Face Mask Detection,

Non-Contact Body Temperature Measurement and Automatic Hand Sanitizer Dispenser System

Abstract

During the pandemic, to monitor if the people are following this basic safety principle, a strategy should be adopted. A face mask detector with body temperature system can be implemented. Face mask detection means to identify whether a person is wearing a mask or not. Recognizing the presence of a mask on the face is to detect a human face and detect masks on those faces. Face detection is one of the applications of object detection and can be used in many areas. Another way to find potential Covid-19 cases is via measuring the human body temperature and checking for fever and other likely symptoms. Our project aims at developing a stand-alone system which models the face detection feature and classifies the observed body temperatures. Our project also supports distribution of hand sanitizer and hence can be used as a substitute for appointed personal at places where the guidelines of facemask, temperature and hand sanitization must be followed.

Table of Contents

Table of Figures

Chapter 1

Introduction

The COVID-19 pandemic has reshaped our lives. Everyone is advised to stay at home, avoid crowd and follow the guidelines to stop spreading of the virus. Science is the pursuit of knowledge and understanding and it inevitably changes the way we see the world. Thanks to the tireless efforts of scientists everywhere, we have compressed years of research on the COVID-19 virus into months. This has led to a rapid evolution of policies and recommendations, and not surprisingly some scepticism about the advice of experts.

Health Organization (WHO), Centres for Disease Control and Prevention (CDC),healthcare officials and native institutions from all round the world are urging people to wear face masks, as it's one of the ways to prevent the transmission of the virus as not wearing a face mask can increase the outbreak of coronavirus in world.

Various measures are taken by the government and the people to bring this spread of virus under control. We need to adopt few habits such as wearing a mask or face covering whenever we are in a public space, regularly sanitizing our hands or wash hands if possible.

The main aim of this project is to develop an efficient face mask detection system to maintain the rules during these tough times of Covid-19 outbreak and to help people protect themselves from the virus.

We propose a dynamic Computer Vision based automated solution system focused on the real-time face monitoring of people to detect both face masks and body temperature in public places to detect face mask protocol violations through an integrated Webcam and to monitor body temperature with the help of temperature sensor. An automatic hand wash dispenser which dispenses a specific amount of alcohol based hand sanitizer on the hands when placed under a specific distance of the Infrared sensor module. In this project we have embedded this model with Raspberry pi 3 to perform real-time mask detection, where ML (Machine Learning) , OpenCV (Open Source Computer Vision Library), TensorFlow, Deep Learning and Computer Vision to recognize face masks. This model can be utilized for security purposes since it is a very resource efficient to deploy.

"The first rule of any technology used in a business is that automation applied to an efficient operation will magnify the efficiency. The second is that automation applied to an inefficient operation will magnify the inefficiency." - Bill Gates

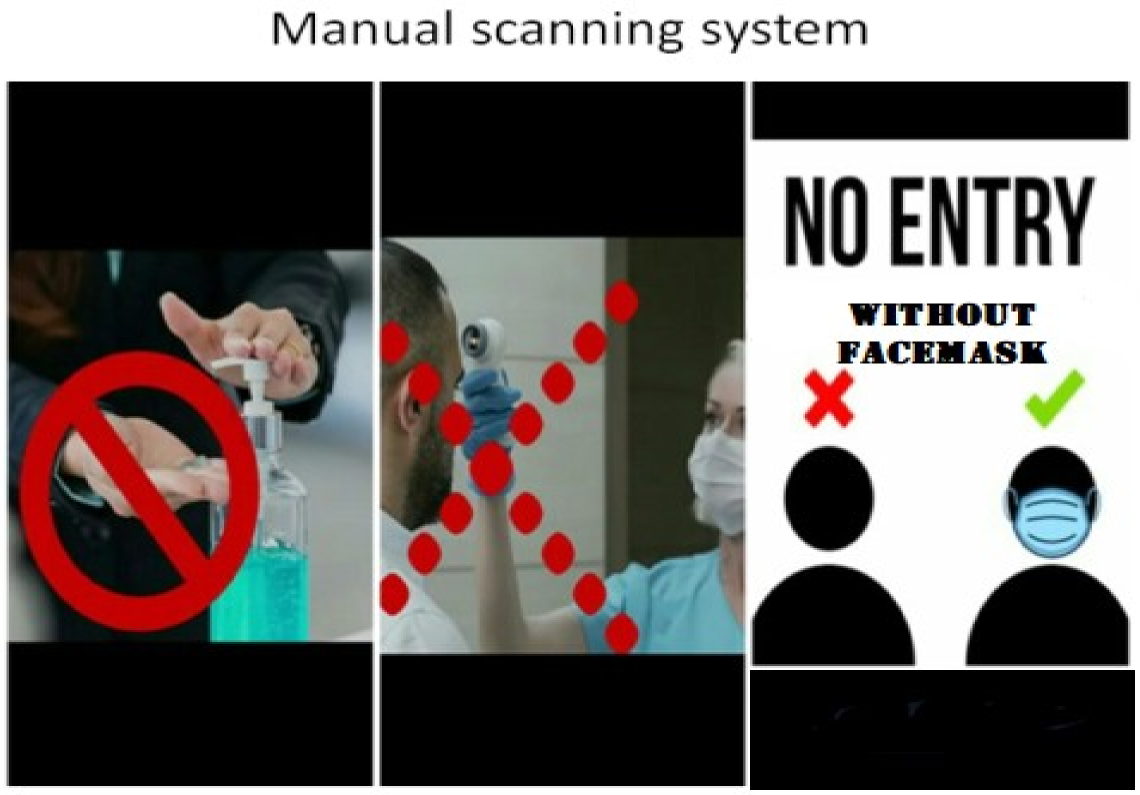


Fig 1. Current Manual scanning system

Chapter 2

Problem Statement and Objectives

2.1 Problem Statement

To limit the spread of the Covid-19 respiratory disease, mandatory rules of wearing face-mask and checking human body temperature are now becoming common in public settings around the world. Public service providers require customers to wear face-masks in accordance with predefined rules when using public services. People are doing all they can for their own and the society's safety. Face masks are one of the personal protective equipment. However, there are at times where some people are found to visit public spaces or using public services without following the necessary guidelines and are seen violating the rules. For this, many service places' authorities appoint a separate person who would manually check if the person entering the building is wearing a mask and the body temperature is in the normal range. This introduces human interaction and additional cost. If the person appointed is unknowingly a career of the disease, then the purpose fails.

2.2 Objectives

Our project's main objective is to provide a stand-alone system which models the face detection feature and classifies the observed body temperatures. Our project also supports distribution of hand sanitizer and hence can be used as a substitute for appointed personal at places where the guidelines of facemask, temperature and hand sanitization are must.

- Provide a stand-alone system which would give real time results. Detect whether a person is wearing face mask and display a 'Yes' or 'No' with measured body temperature.
- Provide data collected to the control authority wirelessly.
- Remove the need to appoint a team for checking on people.
- Zero contact testing and dispensing of alcohol based hand sanitizer.
- Record the real time data for analysis purpose.
- To provide a real world prototype that can be applied and utilized. We want to be able to show a model that can provide a base for the idea and allow the flexibility for further advancement.

Chapter 3

Work Done

3.1 Methodology

The proposed this project is divided into *three* segments.

A) *Automatic Sanitizer Dispenser*
B) *Contact less temperature measurement*
C) *Face Mask Detection using Raspberry pi 3*

The entire project is a stand-alone system which requires a power source to turn on the Raspberry Pi 3 board and a working internet connection. The entire project is placed on a stand. The webcam and display is attached to the upper side to capture the face data and display it. The Raspberry Pi is placed below it which is connected to the webcam and the display. The Pi is also connected to the temperature sensor which returns the body temperature reading measured. The IR sensor placed below the board senses if the hand is placed and squirts the sanitizing liquid on the hand. The hand sanitizer block is instantaneous whereas the temperature and face mask detection requires processing. Hence, the former is kept independent and the other blocks are bundled to have the output on a single screen i.e. Face mask detected output and measured body temperature output.

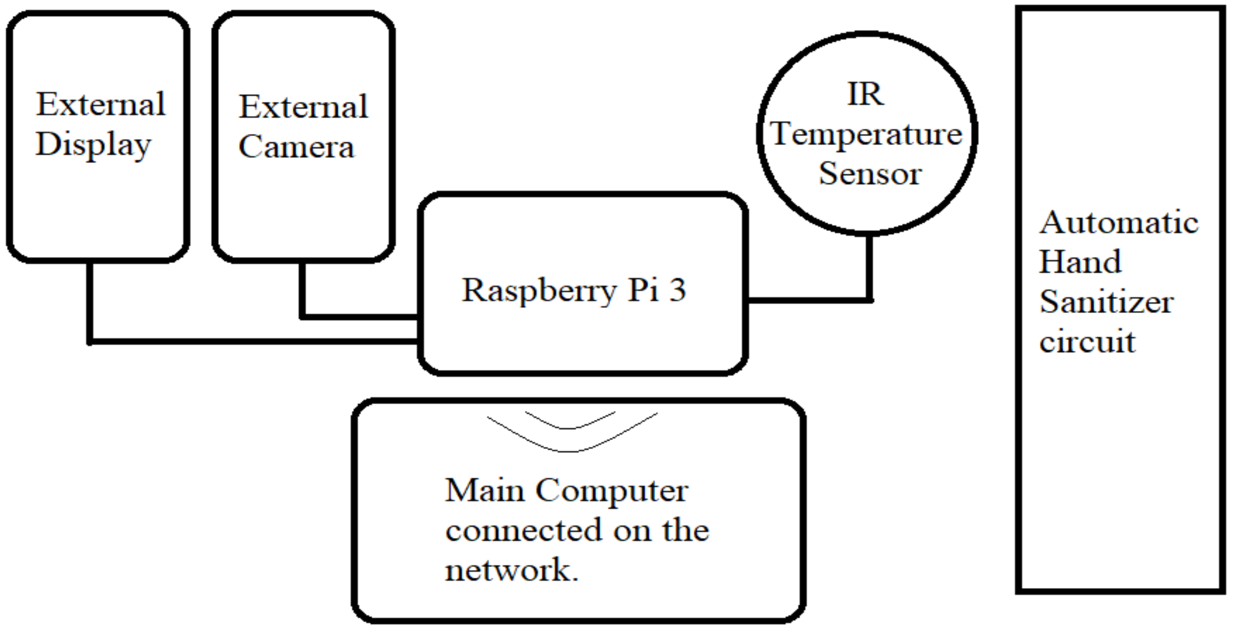

External Display
External Camera
IR Temperature Sensor
Automatic Hand Sanitizer circuit
Raspberry Pi 3
Main Computer connected on the network.

Fig. Block Diagram for the Project.

3.2 Design and working

Part A) Automatic Hand Sanitizer Dispenser

Circuit Design

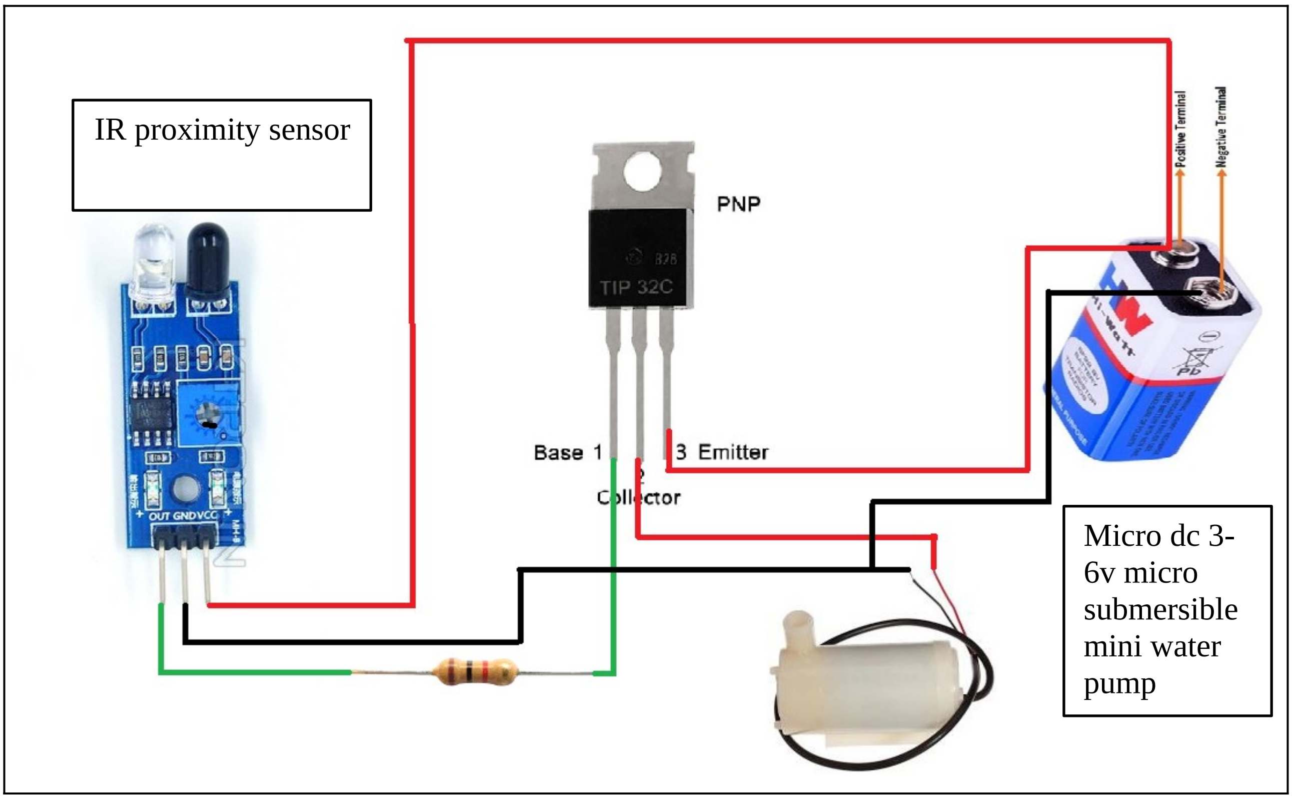


Fig 3. Automatic Hand Sanitizer Dispenser

After the above connections are made, the pump with plastic tube attached can be placed inside the plastic jar while the electronic components can be taped on top of the lid of the Jar.

Two holes are made on top of the jar such that plastic pipe and the electrical wires can be brought out of the jar. Make sure all exposed wires are covered with electrical tape so as to avoid short circuits. A nozzle can also be added to the end of the plastic tube connected to the pump, to restrict the flow of the sanitizer. Once all the connections and set up is ready, the main power can be given and you should see the IR having their power LEDs glow on.

Everything went smoothly and correctly, when powered on, as soon as the IR detects an object, signal from the IR sensor will turn on the transistor that will turn on the water pump motor.

In case it doesn’t, verify the connections and try again. Once ensured that the system works as it is supposed to, the jar can be filled with sanitizer and the set up can be deployed wherever necessary.

Working

As soon as a person places their hand in front of the IR sensor, the sensor will detect it and will give an output signal to the data pin.

The data pin is connected to the transistor signal (base) terminal.

Signal received will turn on the transistor that will turn on sanitizer pump.

Until the hand is removed, the pump will remain on continuously dispensing the sanitizer.

B) Contact less temperature measurement

MLX90614 is an IR Temperature sensor for non-contact temperature measurements. It has an I2C Interface to communicate with Raspberry pi model 3 B. This temperature sensor can measure the temperature without touch the object. It has 0.5 degree Celsius over a wide range of temperature.

Circuit Design

The figure below shows the circuit connection of the MLX90614 IR temperature sensor to the Raspberry pi module.It is I2C supported and is connected with the I2C bus of the Raspberry pi through pin3 and pin5. Pin3 is the SDA and Pin5 is the SCL.

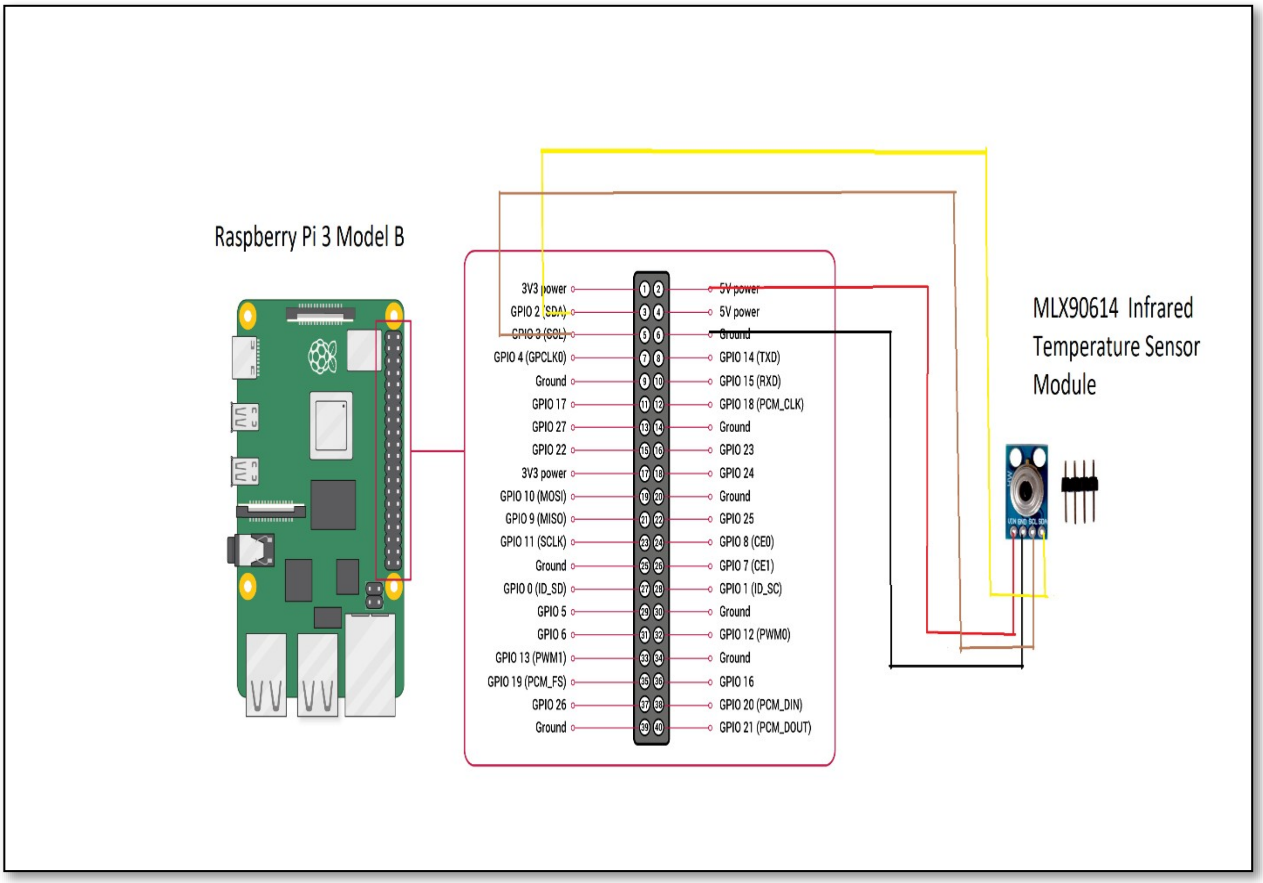


Fig 4. Interfacing MLX90614 Temperature Sensor With Raspberry Pi 3

Working

MLX90614 is a contactless IR temperature sensor that interfaced with a raspberry pi 3 model B and used to detect an object's temperature. The operating distance of this specific IR sensor is 2cm-5cm. The detection software program utilizes the python library smbus2 and the locally stored folder mlx90614 to read the object temperature from the IR sensor.

Steps for detection

- Person places their hand in front of the MLX90614 IR temperature sensor.
- When the hand is in range of the temperature sensor, it will check their temperature of the hand.
- The detected temperature is displayed on output display screen along with the face mask detected output. (Part C explained below)

C) Face Mask Detection using Raspberry pi 3

Working

The code of the convolutional neural network created using machine learning is written in Python language. The CNN or the model so created must be done only once and can be reused across various other such stand alone projects. Project requires a wireless connection to the internet via which the authority computer and the project computer (Raspberry pi 3) can exchange data. The output for the face mask is rather in probabilities of wearing and not wearing mask as the model was trained with only 7500 test images. The more test images ,the greater the accuracy.

The objective of the model is to classify facial photos, where it will divide the received data into two classes to answer the question : If the test person is wearing a mask or not.

The webcam attached to the stand captures the video and sends the video stream to the computer connected on the wireless network. The computer fetches the stream and captures the temperature data from the MLX90614 temperature sensor and runs it through the trained model. The output of the code is presented to the Raspberry pi which displays it on the display attached to the stand itself.

The procedure is divided into two phases. Phase 1 would be to train a face mask detector using the dataset. This procedure would be done only once and a model can be extracted. In phase 2, we would apply this model in our code. Our project will be using the trained model which was done in phase 1 in phase 2. This application of phase 2 would be repetitive. Every time it receives an input video stream it would try to detect human faces and apply the face mask detector.

Algorithm

Phase 1:

- Load the data set in the environment (set of separated pictures of human faces wearing masks and no masks)
- Separate training data and test data and import all the required libraries.
- Use *tf.io.read_file(path)* to create a batch of data before training the model, so that the training process can run faster.
- Build the neural network using Keras.
- Train the model. Use Early Stopping so that the training process can be stopped automatically if the model doesn't improve for 3 epcohs to avoid overfitting.
- Save the model to disk.

Phase 2:

- Create a new file and import all required libraries and the model we created in Phase 1.
- Create a function which would print Mask detection result and measured body temperature on every frame of the video output.
- Import the video stream from the webcam and import the temperature sensor data.
- Run the input stream through the model and call the function created to print results on the output video frame.
- Repeat above steps till the program is terminated manually.
- Release all resources and stop.

Pictorial form for the phases of the Fask Mask Detector code.

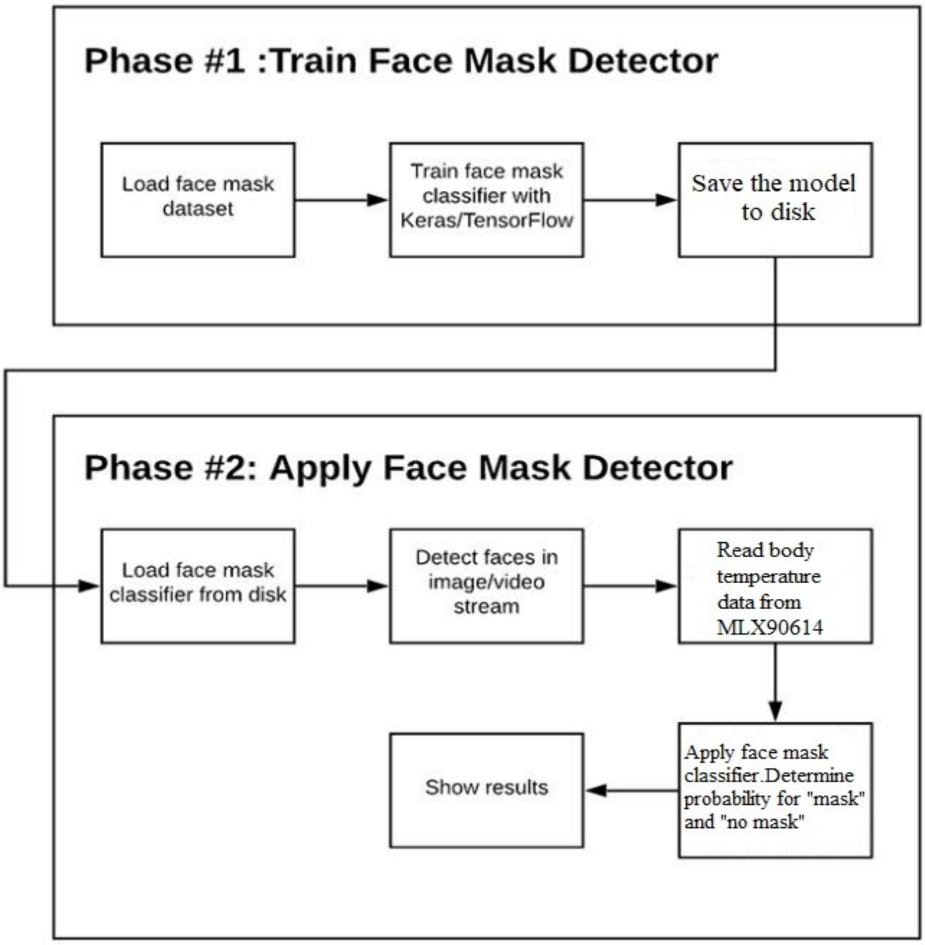

Phase #1 :Train Face Mask Detector
Load face mask dataset
Train face mask classifier with Keras/TensorFlow
Save the model to disk
Phase #2: Apply Face Mask Detector
Load face mask classifier from disk
Detect faces in image/video stream
Read body temperature data from MLX90614
Apply face mask classifier.Determine probability for "mask" and "no mask"
Show results

Fig 5. Face Mask Detector phase diagram

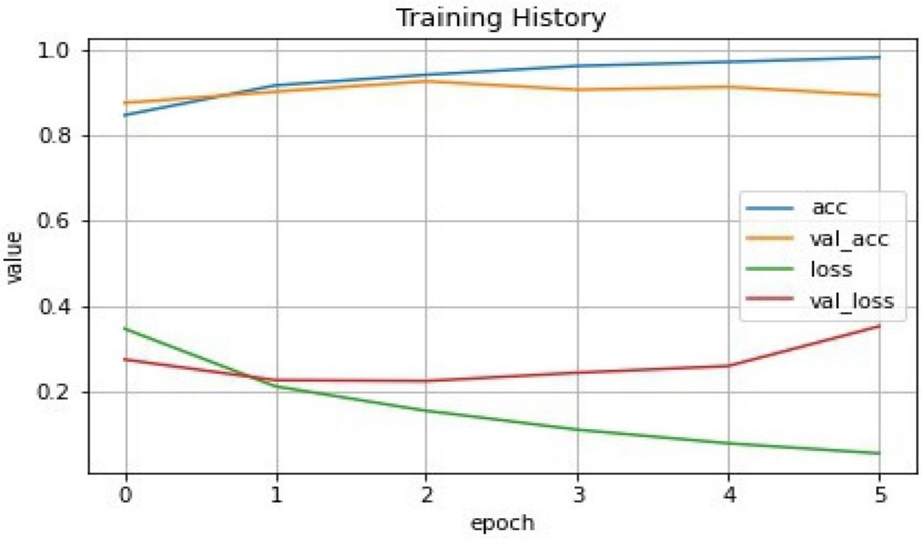


Fig 6. Training process graph in 5 epochs

3.3 Code and libraries used

Libraries

- OpenCV

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 products. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions.

- Numpy

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

- Urllib3

urllib3 is a powerful, *user-friendly*HTTP client for Python. It supports Thread safety, Connection pooling, Client-side SSL/TLS verification, File uploads with multipart encoding, Helpers for retrying requests and dealing with HTTP redirects, Proxy support for HTTP and SOCKS.

- Tensorflow and Keras

TensorFlow is an open-source platform for machine learning and a symbolic math library that is used for machine learning applications.It is an Open Source Neural Network library that runs on top of Theano or Tensorflow. It is designed to be fast and easy for the user to use. It is a useful library to construct any deep learning algorithm of whatever choice we want.

Working Code for Face Mask Detection

```
import cv2
import numpy as np
import urllib3
import tensorflow as tf
keras = tf.keras
model = keras.models.load_model('face_mask_ai.h5')
def blit(text,frame,color,position=(100, 100)):
    cv2.putText(frame, text, position, cv2.FONT_HERSHEY_SIMPLEX, 1,
color, 4, cv2.LINE_4)
http = urllib3.PoolManager()
#cam = 'http://192.168.1.40:8080/video'  #stream ip changes with network
cap = cv2.VideoCapture(0)
#cap2 = cv2.VideoCapture('bb.mp4')
fps = cap.get(cv2.CAP_PROP_FPS)
size = (int(cap.get(cv2.CAP_PROP_FRAME_WIDTH)),
        int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT)))
print(f'Video fps : {fps}')
print(f'Video size : {size}')
while(cap.isOpened()):
    res, frame = cap.read()
    #res,frame1 = cap2.read()
    t = http.request('GET','http://192.168.1.55:8080') #ip changes with
network
    tempB = t.data
    temp = tempB.decode('utf-8')
    img = cv2.resize(frame, (224,224), interpolation = cv2.INTER_AREA)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    prediction = model.predict(np.expand_dims(img,axis=0))
    blit(f'Prediction Probability',frame,(255,255,255),(20,50))
    blit(f'Wearing mask : {prediction[0][0]}',frame,(0, 255, 0),(20,150))
    blit(f'No mask : {prediction[0][1]}',frame,(0, 0, 255),(20,250))
    blit(f'Temperature : {temp}',frame,(255,0,0),(20,350))
    cv2.imshow('output',frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()              #release resources
cv2.destroyAllWindows()
```

Chapter 4

Components used

1) Raspberry Pi 3 Model B

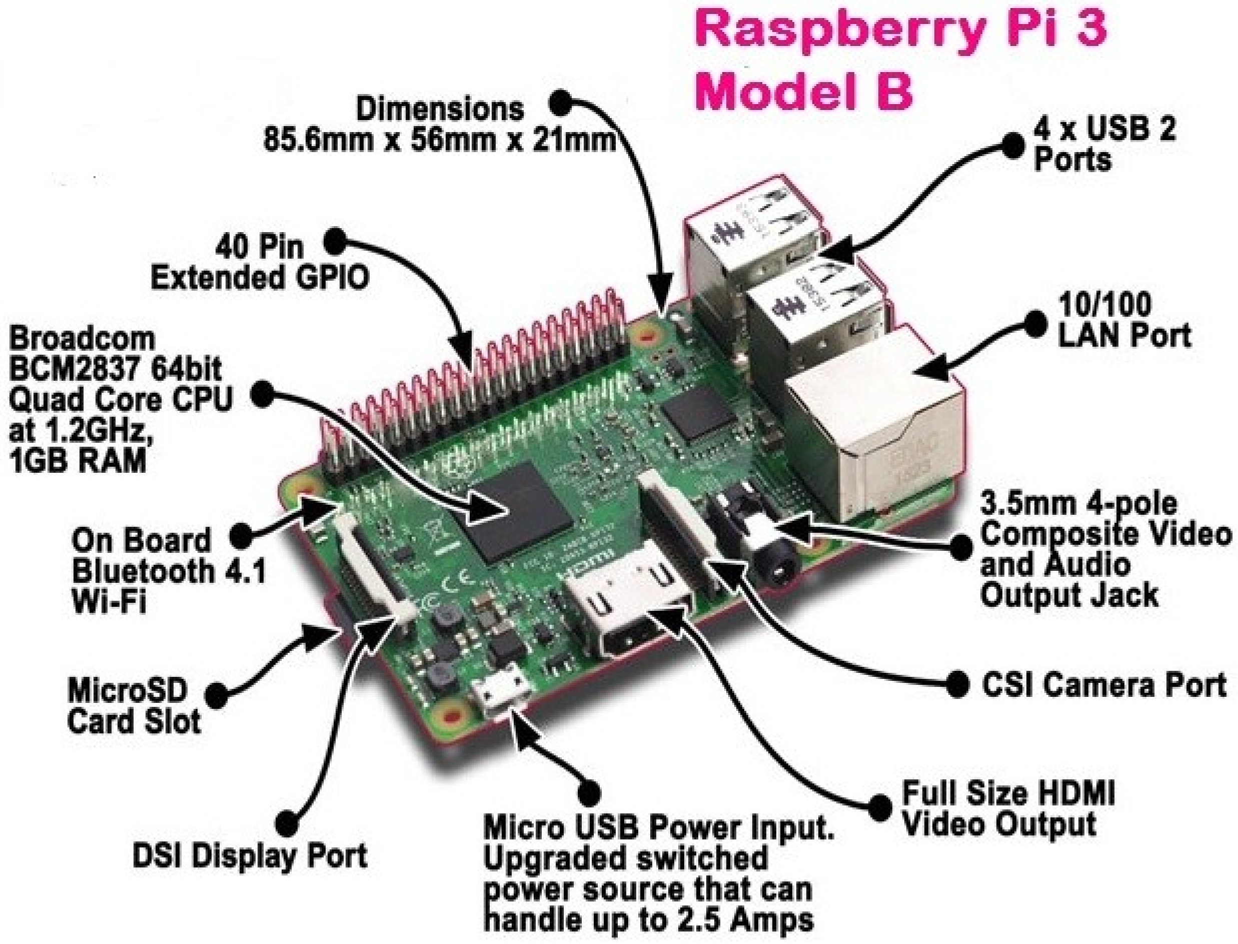


Fig 7. Raspberry Pi 3 Model B

The 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. It 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.

Specification:

- Quad Core 1.2GHz Broadcom BCM2837 64bit CPU.
- 1GB RAM.
- BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board.
- 100 Base Ethernet.
- 40-pin extended GPIO.
- 4 USB 2 ports.
- 4 Pole stereo output and composite video port.

2) MLX90614 Non-Contact IR Temperature Sensor

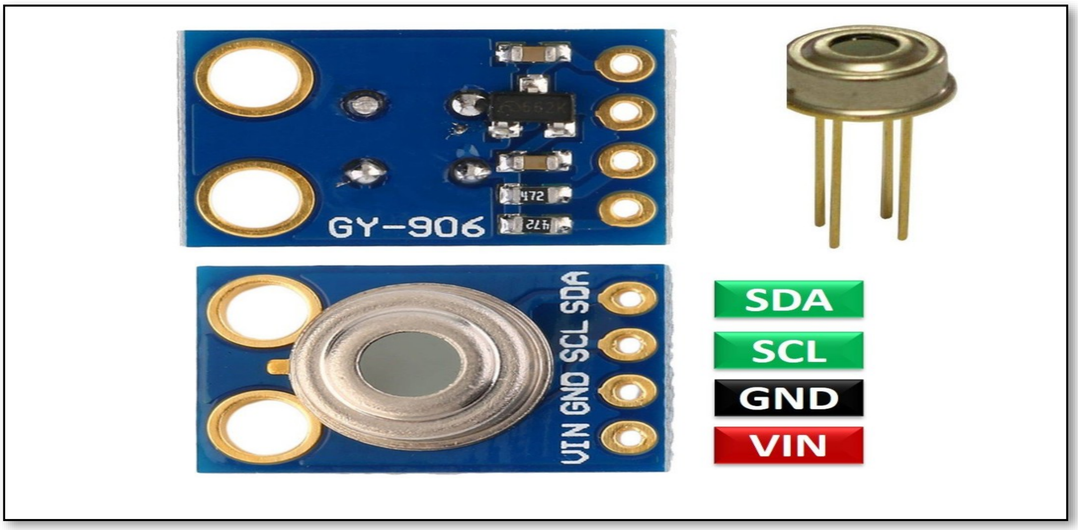


Fig 8. MLX90614 Non-Contact IR Temperature Sensor

The MLX90614 is a Contactless Infrared (IR) Digital Temperature Sensor that can be used to measure the temperature of a particular object ranging from -70° C to 382.2°C. The sensor uses IR rays to measure the temperature of the object without any physical contact and communicates to the microcontroller using the I2C protocol. It can measure the temperature of an object without any physical contact with it. This is made possible with a law called Stefan-Boltzmann Law, which states that all objects and living beings emit IR Energy and the intensity of this emitted IR energy will be directly proportional to the temperature of that object or living being. So the MLX90614 sensor calculates the temperature of an object by measuring the amount of IR energy emitted from it.

Specification:

- Operating Voltage: 3.6V to 5V (available in 3V and 5V version)
- Supply Current: 1.5mA
- Object Temperature Range: -70° C to 382.2°C
- Ambient Temperature Range: -40° C to 125°C
- Accuracy: 0.02°C
- Field of View: 80°
- Distance between object and sensor: 2cm-5cm (approx.)

3)IR Proximity Sensor

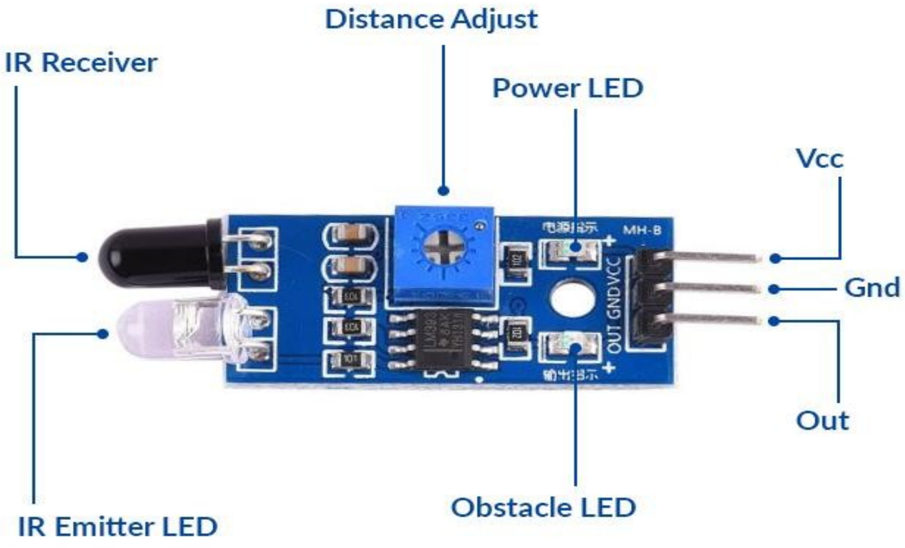


Fig 9. IR Proximity Sensor

IR Sensor module has great adaptive capability of the ambient light, having a pair of infrared transmitter and the receiver tube, the infrared emitting tube to emit a certain frequency, encounters an obstacle detection direction (reflecting surface), infrared reflected back to the receiver tube receiving, after a comparator circuit processing, the green LED lights up, while the signal output will output digital signal (a low-level signal), through the potentiometer knob to adjust the detection distance, the effective distance range 2 ~ 10cm working voltage of 3.3V-5V. The detection range of the sensor can be adjusted by the potentiometer, with little interference, easy to assemble, easy to use features, can be widely used robot obstacle avoidance, obstacle avoidance car assembly line count and black-and-white line tracking and many other occasions.

Specification:

- I/O pins are 5V and 3.3V compliant.
- Range: Up to 20cm.
- Adjustable Sensing range.
- Built-in Ambient Light Sensor.
- 20mA supply current

4) MICRO DC 3-6V MICRO SUBMERSIBLE PUMP MINI WATER PUMP

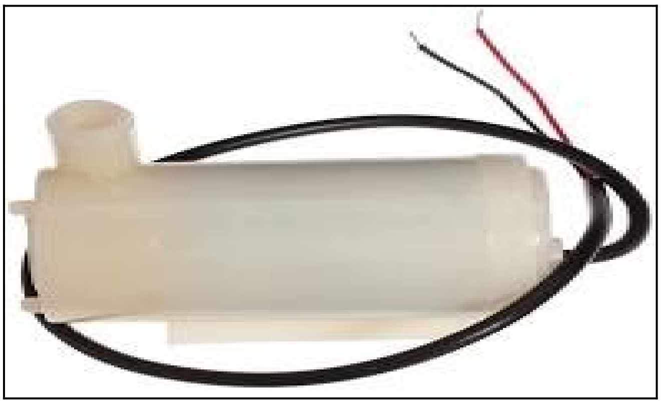


Fig 10. MICRO DC 3-6V MICRO SUBMERSIBLE PUMP MINI WATER PUMP

This is a low cost mini submersible type water pump that works on 3 - 6V DC supply which is used for dispersing liquid sanitizer. It is extremely simple and easy to use. Just immerse the pump in water, connect a suitable pipe to the outlet and power the motor with 3 - 6V to start pumping sanitizer. This motor is small, compact and light. It can be controlled from a micro controller using our DC Motor Drivers or one of the Relay Boards. You may use our 5V Switch Mode Power Supply Adapter to run this pump. You may also use 6V Solar Panel to run the pump with appropriate a 6V voltage regulator.

Specification:

- Operating Voltage : 3 ~ 6V.
- Operating Current : 130 ~ 220mA.
- Flow Rate : 80 ~ 120 L/H.
- Maximum Lift : 40 ~ 110 mm.
- Outlet Outside Diameter: 7.5 mm.
- Outlet Inside Diameter: 5 m

5) TIP32C PNP TRANSISTOR

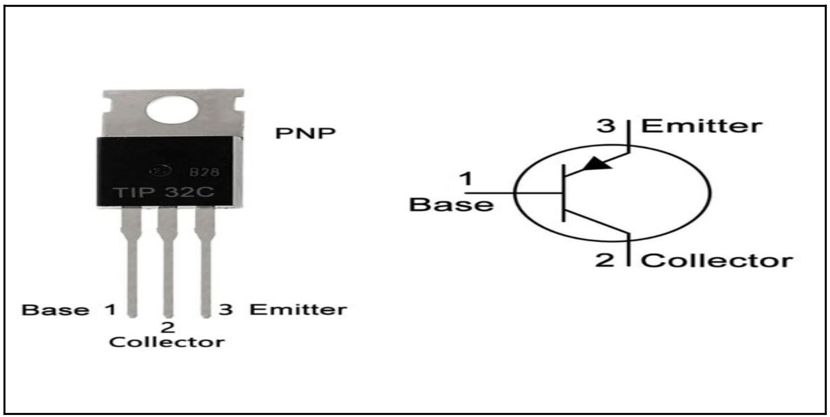


Fig 11. TIP32C PNP TRANSISTOR

The TIP32C is a silicon Epitaxial-base PNP power transistor in Jedec TO-220 plastic package. It is intended for use in medium power linear and switching applications. The complementary NPN type is TIP31C. The PNP transistor turns on when a small current flows through the base. The direction of current in PNP transistor is from the emitter to collector. The letter of the PNP transistor indicates the voltage requires by the emitter, collector and the base of the transistor.

Specification:

- Package of – TO-220 and PNP type
- Collector & Emitter Voltage: -100 Volt
- Collector-Base Voltage: -100 Volt
- Emitter-Base Voltage: -5 Volt
- Maximum Collector Current: -3 A
- Collector Dissipation Max – 40 Watts
- Maximum DC Current Gain (h_{fe}) – 10 – 50

6) Web camera module

Fig 12. ZEBRONICS web camera module

A 5MP High quality lens web camera module is used for this setup. Any USB webcam can use with Raspberry pi 3. Face mask detection system uses visible streams from the camera to detect whether people wearing mask or not.

Specification:

- Interface : USB
- Image sensor :CMOS
- 5P High Quality Lens with Video Resolution: 1920*1080 (30fps)

7) TFT LCD DISPLAY

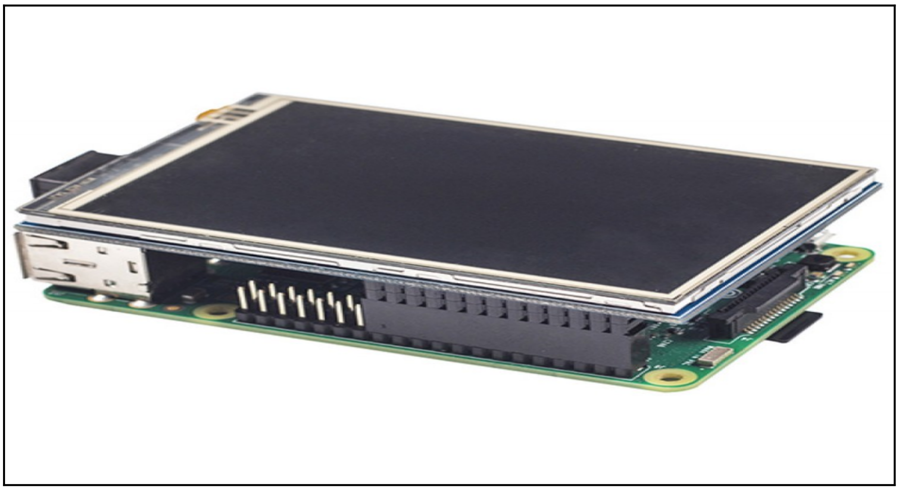


Fig13 . TFT LCD DISPLAY

TFT LCD Display for Raspberry pi has 5" screen size. It is used to display camera feed from the Raspberry pi to display face mask and body temperature readings. Any compatible display can be used in this project.

Specification:

- Resolution -800*400 Pixel
- Touch screen type – Resistive
- LCD type- TFT

8) Jumper Wires

Fig 14. Jumper wires male to female

Jumper wires are used to connect component to each other on the breadboard or other prototypes, internally or with other equipment or components ,without soldering.

Chapter 5

Result

We have realised the following blocks.

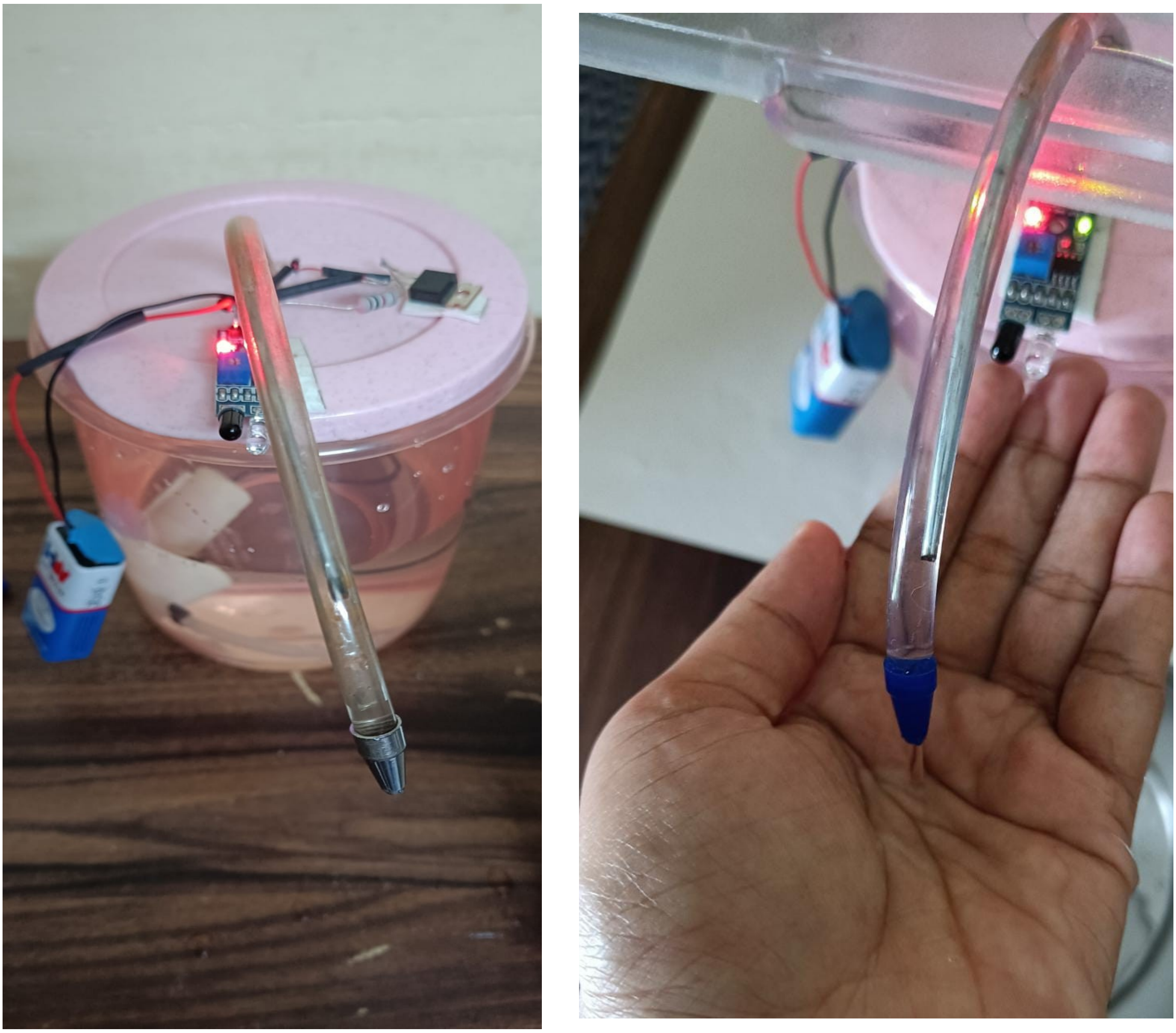
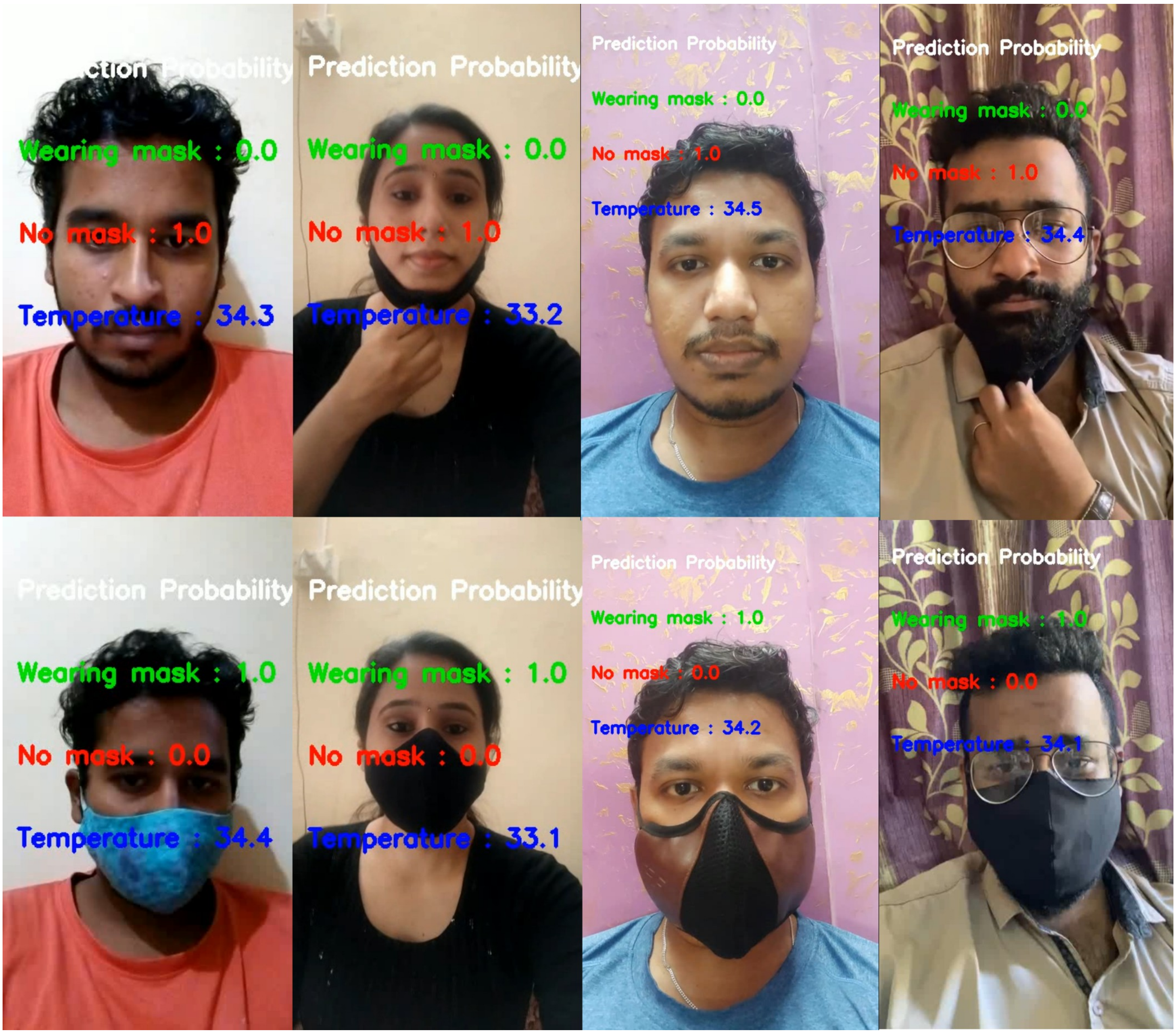


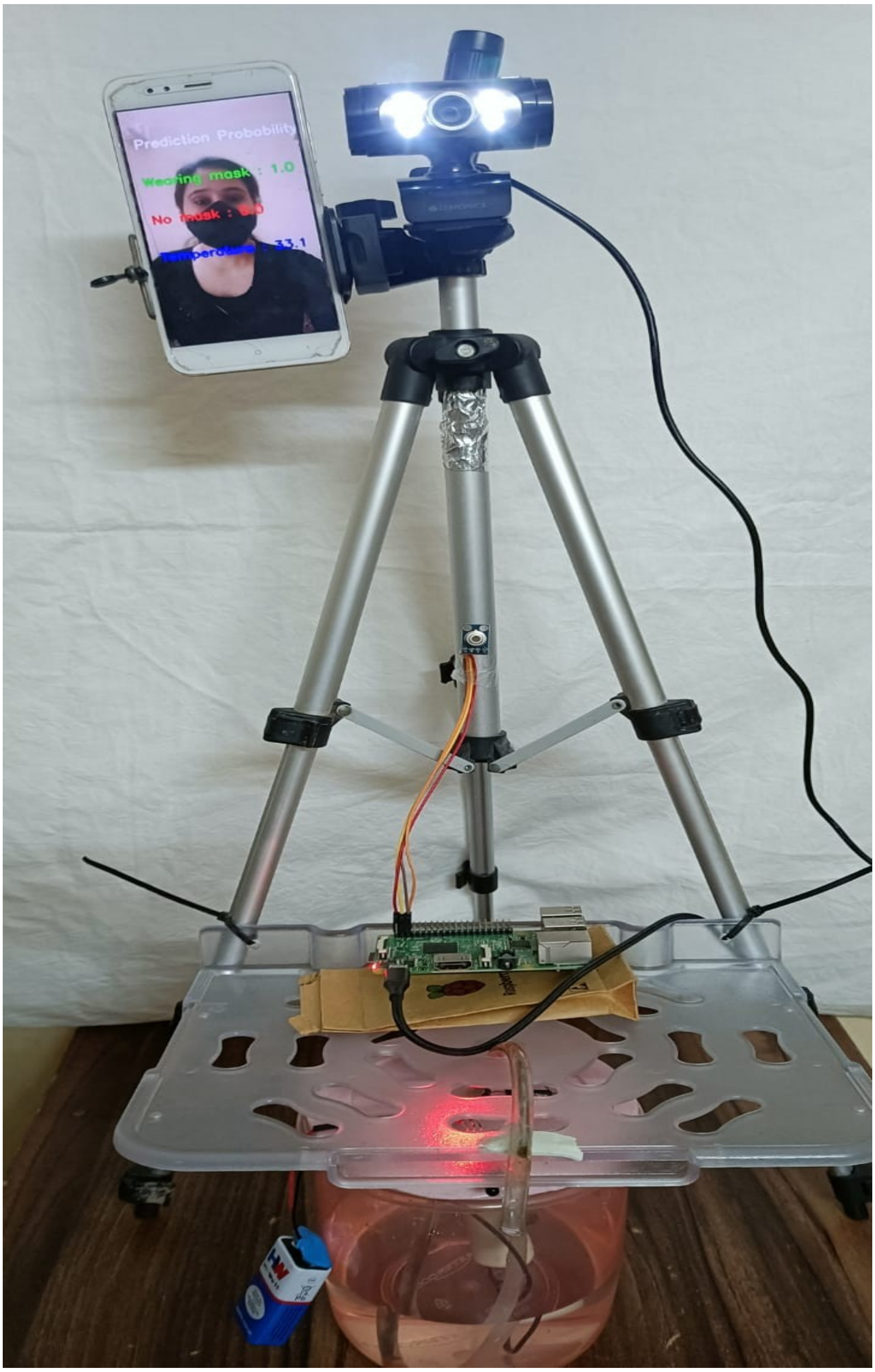
Fig 15. Automatic Hand Sanitizer block



[Face mask detector output screen picture]

Fig 16. Face mask detector and temperature reading output on display screen

The font in the above depicted screen outputs is different since different cameras were used to for detection testing. However, once the setup is installed with a camera with constant resolution and frame size, the font in the output screen will cover the screen uniformly.



[Overall project hardware picture]
Fig 17. Entire project

The display, project stand height, raspberry pi board base and proximity sensor height can be adjusted according to the user needs. The display can be used either in portrait mode or landscape mode.

Chapter 6

Cost Estimation

Sr. no.	Component	Cost
1	Raspberry Pi 3 B	2999.00
2	Webcam	829.00
3	Display	2695.00
4	Mlx90614 Temperature Sensor	1000.00
5	IR Proximity Sensor	70.00
6	DC 3-6V Micro Submersible mini water pump	20.0
7	TIP32C PNP Transistor	40.0
8	Jumper wires (male to female) and sleeves	45.0
	TOTAL	**7698.00**

Chapter 7

Conclusion

Our project has been concluded with the required output and meets the conditions aforementioned. It satisfies our project objectives. The main purpose of this project to automate the manual work of the Covid-19 protocols of sanitizing hands, checking the temperature and ensuring if people wear mask is satisfied. This system will successfully help to reduce man power while also provide an extra layer of protection against the spread of infection. The model uses a real – time deep learning system using Raspberry pi to detect face mask and temperature. Automatic hand sanitizer dispenser can facilitate the use of the hand sanitizer without touching the hand sanitizer nozzle. By using this project, people can feel more secure and prevent from getting affected from the life-threatening situations in the pandemic.

Chapter 8

Limitations and Future aspects

Limitation

- The system would just show and store data. The necessary action should be taken manually.
- Does not have an automatic sanitizer refill system. It needs to be refilled depending on the use.
- Provide masks for people who do not have one at that the moment.
- Alert police when necessary guidelines are not being followed.
- Request people to maintain social distance.

Future aspects

- A solenoid door lock mechanism can be interfaced which unlocks only when the criteria is fulfilled.
- An emergency alert system for anyone who does not obey the rules and need immediate action.
- Can add a speaker and a TTS system to ensure proper social distancing is maintained.
- A mask vending mechanism for the people who need a mask on urgent basis.

Chapter 9

References

- Covid-19 facemask detection with deep learning and computer vision Vinitha.V1, Velantina.V2, International Research Journal of Engineering and Technology
- A. Das, M. Wasif Ansari and R. Basak, "Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV," 2020 IEEE 17th India Council International Conference (INDICON), New Delhi, India, 2020, pp. 1-5, doi: 10.1109/INDICON49873.2020.9342585.
- Automatic Facial Mask Detection using Deep Learning, Smaranjit Ghose, Suhrid Datta, Department of Computer Science and Engineering, SRM IST, Chennai, India, Mukt Shabd Journal.
- Wei Bu, Jiangjinn Xiao, Chuanhong Zhou, Minmin yang and Chengbin “A Cascade Framework for Masked Face Detection”. 2017, IEEE 8th International Conference on CTS & RAM Ningbo.
- Kaihan Lin, Xiaoyong Liu, Huimin “Face Detection and Segmentation based on Improved Mask R-CNN”, 2017.
- ArtiMahore, MeenakshiTripathi “Detection of 3D Mask in 2D Face Recognition System Using DWT and LBP”. 2018, IEEE 3rd International Conference on Communication and Information System.
- https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/
- https://components101.com/sensors/melexis-mlx90614-contact-less-ir-temperature-sensor
- https://opencv.org/about/
- https://numpy.org/doc/stable/user/whatisnumpy.html
- https://pypi.org/project/urllib3/
- https://keras.io/
- https://www.kaggle.com/omkargurav/face-mask-dataset