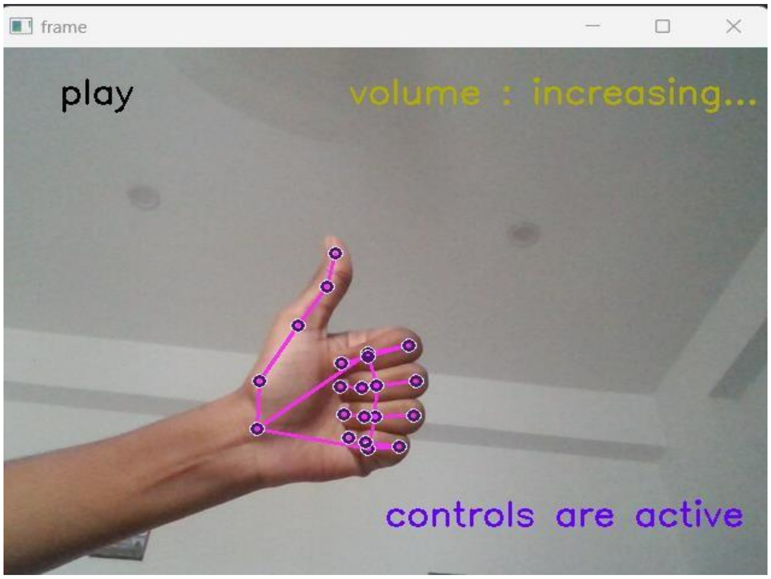


FIGURE 6. Increasing volume

To decrease the volume of the video, the gesture we used is pointing the thumb finger downward with the remaining fingers closed of the user’s right hand.

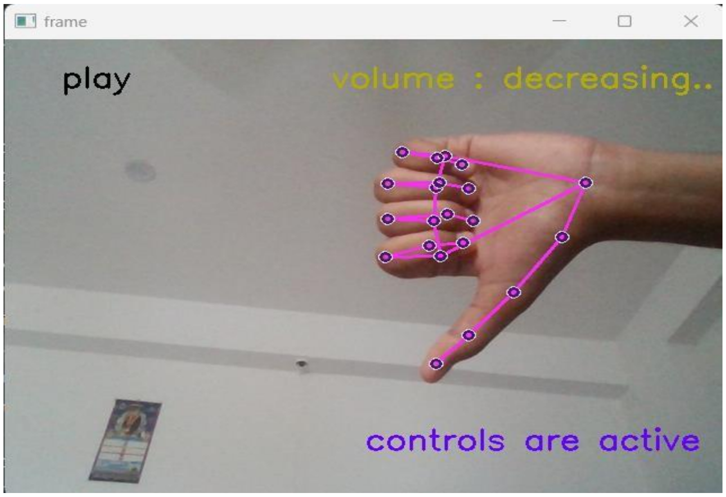


FIGURE 7. Decreasing volu

To pause the video we are closing all the fingers in our hand and making it a fist. By making that gesture thevideo will stop playing.

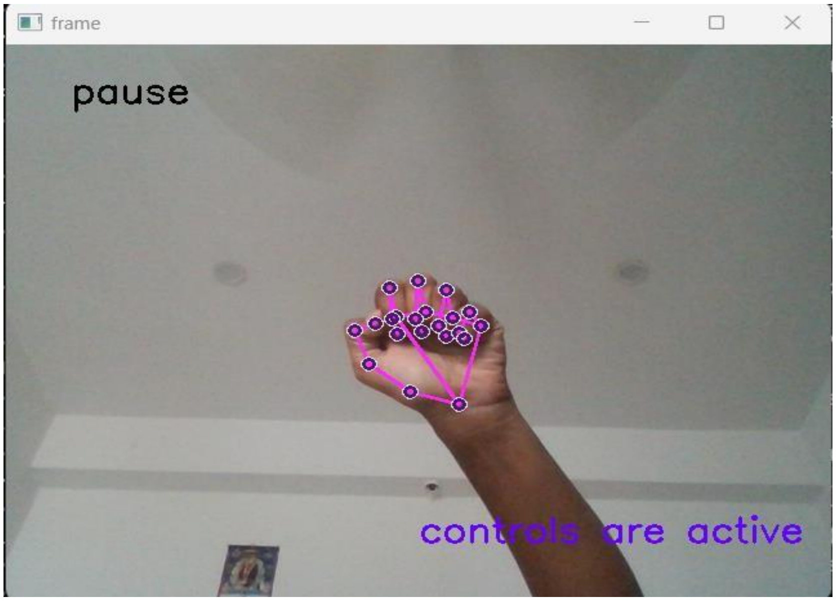


FIGURE 8. Pausing the video

To play the video we are opening all the fingers in our hand. By making that gesture the video will start playing.

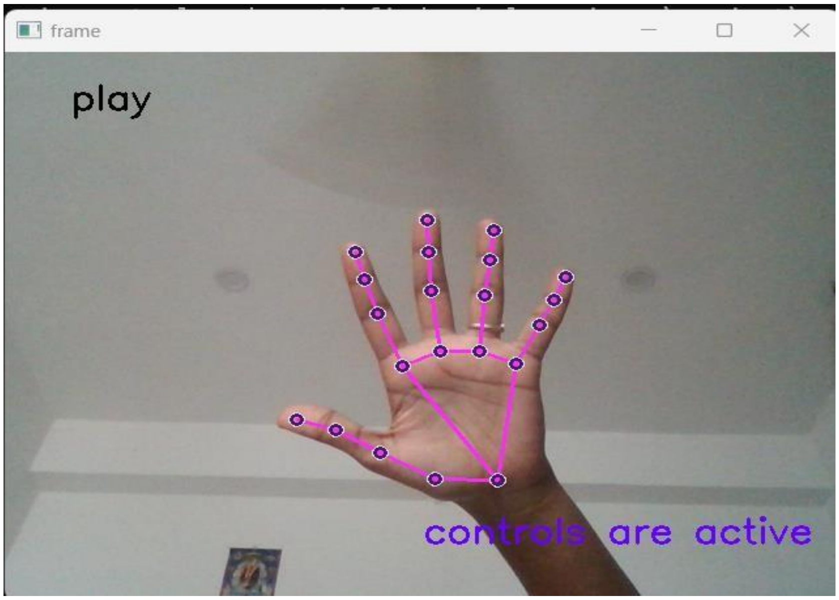
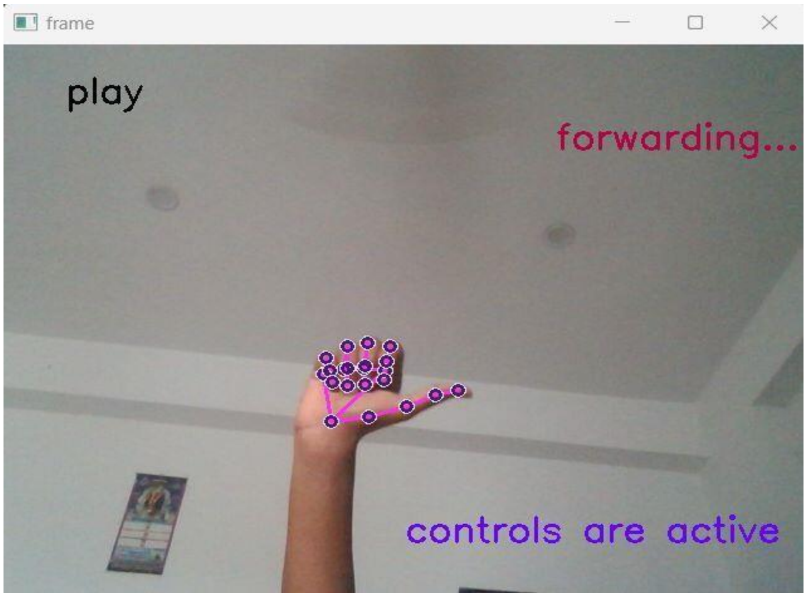


FIGURE 9. Play the video

To forward the video, the user should use their left hand with opened thumb finger pointing right side with the remaining fingers closed.



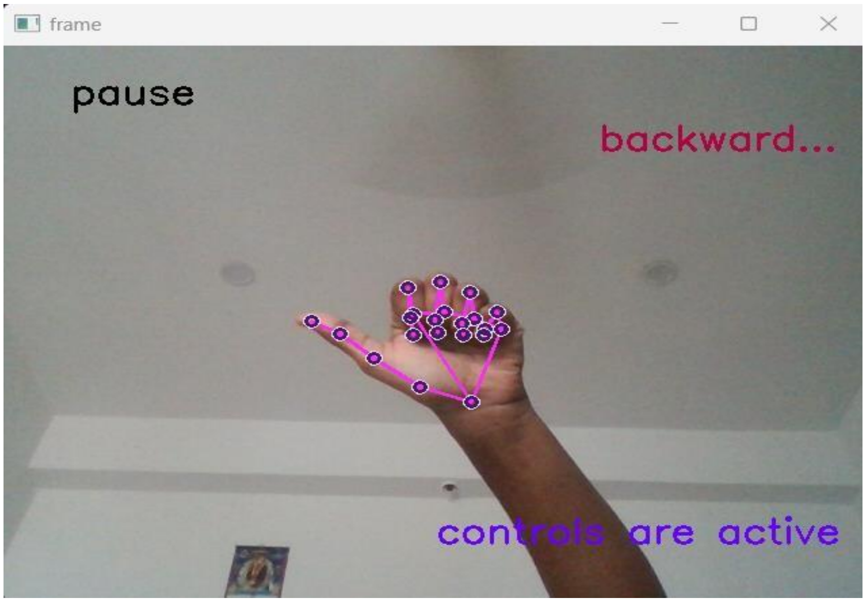
play

forwarding...

controls are active

FIGURE 10. Forward operation

To forward the video backward, the user should use their right hand with opened thumb finger pointing left sidewith the remaining fingers closed.



pause

backward...

controls are active

FIGURE 11. Backward operation

As shown above, the video is controlled by different hand gestures. We can also add different hand gestures for the same operations.
Now let us compare the presented model with existing models to compare its accuracy.

TABLE 1. Comparision of performances in SVM, KNN,CNN and Media Pipe.

Sl.No	Classifier Type	Accuracy(%)	Application
1	Support Vector Machine(SVM)	49.2	Robotic Control
2.	K-Nearest Neighbors (KNN)	93.2	Static GestureRecognition
3.	Convolution Neural Network (CNN)	93.09	Static and DynamicGesture Recognition
4.	MediaPipe Hand Solutions	95.7	Controlling MediaPlayer

6. CONCLUSION

We set out to create a system that can control the media player using Media Pipe on the basis of hand gesture commands given to it. Such a system can automate the media player and can provide different commands to perform on the media player using different gestures. Such a strategy for hand sign detection is presented in the proposed method. Our hand will be detected by subtracting our hand from the background-by-background subtraction technique. By creating this system, we can also extend this model for using different hand recognition methods to use for variable applications. Our system also has an easy-to-use interface. By using Media Pipe to recognize our palm, we are getting a precision of 95.7%.

Future Scope:

Hence we are using the inbuilt libraries to capture and progress the image in real-time, we can use this way to control other applications by hand gestures. The user has the flexibility to choose their own hand gestures. They can use other methods to capture the image in real-time.

They can use different technologies to interact with the computer without physically interacting with their input devices.

REFERENCES

[1]. Abhik Swarnakar, Dr. Anuradha Kanade, Ashish Pal, Sonu Kumar Soni, Rajesh Kumar, 2020, Hand Gesture Recognition and Finger Count, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) ICSITS – 2020 (Volume 8 – Issue 05).

[2]. S. Hussain, R. Saxena, X. Han, J. A. Khan and H. Shin, "Hand gesture recognition using deep learning," 2017 International SoC Design Conference (ISOCC), Seoul, Korea (South), 2017, pp. 48-49, doi: 10.1109/ISOCC.2017.8368821.

[3]. Khan, Rafiqul Zaman & Ibraheem, Noor. (2012). Hand Gesture Recognition: A Literature Review. International Journal of Artificial Intelligence & Applications (IJAIA). 3. 161-174. 10.5121/ijaia.2012.3412.

[4]. N. Mohamed, M. B. Mustafa and N. Jomhari, "A Review of the Hand Gesture Recognition System: Current Progress and Future Directions," in IEEE Access, vol. 9, pp. 157422-157436, 2021, doi: 10.1109/ACCESS.2021.3129650.D. J. Reddy and M. R. Kumar, “Crop Yield Prediction using Machine Learning Algorithm” in 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS),2021, pp.14661470, doi:10.1109/ICICCS51141.2021.9432236.

[5]. J. S. Sonkusare, N. B. Chopade, R. Sor and S. L. Tade, "A Review on Hand Gesture Recognition System," 2015 International Conference on Computing Communication Control and Automation, Pune, India, 2015, pp. 790- 794, doi: 10.1109/ICCUBEA.2015.158.

[6]. Deepali N. Kakade, Prof. Dr. J.S. Chitode, 2012, Dynamic Hand Gesture Recognition : A Literature Review, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 01, Issue 09 (November 2012)

[7]. Wang, Xianghan & Jiang, Jie & Wei, Yingmei & Kang, Lai & Gao, Yingying. (2018). Research on Gesture Recognition Method Based on Computer Vision. MATEC Web of Conferences. 232. 03042. 10.1051/matecconf/201823203042.

[8]. Yasen, Mais & Jusoh, Shaidah. (2019). A systematic review on hand gesture recognition techniques, challenges and applications. PeerJ Computer Science. 10.7717/peerj-cs.218.

[9]. Anju S R, Subu Surendran, 2014, A Study on Different Hand Gesture Recognition Techniques, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 03, Issue 04 (April 2014).

[10]. Chong Wang, Sabiha Pathan, Neha Rokade at Uma Annamalai “Hand Gesture Recognition System For Multimedia Applications”, International Research Journal of Engineering and Technology (IRJET) Volume: 03 Issue: 04 | Apr-2016, Apr2016.

[11]. Swapnil D. Badgujar , Amrapali Waghmare , Reshma Ganjewar , Dr. Abhijit Banubakode "A Media Player which operates depending on Human Emotions" International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 5, May 2015

[12]. Viraj Shinde, Tushar Bacchav, "Emotion Detection Using Facial Expression", Journ. International Journal Of Advanced ResearchIn Computer Science & Software Engineering, Vol. 4, PP-465467, April 2016

[13]. J. Hossain Gourob, S. Raxit and A. Hasan, "A Robotic Hand: Controlled With Vision Based Hand Gesture Recognition System," 2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI), Rajshahi, Bangladesh, 2021, pp. 1-4, doi: 10.1109/ACMI53878.2021.9528192.

[14]. M. Garg, P. M. Pradhan and D. Ghosh, "Multiview Hand Gesture Recognition using Deep Learning," 2021 IEEE 18th India Council International Conference (INDICON), Guwahati, India, 2021, pp. 1-6, doi: 10.1109/INDICON52576.2021.9691523.

[15]. J. Shukla and A. Dwivedi, "A Method for Hand Gesture Recognition," 2014 Fourth International Conference on Communication Systems and Network Technologies, Bhopal, India, 2014, pp. 919-923, doi: 10.1109/CSNT.2014.189.

[16]. K. Teachasrisaksakul, L. Wu, G. -Z. Yang and B. Lo, "Hand Gesture Recognition with Inertial Sensors," 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, USA, 2018, pp. 3517-3520, doi: 10.1109/EMBC.2018.8513098.

[17]. Manav Prajapati, Mitesh Makawana, Sahil Hada, "Indian Sign Language Recognition for Static and Dynamic Hand Gestures," International Journal of Computer Sciences and Engineering, Vol.8, Issue.9, pp.54-58, 2020.

[18]. Apoorva M A, Harshitha M S, Chaitra S, Akshitha V, Rajath A N, 2017, An Efficient and Robust System for Hand Gesture Recognition and Interpretation, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCETEIT – 2017 (Volume 5 – Issue 20).

[19]. Cheok, M.J., Omar, Z.B., & Jaward, M.H. (2019). A review of hand gesture and sign language recognition techniques. International Journal of Machine Learning and Cybernetics, 10, 131-153.

[20]. R.M. Kagalkar, S.V Gumaste, "Gradient Based Key Frame Extraction for Continuous Indian Sign Language Gesture Recognition and Sentence Formation in Kannada Language: A Comparative Study of Classifiers," International Journal of Computer