GESTURE CONTROLLED VIDEO PLAYER USING RASPBERRY PI

A Project Report submitted in partial fulfillment of the requirements for the award of degree of

BACHELOR OF TECHNOLOGY

in

ELECTRONICS & COMMUNICATION ENGINEERING

by

DUDEKULA SHABBER	-	19G31A0414
BAPURAM SHAIK SAHEB	-	19G31A0404
VEGULLA VAMSIKRISHNA	-	19G31A0424
MALA NAVEEN KUMAR	-	19G31A0440
GOLLA UDAY KUMAR	-	19G31A0421

Under the esteemed guidance of

Sri P. IMRAN KHAN M.Tech, (PhD),

Assistant Professor

Department of Electronics and Communication Engineering

St. JOHNS COLLEGE OF ENGINEERING & TECHNOLOGY

(Affiliated to Jawaharlal Nehru Technological University, Anantapur, A.P.)
Yerrakota, Yemmiganur-518360, Kurnool(Dist), A.P

2022-2023

St. JOHNS COLLEGE OF ENGINEERING & TECHNOLOGY

(Affiliated to Jawaharlal Nehru Technological University, Anantapur, A.P.)
Yerrakota, Yemmiganur-518360, Kurnool(Dist), A.P

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to Certify that the work which is being presented in the Project entitled **"GESTURE CONTROLLED VIDEO PLAYER USING RASPBERRY PI"** by **BAPURAM SHAIK SAHEB(19G31A0404), DUDEKULA SHABBER(19G31A0414), VEGULLA VAMSIKRISHNA(19G31A0424), MALA NAVEEN KUMAR (19G31A0440),GOLLA UDAY KUMAR(19G31A0421)** in partial fulfillment of the requirements for the award of the Degree of **B.Tech** submitted in the Department of Electronics & Communication Engineering at **St. JOHNS COLLEGE OF ENGINEERING & TECHNOLOGY, Yerrakota,** is an authentic record of their own work carried out during **2022-23**.

Sri. P.IMRAN KHAN M.Tech.,(PhD).
Project Guide
Assistant Professor
Dept. of ECE

Dr. K SUDHAKAR M.Tech.,Ph.D
Professor
Head of the Department
Dept. of ECE

Place: Yerrakota

Date:

Certified that the candidate was examined by me in the viva-voce Examination held at St.Johns College of Engineering & Technology, Yerrakota, on ___________________

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

At the very outset we would like to take this opportunity to express our gratitude to almighty for his blessings showered upon us to complete this project work. We are deeply indebted to our parents and family members for their unconditional love and support. They have been on our side during high and low phases of the project.

We wish to thank our Project Guide **Sri P. IMRAN KHAN** **M.Tech, (Ph.D),** Asst. Professor, St.Johns College of Engineering & Technology, for his incessant guidance, encouragement and patience in not only our project work, but also in our overall academic and professional career. His constant enthusiasm and genuine interest in our academic progress and project work has been a great source of moral support all along and also we thank for his immense insight into the analysis of our project work.

We are deeply indebted to the Project coordinator, **Sri. P. IMRAN KHAN** **M.Tech, (Ph.D),** Asst. Professor, St.Johns College of Engineering & Technology, for his generous guidance, help and useful suggestions, without which it would have been impossible.

We are very much thankful to **Dr. K SUDHAKAR** **M.Tech.,Ph.D.,** Professor, St.Johns College of Engineering & Technology, for providing us the infrastructural facilities to work in and giving us an opportunity to complete our project work.

We would like to thank **Dr. V. VEERANNA** **M.Tech,.,Ph.D** Principal, Head of the Department of ECE, St.Johns College of Engineering & Technology, for his valuable advices and guidance.

We would also like to express our gratitude to the Faculty of Department of Electronics & Communication Engineering**, St.Johns College of Engineering & Technology, Yerrakota,** for their constant support.

We are also thankful to all our friends for their immense help in completion of this project.

Batch - 08

BAPURAM SHAIK SAHEB - 19G31A0404
DUDEKULA SHABBER - 19G31A0414
VEGULLA VAMSIKRISHNA - 19G31A0424
MALA NAVEEN KUMAR - 19G31A0440
GOLLA UDAY KUMAR - 19G31A0421

ABTRACT

In this Project, we are developing the Gesture Controlled Video Player using Raspberry Pi. Here, we are going to use the MediaPipe Python library to detect our hand gestures and control the Raspberry Pi Media player.

A total of six Hand Gestures are used, i.e open & close fist and up, down, left and right movement of the hand. Open and close fist gestures are used to play and pause the video. Up and Down Gestures are used to increase and decrease volume, and left and right gestures are used to fast forward and reverse the video.

CONTENTS

List of Figures

List of Tables

CHAPTER-1

INTRODUCTION

Chapter – 01

INTRODUCTION

1.1. Objective:

To Develop the Gesture Controlled Video Player using Raspberry Pi. In this Project, we have used MediaPipe Python library to detect our hand gestures and control the Raspberry Pi Media player.

1.2. Motivation:

Hand Gesture recognition technology is becoming increasingly popular due to the recent growth and popularity of Virtual and Augumented Reality Technology. Also, Nowadays most electronic devices focus on the hand gesture recognition algorithm and the corresponding user interface. This motivated us in developing this project.

1.3. Documentation output:

The Project report comprises with 11 chapters. Where in the first chapter we have discussed objective and motivation of our project. In the second chapter we explored the Literature review. In the third chapter we have presented Hardware Requirements. In the fourth chapter we discussed the Implementation & Design Process. In the fifth chapter we explained Software required for the project. In the sixth chapter we briefed the execution and Results of the Project. In the seventh chapter we discussed Advantages. In the Eighth chapter we have presented Disadvantages. In the nineth chapter we mentioned few Applications of the project. In the Tenth chapter we have given the Conclusion to the Project. In the Eleventh chapter we have kept the Future enhancements of the project. Finally References are acknowledged.

CHAPTER-2

LITERATURE REVIEW

Chapter – 02

LITERATURE REVIEW

Gesture-controlled video players using Raspberry Pi have become an increasingly popular research topic due to the growing demand for innovative and user-friendly ways of interacting with multimedia content. A literature review of recent studies on this subject reveals several key findings and trends.

One of the earliest works on gesture-controlled video players using Raspberry Pi was reported in a study by Kumar and Bhatt (2016). They developed a video player that could be controlled by hand gestures using a Kinect sensor and a Raspberry Pi. The system was capable of recognizing several hand gestures such as swipe, zoom, and rotate. They also tested the system with various video files and found that it was effective in controlling the playback of video content.

Another study by Deka et al. (2017) presented a similar approach to controlling a video player using hand gestures. They used an ultrasonic sensor to detect hand movements and a Raspberry Pi to process the signals and control the playback of videos. The system was found to be accurate in detecting hand gestures and responsive in controlling the video player.

A more recent study by Wang et al. (2020) proposed a gesture-controlled video player that was specifically designed for use in classrooms. They used a Leap Motion controller to detect hand gestures and a Raspberry Pi to process the signals and control the playback of educational videos. The system was found to be effective in improving student engagement and retention of information.

Another interesting approach was presented in a study by Zhang et al. (2021), who used a combination of Raspberry Pi and Myo armband to develop a gesture-controlled video player. The Myo armband was used to detect hand and arm movements, and the Raspberry Pi was used to process the signals and control the video player. The system was found to be accurate in detecting hand gestures and responsive in controlling the video player.

Overall, these studies demonstrate the potential of using Raspberry Pi as a platform for developing gesture-controlled video players. They also highlight the importance of selecting appropriate sensors and techniques for accurately detecting hand gestures and controlling the video player. Future research in this area could focus on improving the accuracy and responsiveness of the system and exploring its applications in different domains such as gaming, healthcare, and education.

CHAPTER-3

HARDWARE REQUIREMENTS

Chapter – 03

HARDWARE REQUIREMENTS

3.1. Raspberry Pi:

Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

What's more, the Raspberry Pi has the ability to interact with the outside world, and has been used in a wide array of digital maker projects, from music machines and parent detectors to weather stations and tweeting birdhouses with infra-red cameras. We want to see the Raspberry Pi being used by kids all over the world to learn to program and understand how computers work.

Till date, researchers, hobbyists and other embedded systems enthusiast across the planet are making amazing projects using Pi which looks unbelievable and have out-of-the-box implementation. Raspberry Pi since its launch is regularly under constant development cum improvement both in terms of hardware and software which in-turn making Pi a "Full Fledged Computer" with possibility to be considered for almost all computing intensive tasks.

The Raspberry Pi board contains Broadcom based ARM Processor, Graphics Chip, RAM, GPIO and other connectors for external devices. The operating procedure of Raspberry Pi is very similar as compared to PC and requires additional hardware like Keyboard, Mouse, Display Unit, Power Supply, SD Card with OS Installed (Acting like Hard Disk) for operation. Raspberry Pi also facilitates USB ports, Ethernet for Internet/Network-Peer to Peer Connectivity. Like any other computer, where Operating system acts as backbone for operation. Raspberry Pi, facilitates open source operating system's based on Linux. Till date more than 30 operating systems based on different flavors of Linux is being launched. Raspberry Pi foundation has also launched various accessories like Camera, Gertboard and Compute Model Kit for deploying add-on hardware modules.

3.2. History of Raspberry Pi:

Raspberry Pi- A name which is now not a new technical buzzword for researchers in the area of Computer Science, Electronics & Embedded Systems engineering worldwide. Till date, 5 million pieces of various generations of Raspberry Pi ranging from Model A to Model B and even the latest Raspberry Pi Zero (Going Out of Stock in First week of Launch) have been sold and thousands of projects ranging from Robotics, Super Computer development, Gaming Consoles, Portable Tablets, Server based Implementations on lines of Linux and Cloud Computing, Drones and even Astro Pi has been proposed and implemented all over the world. The foundation seed for development for Raspberry Pi journey started in 2006, when researchers named: Eben Upton, Rob Mullins, Jack Lang and Alan Mycroft at University of Cambridge's Computer Laboratory became stunned to see the decline in the skill level of A Level students and students applying for computer science. The main idea behind their stepping stone development was to give kids tiny and affordable computer in the period where computers were expensive and programming practice among kids was not supported by parents of children in U.K. The team lead by Eben Upton developed several versions of working prototypes from year 2006 to 2008 and the final released version was named as "RASPBERRY Pi".

Fig: 3.1: Raspberry pi icon

In 2008, the processors used in mobile devices were becoming cheaper and powerful and have full potential to support and run multimedia and all sorts of programming. The project seemed to be of high potential to the team, so the members Eben Upton, Rob, Jack and Alan got the project joint-ventured by Pete Lomas, MD of Norcott Technologies (Hardware Design and Manufacture Company) and David Braben (Co-Author of Seminal BBC Micro Game Elite) to make Raspberry Pi Foundation. The Raspberry Pi was officially created in year 2012 (February 2012) and the developing stone was laid by Raspberry Pi Foundation and within 3 years, the model B entered mass level production with Element 14 and RS Electronics and within 2 years of official launch of Pi, 2 million pieces are sold till date.

Raspberry Pi, a complete PC in itself started a new movement of Portable and Low powerful computers. And taking Raspberry Pi into consideration various replica boards like Intel Galileo, Dwengo, Beaglebone, ORCID etc. have come up with their boards providing somewhat same or little bit more configuration as compared to Pi.

3.3. Generations and Models:

In 2012, the company launched the Raspberry Pi and the current generations of regular Raspberry Pi boards are Zero, 1, 2, 3, and 4. Generation 1 Raspberry Pi had the following four options:

- Model A
- Model A +
- Model B
- Model B +

Among these models, the Raspberry Pi B models are the original credit-card sized format. On the other hand, the Raspberry Pi A models have a smaller and more compact footprint and hence, these models have the reduced connectivity options. Raspberry Pi Zero models, which come with or without GPIO (general-purpose input output) headers installed, are the most compact of all the Raspberry Pi boards types.

Raspberry Pi Model	Released	USB Ports	CPU	RAM	Bluetooth
Raspberry Pi 1 B	Feb 2012	2x USB 2.0	700MHz	512MB	No
Raspberry Pi 1 B+	July 2014	4x USB 2.0	700MHz	512MB	No
Raspberry Pi 1 A+	Nov 2014	1x USB	700MHz	512MB	No
Raspberry Pi 2	Feb 2015	4x USB	900MHz	1GB	No
Raspberry Pi Zero	Nov 2015	1x Micro USB	1GHz	512MB	No

Raspberry Pi 3	Feb 2016	4x USB	1.2GHz	1GB	4.1 LE
Raspberry Pi Zero W	Feb 2017	1x Micro USB	1GHz	512MB	4.1
Raspberry Pi 3 B+	Mar 2018	4x USB 2.0	1.4GHz	1GB	4.2, BLE
Raspberry Pi 3 A+	Nov 2018	1x USB 2.0	1.4GHz	512MB	4.2, BLE
Raspberry Pi 4	June 2019	2X USB 3.0, 2X USB 2.0	1.5GHz	1GB, 2GB, 4GB, or 8GB	5.0, BLE
Raspberry Pi 400	November 2020	2X USB 3.0, 2X USB 2.0	1.8GHz	4GB	5.0, BLE
Raspberry Pi Pico	January 2021	USB C	133MHz	264KB	No

Table: 3.1: Generations and Models of Raspberry Pi

Raspberry Pi 3 Model B+ (2018)

The Raspberry Pi 3 Model B+ was released in 2018 and introduced numerous improvements across several areas.

While this version of the Raspberry Pi was only marketed as a “plus” model, it brought a wealth of changes that made it well worth upgrading from the original Pi 3 Model B.

The first changes were made to the Pi’s processor, boosting the clocks even higher than the standard Pi 3 pushed them. This would also be the first Raspberry Pi to see a boost made to the GPU clock speeds.

With this release, the processor’s clock speed was increased from 1.2GHz to 1.4 GHz. Additionally, the GPU saw a clock boost from 250MHz to 400MHz.

The Raspberry Pi Foundation made these changes possible thanks to improved thermals on the device. Starting with the Pi 3 Model B+, the processor now has a heat spreader over it. This heat spreader allows heat to be more efficiently drawn away from the processor.

On top of this, the team improved the network capabilities for both wired and wireless connections.

This release saw the Pi Pi finally get a full gigabit Ethernet port. Previous versions of the Raspberry Pi were limited to 100 Mbit/s. Now you can theoretically handle up to 1000 Mbit/s using a Pi 3 Model B+.

In addition to these changes, the Pi foundation also made improvements to the Wi-Fi capability. The Wi-Fi supports b/g/n/ac and is dual-band capable. The new chip means you can now connect to both 2.4 GHz and 5 GHz networks and get improved speeds.

The Raspberry Pi 3 Model B+ is the latest product in the Raspberry Pi 3 range, boasting a 64-bit quad core processor running at 1.4GHz, dual-band 2.4GHz and 5GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and PoE capability via a separate PoE HAT.

The dual-band wireless LAN comes with modular compliance certification, allowing the board to be designed into end products with significantly reduced wireless LAN compliance testing, improving both cost and time to market.

The Raspberry Pi 3 Model B+ maintains the same mechanical footprint as both the Raspberry Pi 2 Model B and the Raspberry Pi 3 Model B.

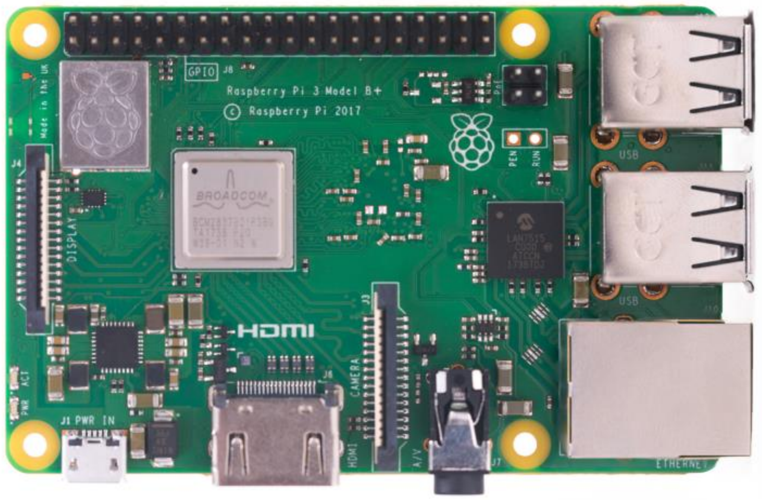

Fig: 3.2: Raspberry pi 3 model B+ Model

Specifications :

Processor: Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz

Memory: 1GB LPDDR2 SDRAM

Connectivity:

- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2,BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput 3000Mbps)
- 4 × USB 2.0 ports

Access: Extended 40-pin GPIO header

Video & Sound:

- ✓ 1 × full size HDMI
- ✓ MIPI DSI display port
- ✓ MIPI CSI camera port
- ✓ 4 pole stereo output and composite video port

Multimedia: H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics

SD card support: Micro SD format for loading operating system and data storage

Input power:

- ✓ 5V/2.5A DC via micro USB connector
- ✓ 5V DC via GPIO header
- ✓ Power over Ethernet (PoE)–enabled (requires separate PoE HAT)

Environment: Operating temperature, 0–50°C

Production lifetime: The Raspberry Pi 3 Model B+ will remain in production until at least January 2023.

3.4. Raspberry pi Display:

Figure: 3.3: Raspberry pi Display

Display is a device used to present visual information. The main aim of any display technology is to simplify the information sharing. Today, there are different types of displays used for different applications. These displays can be categorized as Video Displays, Non- Video Displays and 3D displays.

The Raspberry Pi Touch Display is an LCD display which connects to the Raspberry Pi through the DSI connector. In some situations, it allows for the use of both the HDMI and LCD displays at the same time (this requires software support).

The following image shows how to attach the Raspberry Pi to the back of the Touch Display (if required), and how to connect both the data (ribbon cable) and power (red/black wires) from the Raspberry Pi to the display. If you are not attaching the Raspberry Pi to the back of the display, take extra care when attaching the ribbon cable to ensure it is the correct way round. The black and red power wires should be attached to the GND and 5v pins respectively.

A display is a computer output surface and projecting mechanism that shows text and often graphic images to the computer user, using a cathode ray tube (CRT), liquid crystal display (LCD), light-emitting diode, gas plasma, or other image projection technology. The display is usually considered to include the screen or projection surface and the device that produces the information on the screen. In some computers, the display is packaged in a separate unit called a monitor. In other computers, the display is integrated into a unit with the processor and other parts of the computer. (Some sources make the distinction that the monitor includes other signal-handling devices that feed and control the display or projection device. However, this distinction disappears when all these parts become integrated into a total unit, as in the case of notebook computers.) Displays (and monitors) are also sometimes called video display terminals (VDTs).

The terms display and monitor are often used interchangeably most computer displays use Analog signals as input to the display image creation mechanism. This requirement and the need to continually refresh the display image mean that the computer also needs a display or video adapter. The video adapter takes the digital data sent by application programs, stores it in video random access memory (video RAM), and converts it to Analog data for the display scanning mechanism using an Digital-to-Analog converter (DAC).

Displays can be characterized according to:

- ✓ Colour capability
- ✓ Sharpness and viewability
- ✓ The size of the screen
- ✓ The projection technology

❖ Colour Capability

Today, most desktop displays provide colour. Notebook and smaller computers sometimes have a less expensive *monochrome* display. Displays can usually operate in one of several display modes that determine how many bits are used to describe colour and how many colours can be displayed. A display that can operate in Super VGA mode can display up to 16,777,216 colours because it can process a 24-bit long description of a pixel. The number of bits used to describe a pixel is known as its bit-depth. The 24-bit bit-depth is also known as true colour. It allows eight bits for each of the three additive primary colours - red, green, and blue. Although human beings can't really distinguish that many colours, the 24-bit system is convenient for graphic designers since it allocates one byte for each colour. The Visual Graphics Array (VGA) mode is the lowest common denominator of display modes. Depending on the resolution setting, it can provide up to 256 colours.

❖ Sharpness and Viewability

The absolute physical limitation on the potential image sharpness of a screen image is the dot pitch, which is the size of an individual beam that gets through to light up a point of phosphor on the screen. (The shape of this beam can be round or a vertical, slot-shaped rectangle depending on the display technology.) Displays typically come with a dot pitch of .28 mm (milli meters) or smaller. The smaller the dot pitch in milli meters, the greater the potential image sharpness.

The actual sharpness of any particular overall display image is measured in dots-per-inch (dot per inch). The dots-per-inch is determined by a combination of the screen resolution (how many pixels are projected on the screen horizontally and vertically) and the physical screen size. The same resolution spread out over a larger screen offers reduced sharpness. On the other hand, a high-resolution setting on a smaller surface will Product a sharper image, but text readability will become more difficult.

Viewability includes the ability to see the screen image well from different angles. Displays with cathode ray tubes (CRT) generally provide good viewability from angles other than straight on. Flat-panel displays, including those using light-emitting diode and liquid crystal display technology, are often harder to see at angles other than straight on.

❖ The Size of the Screen

On desktop computers, the display screen width relative to height, known as the aspect ratio, is generally standardized at 4 to 3 (usually indicated as "4:3"). Screen sizes are measured in either milli meters or inches diagonally from one corner to the opposite corner. Popular desktop screen sizes are 12-, 13-, 15-, and 17-inch. Notebook screen sizes are somewhat smaller.

❖ The Projection Technology

Most displays in current use employ cathode ray tube (CRT) technology similar to that used in most television sets. The CRT technology requires a certain distance from the beam projection device to the screen in order to function. Using other technologies, displays can be much thinner and are known as flat-panel displays. Flat panel display technologies include light-emitting diode (LED), liquid crystal display (LCD), and gas plasma. LED and gas plasma work by lighting up display screen positions based on the voltages at different grid intersections. LCDs work by blocking light rather than creating it. LCDs require far less energy than LED and gas plasma technologies and are currently the primary technology for notebook and other mobile computers.

Displays generally handle data input as character maps or bitmap s. In character-mapping mode, a display has a pre-allocated amount of pixel space for each character. In bitmap mode, it receives an exact representation of the screen image that is to be projected in the form of a sequence of bits that describe the colour values for specific X and Y coordinates starting from a given location on the screen. Displays that handle bitmaps are also known as all-points addressable displays.

3.5. VGA to HDMI Converter:

Fig 3.4: VGA to HDMI Converter

VGA to HDMI is a conversion from analog signal to digital signal, which requires that the signal source has a VGA interface, i.e. the input side is a device with a VGA interface, and the display device has an HDMI interface, i.e. the output side.

This VGA to HDMI Converter is used to connect monitor with Raspberry Pi board.

HDMI to HDMI Connector:

Fig 3.5: HDMI to HDMI Connector

HDMI is the modern standard interface for audio/video (A/V) connectivity, which stands for High-Definition Multimedia Interface. Generally, it is a connector that is used for connecting your audio-video devices together.

This cable is used to make connection between Raspberry Pi board and Display.

3.6. Ethernet Cable:

Fig 3.6: Ethernet Cable

These physical cables are limited by length and durability. If a network cable is too long or of poor quality, it won't carry a good network signal. These limits are one reason there are different types of Ethernet cables that are optimized to perform certain tasks in specific situations. ethernet Cable is used to connect Raspberry Pi board with Laptop.

3.7 Camera module:

A camera module is a component of a digital camera or other imaging devices that captures images or videos. It consists of a lens, an image sensor, and various electronic components that process and store the image data. The lens is responsible for focusing light onto the image sensor, which then converts the light into an electrical signal that can be processed by the camera's electronics.

Fig: 3.7: Camera module

There are many types of camera modules, ranging from those found in smartphones and tablets to those used in professional cameras and surveillance systems. Some camera modules are designed to be easily interchangeable, allowing photographers to switch lenses depending on their needs. Others are integrated into devices and cannot be easily removed or replaced.

The quality of a camera module depends on several factors, including the size and quality of the image sensor, the quality of the lens, and the processing capabilities of the camera's electronics. High-end camera modules can produce extremely high-quality images and videos, while lower-end modules may produce lower-quality images with less detail and poorer color accuracy.

CHPTER-4

IMPLEMENTATION & DESIGN

Chapter – 04

IMPLEMENTATION & DESIGN

4.1. Raspberry Pi Setup :

To set up the Raspberry Pi for monitoring, you'll need to follow these general steps:

- Get a Raspberry Pi board, power supply, and a microSD card. You'll also need a keyboard, mouse, and monitor to set it up initially.
- Download the Raspberry Pi OS from the official website, and flash the image onto the microSD card using a tool like BalenaEtcher.
- Insert the microSD card into the Raspberry Pi and connect the power supply, keyboard, mouse, and monitor.
- Boot up the Raspberry Pi and follow the setup process to configure the Wi-Fi or Ethernet network, locale, and other preferences.

With these steps, you should have a Raspberry Pi set up for monitoring various metrics and services. You can customize the setup to suit your needs, such as adding alerting rules or integrating other monitoring tools.

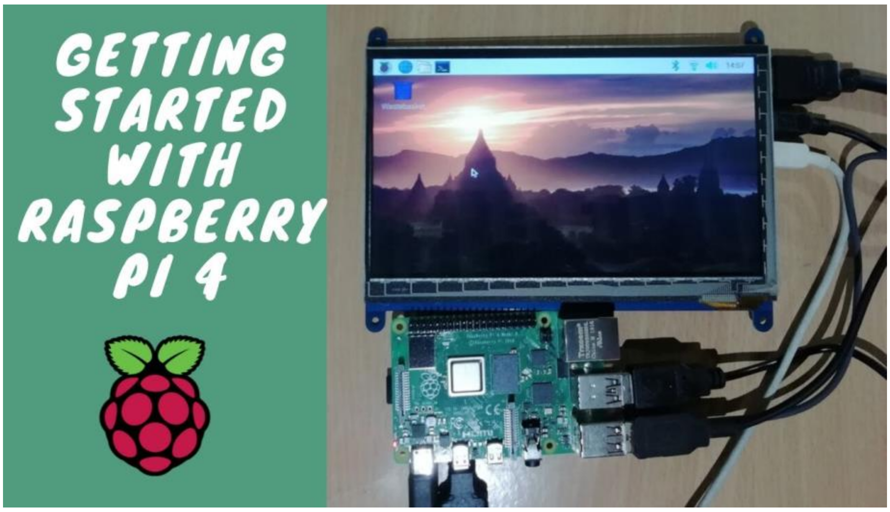

Fig: 4.1: Set Up Your Raspberry Pi

4.2. Connect the camera module:

Identify the camera module and the device you want to connect it to. Make sure that the camera module is compatible with the device and that you have the necessary cables and adapters.

Connect the camera module to the device using the appropriate cable or adapter. This may involve plugging the camera module into a USB port or connecting it to a dedicated camera port.

Turn on the device and the camera module. The device should recognize the camera module and install any necessary drivers or software.

Open the camera app or software on the device and verify that the camera module is working properly. You should be able to see a live video feed from the camera module.

Fig: 4.2: Connect the camera module

4.3. Code for gesture detection:

Write a Python script to detect hand gestures using the camera module and the OpenCV library.

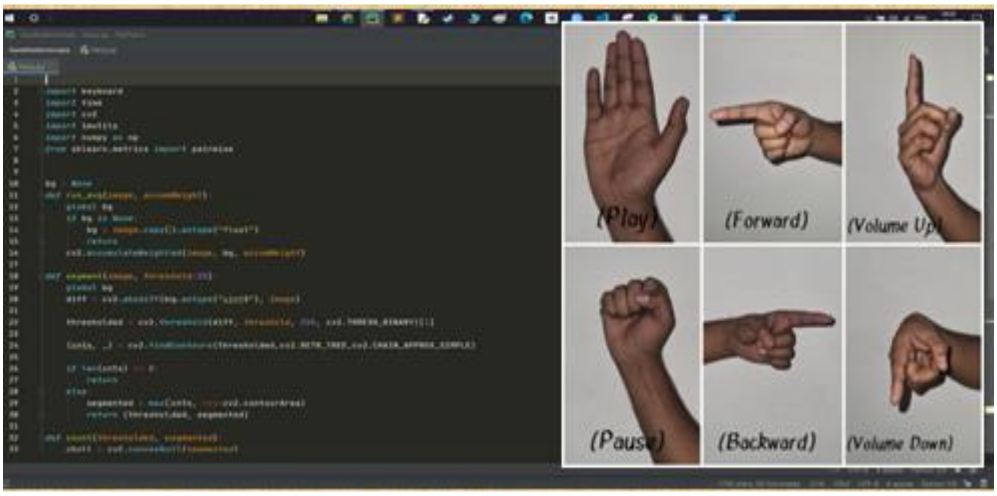

Fig: 4.3: Code for gesture control

4.4. Test and refine:

Test your gesture control system to ensure that it is working correctly. You may need to refine the code or make some adjustments to the hardware configuration to improve the accuracy of the gesture detection.

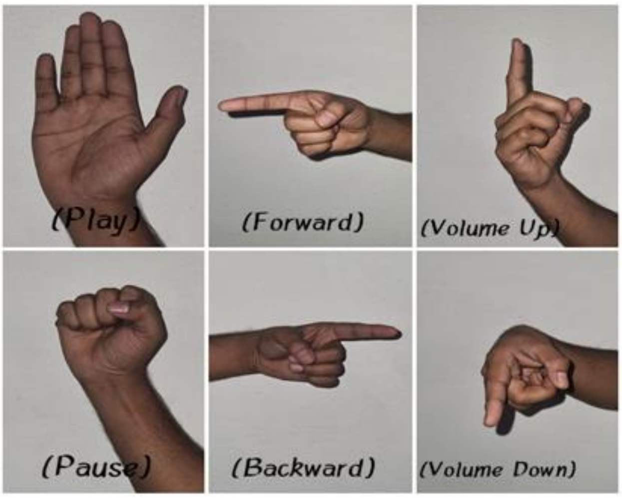

Fig: 4.4: Test and refine

CHAPTER-5

SOFTWARE REQUIRED

Chapter – 05

SOFTWARE REQUIREMENTS

RASPBERRY PI OPERATING SYSTEM

The operating systems that can be used for raspberry pi are:

1.Raspbian

2.Noobs

Raspberry Pi needs an operating system to work. Raspberry Pi OS (previously called Raspbian) is our official supported operating system.

5.1. RASPBIAN OS:

Raspberry Pi OS (formerly **Raspbian**) is a Debian-based operating system for Raspberry Pi. Since 2013, it has been officially provided by the Raspberry Pi Foundation as the primary operating system for the Raspberry Pi family of compact single-board computers.

Raspberry Pi OS was first developed by Mike Thompson and Peter Green as Raspbian, an independent and unofficial port of Debian to the Raspberry Pi. The first build was released on July 15, 2012. As the Raspberry Pi had no officially provided operating system at the time, the Raspberry Pi Foundation decided to build off of the work done by the Raspbian project and began producing and releasing their own version of the software. The Foundation's first release of Raspbian, which now referred both to the community project as well as the official operating system, was announced on September 10th, 2013.

On May 28th, 2020, the Raspberry Pi Foundation announced they were releasing a beta 64-bit version of their official operating system. However, the 64-bit version was not based on Raspbian, instead taking its userland from Debian directly. Since the Foundation did not want to use the name Raspbian to refer to software that was not based on the Raspbian project, the name of the officially provided operating system was changed to Raspberry Pi OS. This change was carried over to the 32-bit version as well, though it continued to be based on Raspbian. The 64-bit version of Raspberry Pi OS was officially released on February 2nd, 2022.

Raspberry Pi OS is highly optimized for the Raspberry Pi line of compact single-board computers with ARM CPUs. It runs on every Raspberry Pi except the Pico microcontroller. Raspberry Pi OS uses a modified LXDE as its desktop environment with the Openbox stacking window manager, along with a unique theme. The default distribution is shipped with a copy of the algebra program Wolfram Mathematica, VLC, and a lightweight version of the Chromium web browser.

For using an OS, we need to create a Secure Digital (SD) or MicroSD card with an OS on it. The prerequisite for setting up the SD or MicroSD is a computer having an internet connection and the ability to write to SD or MicroSD cards.

5.1.1. Features:

User interface:

Raspberry Pi OS has a desktop environment, PIXEL, based on LXDE, which looks similar to many common desktops, such as macOS and Microsoft Windows. The desktop has a background image. A menu bar is positioned at the top and contains an application menu and shortcuts to a web browser (Chromium), file manager, and terminal. The other end of the menu bar shows a Bluetooth menu, Wi-Fi menu, volume control, and clock. The desktop can also be changed from its default appearance, such as repositioning the menu bar.

Package management:

Packages can be installed via APT, the *Recommended Software* app, and by using the *Add/Remove Software* tool, a GUI wrapper for APT.

Components:

PCManFM is a file browser allowing quick access to all areas of the computer, and was redesigned in the first Raspberry Pi OS Buster release (2019-06-20).

Raspberry Pi OS originally used Epiphany as the web browser, but switched to Chromium with the launch of its redesigned desktop.

Raspberry Pi OS comes with many beginner IDEs, such as Thonny Python IDE, Mu Editor, and Greenfoot. It also ships with educational software like Scratch and Bookshelf.

Versions:

Raspberry Pi OS has three installation versions:

- Raspberry Pi OS Lite (32-bit & 64-bit)
- Raspberry Pi OS with desktop (32-bit & 64-bit)
- Raspberry Pi OS with desktop and recommended software (32-bit)

Raspberry Pi OS also has two legacy versions:

- Raspberry Pi OS Lite (Legacy) (32-bit)
- Raspberry Pi OS (Legacy) with desktop (32-bit)

Raspberry Pi OS Lite is the smallest version and doesn't include a desktop environment. Raspberry Pi OS with desktop includes the Pixel desktop environment. Raspberry Pi OS with desktop and recommended software additionally comes pre-installed with additional productivity software, such as Libre Office.

On December 2nd, 2021, the Raspberry Pi Foundation released Raspberry Pi OS (Legacy), a branch of the operating system that continued to receive security and hardware compatibility updates but was based in the older Buster version of Debian.

All versions are distributed as .img disk image files. These files can then be flashed on to microSD cards where Raspberry Pi OS runs. In March 2020, the Raspberry Pi Foundation also published the Raspberry Pi Imager, a custom disk flasher that allows for the installation of Raspberry Pi OS as well as other operating systems designed for the Raspberry Pi, including RetroPie, Kodi OS, and others. The Raspberry Pi documentation recommends at least a 4GB microSD card for Raspberry Pi OS Lite, and at least a 8GB microSD card for all other versions.

5.2. NOOBS Software:

NOOBS means **new-out-of-box software** and it is the easiest way to get started with the Raspberry Pi. It is easy to copy NOOBS to your SD or MicroSD card. Once copied, it provides us with a simple menu for installing various operating systems.

There is an option to buy a card with NOOBS already installed on it, but it is always useful to know how to create your own NOOBS cards.

5.3. Installation steps for Raspberry Pi:

Step1: Go to Raspberry Pi.com/.org. Here click on downloads.

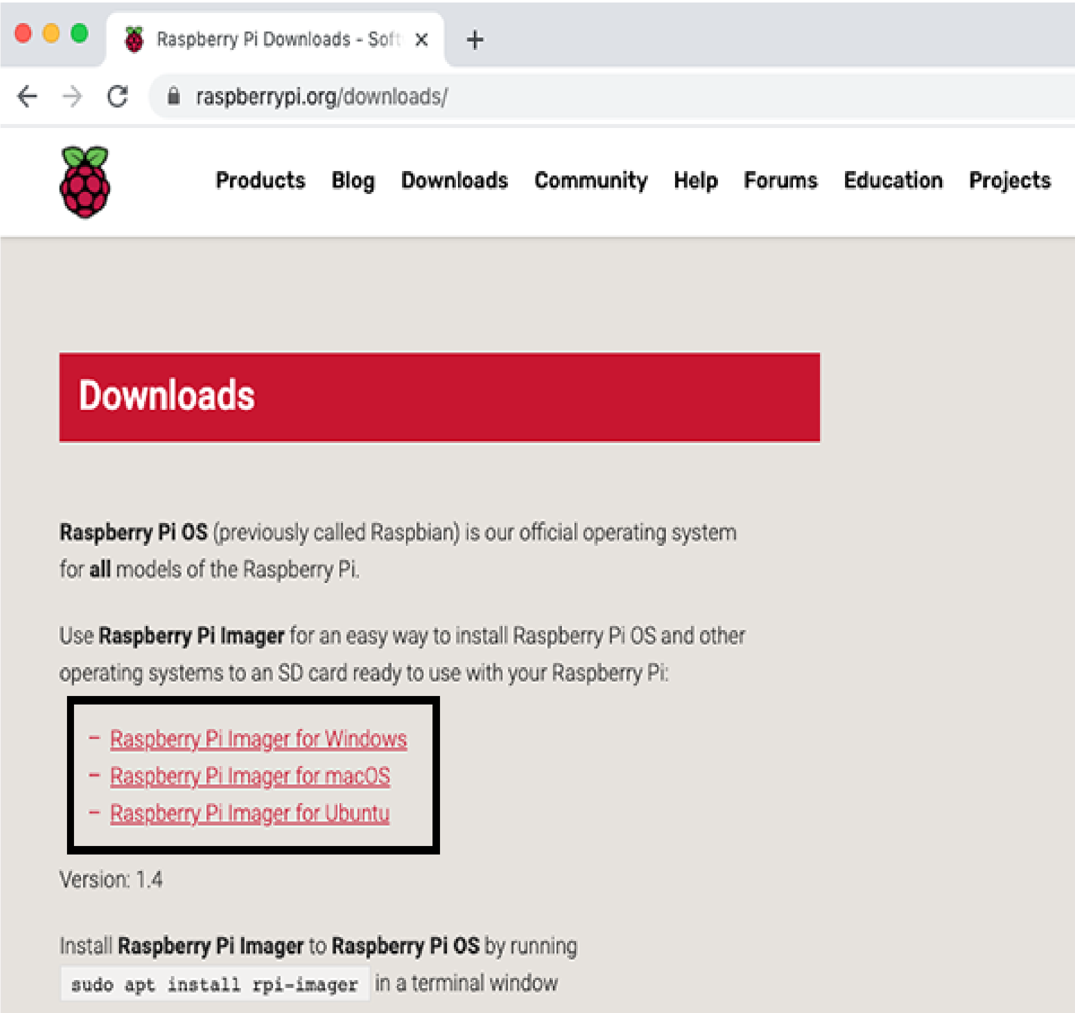

Fig: 5.1: Installation for raspberry pi

Step 2: Here click on download for windows .Then the Raspberry Pi Imager is downloaded .If you are using Mac ,click on download for Mac or if you are using Ubuntu ,click on download for Ubuntu.

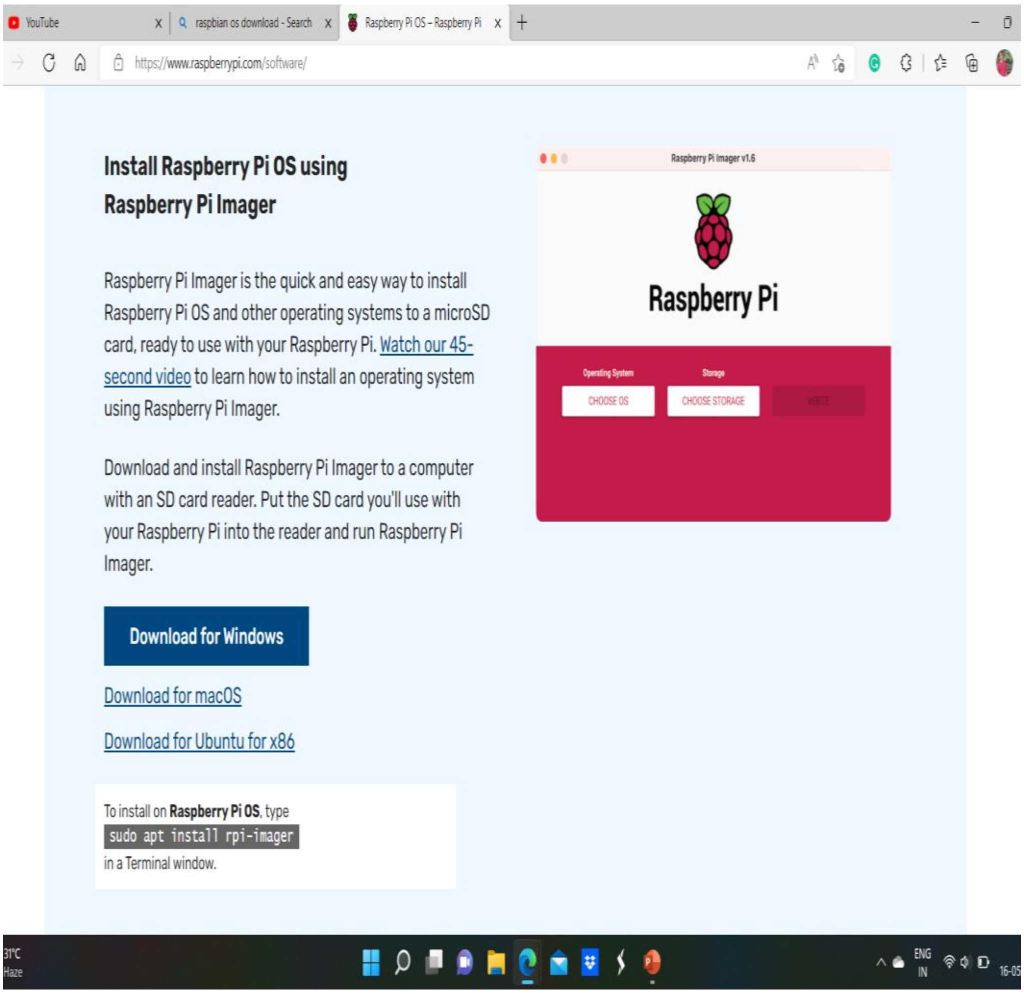

Fig: 5.2: Download for windows

Step 3: Insert SD card to PC though card reader . Download SD card formatter from SD card.com/.org and format the SD card using SD card formatter

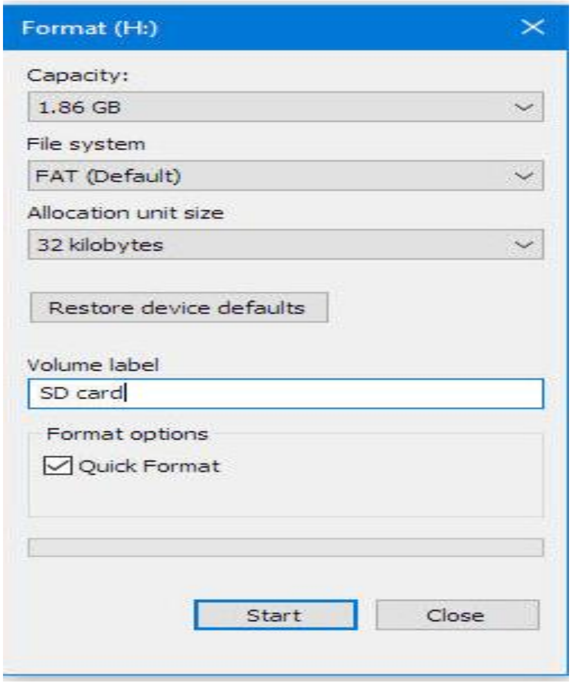

Fig: 5.3: Storage for capable

Step 4 :After formatting, open Raspberry Pi Imager. Click on choose OS as Raspbian and click on choose SD card storage location and then choose write. Now writing up of Raspbian OS takes place on SD card

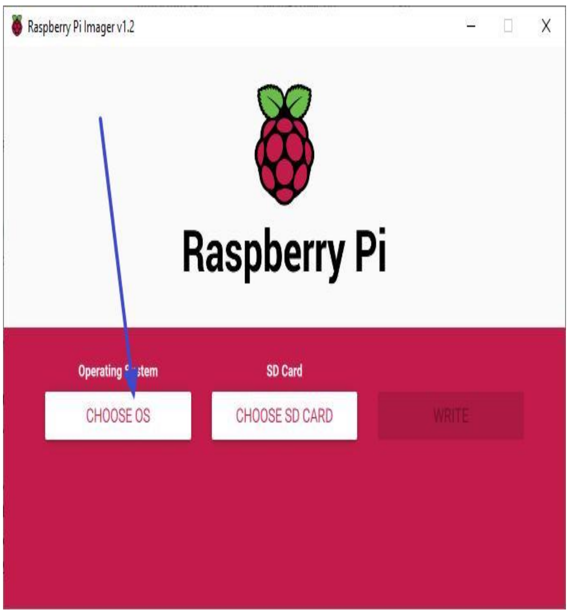

Fig: 5.4: Storage location

Step 5: After successful write up of Raspbian OS onto the SD card, remove the SD card from the PC and insert the SD card into the Raspberry Pi board .

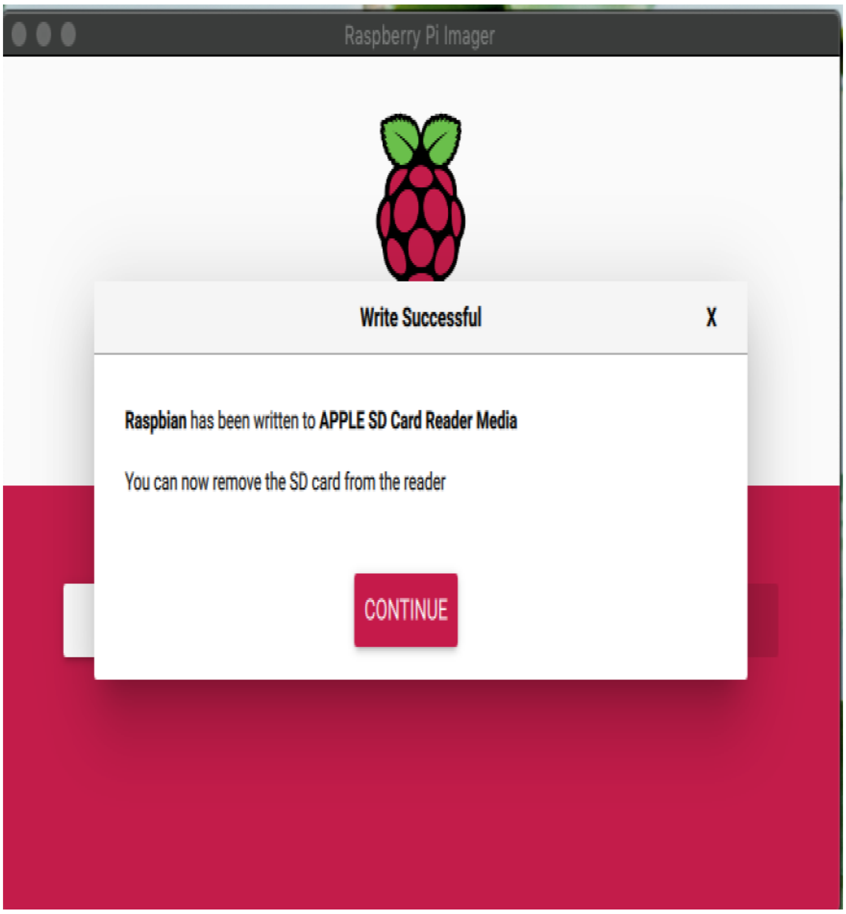

Fig: 5.5: Remove the SD card

Step 6: Make all the necessary connections like 2Amps power supply to Raspberry Pi board , connect the desktop to Raspberry Pi board with the help of VGA to HDMI connector such that VGA port is connect to desktop and HDMI port is connected to Raspberry Pi board. Switch on the power supply ,now booting will start .After booting Raspbian OS is successfully installed on your desktop.

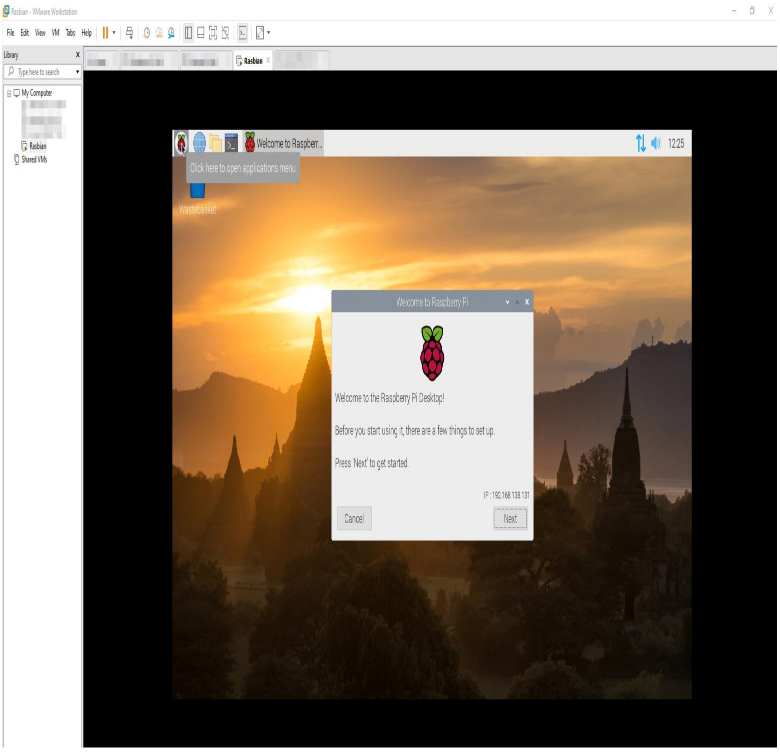

Fig: 5.6: successfully installed

CHAPTER-6

RESULTS

Chapter – 06

RESULT

The Complete Setup of our project is,

Fig: 6.1: Setup of our project

6.1 Testing Gesture Controlled Media Controller :

Initially, check whether the Pi camera is working or not.

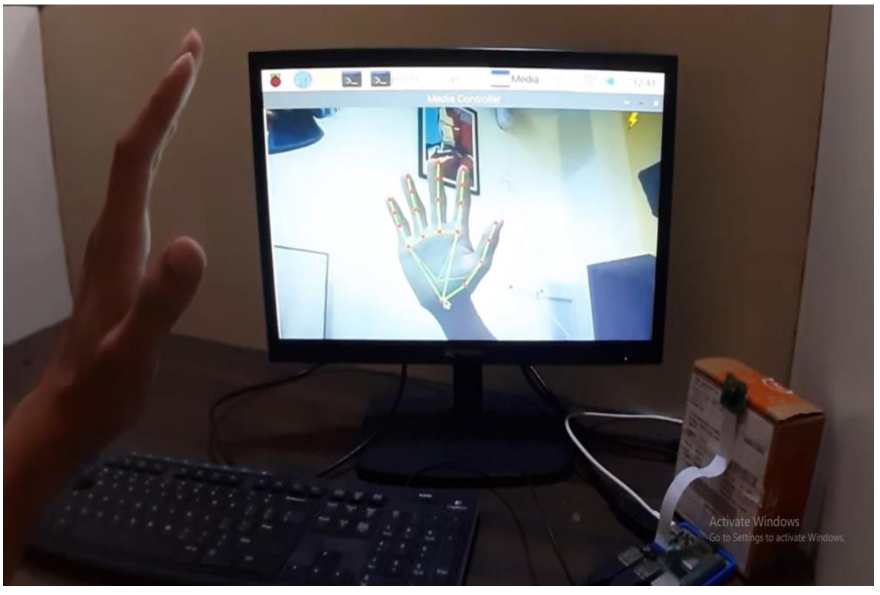

Fig: 6.2: Testing gesture controlled media controller

After reviewing the camera, launch the Python script, and you will find a window popping up with a video feed in it.

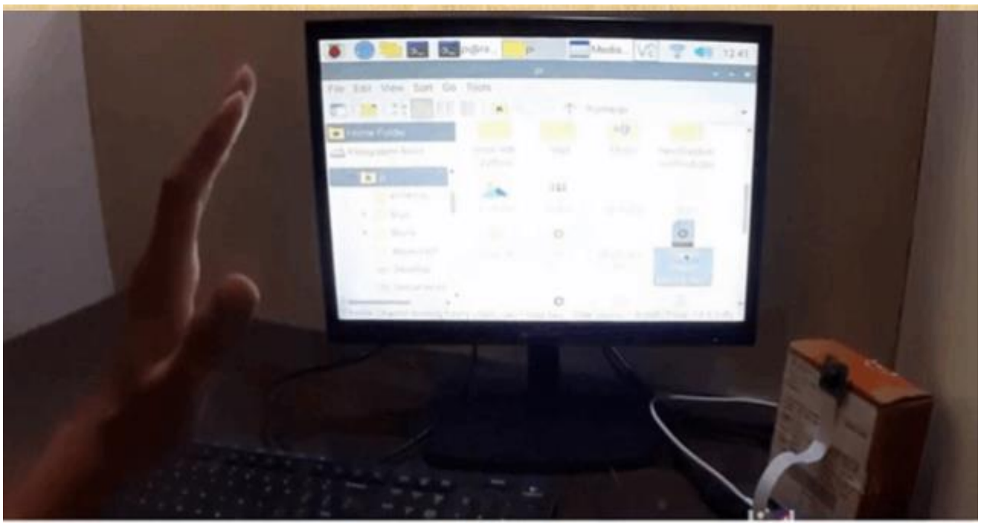

Fig: 6.3: Controlling the video player

Now, you can control the video player by your hand gestures.

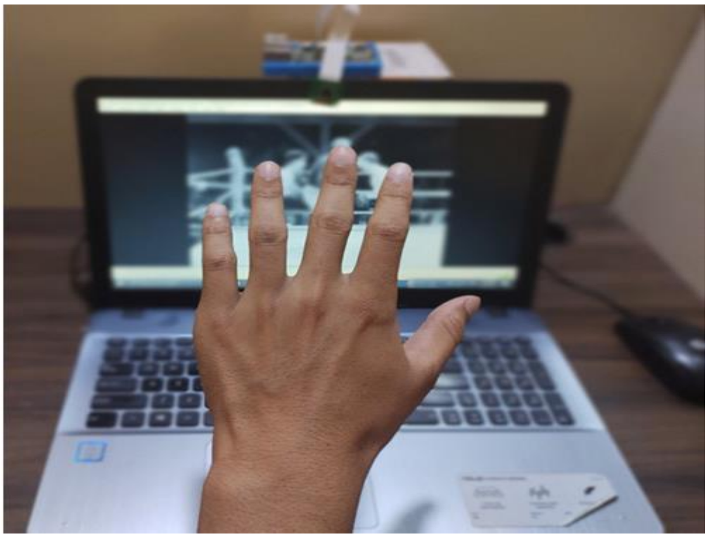

Fig: 6.4: Final output of our project

You can change the video feed to any video of your choice and have fun controlling it with hand gestures.

CHAPTER-7

ADVANTAGES

Chapter – 07

ADVANTAGES

There are several advantages of using a gesture-controlled video player using Raspberry Pi, including:

Hands-free operation:

With gesture control, the user can navigate the video player without having to physically touch any buttons or controls.

Improved accessibility:

Gesture control provides an alternative way of interacting with technology for people with physical disabilities or impairments that may make traditional controls difficult to use.

Greater precision:

Gesture control allows for more precise control than traditional controls. For example, the user can fast-forward or rewind the video with greater accuracy by simply gesturing in the appropriate direction.

Customizability:

It allows users to modify and customize their gesture-controlled video player to suit their specific needs and preferences.

Intuitive user interface:

Gestures are a natural way to interact with technology, and users can easily learn and remember simple gestures to control the video player. This makes the interface more intuitive and user-friendly, reducing the learning curve for new users.

CHAPTER-8

DISADVANTAGES

Chapter – 08

DISADVANTAGES

There are a few disadvantages of a gesture-controlled video player using Raspberry Pi:

Limited Range of Motion: The gesture recognition technology used in these systems may have a limited range of motion, which could make it difficult to perform certain actions or gestures.

Limited Flexibility: Gesture-controlled video players may have limited flexibility compared to traditional remote controls or other input devices, as they may only recognize a limited set of gestures.

Accuracy: Gesture recognition systems may not always be accurate, and there can be errors in interpreting the user's gestures. This could result in the video player not responding correctly to the user's commands.

Limited Gestures: The number of gestures that can be recognized by the system may be limited, which could restrict the functionality of the video player.

Lighting conditions: Gesture recognition can be affected by lighting conditions, and the system may not work well in low light or bright light conditions.

Interruptions: The gesture recognition system may be interrupted by external factors such as other people or objects crossing its line of sight, which could cause the system to misinterpret the user's gestures.

CHAPTER-9

APPLICATIONS

Chapter – 09

APPLICATIONS

A Gesture-controlled video player using Raspberry Pi has many potential applications.

Interactive exhibits:

A gesture-controlled video player could be used in museums or other public exhibits to allow visitors to control video playback without physically touching a screen or buttons. This could help reduce the spread of germs and create a more interactive and engaging experience for visitors.

Home entertainment:

A gesture-controlled video player could be a fun and unique way to control video playback at home. Imagine being able to pause, rewind, or fast-forward a movie just by waving your hand!

Accessibility:

For people with physical disabilities, a gesture-controlled video player could be an important tool for accessing video content. By using gestures instead of physical buttons or controls, people with limited mobility could have more independence and control over their media.

Education:

A gesture-controlled video player could be used in educational settings to help teach concepts like computer vision, programming, and machine learning. Students could build their own gesture recognition systems and use them to control video playback.

Overall, a gesture-controlled video player using Raspberry Pi has many potential applications, from entertainment to education to accessibility. With its low cost and flexible programming capabilities, the Raspberry Pi is an ideal platform for building such a system.

CHAPTER-10

CONCLUSION

Chapter – 10

CONCLUSION

Creating a Gesture-controlled video player using Raspberry Pi can be an engaging project for those interested in exploring the intersection of technology and user experience. With the right hardware components and software tools, it is possible to develop a system that responds to hand movements and allows users to control various aspects of video playback without the need for physical input devices such as keyboards or remote controls.

A Gesture controlled video player using Raspberry Pi provides a convenient and accessible way for users to interact with their video player. With hands-free operation, improved accessibility, greater precision, cost-effectiveness, customizability, and entertainment value, this technology offers several advantages. While there are many different approaches to implementing a gesture recognition system using Raspberry Pi, some key considerations include choosing an appropriate camera module, selecting an appropriate machine learning algorithm for hand tracking and gesture recognition, and designing an intuitive user interface that makes it easy for users to interact with the video player using gestures.

Overall, with the right combination of hardware, software, and design choices, a gesture-controlled video player using Raspberry Pi can offer an innovative and enjoyable user experience that is sure to impress and delight users of all ages and technical backgrounds.

CHAPTER-11

FUTURE ENHANCEMENTS

Chapter – 11

FUTURE ENHANCEMENTS

There are several possible future enhancements for a gesture-controlled video player using a Raspberry Pi, some of which are:

Multi-gesture support: The current implementation of a gesture-controlled video player typically supports only a few basic gestures. Future enhancements could allow for more advanced gestures such as zooming, rotating, or scrolling.

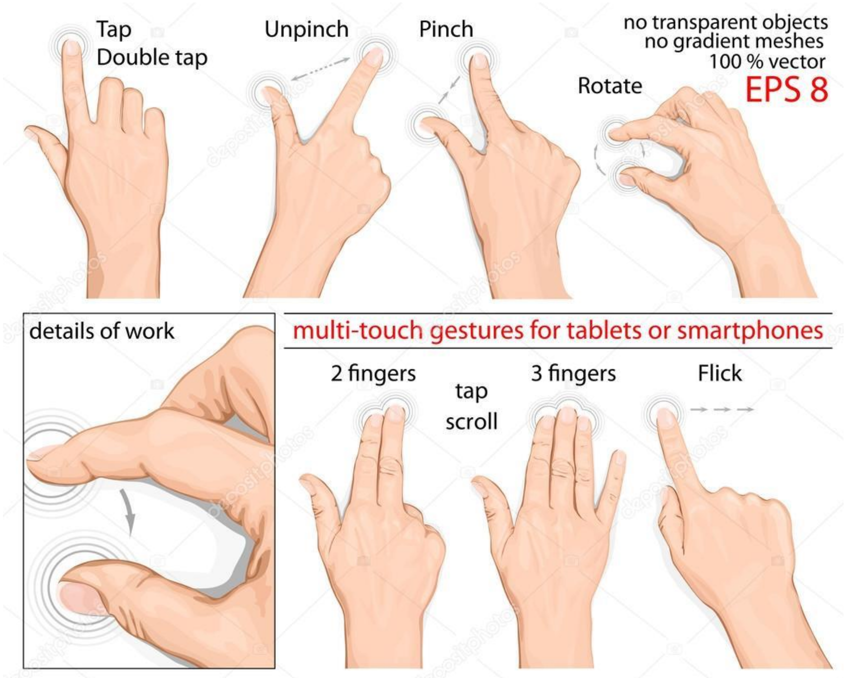

Fig: 11.1: Multi-gesture support

Voice control: Integrating voice control could enable users to interact with the video player using voice commands. This could be especially useful for people with disabilities or those who find gesture-based interaction challenging.

Fig: 11.2: Voice control

Smart gesture recognition: Incorporating machine learning and artificial intelligence techniques could enable the system to learn and recognize new gestures, making it more flexible and adaptable.

Fig: 11.3: Smart gesture recognition

Improved accuracy: Current gesture recognition systems can sometimes be inaccurate or unreliable, especially in low light conditions or with complex backgrounds. Improving the accuracy of the gesture recognition system could make it more robust and reliable.

Fig: 11.4: Improved accuracy

Mobile support: Expanding the gesture-controlled video player to mobile platforms could enable users to control their videos on their smartphones or tablets using hand gestures.

Fig: 11.5: Mobile support

Social interaction: Adding social media features could enable users to share videos and interact with others using hand gestures.

Fig: 11.6: Social interaction

These are just a few examples of possible future enhancements for a gesture-controlled video player using a Raspberry Pi. As technology evolves and new hardware and software capabilities become available, there will be even more possibilities for innovation and improvement.

REFERENCES

REFERENCES

[1] Saransh Sharma; Samyak Jain; Khushboo,”A Static Hand Ges- ture and Face Recognition System for Blind People”,2019 6th In-ternational Conference on Signal Processing and Integrated Net- works,IEEE,March.2019.

[2] Georgi Krastev; Ivan Ralev,”Gesture Based System For User Inter- face”,2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies,IEEE,Oct,2021.

[3] Y. Wu and T. S. Huang, ”Vision-Based Gesture Recognition: A Review”, Lecture Notes in Computer Science, Vol. 1739,pp. 103-115, 1999.

[4] V. Pavlovic, R. Sharma and T. S. Huang, “Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review,” IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. 19(7), pp. 677-695,1997.

[5] Granit Luzhnica, Elizabeth Lex, Viktoria Pammer. A Sliding Window Approach to Natural Hand Gesture Recognition using a Custom Data Glove. In: 3D User Interfaces (3DUI); 2016 IEEE Symposium on ; 2016 Mar 19 ; New York : IEEE; 2016 ; p.81-90.

[6] CH Hung, YW Bai, HY Wu Home outlet and LED array lamp controlled by a smartphone with a hand gesture recognition.Consumer Electronics (ICCE); 2016 IEEE International Conference on ; 2016 Jan 7, IEEE, New York (2016), pp. 5-6.

[7] Meng, H., Furao, S., Jinxi, Z.: Hidden Markov models based dynamic hand gesture recognition with incremental learning method. Paper Presented at the Neural Networks (IJCNN), 2014 International Joint Conference on (2014)

[8] Hasan, H., Abdul Kareem, S.: Fingerprint image enhancement and recognition algorithms: a survey. Neural Comput. Appl. 23, 1606–1608 (2013)

[9] Bretzner, L., Laptev, I., Lindeberg, T.: Hand gesture recognition using multi-scale colour features, hierarchical models and particle filtering. Paper Presented at the Automatic Face and Gesture Recognition, 2002. Proceedings of Fifth IEEE International Conference on (2002)

PYTHON SCRIPT

```
import cv2
import mediapipe as mp
import pyautogui
mp_drawing = mp.solutions.drawing_utils
mp_hands = mp.solutions.hands
##################################
tipIds = [4, 8, 12, 16, 20]
state = None
Gesture = None
wCam, hCam = 720, 640
############################
def fingerPosition(image, handNo=0):
    lmList = []
    if results.multi_hand_landmarks:
        myHand = results.multi_hand_landmarks[handNo]
        for id, lm in enumerate(myHand.landmark):
            # print(id,lm)
            h, w, c = image.shape
            cx, cy = int(lm.x * w), int(lm.y * h)
            lmList.append([id, cx, cy])
    return lmList
# For webcam input:
cap = cv2.VideoCapture(0)
cap.set(3, wCam)
cap.set(4, hCam)
with mp_hands.Hands(
    min_detection_confidence=0.8,
    min_tracking_confidence=0.5) as hands:
  while cap.isOpened():
    success, image = cap.read()
    if not success:
        print("Ignoring empty camera frame.")
      # If loading a video, use 'break' instead of 'continue'.
        continue
    # Flip the image horizontally for a later selfie-view display, and
convert
    # the BGR image to RGB.
```

```
    image = cv2.cvtColor(cv2.flip(image, 1), cv2.COLOR_BGR2RGB)
    image.flags.writeable = False
    results = hands.process(image)
    # Draw the hand annotations on the image.
    image.flags.writeable = True
    image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
    if results.multi_hand_landmarks:
      for hand_landmarks in results.multi_hand_landmarks:
        mp_drawing.draw_landmarks(
            image, hand_landmarks, mp_hands.HAND_CONNECTIONS)
    lmList = fingerPosition(image)
    #print(lmList)
    if len(lmList) != 0:
        fingers = []
        for id in range(1, 5):
            if lmList[tipIds[id]][2] < lmList[tipIds[id] - 2][2]:
                #state = "Play"
                fingers.append(1)
            if (lmList[tipIds[id]][2] > lmList[tipIds[id] - 2][2] ):
               # state = "Pause"
               # pyautogui.press('space')
               # print("Space")
                fingers.append(0)
        totalFingers = fingers.count(1)
        print(totalFingers)
        #print(lmList[9][2])

        if totalFingers == 4:
            state = "Play"
           # fingers.append(1)
        if totalFingers == 0 and state == "Play":
            state = "Pause"
            pyautogui.press('space')
            print("Space")
        if totalFingers == 1:
            if lmList[8][1]<300:
                print("left")
                pyautogui.press('left')
            if lmList[8][1]>400:
                print("Right")
                pyautogui.press('Right')
```

```
        if totalFingers == 2:
            if lmList[9][2] < 210:
                print("Up")
                pyautogui.press('Up')
            if lmList[9][2] > 230:
                print("Down")
                pyautogui.press('Down')
    #cv2.putText(image, str("Gesture"), (10,40),
cv2.FONT_HERSHEY_SIMPLEX,
     #              1, (255, 0, 0), 2)
    cv2.imshow("Media Controller", image)
    key = cv2.waitKey(1) & 0xFF
    # if the `q` key was pressed, break from the loop
    if key == ord("q"):
        break
  cv2.destroyAllWindows()
```