Smart Automated Covid-19 Prevention System

This project report is submitted to

Yeshwantrao Chavan College of Engineering
(An Autonomous Institution Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University)

In partial fulfillment of the requirement
For the award of the degree
Of

Bachelor of Engineering in Electronics & Telecommunication Engineering

by

Pragati D. Lalsare

Vishal R. Bramhankar

Kunal S. Tonge

Ganesh R. singamshettiwar

Shantanu S. Shende

Under the guidance of
Dr. P. D. Dorge
And
Mr. A . P. Deshpande from SSG Embedded solutions pvt. ltd.

DEPARTMENT OF Electronics & Telecommunication Engineering

Nagar Yuwak Shikshan Sanstha's

YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING,
(An autonomous institution affiliated to Rashtrasant Tukdoji Maharaj Nagpur University, Nagpur) NAGPUR 441110

2020-2021

CERTIFICATE OF APPROVAL

This is to Certify that the project report entitled "Smart Automated Covid-19 Prevention System" has been successfully completed by Pragati D. Lalsare, Vishal R. Bramhankar, Kunal S. Tonge, Ganesh R. singamshettiwar and Shantanu S. Shende under the guidance of Dr. P. D. Dorge and Mr.A.P.Deshpande in recognition to the partial fulfillment for the award of the degree of Bachelor of Engineering in Electronics & Telecommunication Engineering, Yeshwantrao Chavan College of Engineering (An Autonomous Institution Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University).

Signature	Signature	Signature
Dr. P.D.Dorge	Mr. A.P.Deshpande	Dr. M.S. Narlawar
(Guide)	(Industry Person)	(HoD, ET Dept.)

Signature of External Examiner
Name:-
Date of Examination:

DECLARATION

We hereby declare that

a. The work contained in this project has been done by us under the guidance of my supervisor(s).

b. The work has not been submitted to any other Institute for any degree or diploma.

c. We have followed the guidelines provided by the Institute in preparing the project report.

d. We have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

e. Whenever we have used materials (data, theoretical analysis, figures, and text) from other sources, we have given due credit to them by citing them in the text of the report and giving their details in the references. Further, we have taken permission from the copyright owners of the sources, whenever necessary.

Signature & Name of the Students

1. Pragati D. Lalsare

2.Vishal R. Bramhankar

3. Kunal S. Tonge

4.Ganesh R. singamshettiwar

5.Shantanu S. Shende

ACKNOWLEDGEMENT

Success is the sum of small efforts, repeated day in and day out. At the very outset, we wish to place on record our deep sense of gratitude and indebtedness to our worthy guide “Dr. P.D. Dorge” Assistant Professor in Department of Electronics and Telecommunication Engineering, Yeshwantrao Chavan College of Engineering, Nagpur and Mr. A. P. Deshpande from “SSG Embedded Solutions pvt. Ltd”. His dynamism and diligent enthusiasm have been highly instrumental in keeping our spirits high. His flawless & forthright suggestions blended with an innate intelligent application have crowned our task with success. We are grateful to his for his comments, constructive criticism and insights in the preparation of this report.

We are highly obliged to Dr. M.S. Narlawar, H.O.D of Electronics & Telecommunication Engineering, Yeshwantrao Chavan College of Engineering, Nagpur for allowing us to carry out our thesis work in this esteemed institution.

We are also thankful to the authors whose work we have consulted and quoted in this work. Last, but not the least, very special thanks to our parents and our friends for their constant encouragement and blessings. Their patience and understanding without which this study would not have been in this present form, is great appreciated.

Nagar Yuwak Shikshan Sanstha's
Yeshwantrao Chavan College of Engineering
(Autonomous Institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University)
Hingna Road, Wanadongri, Nagpur - 441 110
Ph.: 07104-237919, 234623, 329249, 329250 Fax: 07104-232376, Website: www.ycce.edu

Department of Electronics and Telecommunication Engineering

Date:-25/12/2020

To,
Dr. A.P. Deshpande sir

Subject: Guidance for final year UG project in the department of Electronics and Telecommunication Engineering, Yeshwantrao Chavan College of Engineering, Nagpur

Respected Sir,

We are the final year students of Electronics and Telecommunication Department, of Yeshwantrao Chavan College of Engineering, Nagpur. We are a group of 5 students we want to work on the project titled " Smart Covid 19 Prevention System ". We seek your guidance for the same. We request you to kindly accept our request for guidance of our project and it would be very great if you agree to be a co-guide for our project .
This letter is in accordance with the above request and is being sent with the consent of our project guide and HoD. Your acknowledgment for this collaboration would be highly appreciated.

Thanking you.

Projectees:

1.Pragati Lalsare
2.Ganesh Singamshettiwar
3.Vishal Bramhankar
4.Shantanu Shende
5.Kunal Tonge

Dr. M. S. Narlawar
HoD ET Dept.

Dr. A.P. Deshpande

Dr. Prabharkar Dorge

Acceptance letter

SSG Embedded Solutions

Plot no. 25, Vashistha Apartment, NIT Layout Swavalambi Nagar,
Nagpur-440022, Maharashtra, India

Acceptance Letter

Date:-25/12/2020

To,
Head of Department,
Electronics and Telecommunication Department,
Yeshwantrao Chavan College of Engineering, Nagpur.

Subject :- Collaboration of UG project

I the undersigned confirm my acceptance as a co-guide for the UG project titled "Smart Covid19 Prevention System" under the guidance of Dr. Prabhakar Dorge sir. I will provide them necessary guidance for smooth completion of their project work.

Thanking you.

Projectees:

1.Pragati Lalsare
2.Ganesh Singamshettiwar
3.Vishal Bramhankar
4.Shantanu Shende
5. Kunal Tonge

Dr. A. P. Deshpande
Industry Person Name and Sign

Dr. M. S. Narlawar
HoD ET Dept.

List of Tables

List of Figures

Abstract

As we know the COVID-19 is spread all over the world in almost every sector of life. As the Coronavirus is coming with its new strains it is uncertain that how long it will stay. We have already taken major precautionary measures such as nationwide lockdown which hits the economy too badly. The major key factor to reduce the corona is wearing mask, isolation and sanitization time to time. As the precaution is better than cure so we propose a Smart Automated COVID-19 Prevention System which helps us to reduce the number of COVID-19 patients. We aim to measure the temperature of the person which will help us to isolate the person also we are detecting either the person is wearing the mask or not by using image processing along with sanitization of hands. We hoped that our system will help to reduce COVID-19.

Table of Contents

Chapter 1

Introduction

Chapter 1. Introduction

1.1 Overview

This system is mainly focused on three parts. First is mask detection, second is temperature check-up, and third is hand sanitization. These three processes are very important for the prevention of corona-virus. For building the system we are using deep learning and image processing concept. Deep learning has an important role in the field of "Artificial Intelligence" to extract the feature from the image.

Firstly, we create the convolutional neural network and build the model using the training dataset so that it will give the correct output for the testing dataset. Once the model is built, the system is ready to prevent the person from coronavirus. when a person comes in front of the camera, the camera captures an image of that person and processes that image, and detects whether the person wears the mask or not. If the mask is detected then massage is displayed on the screen that mask is detected. After that again massage is displayed on the screen please go for the sanitization. Once they put their hands in the sanitization chamber that time infrared temperature detector detects the temperature and displays it on the display screen. If the temperature of that person is between the normal human temperature, then we all that person to go inside then sanitizer is sprinkles on their hands else message is display on the screen is that you are not allow to go inside.

1.2 Literature Survey

According to the report of cdc (center for Disease control and prevention) cough and fever are the primary symptoms of covid-19. The virus is transmitted through direct contact with respiratory droplets of an infected person (generated through coughing and sneezing). Individuals can also be infected from and touching surfaces contaminated with the virus and touching their face (e.g., eyes, nose, mouth).so to wearing mask is compulsory to all the person and sanitize ourself properly is also compulsory. While considering the current scenario many systems had created for the prevention from covid-19. mask detection system, temperature checkup machine, etc. To developed the prevention system for covid -19 Artificial intelligence and big data plays a important role. Blue dot is the Tornado-based digital health.

1.3 Problem Statement

Severe acute respiratory syndrome coronavirus causes COVID-19 and is spread person-to-person through close contact. We aimed to investigate the effects of physical distance, face masks, and Body Temperature sensing on virus transmission in health-care and non-health-care org (eg, community) settings.

1.4 Thesis Objectives

The basic objective of this report or thesis is that it will prove to be the one and only fundamental document related to our project. The thesis contains all the steps involved in the creation of our final year project which is the "Smart Automated Covid-19 Prevention System". From the very first step of finalizing our problem statement to the last step of creating and testing the implemented model the thesis consists of everything. All the electronic components, methods used are also mentioned in the thesis. An overview of this thesis can easily explain the total idea, working and the whole concept behind this bot. In this thesis we have tried to cover everything related to our project.

1.5 Thesis Organization

This thesis is designed as per the proper formatting. It contains the basic information about the project. certificate of approval, declaration and acknowledgement. After this the thesis has been formed keeping in mind the structure of a book. It contains the table of contents and all the required data in the form of various chapters. The introduction, abstract, literature survey, work done, etc. are all mentioned in the thesis. The design given is on the basis of the format required.

Chapter 2
Review of Literature

Chapter 2. Review of Literature

Due to the COVID-19 pandemic, wearing a mask is mandatory in public spaces, as properly wearing a mask offers a maximum preventive effect against viral transmission. Body temperature has also become an important consideration in determining whether an individual is healthy [2]. In this covid 19 situation main factor is our safety. According to literature survey current system focused only on the sprinklers spraying sanitizer. Basically, that's not a real time system. But for better safety we need a real time system that detects mask, temperature and also provide proper sanitization. To overcome this gap, we are Proposing A Smart & Affordable Automatic Contactless face recognition IR Temperature Check-up and Mask Detection using Facial Landmarking and Deep Learning [1]. Currently, Temperature Checkups are done manually using Contactless Thermometer. Manual Checkups can be Inefficient, Impractical (in places with a large footfall), and Risky [3]. Our aim is to develop a system that could be available to every door step of public places or crowded places in order to maintain necessary precaution from covid19 virus [4].

Chapter 3

Work done

Chapter 3. Work done

3.1 Flow chart

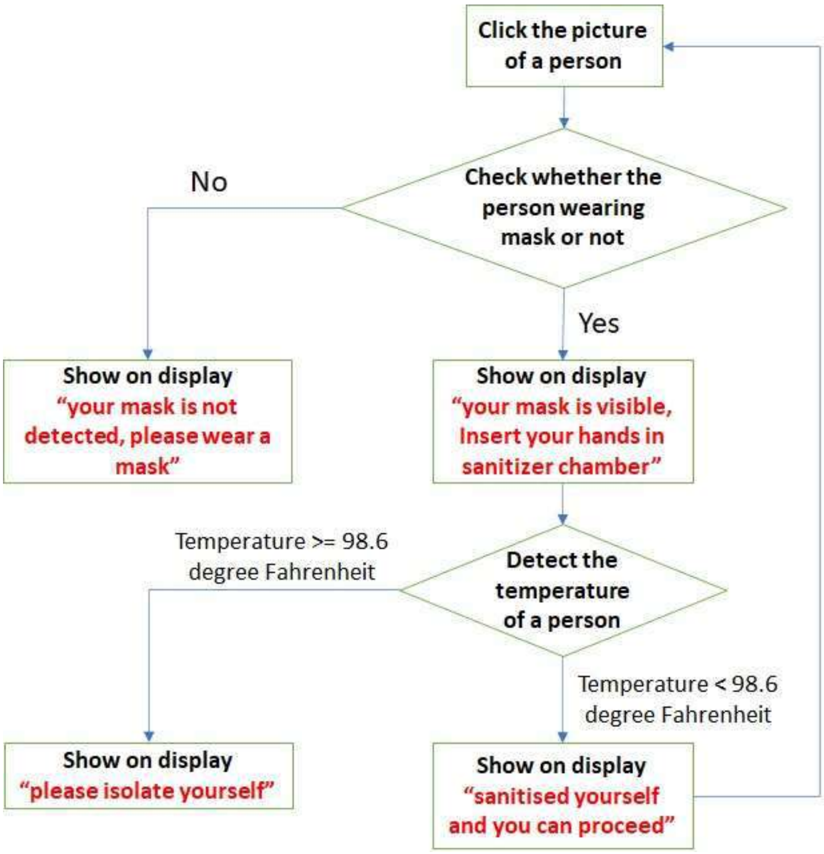


Fig 3.1 flow chart for Smart Automated COVID-19 Prevention System

figure 3.1 illustrates a brief description of the flow chart. When a person comes in front of the raspberry pi camera, the camera takes the image as an input and then sends it to raspberry pi 4B for detection. Using convolution neural network and data set implemented in the system. The input image is then compared with the data set. If the person is not wearing the mask and the mask is not detected the display screen mounted on the system will display " Access Denied ". If the person is wearing the mask the display screen will display the instruction as " Put Your Hands In The Sanitization Chamber ". The LED light will glow in the sanitization chamber. When the person will insert the hands, the IR sensor that is mounted on the sanitization chamber will detect the hand as an object. the non-contact IR temperature sensor (MLX90614) is mounted below the chamber where insertion of hands takes place. Further, the temperature sensor will measure the temperature of the person and the screen will display the temperature If the temperature is greater than 98.6°F then the display screen will display " Access Denied ". The sanitization chamber consists of a vessel with a liquid sanitizer. If the temperature is less than 98.6°F, the relay circuit is triggered and the switch of the solenoid valve gets ON, as soon as the solenoid valve is opened the sanitizer is sprayed over the hand of the person. At this last stage, the display screen will display the instruction as "Access Granted".

3.2 Block diagram

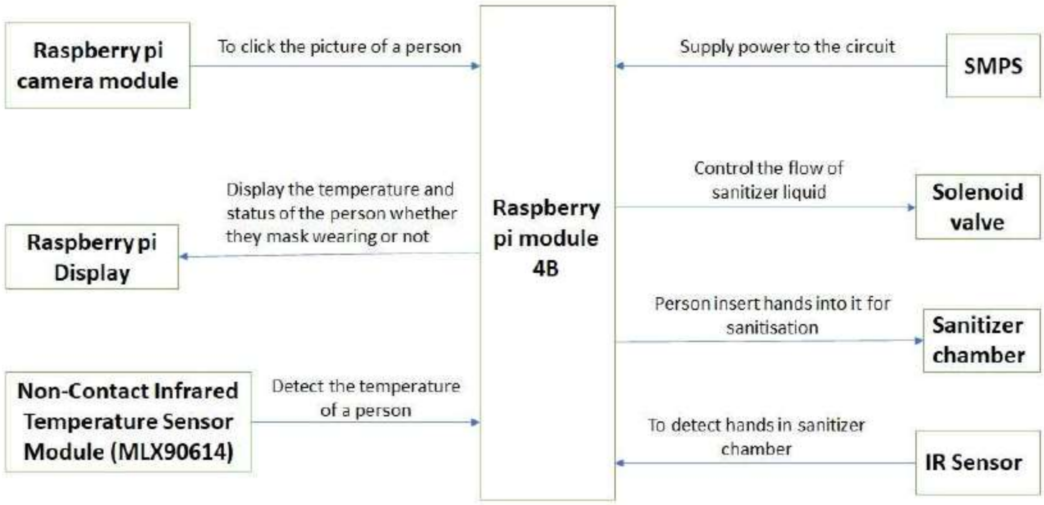


Fig 3.2 block diagram for Smart Automated COVID-19 Prevention System

Figure 3.2 demonstrates the block diagram of the smart covid 19 prevention system, showing the flow of the process. The diagram includes blocks of the various components on the frame. The line which is directed towards the raspberry pi indicates the input line. Raspberry pi camera module, Non-contact infrared temperature sensor, and IR sensor are the input elements. The output of these components is applied to the raspberry pi. Raspberry pi acts as a controlling unit. The output of raspberry pi is given to the raspberry pi display, solenoid valve, and sanitizer chamber.

Figure 3.3 illustrates a brief description of the model building. The first block indicates the loading of face mask dataset which contain the images of a person in which some person wears a mask and some are not. After that, we train our model using deep learning libraries Tensorflow and Keras. are the two most important libraries in Deep Learning that are used to extract the important feature from the image and give a correct prediction. Once the face mask detector is trained, we can then move on to loading the mask detector, performing face detection, and then classifying each face as with or without mask.

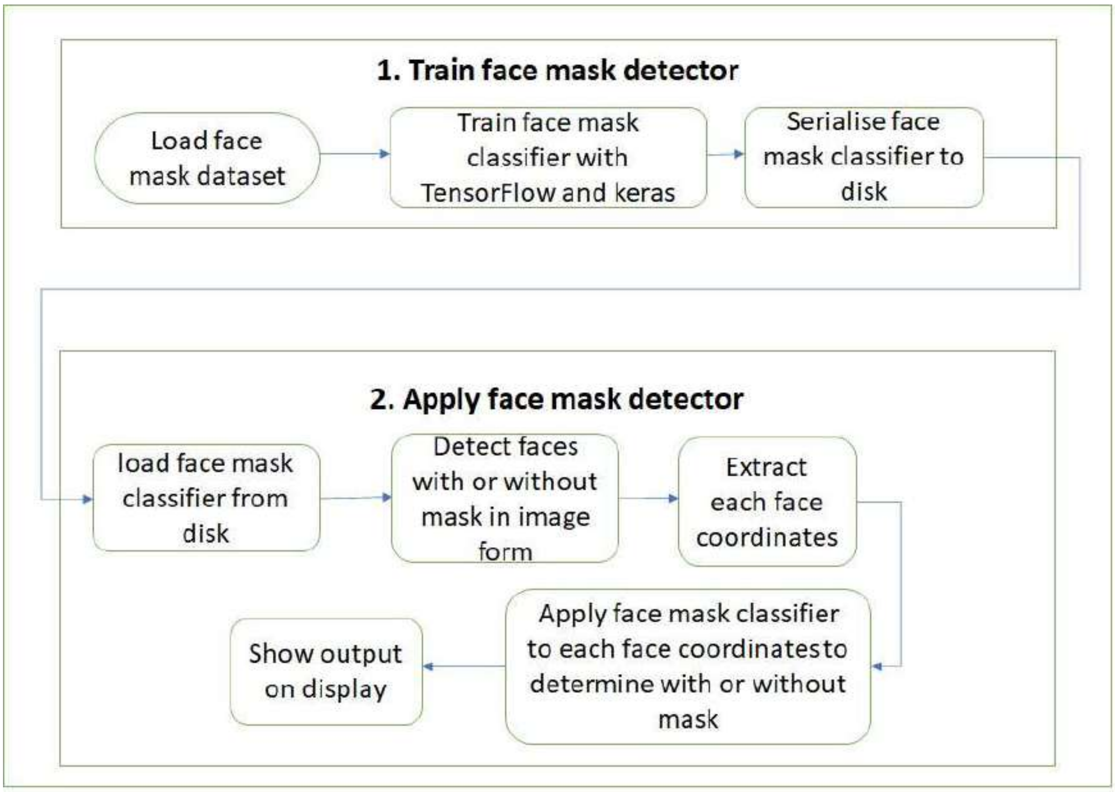


Fig. 3.3 block diagram for model building

3.3 Circuit diagram

Figure 3.4 illustrates the circuit diagram of the smart covid 19 prevention system, It shows the details of visible elements and their connections with each other. the diagram includes the mounting of the various components on the frame. raspberry pi module act as a controlling part in the diagram. Raspberry pi camera module is connected to the raspberry pi with the help of a ribbon cable and enabled camera option in the raspberry pi setting. Non-contact infrared temperature sensor module (MLX90614) is used to calculate the temperature of the body. IR sensors are acting as an obstacle detection system according to that sanitizer sprinkle on the hand. The power supply block gives 12 v constant voltage to the raspberry pi module. Sanitizer indicators are uses to indicate whether the sanitizer machine is on or off. solenoid valve is used to allow the fluid flow.

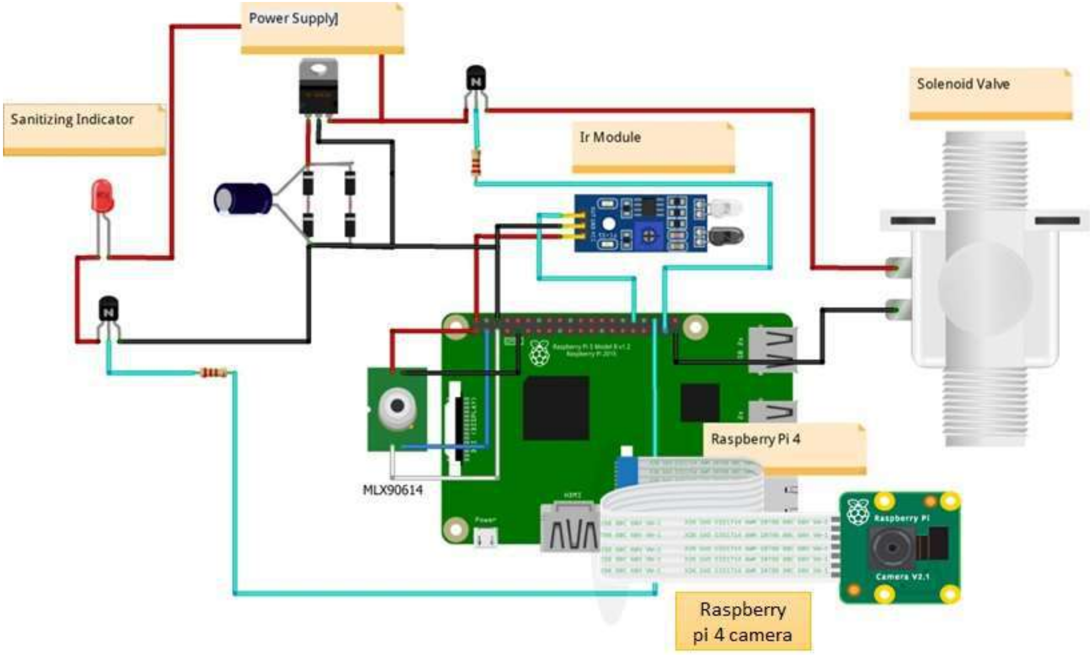


Fig 3.4 circuit diagram for Smart Automated COVID-19 Prevention System

3.4 Description

The invention seeks to develop a smart covid 19 prevention system to detect the face mask, temperature of the body and provide sanitization to the hands. A concept used to detect the face mask is Deep Learning and the concept used for temperature detection and sanitization is an Embedded system.

Image Processing

Image processing is a method to perform some operations on an image, to get an enhanced image, or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be an image or characteristics/ features associated with that image. Image captured by camera required processing before going to the next step. The image captured by the camera is in RGB format we need to convert the image into a grayscale format so that it will take less amount of space as compared to an RGB image. Then we extract important features from the image and convert the pixel of the image into the "0" to "1" range.

Deep Learning

Deep Learning is a Machine Learning technique that learns features and tasks directly from the data where data may be images, text, or sound. This learning feature is used for prediction. to train deep learning architecture, we use a standard dataset from

Github. The feature learning process depends on the Convolutional Neural Network. All the processes of the deep learning algorithm are described below.

1. Dataset: The dataset consists of a total of 4095 images out of which 2065 images are with mask images and 1930 images are without mask images. For Training purposes, we use 80% of the dataset and the rest of the 20% is used for testing purposes.

2) Architecture Development: The learning model is based on a convolutional neural network which is very useful to extract the important feature from an image. The network comprises the input layer, hidden layer, and output layer. The hidden layer consist of CNN consists of convolution layer, max-pooling layer, Flattering layer, and fully connected layer. This layer is used to reduce the size of the image and extract important features from an image. The convolutional layer applies a mask on the image pixel it helps to reduce the size of the image. The function of the pooling layer is to reduce the spatial size of the image and reduce the number of parameters. The flattering layer converts the data into a 1-dimensional array format. Pair of dense and dropout layers learn parameters for classification the dense layer comprises the series of neurons. The dropout layer prevents overfitting. finally, a dense layer containing two neurons distributes the classes.

3) Face mask detection: The main goal of our system is to detect facial masks. The learning architecture identifies whether the person wears a mask or not. The camera captures the image of the person and it sends it to the raspberry pi. Raspberry pi detects whether a person wears a mask or not and according to that message display on the screen. Embedded system: For the temperature measurement, we use a Non-contact Infrared temperature sensor. The infrared temperature sensor can use with any microcontroller that can communicate with it through its I2C interface. It is used to calculate the temperature of the body and it sends to the controller if their body temperature is greater than normal body then message display on the screen "please Isolate Yourself" else sanitizer sprinkle on person hand and massage is displayed on the screen that "You may proceed now".

3.5 Actual model

Figure 3.5.1 Front view

Figure 3.5.2 Top view

Figure 3.5.3 Side view

3.6 Hardwares

3.6.1 Raspberry pi 4B

Overview:

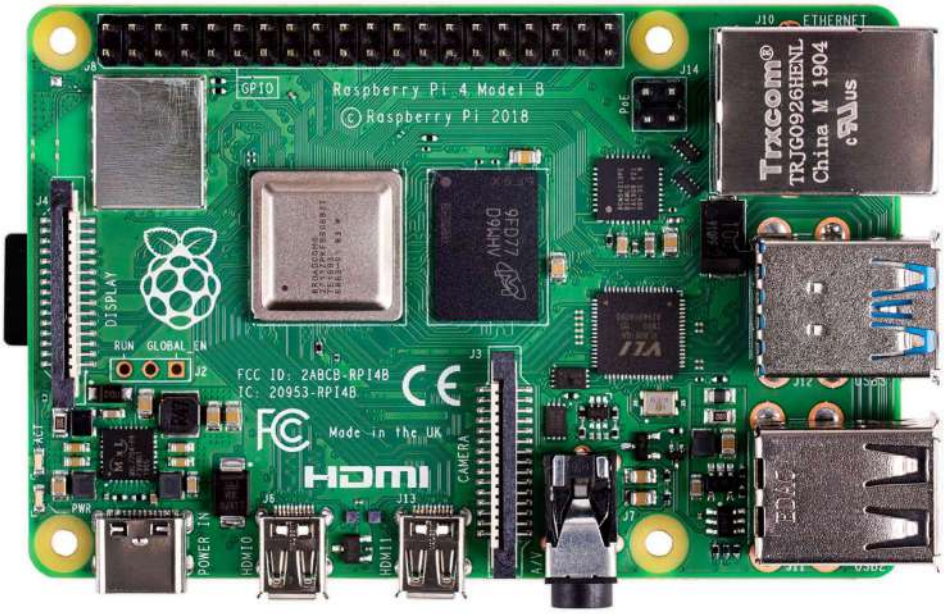


Figure 3.6.1.1 Raspberry pi 4B

Raspberry Pi 4 Model B is the latest product in the popular Raspberry Pi range of computers. It offers ground-breaking increases in processor speed, multimedia performance, memory, and connectivity compared to the prior-generation Raspberry Pi 3 Model B+, while retaining backwards compatibility and similar power consumption. For the end user, Raspberry Pi 4 Model B provides desktop performance comparable to entry-level x86 PC systems.

This product's key features include a high-performance 64-bit quad-core processor, dual-display support at resolutions up to 4K via a pair of micro-HDMI ports, hardware video decode at up to 4Kp60, up to 4GB of RAM, dual-band 2.4/5.0 GHz wireless LAN, Bluetooth 5.0, Gigabit Ethernet, USB 3.0, and PoE capability (via a separate PoE HAT add-on).

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

Specification:

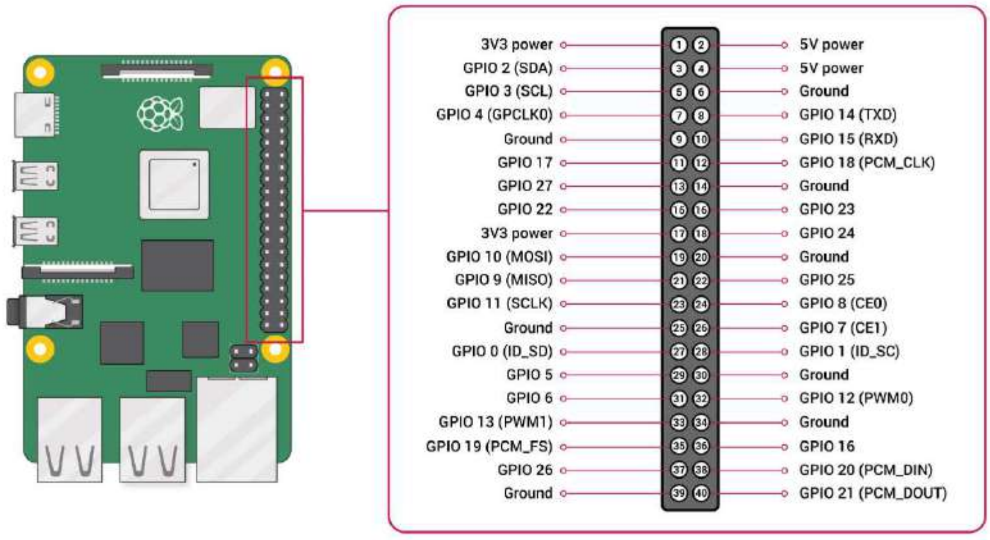


Figure 3.6.1.2 pin diagram

Processor: Broadcom BCM2711, quad-core Cortex-A72 (ARM v8)
64-bit SoC @ 1.5GHz

Memory: 1 GB, 2GB or 4GB LPDDR4
(depending on model)

Connectivity:
2 .4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN, Bluetooth 5.0, BLE
Gigabit Ethernet
2 × USB 3.0 ports
2 × USB 2.0 ports.

GPIO: Standard 40-pin GPIO header
(fully backwards-compatible with previous boards)

Video & sound: 2 × micro-HDMI ports (up to 4Kp60 supported)
2-lane MIPI DSI display port
2-lane MIPI CSI camera port
4-pole stereo audio and composite video port

Multimedia: H .265 (4Kp60 decode);
H.264 (1080p60 decode, 1080p30 encode);
OpenGL ES, 3.0 graphics

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

Input power: 5V DC via USB-C connector (minimum 3A[1])

5V DC via GPIO header (minimum 3A[1])
Power over Ethernet (PoE)–enabled
(requires separate PoE HAT)

Environment: Operating temperature 0–50ºC

Physical Specifications:

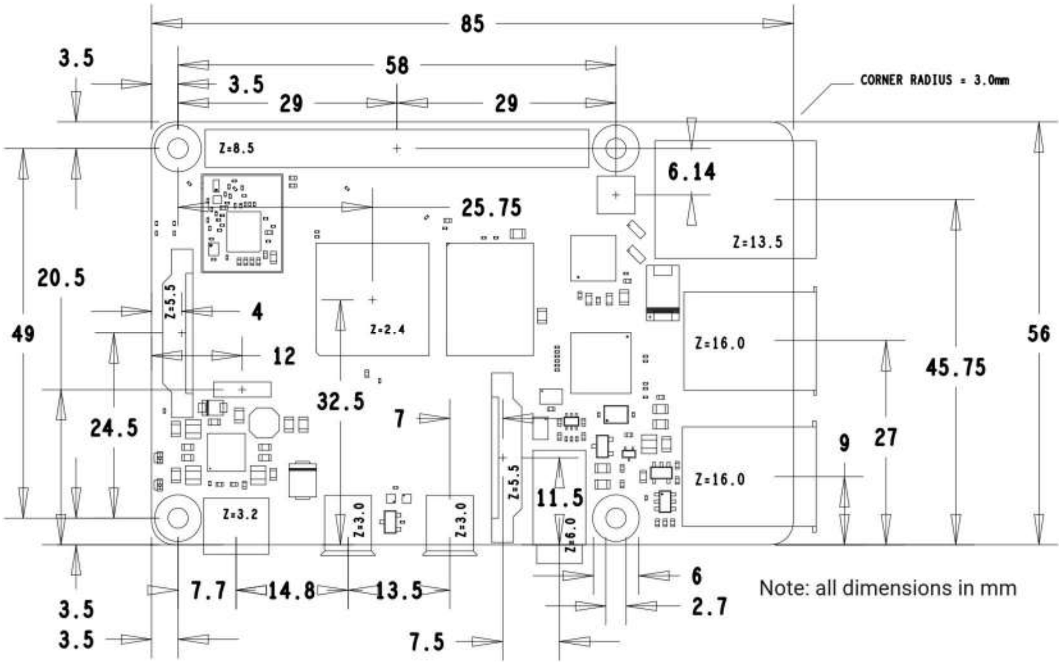


Figure 3.6.1.3 physical specifications

3.6.2 Non-contact Infrared Temperature Sensor module (MLX 90614)

Figure 3.6.2 Non-contact Infrared Temperature Sensor module (MLX 90614)

The conventional temperature sensors need you to come in contact with the sensor. This is where our sensor stands out, MLX90614-DAA is a contactless temperature sensor, where the temperature can be sensed without actually coming in contact with the sensor.

This is an infrared thermometer designed for non-contact temperature sensing. An internal 17-bit ADC and a powerful DSP contribute to the MLX90614's high accuracy and resolution. It has a huge number of applications including body temperature measurement and movement detection. The sensor has a field of view of 90 degrees and returns the average temperature value of all objects within this field of view.

MLX90614ESF-DAA non-contact infrared thermometer for use with Arduino, or any microcontroller that can communicate with it through its I2C interface.

This sensor comes with a breakout board with all of the components needed for operation and two types of pins.

3.6.3 Raspberry Pi Camera Module v2

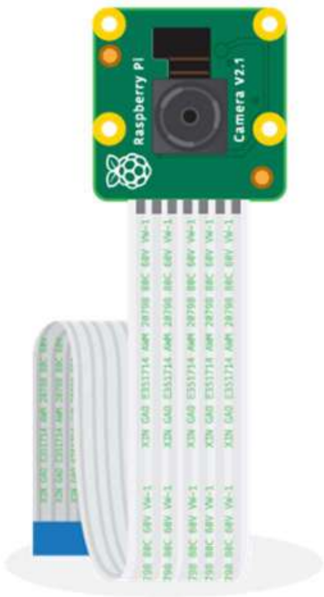
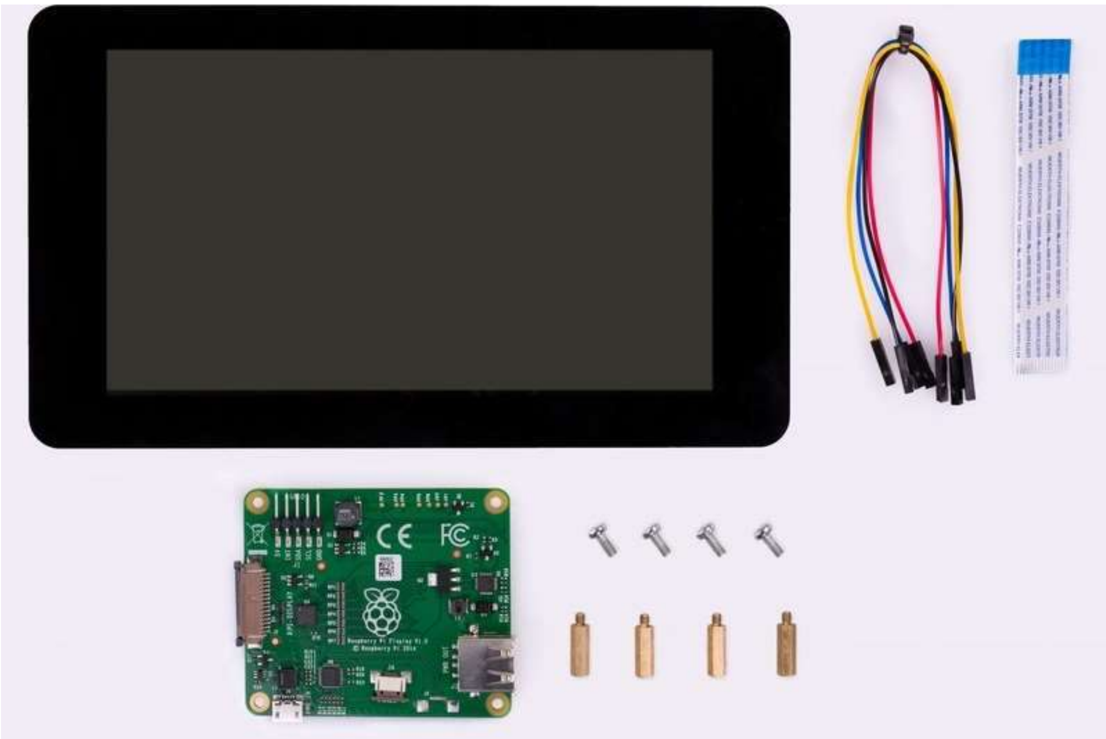


Figure 3.6.3 Raspberry Pi Camera Module v2

The Raspberry Pi Camera Module v2 replaced the original Camera Module in April 2016. The v2 Camera Module has a Sony IMX219 8-megapixel sensor (compared to the 5-megapixel OmniVision OV5647 sensor of the original camera).

The Camera Module can be used to take high-definition video, as well as stills photographs. It’s easy to use for beginners, but has plenty to offer advanced users if you’re looking to expand your knowledge. There are lots of examples online of people using it for time-lapse, slow-motion, and other video cleverness. You can also use the libraries we bundle with the camera to create effects.

3.6.4 Raspberry pi touch display



3.6.4 Raspberry pi touch display

This 7″ touchscreen monitor for Raspberry Pi gives users the ability to create all-in-one, integrated projects such as tablets, infotainment systems and embedded projects. The 800 x 480 display connects via an adapter board which handles power and signal conversion. Only two connections to the Pi are required; power from the Pi’s GPIO port and a ribbon cable that connects to the DSI port present on all Raspberry Pis (except Raspberry Pi Zero and Zero W). Touchscreen drivers with support for 10-

finger touch and an on-screen keyboard will be integrated into the latest Raspberry Pi OS for full functionality without a physical keyboard or mouse.

- Turn your Raspberry Pi into a touch screen tablet, infotainment system, or standalone device.
- Truly interactive – the latest software drivers will support a virtual 'on-screen' keyboard, so there is no need to plug in a keyboard and mouse.
- Make your own *Internet of Things* (IoT) devices including a visual display. Simply connect your Raspberry Pi, develop a Python script to interact with the display, and you're ready to create your own home automation devices with touch screen capability.
- A range of educational software and programs available on the Raspberry Pi will be touch enabled, making learning and programming easier on the Raspberry Pi.

3.6.5 IR Sensors

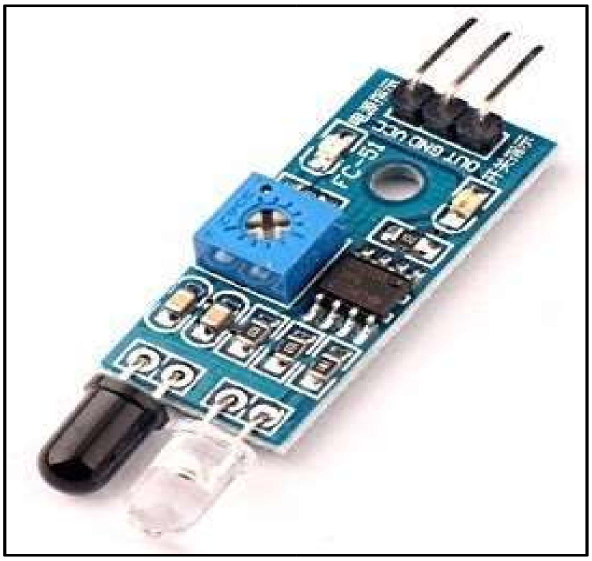


Figure 3.6.5 IR Sensors

The IR Obstacle Detection sensor from Robot Basis specifically designed for robust indoor operation. The sensor module employs a modulated LED light source and a tuned IR receiver to detect if the radiated IR light is reflected back to the sensor. When this happens, the on board LED lights and the output signal S is set to logic low. The Module employs two adjustment potentiometers. One adjustment controls the IR receiver sensitivity. By adjusting the sensor more sensitive, a longer detection distance is available, but more false detections may occur. On the other, reducing detection sensitivity will result in less false detection, but less range. A threshold adjustment used to provide the point at which the sensor determines an obstacle, and to provide adequate range of detection for any given sensitivity setting.

The module may be driven using a three wire interface, or a four wire interface. For the three wire interface, all that is needed is +5VDC, ground and a signal output line. In this case, the Enable Function jumper is required. In four wire mode, the jumper is removed; a fourth signal ENABLE is available. If the enable line is held low, the sensor is enabled, and the light source is activated. When the enable input is held high, the sensor output is placed into a power saving state. The module may be driven using a three wire interface, or a four wire interface. For the three-wire interface, all that is needed is +5VDC, ground and a signal output line. In this case, the Enable Function jumper is required. In four wire mode, the jumper is removed; a fourth signal ENABLE is available. If the enable line is held low, the sensor is enabled, and the light source is activated. When the enable input is held high, the sensor output is placed into a power saving state.

3.6.6 Relay Module

Figure 3.6.6 Relay Module

- Contact pin length: about 3.5mm
- Contact capacity: 10A/250VAC, 10A/125VAC, 10A/30VDC, 10A/28VDC
- Action Type: SPDT,1 NO 1NC
- Nominal coil voltage: DC 5V

- Coil resistance: 70-80 ohm
- Pin Number: 5

A relay is an electrically operated switch. It consists of a set of input terminals for a single or multiple control signals, and a set of operating contact terminals. The switch may have any number of contacts in multiple contact forms, such as make contacts, break contacts, or combinations thereof. The relay module is a separate hardware device used for remote device switching. With it you can remotely control devices over a network or the Internet. Devices can be remotely powered on or off with commands coming from Clock Watch Enterprise delivered over a local or wide area network. One can control computers, peripherals or other powered devices from across the office or across the world. The Relay module can be used to sense external On/Off conditions and to control a variety of external devices. The PC interface connection is made through the serial port.

3.6.7 Micro Sd Card 16GB

Figure 3.6.7 Micro Sd Card 16GB

Here we are using 16GB Sd card for storing the data set of peoples wearing mask and peoples without mask. More storage is required as if more data sets need’s to be added as required thus a vacant space is given to the system.

3.6.8 Rasbery PI Power Adepter

Figure 3.6.8 Rasbery PI Power Adepter

The Rasbery PI Power Adepter Gives the power to the CPU of the system.The Power of : 5 V; 3 A for full power is delivered to USB. AC 100-240V 50Hz/60Hz input DC 5.1V 3A Output 15.3W maximum output power 1.5m 18 AWG captive cable with USB-C output connector

3.6.9 Premium Female/Male Extension Jumper Wires, 40 x 6" (150mm)

Figure 3.6.9 Premium Female/Male Extension Jumper Wires, 40 x 6" (150mm)

These premium jumper wires are 6" (150mm) long and come in a 'strip' of 40 (4 pieces of each of ten rainbow colors). They have 0.1" male header contacts on one end and 0.1" female header contacts on the other. They fit cleanly next to each other on standard-pitch 0.1" (2.54mm) header. This Jumpers plays the important role in data transfer in the system.

Figure 3.6.9 Premium Female/Male Extension Jumper Wires, 40 x 6" (150mm)

3.7 Software's and technologies

3.7.1 Raspberry pi Raspbian (32-bit OS)

Raspbian OS is the operating system provided by Raspberry Pi and is based on Debian. As only the latest Raspberry Pi-boards have a 64-bit chip, the official release of Raspbian OS is 32-bit only. Many operating systems are available for Raspberry Pi, including Raspberry Pi OS, our official supported operating system, and operating systems from other organisations. Raspberry_Pi_Imager is the quick and easy way to install an operating system to a microSD card ready to use with your Raspberry Pi. Alternatively, choose from the operating systems below, available to download and install manually.

3.7.2 Thonny

An integrated development environment (IDE) facilitates computer programmers by integrating fundamental tools (e.g., code editor, compiler, and debugger) into a single software package. Users do not need to install the language's compiler/interpreter on their machines; an IDE provides the environment itself. Thonny is a free, dedicated IDE for Python designed for beginners.

Features:

The following are some of the primary features of Thonny:

- It autocompletes code.
- It inspects code to provide bracket matching and highlight errors.
- It is easy to start with as its installer also installs Python 3.7.
- Its debugger is simple to use as no knowledge of breakpoints is required.
- It enables users to step into a function call by providing details about local variables and displaying the code pointer.
- It has an easy interface to install packages. This makes it very suitable for beginners.

3.7.3 Image Processing:

Image processing is a method to convert an image into digital form and perform some operations on it, in order to get an enhanced image or to extract some useful information from it.

It is a type of signal dispensation in which input is an image, like video frame or photograph and output may be image or characteristics associated with that image.

Usually Image Processing system includes treating images as two dimensional signals while applying already set signal processing methods to them.

Purpose of Image processing

The purpose of image processing is divided into 5 groups. They are :

Visualization - Observe the objects that are not visible.

Image sharpening and restoration - To create a better image.

Image retrieval - Seek for the image of interest.

Measurement of pattern – Measures various objects in an image.

Image Recognition – Distinguish the objects in an image.

Fundamental steps in Digital Image Processing:

1.Image Acquisition

This is the first step or process of the fundamental steps of digital image processing. Image acquisition could be as simple as being given an image that is already in digital form. Generally, the image acquisition stage involves pre-processing, such as scaling etc.

2. Image Enhancement

Image enhancement is among the simplest and most appealing areas of digital image processing. Basically, the idea behind enhancement techniques is to bring out detail that is obscured, or simply to highlight certain features of interest in an image. Such as, changing brightness & contrast etc.

3. Image Restoration

Image restoration is an area that also deals with improving the appearance of an image. However, unlike enhancement, which is subjective, image restoration is objective, in the sense that restoration techniques tend to be based on mathematical or probabilistic models of image degradation.

4. Color Image Processing

Color image processing is an area that has been gaining its importance because of the significant increase in the use of digital images over the Internet. This may include color modeling and processing in a digital domain etc.

5. Wavelets and Multiresolution Processing

Wavelets are the foundation for representing images in various degrees of resolution. Images subdivision successively into smaller regions for data compression and for pyramidal representation.

6. Compression

Compression deals with techniques for reducing the storage required to save an image or the bandwidth to transmit it. Particularly in the uses of internet it is very much necessary to compress data.

7. Morphological Processing

Morphological processing deals with tools for extracting image components that are useful in the representation and description of shape.

8. Segmentation

Segmentation procedures partition an image into its constituent parts or objects. In general, autonomous segmentation is one of the most difficult tasks in digital image processing. A rugged segmentation procedure brings the process a long way toward successful solution of imaging problems that require objects to be identified individually.

9. Representation and Description

Representation and description almost always follow the output of a segmentation stage, which usually is raw pixel data, constituting either the boundary of a region or all the points in the region itself. Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computer processing. Description deals with extracting attributes that result in some quantitative information of interest or are basic for differentiating one class of objects from another.

3.7.4 Deep learning

Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher-level features from the raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces.

Deep learning (also known as deep structured learning) is part of a broader family of machine learning methods based on artificial neural networks with representation learning. Learning can be supervised, semi-supervised or unsupervised.

Deep-learning architectures such as deep neural networks, deep belief networks, graph neural networks, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, speech recognition, natural language processing, machine translation, bioinformatics, drug design, medical image analysis, material inspection and board game programs, where they have produced results comparable to and in some cases surpassing human expert performance.

3.7.5 OpenCV

OpenCV is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then

Itseez. The library is cross-platform and free for use under the open-source Apache 2 License

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 18 million. The library is used extensively in companies, research groups and by governmental bodies.

3.8 Power distribution

As a main supply we are taking 230 V and passing it to Transformer and SMPS.

SMPS takes 230 V as Input and gives (3 A; 5 V) DC as Output

1. Raspberry Pi takes 2A; 5V supply and operates
    a) Temperature Sensor 3.5 V
    b) Infrared Sensor 5V
    c) Camera 3V
2. Display Requires (1 A; 5 V) supply to operate it is obtained and it is obtained by SMPS.

The transformer takes 230 V as a input and gives 12 V; 3 A Ac current and further it is given to Rectifiers as an Input. The Rectifier then gives output to the solenoid valve (1 A,12 V) and to the LED (200 mA, 12 V).

Chapter 4
Programming

Chapter 4. Programming

```
# import the necessary packages
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import load_model
#from imutils.video import VideoStream
from picamera.array import PiRGBArray
from picamera import PiCamera
from mlx90614 import MLX90614
from smbus2 import SMBus
import RPi.GPIO as GPIO
import numpy as np
import argparse
#import imutils
import time
import cv2
import os

GPIO.setwarnings(False)

sensorPIN = 32
valvePIN = 37
LED = 36
fanPIN = 38

GPIO.setmode(GPIO.BOARD)
GPIO.setup(sensorPIN,GPIO.IN)
GPIO.setup(valvePIN,GPIO.OUT)
GPIO.setup(LED,GPIO.OUT)
GPIO.setup(fanPIN,GPIO.OUT)

GPIO.output(valvePIN,False)
GPIO.output(LED,False)
```

```
GPIO.output(fanPIN,False)

camera = PiCamera()
# camera.resolution = (640, 480)
camera.framerate = 30
camera.rotation = 180
rawCapture = PiRGBArray(camera)
# allow the camera to warmup
time.sleep(0.1)


mask_detection_completed = False

temperature_check_completed = False

operation_status_failed = False

Del = False

def detect_and_predict_mask(frame, faceNet, maskNet):
	# grab the dimensions of the frame and then construct a blob
	# from it
	(h, w) = frame.shape[:2]
	blob = cv2.dnn.blobFromImage(frame, 1.0, (300, 300),
		(104.0, 177.0, 123.0))

	# pass the blob through the network and obtain the face detections
	faceNet.setInput(blob)
	detections = faceNet.forward()

	# initialize our list of faces, their corresponding locations,
	# and the list of predictions from our face mask network
	faces = []
	locs = []
```

```
preds = []

# loop over the detections
for i in range(0, detections.shape[2]):
	# extract the confidence (i.e., probability) associated with
	# the detection
	confidence = detections[0, 0, i, 2]

	# filter out weak detections by ensuring the confidence is
	# greater than the minimum confidence
	if confidence > args["confidence"]:
		# compute the (x, y)-coordinates of the bounding box for
		# the object
		box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
		(startX, startY, endX, endY) = box.astype("int")

		# ensure the bounding boxes fall within the dimensions of
		# the frame
		(startX, startY) = (max(0, startX), max(0, startY))
		(endX, endY) = (min(w - 1, endX), min(h - 1, endY))

		# extract the face ROI, convert it from BGR to RGB channel
		# ordering, resize it to 224x224, and preprocess it
		face = frame[startY:endY, startX:endX]
		face = cv2.cvtColor(face, cv2.COLOR_BGR2RGB)
		face = cv2.resize(face, (224, 224))
		face = img_to_array(face)
		face = preprocess_input(face)

		# add the face and bounding boxes to their respective
		# lists
		faces.append(face)
		locs.append((startX, startY, endX, endY))
```

```
		# only make a predictions if at least one face was detected
		if len(faces) > 0:
			# for faster inference we'll make batch predictions on all
			# faces at the same time rather than one-by-one predictions
			# in the above `for` loop
			faces = np.array(faces, dtype="float32")
			preds = maskNet.predict(faces, batch_size=32)

		# return a 2-tuple of the face locations and their corresponding
		# locations
		return (locs, preds)

# construct the argument parser and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-f", "--face", type=str,
		default="face_detector",
		help="path to face detector model directory")
ap.add_argument("-m", "--model", type=str,
		default="mask_detector.model",
		help="path to trained face mask detector model")
ap.add_argument("-c", "--confidence", type=float, default=0.5,
		help="minimum probability to filter weak detections")
args = vars(ap.parse_args())

# load our serialized face detector model from disk
print("[INFO] loading face detector model...")
prototxtPath = os.path.sep.join([args["face"], "deploy.prototxt"])
weightsPath = os.path.sep.join([args["face"],
		"res10_300x300_ssd_iter_140000.caffemodel"])
faceNet = cv2.dnn.readNet(prototxtPath, weightsPath)

# load the face mask detector model from disk
print("[INFO] loading face mask detector model...")
maskNet = load_model(args["model"])
```

```
# initialize the video stream and allow the camera sensor to warm up
print("[INFO] starting video stream...")
#vs = VideoStream(src=0).start()
#vs = VideoStream(usePiCamera=True).start()
#time.sleep(2.0)

bus = SMBus(1)
sensor = MLX90614(bus, address=0x5A)
temp = None

# loop over the frames from the video stream
for fram in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
    # grab the frame from the threaded video stream and resize it
    # to have a maximum width of 400 pixels
    im = fram.array
    frame = cv2.flip(im,1)
    #fram = vs.read()
    #frame = cv2.flip(fram,1)
    frame = cv2.resize(frame, (1024,735))

            # detect faces in the frame and determine if they are wearing a
            # face mask or not
    (locs, preds) = detect_and_predict_mask(frame, faceNet, maskNet)

   # if cpu_temp > 75:
      # print("CPU Temperature too high!")
      # GPIO.output(fanPIN,True)

    #if cpu_temp < 40:
      # print("CPU Temperature Is Cooled Now!")
      # GPIO.output(fanPIN,False)

            # loop over the detected face locations and their corresponding
```

```
        # locations
for (box, pred) in zip(locs, preds):
    if Del:
        time.sleep(8)
        Del = False
    # unpack the bounding box and predictions
    (startX, startY, endX, endY) = box
    (mask, withoutMask) = pred

                # determine the class label and color we'll use to draw
                # the bounding box and text
    if mask_detection_completed is False:
        mask < withoutMask
        label = "No Face Mask Detected"
        color = (0, 0, 255)

        if mask > withoutMask:
            label = "Face Mask Detected"
            color = (0, 255, 0)
            cv2.putText(frame, 'Please! Put Your Hand Into', (200, 500),
                        cv2.FONT_HERSHEY_COMPLEX, 1.3, (0, 255, 255), 3)
            cv2.putText(frame, 'Sanitizer Chamber For Few Seconds', (100, 575),
                        cv2.FONT_HERSHEY_COMPLEX, 1.3, (0, 255, 255), 3)
            mask_detection_completed = True

    elif mask_detection_completed:
        label = "Face Mask Detected"
        color = (0, 255, 0)
        time.sleep(0.5)
        GPIO.output(LED,True)

        if temperature_check_completed is False:
            if GPIO.input(sensorPIN):
                time.sleep(3)
```

```
        temp = 9/5*sensor.get_object_1()+42
        time.sleep(0.3)
        cv2.putText(frame, 'Body Temp: {:.2f}F '.format(temp), (200, 400),
        cv2.FONT_HERSHEY_COMPLEX, 2, (0, 255, 0), 5)
        while GPIO.input(sensorPIN):
            time.sleep(0.2)

        if temp > 100:
            cv2.putText(frame, 'Body Temperature too High! ', (200, 600),
            cv2.FONT_HERSHEY_COMPLEX, 2, (0, 0, 255), 5)
            time.sleep(3)
            operation_status_failed = True
            Del = True

        else:
            time.sleep(1)
            GPIO.output(valvePIN,True)
            time.sleep(2)
            GPIO.output(valvePIN,False)
            cv2.putText(frame, 'Sanitized', (330, 500),
                    cv2.FONT_HERSHEY_COMPLEX, 2, (0, 255, 255), 5)
            cv2.putText(frame, 'Please! Remove Your Hand', (60, 600),
                    cv2.FONT_HERSHEY_COMPLEX, 2, (0, 255, 255), 3)
            cv2.putText(frame, 'YOU MAY PROCEED !', (100, 700),
                    cv2.FONT_HERSHEY_COMPLEX, 2.5, (0, 255, 0), 5)
            time.sleep(1.5)
            GPIO.output(LED,False)
            temperature_check_completed = True
            Del = True

    elif temperature_check_completed:
        mask_detection_completed = False
        temperature_check_completed = False
        temp = None
```

```
                GPIO.output(LED,False)

            elif operation_status_failed:
                # Reset
                mask_detection_completed = False
                temperature_check_completed = False
                temp = None
                operation_status_failed = False
                GPIO.output(LED,False)

        #else:
            #label = "No Face Mask Detected"
            #color = (0, 0, 255)

                        # display the label and bounding box rectangle on the output
                        # frame
        cv2.putText(frame, label, (startX-50, startY - 10),
            cv2.FONT_HERSHEY_SIMPLEX, 0.7, color, 2)
        cv2.rectangle(frame, (startX, startY), (endX, endY), color, 2)

    # show the output frame
    cv2.imshow("Face Mask Detector", frame)
    rawCapture.truncate(0)
    key = cv2.waitKey(1) & 0xFF

    # if the `q` key was pressed, break from the loop
    if key == ord("q"):
        break
# do a bit of cleanup
cv2.destroyAllWindows()
#vs.stop()
GPIO.cleanup()

bus.close()
```

Chapter 5
Results and Discussions

Chapter 5. Results and Discussions

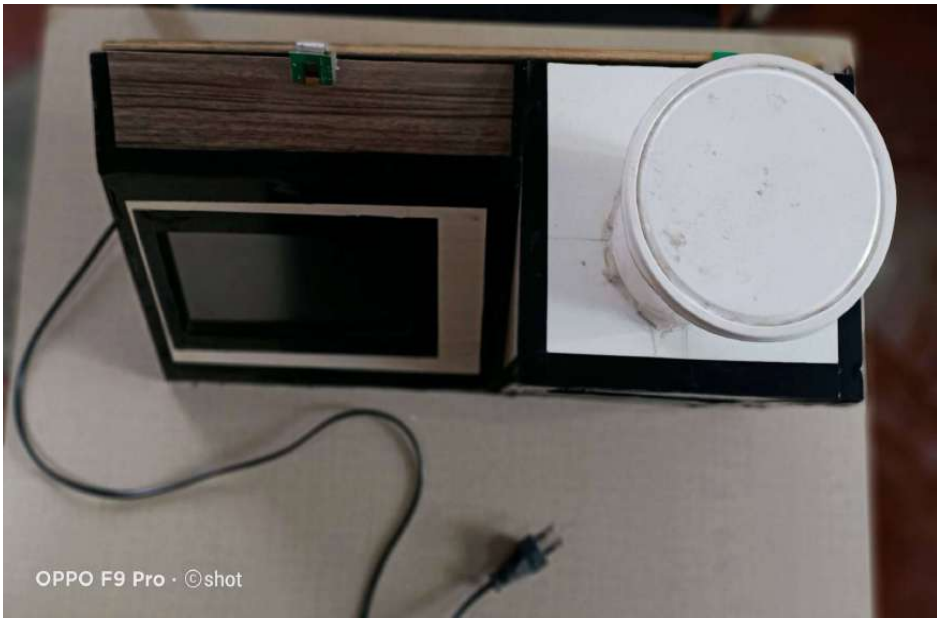
5.1 Result

- This paper presents a system for a smart automated system to reduce the spread of coronavirus by detection of a person who is not wearing a facial mask that is a precautionary measure of COVID-19.
- The motive of the work comes from the people disobeying the rules that are mandatory to stop the spread of coronavirus. The symptoms of covid includes fever, chest conjunction and cold which effects the temperature of the body Therefore temperature checkup stand as an important stage.
- According to WHO, the precautionary measures include face mask social distancing, and proper sanitization Therefore, sanitization Chamber does the work as the last stage of the process that includes Hand sanitization.
- The system contains a face mask detection architecture where a deep learning algorithm is used to detect the mask on the face and also to detect the accurate temperature of the body and proper sanitization.
- To train the model, labeled image data are used where the images were facial images with masks and without a mask. The proposed system detects a face mask with an maximum accuracy.
- And use of infrared sensor to detect the temperature of the person The system proposed in this study will act as a valuable tool to strictly impose the use of a facial mask in public places for all people.
- This project can be deployed in many areas like shopping malls, airports and other crowded areas to avoid the spread of the disease by checking who is following precautions on a daily basis.


F9 Pro · ©shot


OPPO F9 Pro · ©shot


OPPO F9 Pro · ©shot


OPPO F9 Pro · ©shot

5.2 Advantages

- It can be used to isolate the person having a temperature greater than 98.6 degree Fahrenheit
- It helps to reduce the spread of COVID-19
- Hand sanitization is also done using this
- Easy to use
- Simple to handle
- Compact
- Easy to transport

5.3 Disadvantages of existing product

- Currently, the available solutions are only focusing on mask detection
- There is not any system that combined detecting mask and temperature along with hand sanitization
- Available solutions are not more user friendly

5.4 Limitation

- This machine cannot be used for long time it can be used upto 6 hours continuously as this can lead to heating issues in the system

5.5 Cost table

Sr. no.	List of components	Quantity	Cost
1	raspberry pi 4b.	1	5200
2	camera	1	500
3	display	1	1900
4	adapter /kit	1	
5	sd card	1	500
6	fan / heatsink	1	600
7	smps.	1	600
8	ply / wires / tapes/ others	-	2000
9	shouldering iron	1	150
10	ic's / Transformers/ sensors	-	900
11	Relay	1	100
12	IR sensor	1	100

Table 5.5 cost table

Chapter 6
Summary and Conclusions

Chapter 6. Summary and Conclusions

6.1 Summary

This system is mainly focused on three parts. First is mask detection, second is temperature check-up, and third is hand sanitization. These three processes are very important for the prevention of corona-virus. For building the system we are using deep learning and image processing concept. Deep learning has an important role in the field of "Artificial Intelligence" to extract the feature from the image. Firstly, we create the convolutional neural network and build the model using the training dataset so that it will give the correct output for the testing dataset. Once the model is built, the system is ready to prevent the person from coronavirus. when a person comes in front of the camera, the camera captures an image of that person and processes that image, and detects whether the person wears the mask or not. If the mask is detected then massage is displayed on the screen that mask is detected. After that again massage is displayed on the screen please go for the sanitization. Once they put their hands in the sanitization chamber that time infrared temperature detector detects the temperature and displays it on the display screen. If the temperature of that person is between the normal human temperature, then we all that person to go inside then sanitizer is sprinkles on their hands else message is display on the screen is that you are not allow to go inside.

6.2 Application

- For small scales family Gatherings
- For large events in big venue
- for indoor and outdoor stadiums
- It can be used in government and private offices

6.3 Conclusion

- In this model we successfully design a system by using deep learning and image processing which measure temperature of the body and identify a person with or without mask.
- In this model we propose an innovative approach consisting a sanitization with temperature and mask detection system.

- Main motive for making this model is to reduce coronavirus and for that everyone should be wearing a mask, isolate if required and sanitize time to time.
- To train the model labelled image data are used where the images of person with mask and without mask.
- The system is fully automated as well as cost effective.
- Finally, the proposed system achieved 98% accuracy and give accurate result.

6.4 Future scope

- We can use an automated door lock system to restrict the entry of the people who do not satisfy the conditions.
- It will remain as a precaution for years and reduce the panic among peop

References

[1] Mohammad Marufur Rahman, Md. Motaleb Hossen Manik, Md. Milon Islam, Saifuddin Mahmud, Jong-Hoon Kim, “Mask Classification and Head Temperature Detection Combined with Deep Learning Networks”, 2020 2nd International Conference on Broadband Communications, Wireless Sensors and Powering (BCWSP), Computer Science and Engineering, Khulna University of Engineering & Technology, Khulna, Bangladesh, 28-30 Sept. 2020

[2] Isack Farady, Chih-Yang Lin, Amornthep Rojanasarit, Kanatip Prompol, Fityanul Akhyar, “An Automated System to Limit COVID-19 Using Facial Mask Detection in Smart City Network”, 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Yuan Ze University, Taoyuan, Taiwan School of Electrical Engineering Mercu Buana University, Jakarta, Indonesia, 9-12 Sept. 2020.

[3] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition", Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.

[4] M. Z. Islam, M. M. Islam and A. Asraf, "A Combined Deep CNN-LSTM Network for the Detection of Novel Coronavirus (COVID-19) Using X-ray Images", *Informatics in Medicine Unlocked*, vol. 20, pp. 100412, Aug. 2020.

[5] L. J. Muhammad, M. M. Islam, S. S. Usman and S. I. Ayon, "Predictive Data Mining Models for Novel Coronavirus (COVID-19) Infected Patients’ Recovery", *SN Comput. Sci.*, vol. 1, no. 4, pp. 206, Jun. 2020.

[6] L. Calavia, C. Baladrón, J. M. Aguiar, B. Carro and A. Sánchez-Esguevillas, "A Semantic Autonomous Video Surveillance System for Dense Camera Networks in Smart Cities", *Sensors*, vol. 12, no. 8, pp. 10407-10429, Aug. 2012.

[7]https://www.who.int/publications/m/item/who-convened-global-study-of-the-origins-of-sars-cov-2

[8] https://www.worldometers.info/coronavirus/

[9]https://www.who.int/emergencies/diseases/novel-coronavirus-2019/question-and-answers-hub/q-a-detail/coronavirus-dis ease-covid-19

Appendix

We are going to apply for the patent session 2021.

Project group description

Sr. no.	Name	Phone no.	Email id
1	Pragati Lalsare	9552066919	pragatilalsare26@gmail.com
2	Vishal Bramhankar	8975206583	vishalbramhankar18@gmail.com
3	Kunal Tonge	9518597354	kunaltonge555@gmail.com
4	Ganesh Singamshettiwar	9158684039	ganeshsingamshettiwar@gmail.com
5	Shantanu Shende	8446388427	shantanushende44@gmail.com

Table 8: Project Group Information