GESTURE CONTROLLED MEDIA PLAYER

A PROJECT REPORT

Submitted

in the partial fulfillment of the requirements for

the award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

D. JAYAPRADA (18001A0504)

B. SUDHA RANI (18001A0503)

P. MADHAVAN (19005A0501)

Under the guidance of

Dr. R. RAJASEKHAR **M.Tech., Ph.D.**

Associate Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
COLLEGE OF ENGINEERING *(Autonomous)*

ANANTHAPURAMU - 515002

ANDHRA PRADESH

2022

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

COLLEGE OF ENGINEERING *(AUTONOMOUS)*

ANANTHAPURAMU – 515002

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled **"GESTURE CONTROLLED MEDIA PLAYER"** is a bonafide work of **D.JAYAPRADA**, bearing Admission No: **18001A0504**, **B.SUDHA RANI**, bearing Admission No: **18001A0503** and **P. MADHAVAN**, bearing Admission No: **19005A0501** is submitted to the faculty of Computer Science and Engineering, in a partial fulfillment of the requirements for the award of degree of **BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING** from Jawaharlal Nehru Technology University Anantapur, College of Engineering *(Autonomous)* Ananthapuramu.

SIGNATURE OF GUIDE

Dr. R.RAJASEKHAR M.Tech., Ph.D.,

Associate Professor

Department of CSE

JNTUA College of Engineering

ANANTHAPURAMU-515002

SIGNATURE OF H.O.D

Dr. K. MADHAVI M.Tech., Ph.D.,

Associate Professor & H.O.D

Department of CSE

JNTUA College of Engineering

ANANTHAPURAMU-515002

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success.

We are highly indebted to **Dr. R. Rajasekhar**, Associate Professor, Computer Science and Engineering, JNTUA College of Engineering, Ananthapuramu for his guidance and constant supervision as well as for providing necessary information regarding the project. His support has helped us in the completion of the project.

We would like to express special thanks to **Dr. K. Madhavi**, Head of the Department, Computer Science and Engineering, JNTUA College of Engineering, Ananthapuramu. Her wide support, knowledge and enthusiastic encouragement has inspired us to get better involved with our project and technical design.

We want to express our gratitude to **Prof. P. Sujatha**, Principal of JNTUA College of Engineering Ananthapuramu, for her co-operation and her timely help in the successful completion of the project.

We express our sincere thanks to the project committee members, faculty and staff of Computer Science and Engineering, JNTUA College of Engineering, Ananthapuramu, for their valuable guidance and technical support. Last but far from least, we also thank our family members and our friends for their moral support and constant encouragement, we are very thankful to one and all who helped us for the successful completion of the project.

With Gratitude,

D. JAYAPRADA (18001A0504)

B. SUDHA RANI (18001A0503)

P. MADHAVAN (19005A0501)

ABSTRACT

Computer is becoming more important in our daily life. Computer applications require interaction between human and computer. The primary goal of this project is to create a system, which can identify hand gestures to convey information for controlling VLC media player. The gestures would serve as the direct command for operations of VLC media player.

A hand Gesture is basically the movement, position or posture of hand used extensively in our daily lives as part of non-verbal communication. This system uses simple gestures to control the video .The users can recollect a group of gestures to control the video playback. This project implements computer vision and gesture recognition techniques.

Generally, media player has functions like volume up, volume down, play, pause, forward, backward operations. This gesture-based interaction controls those functions of a media player through webcam. These operations can be achieved through Image Processing and Pyautogui. The image processing is done by using OpenCV.

TABLE OF CONTENTS

ABSTRACT i

TABLE OF CONTENTS ii

LIST OF FIGURES iii

CHAPTERS **PAGE No**

1. INTRODUCTION ……1-3

1.1.Gesture Controlled media player…….2

1.2.Problem Statement……...2

1.3.Objectives……3

1.4.Scope……3

2. LITERATURE SURVEY ……...4-6

2.1 Existing System……5

2.2 Proposed System…….6

3.SYSTEM ANALYSIS ……7-14

3.1 What is Gesture Controlled Media Player.?.....8

3.2 Software Requirement Specifications……..8-10

3.3 Technology Tools and Programming Languages Used…....10-14

4.SYSTEM DESIGN …….15-19

4.1 Use Case Diagram……....16-17

4.2 Activity Diagram…….17-18

4.3 Sequence Diagram…….....18-19

5.SYSTEM IMPLEMENTATION …….20-27

5.1 Interface Design……...21-24

5.2 Input Design…….24-27

6. **CONCLUSIONS…..……..28-29**

REFERENCES ……30

LIST OF FIGURES

FIG 4.1.1 Use Case Diagram for Gesture Controlled Media Player…..17

FIG 4.2.1 Activity Diagram for Gesture Controlled Media Player……18

FIG 4.3.1 Sequence Diagram for Gesture Controlled Media Player…..19

FIG 5.1.1 Welcome Page……23

FIG 5.1.2 Instructions and Gestures to be Use……24

FIG 5.2.1 Forward Action using Hand Gesture……..25

FIG 5.2.2 Backward Action using Hand Gesture……25

FIG 5.2.3 Volume Up Action using Hand Gesture…….26

FIG 5.2.4 Volume Down Action using Hand Gesture……26

FIG 5.2.5 Play/Pause Action using Hand Gesture……..27

CHAPTER 1

INTRODUCTION

In today's world, the computers have become an important aspect of life and are used in various fields. So, there must be an interaction between the computer and the user. Although, computers have made numerous advancement in both fields of Software and Hardware, Still the basic way in which Humans interact with computers remains the same, using basic pointing device such as mouse, or through Keyboard or through Voice, or through gestures to make this communication more user friendly.

1.1 GESTURE CONTROLLED MEDIA PLAYER

Gesture Controlled Media Player uses hand gestures to perform various operations such as play, pause, seek forward, seek backward, volume up, volume down in VLC media player. The proposed system can be used to control various operations using hand gestures with the help of programming in python to facilitate interaction within different functions of computer through the Camera to capture video frames. Gesture Controlled Media Player recognizes the gestures and performs functions of VLC media player.

1.2 PROBLEM STATEMENT

The problem Gesture Controlled Media Player using Hand Gesture Recognition solves to facilitate the process of controlling VLC media player functions (Pause, Play, Volume up, Volume Down, Seek Forward, Seek Backward) by mere hand gestures without pressing buttons or tapping onto the screen. This can be used in our daily life while watching videos.

1.3 OBJECTIVES

The main intention of this project is to operate the functions of media player using hand gestures. The objectives of Gesture Controlled Media Player are as follows:

- Develop a computer vision application for simple gesture recognition.
- GUI of media player should be user friendly and provide efficiency

- The hand gestures should be captured precisely and actions related to them should be performed accurately.

1.4 SCOPE

The scope of our project is to develop a real time gesture recognition system which ultimately controls the media player (i.e. VLC Media Player). During the project, five gestures were chosen to represent five navigational commands for the media player, namely Volume Up, Volume Down, Move Forward, Move Backward, Play, and Pause. A simple computer vision application was written for the detection and recognition of the five gestures and their translation into the corresponding commands for the media player. The appropriate OpenCV functions and image processing algorithms for the detection and interpretation of the gestures were used. Thereafter, the program was tested on a webcam with actual hand gestures in real-time and the results were observed.

This enhanced media player can help in minimizing human efforts. In future, this technique can be used to control systems like pdf reader, power point etc.

- Get better experience of using media player
- We have tried to achieve this goal by automating it to a wide extent.
- We are doing this by using hand gestures for controlling various features of the media player.

CHAPTER 2

LITERATURE SURVEY

To develop a convenient human-machine Interface for each consumer electronics product has become an important issue. The traditional electronic input devices, such as mouse, keyboard, and joystick are still the most common interaction way. However, it does not mean that these devices are the most convenient and natural input devices for most users. Since ancient times, gestures are a major way for communication and interaction between people. People can easily express the idea by gestures before the invention of language. In recent years, the gesture control technique has become a new developmental trend for many human based electronics products, such as computers, televisions, and games. This technique let people can control these products more naturally, intuitively and in case of existing system.

2.1 EXISTING SYSTEM

Gesture Controlled Media Player is mainly created to develop an environment where man-machine interactions can be done efficiently with much ease. There are many approaches that were followed peculiarly by different researchers like vision based, data glove based, Artificial Neural Network, Fuzzy Logic, Genetic Algorithm, Hidden Markov Model, Support Vector Machine and Vision based approaches for identifying hand gestures. Researchers classified gesture recognition in to three major steps; they are extraction, features estimation, and classification or recognition. The required portion of the hand was extracted using this colour model and filtered using median filter and smoothing filter. The edges were detected and features such as hand perimeter, aspect ratio, hand area are extracted. Accuracy of this approach was found to be 87.4%. Some researchers have scanned the skin filtered image in all direction to find out the edges of the fingers. Then the edge tips were assigned with highest pixel value and thereby fingertips were detected. Some researchers have used skin-color histograms to perform segmentation process for hand gesture recognition. But it did not yield the desired results.

2.2 PROPOSED SYSTEM

The proposed system usually contain three major stages. The first stage is the object detection. The target of this stage is to detect hand objects in the digital images or videos. Many environment and image problems are needed to solve at this stage to ensure that the hand contours or regions can be extracted precisely to enhance the recognition accuracy. The media pipe module is a better solution to construct an adaptive and robust gesture recognition system. The second stage is object recognition. The detected hand objects are recognized to identify the gestures. The third stage is to analyse sequential gestures to identify users gestures. This project availed of algorithms commonly used in computer vision. These include those used in detecting hand landmarks, counting the number of fingers that are opened by the user with Mediapipe library and controlling media player using pyautogui. Flask is used to build the User Interface of the system.

CHAPTER 3

SYSTEM ANALYSIS

Analysis can be defined as breaking up of any whole so as to find out their nature, function etc. It defines design as to make preliminary sketches to sketch a pattern or outline for plan. To plan and carry out especially by artistic arrangement or in a skill full wall. System analysis and design can be characterized as a set of techniques and processes, a community of interests, a culture and an intellectual orientation. The various tasks in the system analysis include the following.

1. Understanding application.
2. Planning.
3. Scheduling.
4. Developing candidate solution.
5. Performing trade studies.
6. Performing cost benefit analysis.
7. Recommending alternative solutions.
8. Selling of the system.
9. Supervising, installing and maintaining the system

3.1 What is Gesture Controlled Media Player.?

Gesture Controlled Media Player is a non tangible approach to perform functions of video player based on human hand gestures by using openCV. Gesture control devices can detect and decipher movements or actions and translate them into its functions. Gesture control technology is based on gesture recognition. Compared to the Primitive user interfaces, such as keyboard and mouse this builds a richer bridge between the computers and humans.

3.2 Software Requirement Specifications

The software requirements specification is produced at the culmination of the analysis task. The function and performance allocated to the software as a part of system engineering are refined by establishing a complete information description, a detailed functional and behavioural description, and indication of performance requirements and design constraints, appropriate validation criteria and other data pertinent to requirements.

Software and Hardware Requirements Specifications

Designing or developing a system is on something, another is implementing, not just implement in git , but the successful implementation. Implementation of any information system or automated system depends on the hardware, software and people. Hence, the hardware being mainly a personal computer(PC) with printers, scanners and any peripheral that is needed. All the users must be trained and computer literates else the system's aim and objective will be defeated. An outline of the Software Requirements Specification: A simplified outline can be given for the framework of the specifications.

HARDWARE used in the system:

The following components are used:

- Personal Computer (PC) / Laptop (i3 Processor or Higher, 8GB RAM or Higher with Inbuilt Webcam)

SOFTWARE used in the system:

- Microsoft Windows 10 or 11
- Spyder (Python 3.7)

- Libraries:
    - Flask(1.1.1)
    - Pip(20.0.2)
    - Mediapipe(0.8.9)
    - Pyautogui(0.9.53)
    - Cv2(4.5.1)

Non-functional Requirements

- **Usability:** The user is facilitated with the control section for the entire process in which they can arrange the position of hand in the area such that hand gesture is visible to camera. The implementation and calibration of camera and its resolution can also be done as per quality and preciseness requirement.The frame size, flow rate and its command variation with respect to threshold developed and colour component of hand colour, can be easily calibrated by means of certain defined thresholds.
- **Security and support:** Application will be permissible to be used only in secure network so there is less feasibility of insecurity over the functionality of the application. On the other hand, the system functions in a real time application scenario, therefore the camera, colour and platform compatibility is must in this case. IN case of command transfer using certain connected devices or wireless communication, the proper port assignment would also be a predominant factor to be consider.
- **Maintainability:** The installation and operation manual of the project will be provided to the user.
- **Extensibility:** The project work is also open for any future modification and hence the work could be define as the one of the extensible work.

3.3 TECHNOLOGY, TOOLS AND PROGRAMMING LANGUAGE USED

PYTHON OVERVIEW

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently

where as other languages use punctuation, and it has fewer syntactical constructions than other languages. Python can be used for web development frameworks such as Flask and Django.

OPENCV

OpenCV is a Python open-source library, which is used for computer vision in Artificial intelligence, Machine Learning, face recognition, etc .In OpenCV, the CV is an abbreviation form of a computer vision, which is defined as a field of study that helps computers to understand the content of the digital images such as photographs and videos .The purpose of computer vision is to understand the content of the images.

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features.

The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When OpenCV was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.

FLASK

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself.

Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

MEDIAPIPE

MediaPipe offers cross-platform, customizable ML solutions for live and streaming media. MediaPipe Hands is a high-fidelity hand and finger tracking solution. It employs machine learning (ML) to infer 21 3D landmarks of a hand from just a single frame. Whereas current state-of-the-art approaches rely primarily on powerful desktop environments for inference, our method achieves real-time performance on a mobile phone, and even scales to multiple hands.

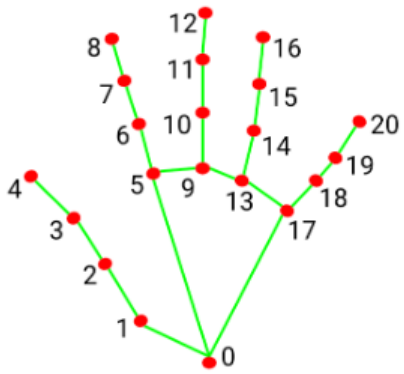
MediaPipe Hands utilizes an ML pipeline consisting of multiple models working together: A palm detection model that operates on the full image and returns an oriented hand bounding box. A hand landmark model that operates on the cropped image region defined by the palm detector and returns high-fidelity 3D hand keypoints.

Palm Detection Model

To detect initial hand locations, a single-shot detector model is optimized for mobile real-time uses in a manner similar to the face detection model in MediaPipe Face Mesh. Detecting hands is a decidedly complex task: our lite model and full model have to work across a variety of hand sizes with a large scale span (~20x) relative to the image frame and be able to detect occluded and self-occluded hands. Whereas faces have high contrast patterns, e.g., in the eye and mouth region, the lack of such features in hands makes it comparatively difficult to detect them reliably from their visual features alone. Instead, providing additional context, like arm, body, or person features, aids accurate hand localization.

Hand Landmark Model

After the palm detection over the whole image our subsequent hand landmark model performs precise keypoint localization of 21 3D hand-knuckle coordinates inside the detected hand regions via regression, that is direct coordinate prediction. The model learns a consistent internal hand pose representation and is robust even to partially visible hands and self-occlusions.



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

Based on the above 21 landmarks, the finger can be detected whether it is opened or closed. There are four landmarks on index finger, middle finger, ring finger, little finger except thumb which as 3 landmarks. So, to detect whether a finger is open or close, take two landmarks such as 6,8 on index finger. If landmark 8 is above the landmark 6,then the finger is open, else the finger is closed. To find the total number of fingers opened, the same process must be followed. The tip ids of all the fingers are 4,8,12,16,20. Using these tip ids and the landmarks below these, we count the total number of fingers opened. Based on total number of fingers the following functions of VLC Media Player happens.

No of fingers	Function performed
0	No Action
1	Forward
2	Backward
3	Volume Up
4	Volume Down
5	Play/Pause

PYAUTOGUI

The pyautogui module is a Python package that comes with multiple functions that can be used to automate the GUI of our system .We can also use the functions of this module to automate the functioning of the keyboard and mouse of our system.

The API is designed to be simple. Pyautogui works on Windows, macOS, and Linux, and runs on Python 2 and 3. Pyautogui has several features: Moving the mouse and clicking in the windows of other applications. Display alert and message boxes.

To press these keys, call the press() function and pass it a string from the pyautogui. KEYBOARD_KEYS such as enter , esc.The press() function is really just a wrapper for the keyDown() and keyUp() functions, which simulate pressing a key down and then releasing it up. These functions can be called by themselves.

CHAPTER 4

SYSTEM DESIGN

System Design focuses, on the requirements for this automated system implementation, moreover the automated system is described in UML(Unified Modelling Language) and the use case diagram is given. An automated system that is not implemental or if not successful implemented cannot be used to perform well whatever the case may be. “Even after the development phase is over, the system project isn’t complete. The new system must be implemented or installed”. The UML is used to describe the system and the Use Case diagram shows the system design (Detail Design or design Overview).UML is a language for specifying, visualizing and constructing the artifacts of software system. A Use case diagram is one of the UML diagrams. Others are sequence diagram, state diagram, collaboration diagram and so on.

4.1 USE CASE DIAGRAM

A use case diagram shows typical interactions between a user(actor) and the working system. According to the principle of system analysis and design, use case are included in UML and it is a narrative document that describes the sequence of event of an act or using a system to complete a process. The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams.

The use case diagram of this system – Gesture controlled media player is given below

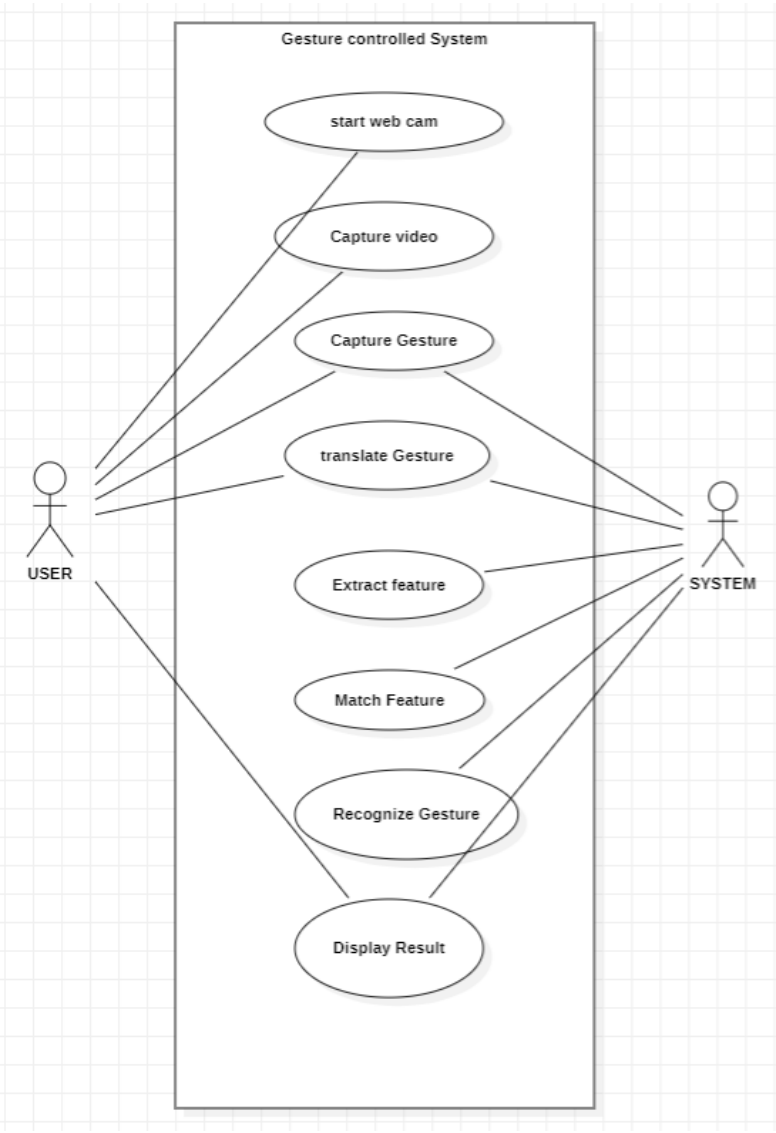


Fig 4.1.1**:** use case diagram for gesture controlled media player

4.2 ACTIVITY DIAGRAM

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. It captures the dynamic behaviour of the system.

Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and

reverse engineering techniques. It does not show any message flow from one activity to another.

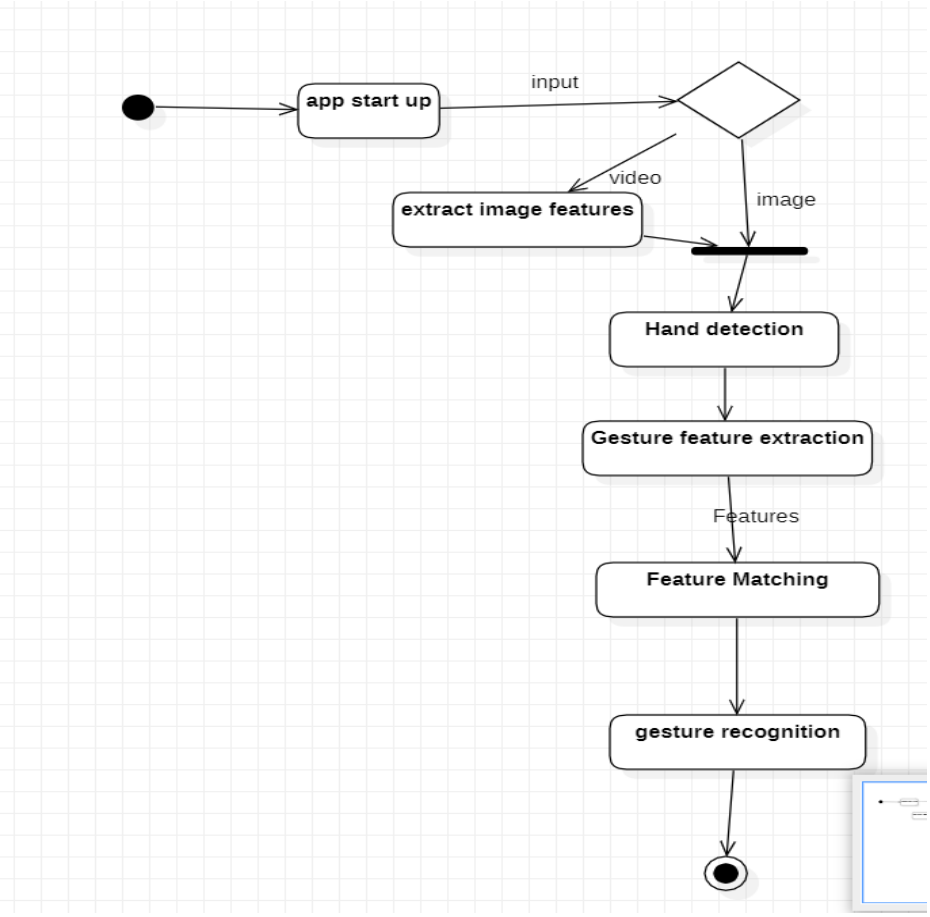


Fig 4.2.1: Activity diagram for gesture controlled media player

4.3 SEQUENCE DIAGRAM

Sequence diagram is an easy and intuitive way of describing the behaviour of a system by viewing the interaction between the system and the environment. A sequence diagram shows an interaction arranged in a time sequence. A sequence diagram has two dimensions: vertical dimension represents time; the horizontal dimension represents the objects existence during the interaction.

The shown Sequence diagram explain the flow of the program. As this System is based on Human Computer Interaction so it basically includes the user, computer and

the medium to connect both digitally that is web camera. As the execution of the program starts, it firstly invoke the web camera to detect the image of the hand. Then the image is recognised using the landmarks on the hand by using Media pipe library.After that, hand gestures are detected i.e the number of fingers raised are found.By using conditionals statements in the program the gestures are matched .As the gestures matches ,it automatically controls the media player and gives us the required results.

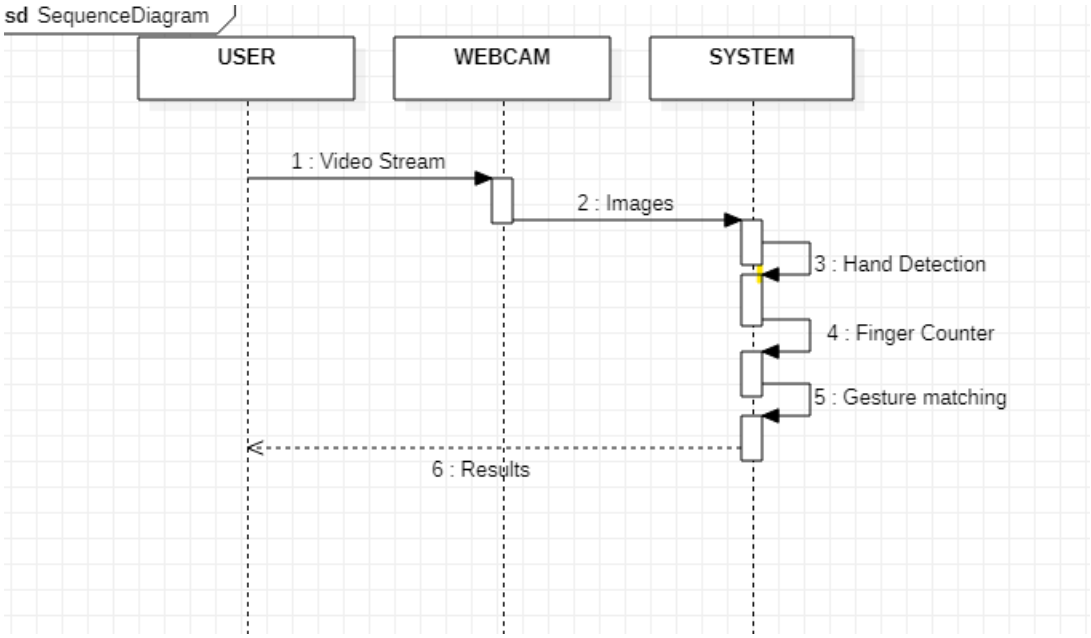


Fig 4.3.1 Sequence Diagram for Gesture Controlled Media Player

CHAPTER 5

SYSTEM IMPLEMENTATION

The implementation is the final and important phase. It involves user training, system testing and successful running of the developed proposed system. The user tests the developed system and changes are made according to their needs. The testing phase involves the testing of developed system using various kinds of data. An elaborate testing of data is prepared and the system is tested using the test data. While testing, errors are noted and corrections are made. The corrections are also noted for future use. The users are trained to operate the developed system.

INTERFACE MODULE:

In this module, user can interact with the application with instruction which are in instruction page and get the proper response from the application.

Start

Open the home page

Start button Displays

Click on start button

If Instructions page open

{

User needs to read the instructions properly

}

Now Camera opens

If Camera works

{

User give the Gestures

}

Else

Restart page

END

GESTURES MODULE:

In this module user can give the gestures and get the proper response from the Gesture Controlled Media player application.

Start

If user opens VLC player

{

User needs to select and click on the video

User needs to give the gestures

}

Switch(choice)

Case:	Swipe right with One finger
	Go forward ()
	Break
Case:	Swipe left with Two finger
	Go backward ()
	break
Case:	Swipe up with Three finger
	Volume increase ()
	break
Case:	Swipe down with Four finger
	Volume decrease ()

break

Case: Open hand

Play/Pause ()

break

Default Gesture not correct

break

END

5.1 INTERFACE DESIGN

This is the physical structure of the visible side of the system, mainly to the users. That is the front end of the system, which the users of the system interact with, it comprises of both inputs and output forms. The interface of the system is a graphical user interface and not a command type.

- Click on ‘start’ button to view the instructions to control the Gesture Controlled Media Player.

,

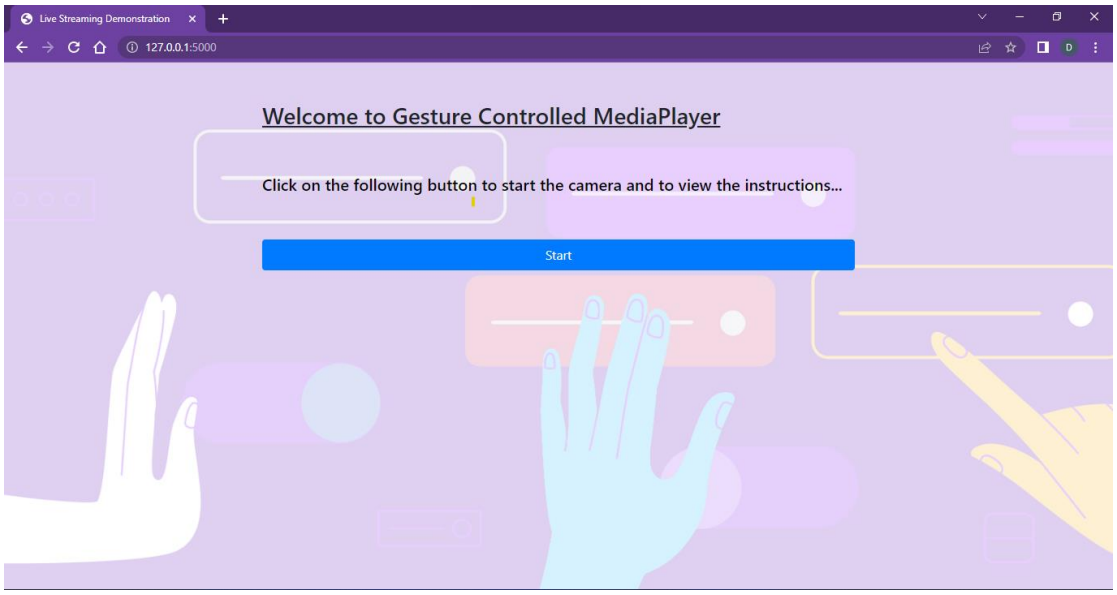


Fig 5.1.1 Welcome page.

Pages used are: index.html, app.py

- Next understand the instructions carefully and remember the Gestures to be shown.

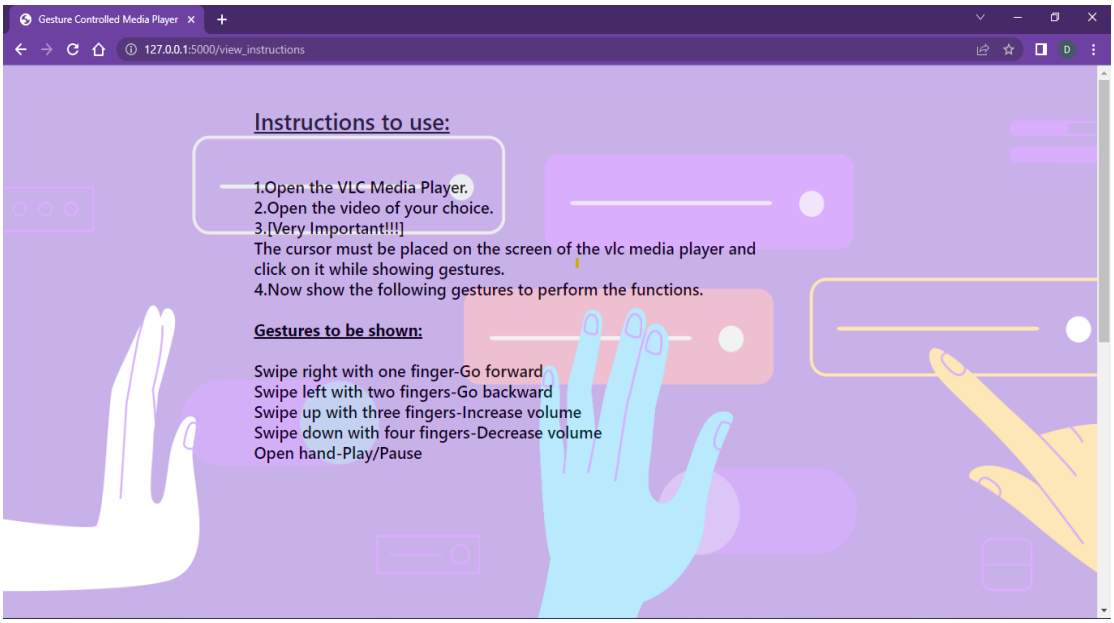


Fig 5.1.2 Instructions and Gestures to be use

Pages used are: move.html, app.py

5.2 INPUT DESIGN

An Input from the User is the most essential part to interact with the VLC player. This Media Player gives output according to the user input.

Input fields are:

- Camera
- Gestures through webcam

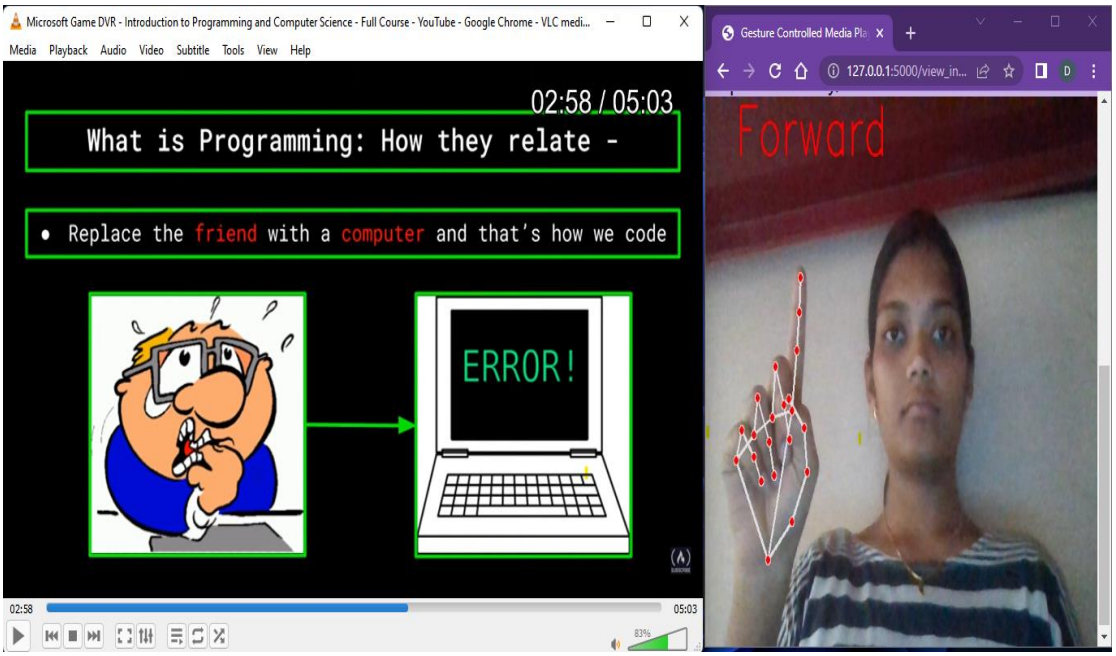


Fig 5.2.1 Forward Action using Hand Gesture

The system is successful in detecting the hand gesture for Forward and recognizing the action to be performed, so the corresponding action of forwarding the video is functioned.

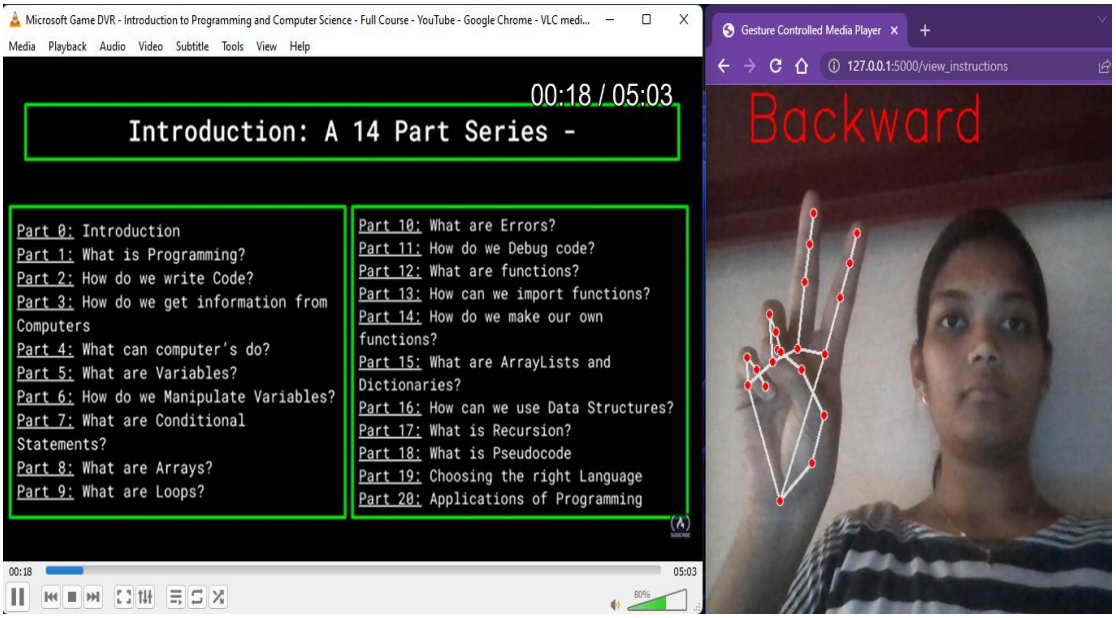


Fig 5.2.2 Backward Action using Hand Gesture

The system is successful in detecting the hand gesture for Backward and recognizing the action to be performed, so the corresponding action of backward the video is functioned.

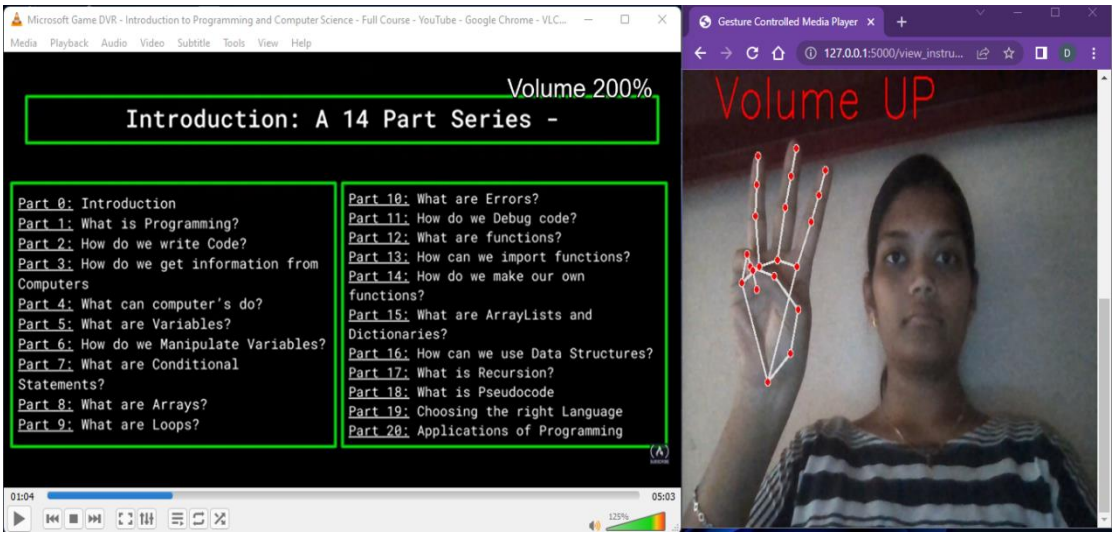


Fig 5.2.3 Volume Up Action using Hand Gesture

The system is successful in detecting the hand gesture for Volume up and recognizing the action to be performed, so the corresponding action of increasing the volume for the video is functioned.

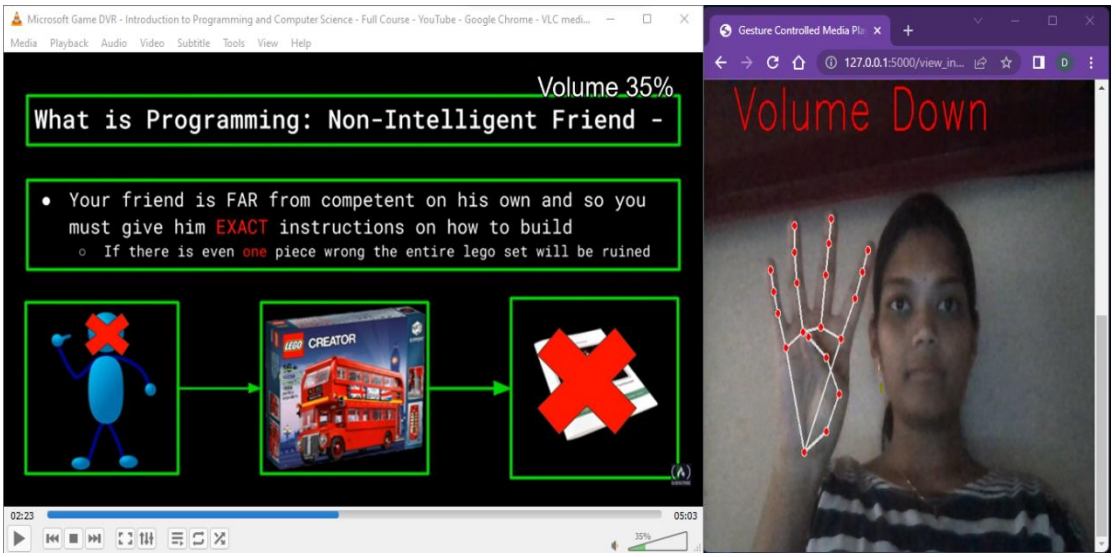


Fig 5.2.4 Volume Down Action using Hand Gesture

The system is successful in detecting the hand gesture for Volume down and recognizing the action to be performed, the corresponding action of decreasing the volume for the video is functioned.

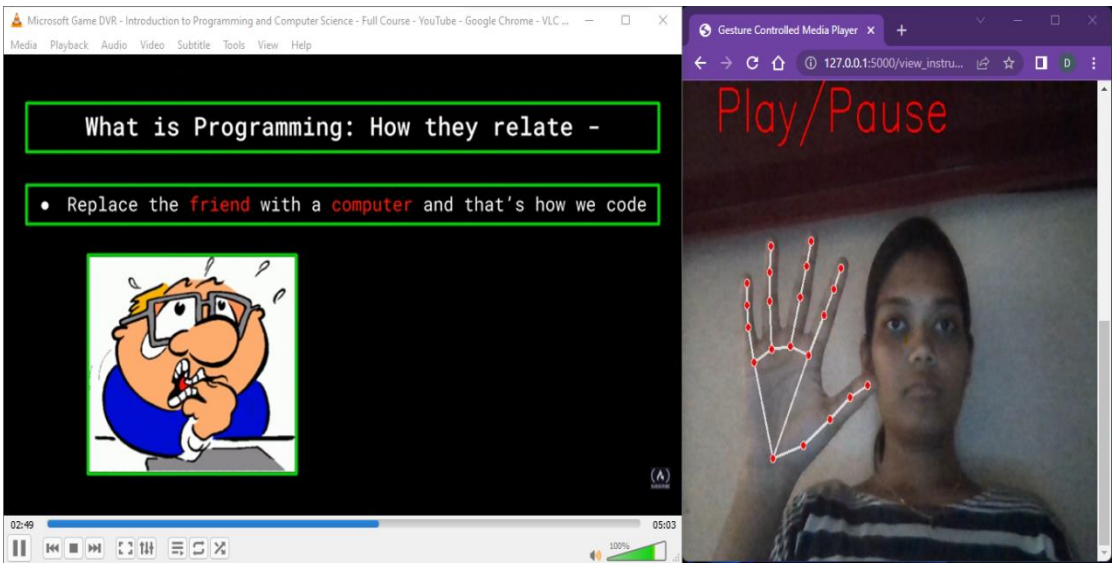


Fig 5.2.5 Play/Pause Action using Hand Gesture

The system is successful in detecting the hand gesture for Play/Pause and recognizing the action to be performed, the corresponding action of Playing for the video is functioned.

CHAPTER 6

CONCLUSIONS

The main aim of the project Gesture Controlled Media player system using Python and OpenCV, which is easy to use and it will help you to control the media player by using Hand Gestures. It provides accurate interaction between human and computer based on gestures by the User. The gesture would serve as the direct command for operations such as play, pause, volume up/down based on the user's gestures onto the screen. The Hand Gesture recognition is moving at tremendous speed for futuristic products and services and major companies are developing a technology based on the hand gesture system and that includes companies like Microsoft, Samsung, Sony and it includes the devices like Laptop, Handheld devices Professional and LED lights. Smartphones have been experiencing an enormous amount of Gesture Recognition Technology with look and views and working to manage the smartphone in reading, viewing and that includes what we call touch less gestures. For the future recommendation this system will include the execution of additional gestures that will allow user to perform more functions easily.

REFERENCES

[1] E. Joy, S. Chandran, C. George, A. A. Sabu and D. Madhu, "Gesture Controlled Video Player A Non-Tangible Approach to Develop a Video Player Based on Human Hand Gestures Using Convolutional Neural Networks," 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, pp. 56-60, doi: 10.1109/ICCONS.2018.8662901.

[2]Rautaray, Siddharth & Agrawal, Anupam. (2019). A Vision based Hand Gesture Interface for Controlling VLC Media Player. International Journal of Computer Applications. 10. 10.5120/1495-2012.

[3] Bilvika, K & M, Sahana & Patil, Tejaswini. (2021). Face and Hand Gesture Recognition System for Controlling VLC Media Player. International Journal of Scientific Research in Science and Technology. 119-124. 10.32628/CSEIT217423.

[4]Stella Nadar, Simran Nazereth, Kevin Paulson,Nilambari Narkar, “Video Controller using Hand Gesture for Disabled People” In: IJTRET(2021).

[5] S. Padmapriya & K. Sumalatha – Digital Image Processing Real Time Application;International Journal Of Engineering Science Invention(IJESI),2018