JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

DETECTION OF HELMET USING YOLOV4 AND GENERATION OF AN E-CHALLAN

Anirban Ashok Rudra
Department of Information Technology,
Universal College of Engineering, Vasai
University of Mumbai, India
anirbanrudra45@gmail.com

Shrish Kiran Vaidya
Department of Information Technology,
Universal College of Engineering, Vasai
University of Mumbai, India
shrish.vaidya@gmail.com

Kaushal Rajbahadur Singh
Department of Information Technology,
Universal College of Engineering, Vasai
University of Mumbai, India
kaushal1910singh@gmail.com

***Abstract*- Motorcycle accidents have been on the rise in several countries over the years. Any smart traffic system must include automated detection of offenders of traffic rules. Motorcycles are one of the main ways of transportation in a country like India, where population density is considerable in all major cities. Over 37 million people in India ride two-wheelers. Most motorcyclists do not wear helmets in the city or even on highways, according to reports. In most motorcycle accident scenarios, wearing a helmet can lower the likelihood of a biker suffering a head or severe brain injury. As a result, a technology for automatically detecting helmets is required for road safety. As a result, using a CNN-based algorithm (YOLOv4), custom object detection models are built. The License Plate is retrieved and the License Registration number is recognized using an OCR whenever a Helmetless rider is detected. This project aims to develop a CNN-based automated detection system for helmet identification utilizing custom-trained models and datasets that will aid police departments in enforcing the law for the greater good of society.**

*Keywords: **YOLOv4(You Only Look Once) object detection algorithm, CNN(Convolutional Neural Network), OCR(Optical Character Recognition).***

I. INTRODUCTION

The helmet is the most important piece of motorcycle safety equipment. The helmet protects the motorcyclist against accidents. Although helmet use is mandatory in many countries, there are motorcyclists that do not use it. The majority of deaths in accidents in the past couple of years have been caused by head injuries. Because of this, wearing a helmet is mandatory as per traffic rules, violations of which attract hefty fines. Despite this, a considerable number of motorcycle riders disobey the law. Currently, all major cities have massive video surveillance networks in place to keep an eye on a variety of threats. Thus using such an already existing system will be a cost-efficient solution, however, these systems involve a large number of humans whose performance is not sustainable for long periods of time.

Human surveillance has been demonstrated to be unsuccessful in recent research, with the number of human errors increasing as the period of video monitoring grows. Machine learning (ML) is the field of Artificial Intelligence in which a trained model works on its own using the inputs given during the training period. Machine learning algorithms create a mathematical model of sample data, referred to as "training data," in order to generate predictions or decisions, and they are also utilized in object detection applications. As a result, a Helmet detection model may be built by training it with a given dataset. Helmet-less riders can be easily detected with our helmet detection model. The rider's license plate is cropped off and saved as an image based on the recognized classes. This image is fed into an Optical Character Recognition (OCR) model, which reads the text and outputs the License Plate number in the machine-readable text.

II. PROBLEM DEFINITION

The helmet is the most essential element of safety equipment for motorcyclists, even though many do not use it. The main aim of this project is to construct an automatic detection of the motorcyclist without a helmet from video using a YOLOV4 trained model. If they are not wearing the helmet, from the registered number plate we extract the registration number. After extraction of the registration number, we can either generate a direct e-challan or send the registration details to the traffic police database.

III. LITERATURE REVIEW

Rising cases of accidental deaths due to not wearing the helmet are a serious concern and this concern should be solved by tightening the laws and imposing heavy fines on the ones disobeying it. To punish these rule offenders using the latest technology there were multiple approaches made to detect the helmets.

In [1], the researchers used yolov3 for detection of five classes, helmet, nohelmet, person, motorbike, licenseplate. All these classes are identified using a single trained weights.
And detection is run on frames of video. If nohelmet classes is detected then it searches for its person classes and motorbike class and finally detects numberplate, crops it and then finally saves it. But all this testing is done in sparsely populated riding environment. And according to us it's accuracy will decrease if it is done in a densely populated riding environment as there will be multiple detections overlapping each other. To tackle this issue what we did is that we divided the detection process in 3 parts first rider will be detected from video and its cropped image will be saved then another mode will detect the helmet if present else it will forward the rider image to next model that will detect numberplate and crop it and finally it will be sent forward for E-challan generation.

In [2] the researchers used technologies like CNN, Gausian mixture model. Their view was to detect motorcyclist without helmet in videos using CNN. They approached this problem by first separating the moving objects. Next they used CNN to find difference between motorcyclists and non-motorcyclists. Next they did head localization; this was done by cropping the motorcyclists image one fourth from the top and then using CNN to separate without-helmet and with-helmet. From this we got the idea to first separate all the riders from the moving video file but we didn't use the Guasian mixture model to do so. Instead we used the yolov4 object detection algorithm which is based on CNN. Also for further hekent detection we didn't use CNN as it is slower than yoloV4.

In [3] the main Objective of this paper is to detect helmet and no-helmet. But they have used a different approach. They first used yolov3 to detect the rider class and then directly they used CNN to detect the helmet and no-helmet class. From this we got an idea to use yolov4 instead of CNN as Yolov4 is faster than CNN.

IV. PROPOSED FLOW OF SYSTEM

Fig 1 shows System Flow, detailed description is as follows

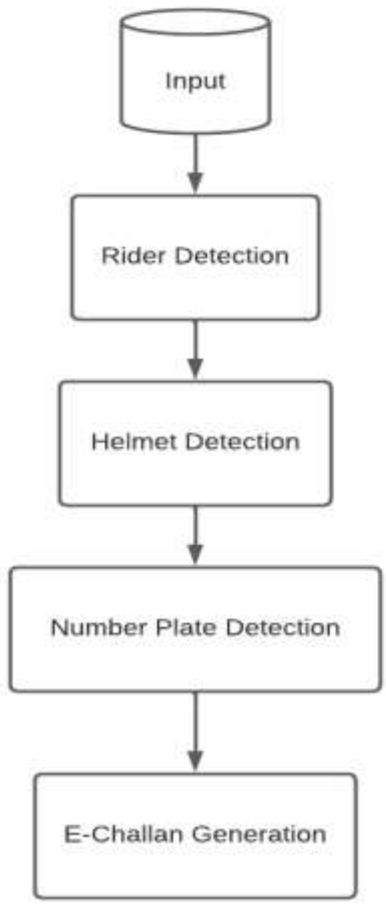

Fig. 1 System Flow

1. Rider detection : Riders are detected framewise from a video and the detections are cropped and saved for further usage.

2. Helmet detection: It detects if the rider detected is wearing a helmet or not if not again the image is saved for further procedure.

3. Number plate detection: The number plates are detected by those riders who aren't wearing a helmet.

4. E-Challan generation: Lastly, from the number plates detected the license number is extracted and then an e-challan is generated as an entry in a challan database.

Fig 2. Shows flowchart of the proposed system, the path taken by data in the system and the decisions made during different levels.

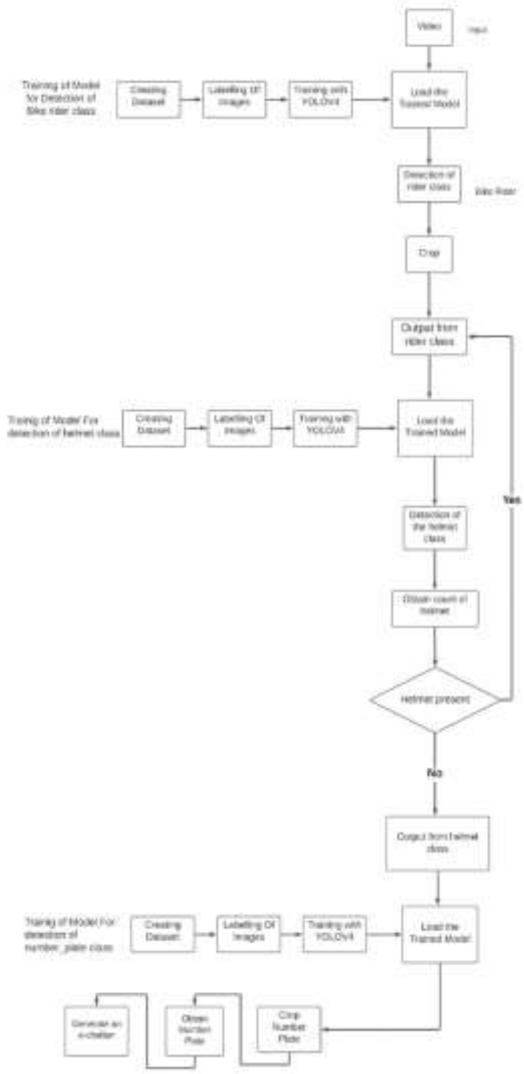

Fig. 2 Flowchart

YoloV4 : You Only Look Once is an object detection algorithm and V4 is its 4th version which gives more accuracy than previous ones. Yolo gives a pre-trained model named “COCO” which helps identify 80 different classes. It also is used to define custom objects and to train the model on custom dataset according to the user's needs. So, for us we trained a custom model as classes required by us were not in the “COCO” model.

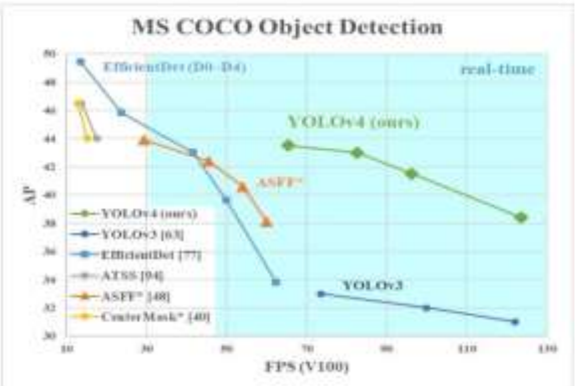

Fig. 3 Yolov4 efficiency

V. METHODOLOGY

Step 1: Run the following command on the Anaconda prompt to run the Rider class Model

python detect_video.py --weights ./checkpoints/custom-416 --size 416 --model yolov4 --video ./data/video/test1.mp4 --output

This will initiate the processing of the input mp4 video file. That would detect and crop any Rider(bikes) present in the input file. The detection frequency of the model is set to every 20FPS(frames per second) and the cropped images of the Rider along with the output file will get stored as a .png and .mp4 file respectively.

Eg:

Fig. 4 Cropped image of Rider

Note: The output of the previous model would act as an input for the next model respectively.

Step 2: After separating the Riders from the input file we execute the following command for the Helmet Model

python detect.py --weights ./checkpoints/custom-416 --size 416 --model yolov4 --count

The Helmet Model will receive the Rider’s image in .png format as the input and it will loop through the images and detect if the Rider is wearing a helmet or not.
If the Rider is wearing a helmet it will detect it and set the FLAG to 1 and if no helmet is detected in the image the FLAG value would be unchanged i.e 0 and at the end the images with the FLAG value 0 will be forwarded to the next stage where the number plate detection of the Riders not wearing helmet will take place.

Eg:

Fig. 5 Cropped image of helmet detected

Step 3: The very next step after successfully separating the bike Rider's those who are not wearing helmets from the others who are wearing helmets is to detect the number plate(license plate) of these Riders with no-helmets.
Executing the following command would run the number_plate Model.

python detect.py --weights ./checkpoints/custom-416 --size 416 --model yolov4 --crop

The number_plate Model would simply loop through the images that it has received as input in .png format from the previous Model and would detect the number plates(license plates) of the bikes (if visible) and separately crop the image of the number plate and store it as an output as shown in the below Fig 6.

Eg:

Fig. 6 Cropped image of number_plate detected of a Rider not wearing Helmet

Step 4: After getting the cropped images of the number plate the next step is to convert the image to text(OCR).
This would help to generate an e-challan for the registered owner of the number plate.

Python VisionAPIDemp.py

The above command would initiate the OCR conversion process. And after the process of OCR conversion is completed the numbers of the number_plate would be inserted into the E-Challan database and the owner of the number plate would receive an Email regarding the same on their registered Email-id.

For simplicity purpose we've created a dummy database for the number plates that are present in the input video.
The database consists of the number plates(which are to be inserted after the OCR conversion), registered email-id of the owner of the specific number plate, date on which the following detections took place.

Fig. 7 Cropped image of OCR conversion of the detected number_plates(that are been inserted into the e-challan database)

When the OCR conversion starts for a particular batch of inputs, at first it asks for the date on which the detections took place(so that it doesn't generate more than one e-challan to a particular vehicle on the same day).

After this step the OCR conversion takes place using Google Vision API and the data is inserted into the dummy database that we've created.

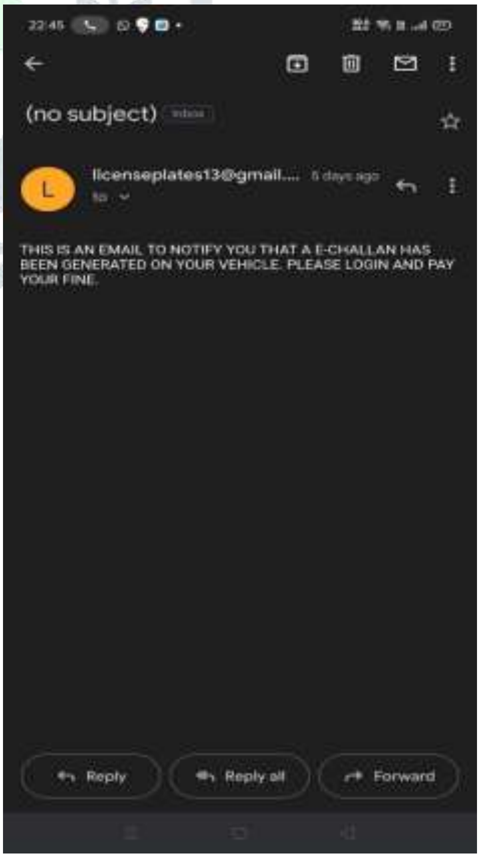

Fig. 8 Email notification to the owner of the number plate.

The owner of the number plate would receive an email stating that an e-challan has been generated on his registered number plate.
After which the user can visit the E-challan website and enter his number_plate and search if any records of e-challan is present or not.

Fig. 9 E-challan website

If there is a record present then the user can pay the amount by clicking on the PAY button and the record will get deleted from the database.

Proposed System Analysis:

We executed the system using some datasets. Some of the dataset were videos and some were images.

Accuracy = (True Positives / Total items tested)*100
Average accuracy = sum of all accuracy of different datasets tested/ no, of datasets tested.

Model	Tested dataset type Images or Videos	Accuracy	Details
Rider model	Videos	Average accuracy ~ 98%	In most videos all the riders were accurately detected but in one video it detected a cycle-rickshaw but with less accuracy.
Helmet detection model	Images	Average accuracy ~ 95%	Most of the time it detected all the riders with helmets accurately. From 100 riders it detected 5 false negatives. False negatives were the people with masks and a lady with full face cover. Due to covid-19 regulations people started wearing masks.
Number Plate detection	Images	Average accuracy : 100%	The model detected the number plates in the images with 100% accuracy. There were no false negatives. Around 60 - 70 images were tested but the model detected numberplate accurately
OCR conversion	Images	Average accuracy ~ 98%	The accuracy was affected as some riders didn't had numberplates and some were having numberplates written in design in such a way that human also cannot tell the number in one go.

Reference papers System Analysis:

Accuracy of helmet detection models in other papers:

Paper	Accuracy
[1]	Mean Average Precision of 5 classes detected : 75% OCR model accuracy : 85%
[2]	Classification performance (motorcyclist v/s non-motorcyclist) : 99.24% Classification performance (Helmet v/s Non-helmet) : 98.63%
[3]	Helmet detection accuracy : 96.23%

VI. CONCLUSION AND FUTURE SCOPE

The following proposed system helps the traffic department to detect multiple bike riders who are not wearing helmet from traffic surveillance videos. This is done by passing the surveillance video through multiple Machine Learning Models (i.e Rider detection Model, Helmet detection Model and number plate detection Model). The system also helps to automate the process of E-challan generation. The accuracy of the system is more than 90% in areas having sparsely populated traffic and ~ 80% in densely populated traffic. This architecture works well compared to other CNN models and can be further improved and updated when other versions of yolo object detection Models are released.

In future we plan to add linguistic OCR detection.
Find the best way to integrate the model without decreasing the accuracy.
Add payment option to the portal where the rule violator can see the challan.

VI. REFERENCES

[1] L Allamki, M Panchakshari, A Sateesha,and K S Pratheek, “Helmet Detection using Machine Learning and Automatic License Plate Recognition”,Int. Research J of Technology and Research(IRJET), Vol: 6, Issue: 12, Dec 2019.

[2] C. Vishnu, D Singh, C. K Mohan ,and S Babu,“Detection of Motorcyclists without Helmet in Videos using Convolutional Neural Network”, Institute of Electrical and Electronics Engineers(IEEE), July 2017.

[3] Madhuchhanda Dasgupta, Oishila Bandyopadhyay, Sanjay Chatterji, “Automated Helmet Detection for Multiple Motorcycle Riders using CNN”,, Institute of Electrical and Electronics Engineers(IEEE) Dec 2019.

[4] Google Vision API, Access date: Oct 2021, Online [Available]: “https://cloud.google.com/vision”

[5] Darknet framework, Access date: Mar 2021, Online [Available]: “http://pjreddie.com/darknet/”