Covid 19 Safety Detection in Riders using Machine Learning Techniques

VANI VASUDEVAN, B.V SHRUTI, V. LAXMI DEEPAK, C. VINEETH, N. NAGA JAYANTH, K. MADHAN,

Department of computer science and engineering, NMIT, Bengaluru, India.
Department of computer science and engineering, NMIT, Bengaluru, India.
Department of computer science and engineering, NMIT, Bengaluru, India.
vani.v@nmit.ac.in , shrutibv@nmit.ac.in , deepakong28@gmail.com, chitravineeth64314@gmail.com , jayanthnallagatla7@gmail.com , kothamadhan001@gmail.com ,

Abstract. The main objective of this paper is to detect the person with or without a helmet, with or without a mask, and extract number plate details. The COVID-19 pandemic has bought many changes in our day-to-day life. To protect and save ourselves from getting infected by COVID-19, we need to follow certain precautions. There are some precautions to follow to safeguard ourselves. Use a face mask, maintain social distancing, use sanitizer, etc. One such important and basic safety precaution is wearing a face mask to protect ourselves. Viruses spread from one person to another person with respiratory droplets. So, when a virus in-person talks, shouts, sneezes, or coughs, the droplets enter the mouth or nose of another person who is nearby. So, wearing a face mask over the mouth and nose acts as a barrier between the persons to prevent the transmission of respiratory droplets. But so many people were ignoring wearing a mask in public areas. It is good to have a software Or system to identify people who are not wearing masks (without masks) in public places. Here, our project aims to detect people whoever not wearing a face mask on a video by developing an AI and machine learning algorithm that identifies the persons without a face mask. Machine learning is a technique of Artificial Intelligence used to extract useful and valuable data from large databases. We collected the training sets manually by capturing people's faces, from Kaggle, from google where people consist of faces with and without a face mask, and from Kaggle, the datasets contain the images of people with and without a mask.

1. INTRODUCTION

Over the past few years, face detection has been an increasing and trending topic in CV (computer vision). It has been used in cameras that make sure faces are focused before taking a picture, while uploading on social media sites people will get tagged automatically. Face detection has been used very widely for security purposes. ATMs with facial detection and recognition software have been introduced. Artificial intelligence and Machine learning can do functions like learning and problem solving, though AI, and computers can simulate reasoning using logic and math. Machine learning helps computers to learn without any instruction. This makes the computers learn and improve based on experience, it will keep improving until the model's accuracy is high enough to finish the task. Object recognition and pattern learning are the types of tasks for the computer vision (CV) technique. Object detection and image classification can be done by Object recognition.

2. RELATED WORK

The face recognition technique was proposed in 1960 by Woody Bledsoe, Helen Chan, and Charles Bisson. They worked on computers to detect human faces. In the same way, Face mask detection was discovered by Natraj Mishra faculty in the department of mechanical engineering [UPSC member]. He developed using machine learning techniques i.e., supervised learning technique and PC web camera to identify and detect the people whether they are wearing a mask or not. This has been used in various platforms like colleges, schools, airports, hospitals, and offices. In which there is a chance of increasing the covid-19 virus, the spreading of the covid-19 virus is high.[3]

Many old studies are done on older CNN with less accuracy and less performance. This system has image processing and frame separation with advanced machine-learning models. For YOLO like other region proposal classification networks (fast RCNN) which perform detection on various region proposals networks and end up performing prediction multiple times for various regions in an image, YOLO architecture is more like FCNN (fully convolutional neural network) and passes the image once through the FCNN and output is prediction.[7]

Object detection is one of the leading technologies in the world. It is an application of OpenCV. Using object detection, we will be able to detect multiple objects in a single image. It is easy for us as humans to recognize objects in images, however, things become difficult for computers. Object detection applications include unlocking the mobile phone using face identification, self-driving cars, etc. In the same way, face masks and vehicle number plate detection come.[12]

Over the years, number plate detection was done in many ways. In [13], ANPR (Automatic number plate recognition) was done using OCR (Optical Character Recognition) and Binary Image Processing. In addition to that, ANPR system was also done by OpenCV library along with python language used for image processing techniques using py tesseract. TESSERACT is used as an Optical Character Recognition (OCR).[14]

3. PROPOSED METHODOLOGY

The objective of this segment is to provide detailed information about the proposed system. Our system was developed using the **COCO** (Common Objects in Context) dataset which uses a JSON format that provides information about each dataset and all the images within it. Face mask & helmet classifiers were done by YOLO (You Only Look Once) v4 weights using the Caffe model (pre-trained model). Extracting the characters from the numberplate using **API** (Application programming interface) named **plate recognizer** and sending alert mail to the rider's registered mail. This is what makes us different from existing systems.

4. DESIGN

Flow Diagram

The high-level design of the flow chart represents the working of the system from start to end with all conditions and possibilities. The flow diagram is shown in Figure 1

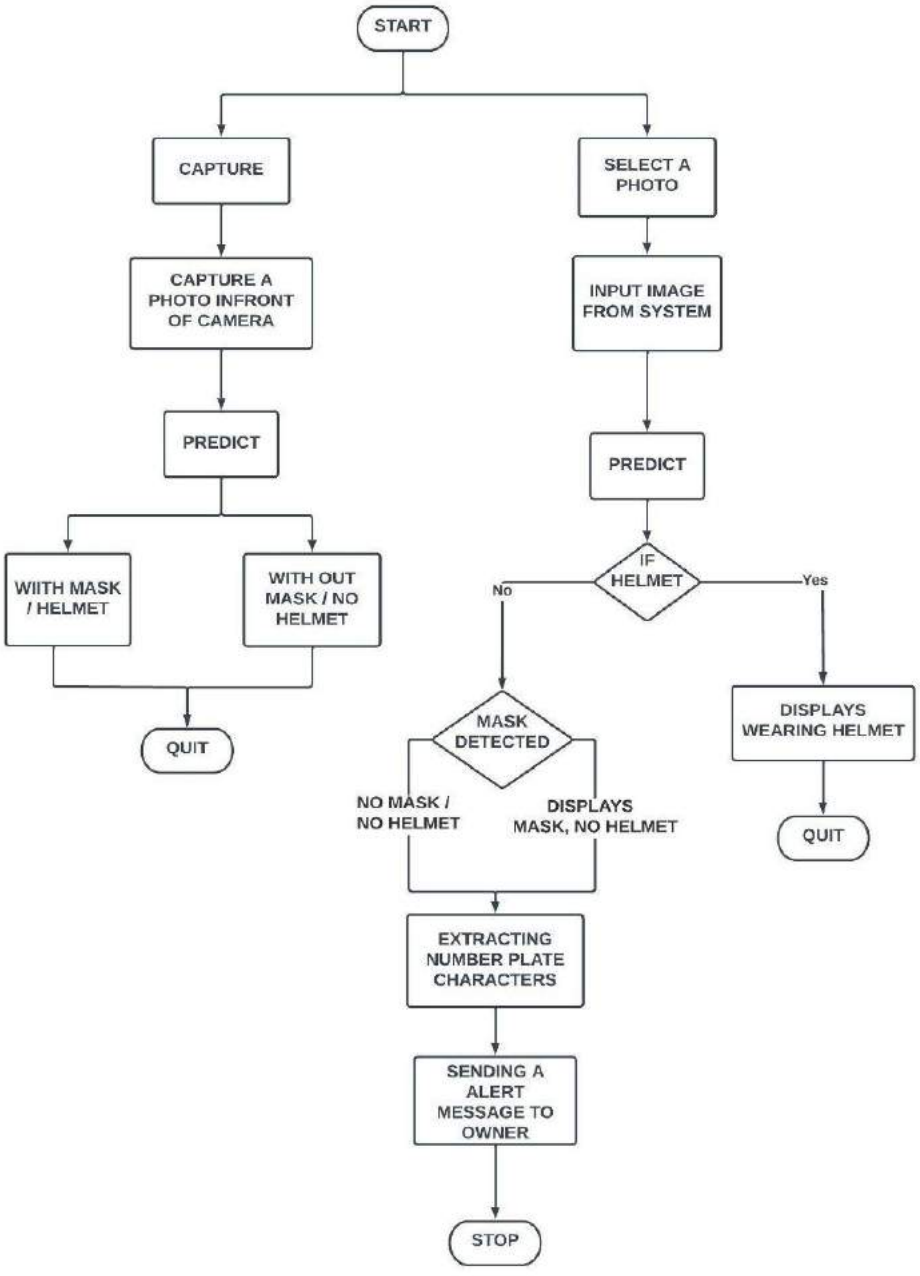


Fig 1.1: Flowchart of the system working

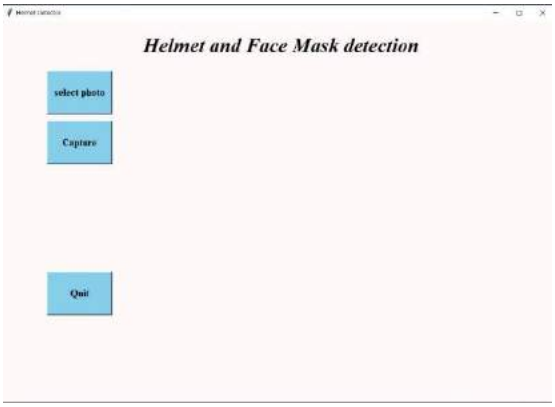


Fig 1.2: User Interface

Fig 1.1 represents how system works, after successful execution of system, an interface (as shown in fig 1.2) with two options pops out. **Capture** option enables the main camera of the system and takes the live photo as input to predict and the **Select** option enables the user to upload an image from the device as input to predict whether the rider is wearing a helmet or not, then follows wearing a mask or not and extract the number plate details to send an alert mail/SMS to the registered details. **Quit** button closes the application.

5. IMPLEMENTATION

1 Data Collection (Images):

The sample image collection was the most difficult task in this project. The sample images of two-wheeler riders with, without helmets, and with a mask with proper visibility of two-wheelers' registration number plate. After image collection, the images must be classified into two types. The Positive comprises the images to be detected by the system in real-time and The Negative is to be ignored by the system in real-time. The system may not be able to detect characters because of weather conditions.

2 Machine Training:

Transfer learning is a machine learning technique where a model trained on one task is re-purposed on a second related task. COCO (Common Objects In Context) dataset contains the large-scale object detection, segmentation, and captioning dataset published by Microsoft. The COCO model contains a vast number of pre-trained data. YOLO (You Only Look Once) is also a transfer learning which is used to train the data in both YOLOV3 and YOLOV4. We use YOLO weights and cfg files where the cfg file contains the configuration of convolution layers information like batches, subdivisions, learning rate, etc. HAAR (haar cascade frontal face detection) file which could be used by any scripting language among Java, Python, and CPP and used for deciding the real-time environment**.**

3 Extracting the characters from API (Plate recognizer):

API (Application Programming Interface) is a software code that helps two different software's communicate and exchange data. In our case, we will get the characters from the number plate using this API named plate recognizer.

4 Sending an Email and SMS:

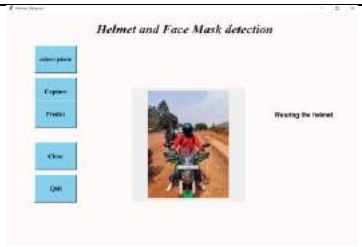
When the system detects that the rider is not wearing a helmet or not wearing a mask, characters are extracted from the number plate characters using API stored it. An email was sent Using the python libraries like smtplib, SSL, and requests to the concerned person with the image.

6. RESULT & DISCUSSION

TABLE: RESULTS FOR THE TEST CASES

Below are the results of some of the test cases for different possibilities.

<table>
<tr>
<td> 

Test case 1</td>
<td> 

Test case 2</td>
<td> 

Test case 3</td>
</tr>
<tr>
<td> 

Test case 4</td>
<td> 

Test case 5</td>
<td> 

Test case 6</td>
</tr>
<tr>
<td> 

Test case 7</td>
<td> 

Test Case 8</td>
<td> 
Test case 9</td>
</tr>
</table>

TEST CASES	SPEED OF THE VEHICLE in (Km/h)	WEARING HELMET OR NOT	WEARING MASK OR NOT	NUMBER PLATE EXTRACTED OR NOT	OUTPUT
Test case 1	NA	NO HELMET	NO MASK	NA	NO MASK NO HELMET-EMAIL SENT
Test case 2	NA	HELMET	NO MASK	NA	HELMET -89%
Test case 3	NA	NO HELMET	MASK	NA	MASK DETECTED NO HELMET
Test case 4	10	HELMET	NO MASK	NA	WEARING HELMET-94%
Test case 5	NA	NO HELMET	MASK	YES	NO HELMET – EMAIL SENT
Test case 6	NA	NO HELMET	NO MASK	YES	NO MASK NO HELMET-EMAIL SENT
Test case 7	5	NO HELMET	NO MASK	YES	NO MASK NO HELMET-EMAIL SENT

Test case 8 (Back view)	**NA**	**HELMET**	**NA**	**NO**	**WEARING THE HELMET-96%**
Test case 9	**30**	**NO HELMET**	**NO MASK**	**NO**	**NOT DETECTED**

The table represents the test cases with each different case. In most cases, using the select button, the rider's motion was at rest.

All the above testcase images are of real-time.We collected the data of different vehicles with different riders.

As specified in the table the test cases are collected along with the speed and various tests are done. In Test case 2, Test case 4, and Test case 8 the system detected and gives an accuracy of 89%,94%, and 96%. In Test case 1, Test case 5, Test case 6, Test case 7 the seystem detects that rides is not wearing a helmet and mask so an alertmessage is sent to the vehicle registered number.In Test case 3 rider wears only mask so it detects and sent an alert message and displays that the rider is wearing only mask but no helmet as shown in Test case 3.In Test case 9,system fails to detects the rider.

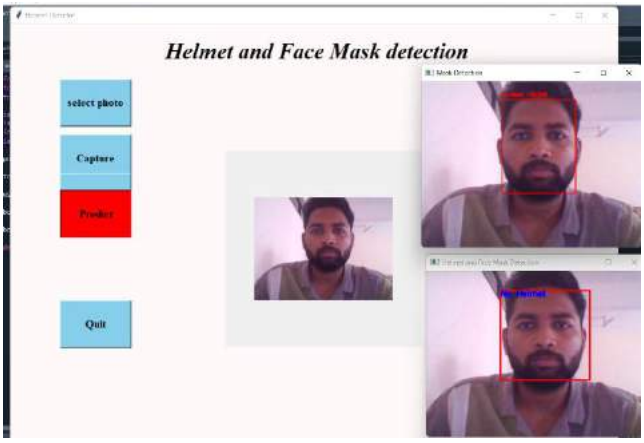


Fig 2.1: Detecting person using capture button

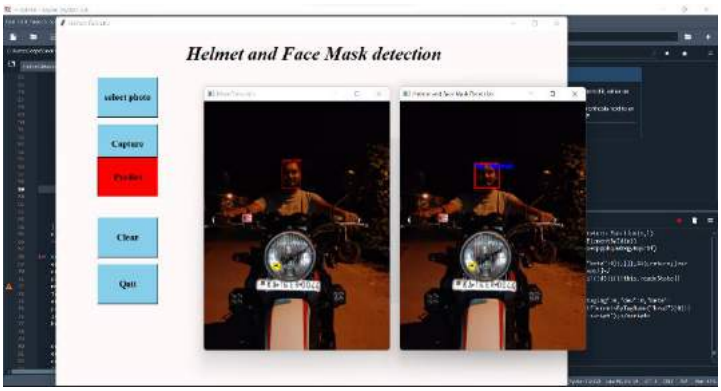


Fig 2.2: Detecting rider status using the select button

Fig 2.1 represents the result of the person as he is not wearing both a helmet & mask.
Fig 2.2 represents the result of the rider at rest as he is not wearing a helmet.

CHALLENGES FACED:

- It is difficult to get accuracy when we upload an image of riders in motion.
- It is tough to detect the number plate for some motorcycles considering dull light.
- In some cases, the system detects multiple helmets due to the windshield and headlight.
- It fails to give the desired results when we upload an image of some exceptional cases.

FUTURE WORK TO PROCESS:

- Should consider more testcases and execute.
- Should get more accuracy.

7. CONCLUSION

The target of detecting the two-wheeler riders has been accomplished. This system can detect whether the two-wheeler rider is wearing a helmet or not and if the rider is not wearing the helmet the system proceeds to detect the number plate and the image of this number plate will be passed to the API (plate recognizer) for text extraction which would be the vehicle number and we will generate the notification to concerned person through e-mail and SMS about the rule violation.

- People not wearing helmets or masks is considered a rule violation.