Motorcycle Apprehension using Deep Learning and K-Nearest Neighbor Algorithm

Maria Rosario T. Garcia
Gokongwei College of Engineering
De La Salle University
Manila, Philippines
maria_rosario_t_garcia@dlsu.edu.ph

Argel A. Bandala
Gokongwei College of Engineering
De La Salle University
Manila, Philippines
argel.bandala@dlsu.edu.ph

Elmer P. Dadios
Gokongwei College of Engineering
De La Salle University
Manila, Philippines
elmer.dadios@dlsu.edu.ph

Abstract — **Road violations that lead to accidents and deaths are increasing significantly. There are about 1.35 million people who die every year because of road accidents, and more than half of these involve a motorcycle. Authorities are strictly implementing traffic laws and making some innovations to capture those motorists violating laws easily. Researchers are also doing their part to help solve the problem; indeed, their studies give a vast contribution and solve road safety issues. However, the papers on road violations were focused more on on-road violations involving four-wheeled vehicles.**

For this reason, a motorcyclist violation detection and plate recognition with e-mail notification using a Deep Learning algorithm were developed to apprehend motorcyclists violating traffic laws. Tensorflow Object Detection API was used as a framework along with the Faster R-CNN model. The system was developed using Anaconda Environment, Python Scripting, KNN, and MySQL Connector. The conditions and criteria for detecting a violation are based on motorcycle detection, including motorcycle tracking. After violation detection and plate recognition, the violation's image is sent through e-mail together with the details of the offense.

Keywords— motorcycle, machine learning, Tensorflow Object Detection Classifier, Faster R-CNN, KNN, MySQL, traffic violation

I. INTRODUCTION

At the present daily process, commuters usually spend a relatively extent of their time and money traveling point to point due to the traffic congestion on public streets. Constructed daily automobile drivers' strategies to reduce this struggle's damage became apparent, such as utilizing a motorcycle for an efficient journey. It might be a useful answer to the primary situation. However, motorcyclists are much prone to accidents because of the vulnerability it projects. Suppose a motorist loses his driving ethics in the heat of the moment. In that case, it could result in law violations that lead to accidents, road destruction, or demise [1].

There are already automatic ways to detect violations on the road to lessen the casualties that may happen. In the Philippines, the "No Contact Apprehension" was proposed by Metropolitan Manila Development Authority or MMDA. The agency uses CCTV or any other gadgets that can capture videos and images and use them to apprehend motorists violating traffic laws [2]. There is also a volume of researches that uses technology to capture road violations automatically. Some of these are [3]–[6] that use images or videos to detect road violations with different approaches: machine learning, deep learning, convolutional neural network, or image processing, to capture or apprehend violating motorists. Their studies give vast contributions to traffic management and evidently solve safety issues.

The growing number of traffic injuries and policy violations are just two of the critical issues the road operators and government authorities ought to diminish. Past studies came up with road vehicle apprehension and violation detection; however, it only centered on automobiles. Some studies suggested motorcycle apprehension wherein they solely focused on helmet detection or no-helmet.

According to the World Health Organization, more than fifty percent of road deaths involve motorcycles or other two- or three-wheeled vehicles. The highest traffic casualty cause is refusing to wear safety gear, specifically helmets [7]. The existing papers [8]–[11] detect if the rider has a helmet or no helmet. But, many other motorcycle violations are still commonly occurring or rampant. Although the papers furnished a great deal of proposals, recent traffic complications demand in-depth solutions such as sidewalk issues and overloading. Negligence such as these should also be condemned since they could be a factor in adding up to awful collisions.

To address the issue, we developed a system for automatic apprehension of a motorcyclist with the following violations: No Helmet, Beating the Red-Light, Overloading, Wrong Lane, and Over-speeding. This study addressed a broader range of law violations for a motorcycle by applying existing algorithms and deep learning approaches to develop a motorcycle apprehension. We created a strategy divided into five parts: object detection such as motorcycle, rider, traffic light, and plate; tracking a motorcycle; detection of the violation; plate recognition; and sending a notification to the violator.
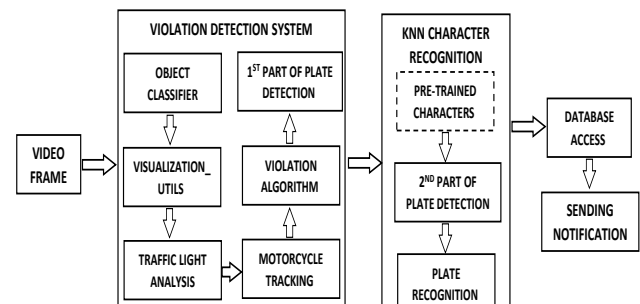
II. METHODS

Fig. 1. Motorcycle Apprehension Process

Fig. 1 shows the process of the system from the video stream to the sending of a notification. A video from the

CCTV footage was used as the input of the system. The object detection classifier identified the motorcycle, the motorcycle's plate, the pedestrians' upper body, and the traffic light present in the video, as it continuously receives video frames.

On each frame of the video used, the system evaluated a group of classes inside the motorcycle boundary along with the classified traffic light to determine if there is a violation present. After a violation is detected, the cropped plate is sent to another algorithm called the KNN algorithm. This algorithm determines or recognizes the plate number.

The outputs of the system were the detected violation and recognized plate displayed on the screen. Additionally, the recognized plate was used to find a match to the rider's database to get the corresponding e-mail address. An e-mail notification of the details of the violation is sent to the rider.

A. Dataset Preparation and Training Object Classifier

Evaluation of images is required to prepare the dataset needed in training the object detection classifier. Images extracted from the CCTV footage located at the front gate of the South entrance of De La Salle University have been labeled or annotated with seven classes: motorcycle, plate, with helmet, no helmet, red, green, and yellow. These classes are used as one of the bases for detecting motorcyclist's violations.

Since the camera that the system is using was installed on the motorcycle's rear-right side, the whole motorcycle was annotated, both with the back and side views. However, the plate's annotation size was doubled to meet the minimum size requirement, which is not less than 33px. Additionally, all pedestrians were annotated as riders except those facing upfront. The annotation starts from the head up to the hips of a person. The label used was either with a helmet or no helmet. A total of 19825 images were used; 80% of them (15860 images) were set as training images, while 20% (3965 images) were designated as evaluation or test images. There were also 4856 annotations for each class in the train folder while 1214 in the test folder.

We used the Tensorflow Object Detection API framework, along with the Faster R-CNN model, to train the prepared datasets. Every annotated image in the dataset produced an xml file that contains the characteristics of each annotated class: filename, path, and the coordinates of each bounding box. All xml files on train and test folders were compressed into a single file, train.csv, and test.csv files, respectively. The two csv files were the equivalent tabular form of all xml files on train and test folders and used to generate a tfrecords: train.record and test.record. The two records were used as inputs in the Tensorflow training model.

The training process required a Tensorflow-GPU since it needed extra processing power [12], though it worked even there's none. For my training using Intel i5 processor with 12GB RAM and NVIDIA GTX 1050 6GB, it took 22 hours to reach less than 0.0130 classification loss in 50887 steps. Fig. 2 shows the training process and its classification loss graph.
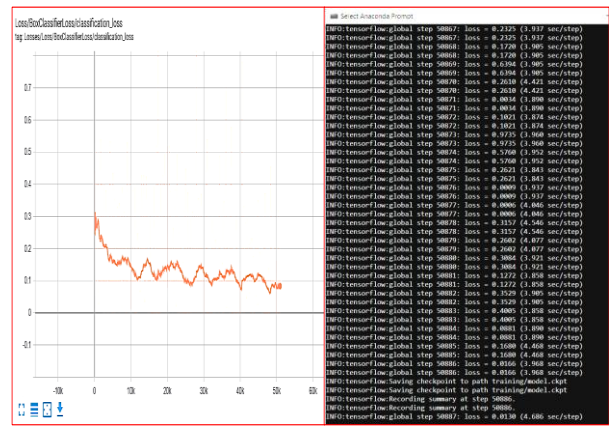

Fig. 2. (a) Classification Loss (b) Training (Steps) Process

B. Violation Detection

If the system detects an object/s included in the seven classes, it draws a bounding box around it as seen in fig. 3. The motorcycle's bounding box is based on other conditions and criteria for detecting the violation, including the tracking of the motorcycle. It is then safe to say that violations depend on the detected motorcycle.
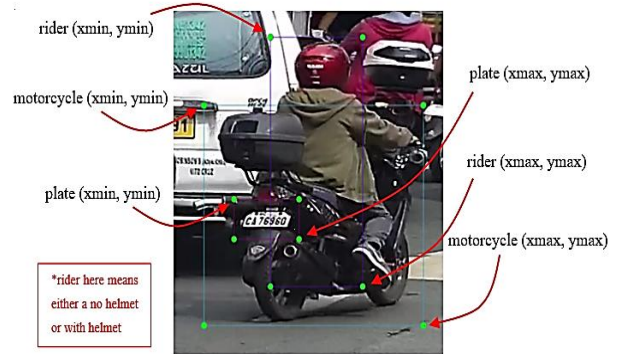

Fig. 3. Bounding Box of a Motorcycle, Rider and Plate

- *No Helmet Violation*

Any person captured in the video can be considered a rider, either with a helmet or no helmet. However, any person captured by the camera face up or upfront is not viewed as such. The system determines if the person detected is riding a motorcycle if its bounding box is located inside the detected motorcycle's bounding box. If the class of at least one of the detected riders was no helmet, then a violation is detected.

Variables Used for the Bounding Box:

Mx1/Mx2, Motorcycle's Upper/Lower x-coordinate

My1/My2, Motorcycle's Upper/Lower y-coordinate

Rx1/Rx2, Rider's Upper/Lower x-coordinate

Ry1/Ry2, Rider's Upper/Lower y-coordinate

Px1/Px2, Plate's Upper/Lower x-coordinate

Py1/Py2, Plate's Upper/Lower y-coordinate

TMx1/TMy1, Tracked Motorcycle's Upper Left x/y-coordinate

Algorithm 1: No Helmet Violation Detection

```
Input:          V, video stream
Output:         T, No Helmet Violation

1.   for each frame f in V
2.      extract the set of motorcycle bounding box M
3.      extract the set of no helmet and with helmet bounding box R
4.   for each element m in M
5.      LOOP while TRUE:
           #Compare Rider Coordinate to Motorcycle Coordinate
6.         if ((Rx2, Ry2) > (Mx1, My1)) AND ((Rx2, Ry2) > (Mx2, My2)) AND (Rx1) > (Mx1)
7.            if R(class) = "No Helmet"
8.               Display T
9.      End LOOP
```

- *Overloading Violation*

The system counts the rider where its bounding box is inside the detected motorcycle's bounding box. An overloading violation is detected if the number of riders exceeds two.

Algorithm 2: Overloading Detection

```
Input:          V, video stream
Output:         T, Overloading Violation

1.   for each frame f in V
2.      extract the set of motorcycle bounding box M
3.      extract the set of no helmet and with helmet bounding box R
4.   count = 0
5.   for each element m in M
6.      LOOP while TRUE:
           #Compare Rider Coordinate to Motorcycle Coordinate
7.         if ((Rx2, Ry2) > (Mx1, My1)) AND ((Rx2, Ry2) > (Mx2, My2)) AND (Rx1) > (Mx1)
8.            count = count + 1
9.      End LOOP
10.  If count > 2
11.     Display T
```

- *Beating the Red-Light Violation*

The traffic light was classified as red, green, and yellow. The object detection classifier determined what traffic light class is present in the frame.

A red line was drawn near the pedestrian lane/crossing to detect a beating the red-light violation, see fig. 4. The motorists should not cross/pass-through this line when red class is present; otherwise, a violation will be given.

Algorithm 3: Beating the Red-Light Detection

```
Input:          V, video stream
Output:         T, Beating the Red-Light Violation

1.   for each frame f in V
2.      extract the set of motorcycle bounding box M
3.      extract the set of traffic light bounding box TL
4.      extract the set of red-line points RL
5.   if  TL(class) = "Red"
6.      for each element m in M
7.         if (Mx2 = RLx) AND (My2 < RLy)
8.            Display T
```

- *Wrong Lane (Sidewalk) Violation*

A line was drawn in the sidewalk's pavement (purple color), as shown in fig. 4. Once the rider/motorcycle crosses the line; a Wrong Lane violation is given to the motorcyclist.

Algorithm 4: Wrong Lane Detection

```
Input:          V, video stream
Output:         T, Wrong Lane Violation (Sidewalk)

1.   for each frame f in V
2.      extract the set of motorcycle bounding box M
3.      extract the set of sidewalk-line points SL
4.   for each element m in M
5.      if (Mx2 > SLx) AND (My2 = SLy)
6.         Display T
```

- *Over-speeding Violation*

The formula for speed is $s = d/t$, where t is computed by dividing the frames counted by the system and the camera's frame rate. The distance between the two lines is 10 meters, and the frame rate of the camera used is 30fps. The test area is along Taft Avenue in front of DLSU South Gate. According to RA4136 - Land Transportation and Traffic Code [13], the set speed limit passing school zone is 20kph. Thus, if the motorcycle's speed passing through a school zone is greater than 20kph, a Speeding Violation is given.

The system starts tracking a motorcycle if, after detection, it passed through between the two tracking lines, T1 and T2. T1 is the color green vertical line beside the post on the left, while T2 is the turquoise color line after the pedestrian lane, see fig. 4. The set condition is if motorcycle (xmin) > T1 line (x) and motorcycle (xmin) <T2 line (x) then tracking will begin.

Algorithm 5: Over-speeding Detection

```
Input:          V, video stream
Output:         T, Overspeeding

1.   frame = 0
2.   distance = 10
3.   frame_rate = 30
4.   for each frame f in V
5.      extract the set of tracked bounding box TBB
6.      extract the set of tracklineA points LA
7.      extract the set of tracklineB points LB
8.      if (TMx1 = Ax) AND (TMx1= Bx) AND (TMy1 > Ay) AND (TMy1 < By)
9.         frame = frame + 1
        #compute for the value of speed in kph
10.  time = frame / frame_rate
11.  speed = (distance / time) * 3.6
12.  if (speed > 20)
13.     Display T
```


Fig. 4. All lines drawn on the image that use as basis for violation (Tracking Lines, Pavement Line, Beating Line)

C. Plate Detection and Recognition

There are two sections of detection in processing Motor Plate Detection and Character Recognition. In the first section, the system crops the image where the plate of the motorcycle is found. This cropping is done on the first stage of detection using TensorFlow's Object Detection API. If there is a motorcycle violation detected, the system crops the plate's Region of Interest (ROI) then save it as a jpg file in C:\MotorViolationDetection\models\research\object_detection\motorcycle_detection\cropped_images folder.

The cropped image is analyzed in the second section using the KNN algorithm. Its plate recognition flowchart is shown in fig. 5. The system detects all possible plates on the new image file by grouping all contours near each other. It then takes the most extended plate possible then runs the character recognition process.
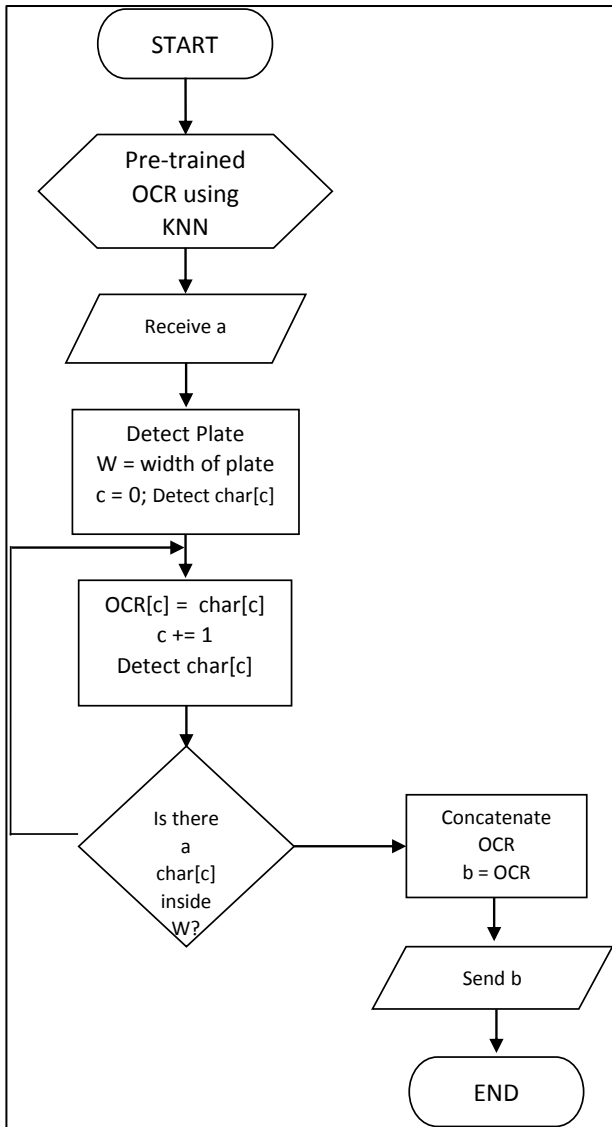
Fig. 5. KNN Plate Recognition Flowchart

process is exemplified in the images or photos shown in fig. 6.
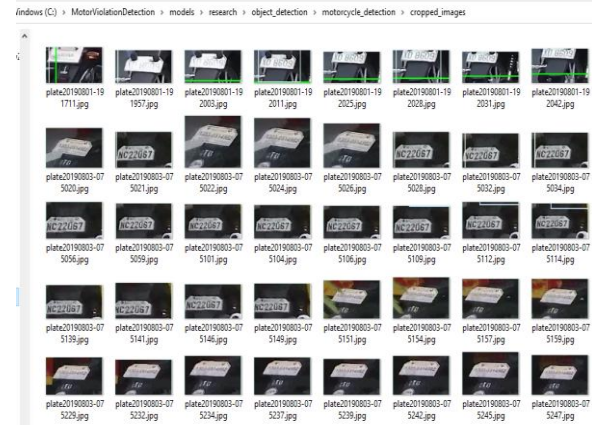

Fig. 6. Cropped Plates from the 1st Detection

A. *Motorcycle Detection and Tracking*

The motorcycle's bounding box is used on all violation algorithms, including tracking a motorcycle. It is then safe to say that violations depend on the detected motorcycle. Figure 7 shows the graph of the number of actual motorcycles, detected motorcycles, and tracked motorcycles. Based on the data gathered, the average accuracy of motorcycle detection is 82.1%, while tracking accuracy is 57.54%. The low accuracy rate of motorcycle detection and tracking is due to the traffic condition or crowded testing environment.
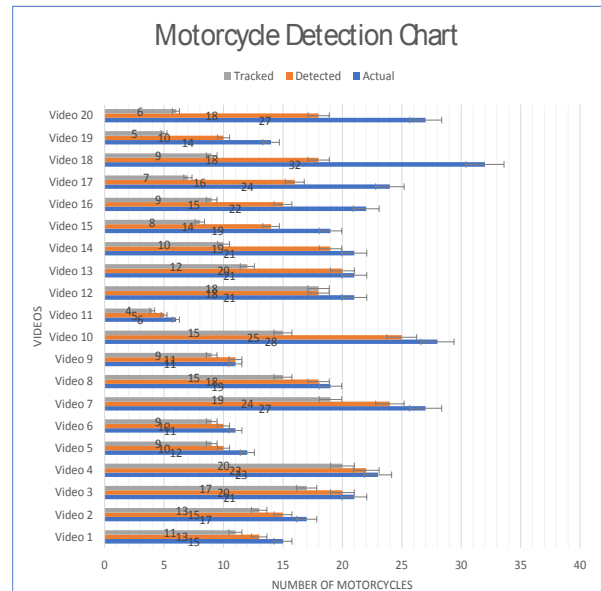

Fig. 7. Graph of Detected and Tracked motorcycles

A. *Rider, Traffic Light, and Plate Detection*

The system determines if the person detected is riding a motorcycle if its bounding box is located inside the detected motorcycle's bounding box. Rider classification accuracy: with a helmet or no helmet, is ranging from 75% to 98%. Meanwhile, the accuracy of detection on traffic light for all videos used is 100% with 85% to 99% accuracy on each frame's classification. Figure 8 shows some images of object detection with its corresponding accuracy.

D. *Pre-recorded Database and E-mail Notification*

The motorcycle information, like plate number, the name of the owner or user, and its e-mail address, were stored in a database through MySQL. The researchers gathered pre-encoded data from motorists via a google form. They used the database to determine the e-mail address of the apprehended rider. The recognized plate is matched on each plate number recorded in the database.

We created an e-mail account, *hulikomotormo@gmail.com,* to be used as the default sender of violation detected. The system looks in the database plate field to find or match the recognized plate number. If there is a matched plate, the system gets the corresponding e-mail address and then sends the violation's details.

III. RESULTS AND DISCUSSION

The camera's location and projection should consider the traffic light, the sidewalk, and the rear part of the motorcycle, making sure that the riders are visible. The camera was zoomed so that the motorcycle's plate can be captured since the Philippines' motorcycle plate is smaller than the plate of other vehicles such as cars, trucks, and buses [14]. This
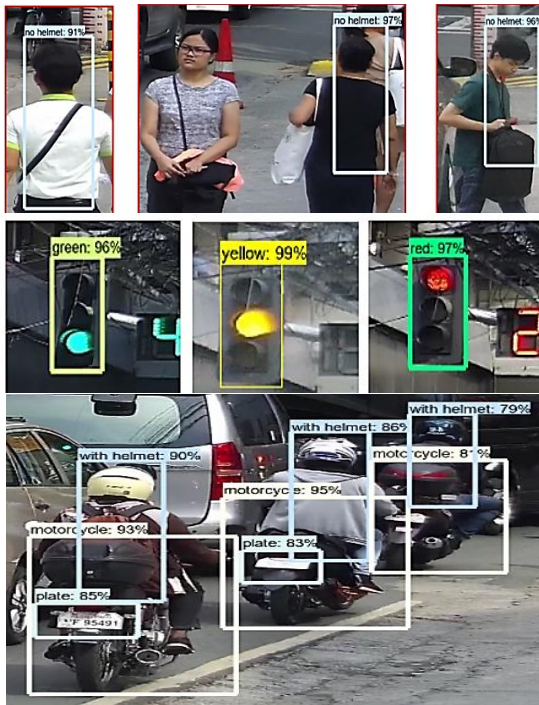
Fig. 8.   Sample Detection on each Class

From a random of 100 motorcycles with plates installed, there are 94 plates detected and six plates not seen, which gives accuracy detection of 94%. In the second stage, using the KNN algorithm, the plate detection is 92% from a random of 50 clear cropped images.

B. Violation Detection

We listed the number of actual violations from 20 videos gathered and compared them with the system's number of violations. Each video has a length of approximately 2 minutes.

A confusion matrix is used to determine the accuracy of violation detection. The accuracy of No Helmet Detection, Overloading Detection, Beating the Red-Light Detection, and Wrong Lane (Sidewalk) Detection. The results are 93.09%, 94.12%, 94.37% and 94.68%, respectively as seen in Table 1. The formula is also shown in the said table. Beating the red light and sidewalk violations shows low precision results because of the occlusion problem during heavy traffic situations.

TABLE I.   DETECTION ACCURACY - CONFUSION MATRIX

NO HELMET			ACCURACY
Total Number of Motorcycles = 391			
Total Number of Violations = 21			
	Detected With Violation	Detected No Violation	
Actual With Violation	17	4	93.09
Actual No Violation	23	347	

OVERLOADING			ACCURACY
Total Number of Motorcycles = 391			
Total Number of Violations = 3			
	Detected With Violation	Detected No Violation	
Actual With Violation	2	1	94.12
Actual No Violation	22	366	

BEATING THE RED LIGHT			ACCURACY
Total Number of Motorcycles = 391			
Total Number of Violations = 50			
	Detected With Violation	Detected No Violation	
Actual With Violation	28	22	94.37
Actual No Violation	0	341	

SIDEWALK			ACCURACY
Total Number of Motorcycles = 391			
Total Number of Violations = 59			
	Detected With Violation	Detected No Violation	
Actual With Violation	38	21	94.63
Actual No Violation	0	332	

$$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

Where:

`   TP   With Actual Violation and with Detected Violation

FP   With No Actual Violation but with Detected Violation
FN   With Actual Violation but No Detected Violation
TN   With No Actual Violation and No Detected Violation

The system was able to compute the speed of the tracked motorcycle through the speed measurement algorithm. Table 2 shows the results of speed detection from a random of 25 tracked motorcycles. Out of these, 25 were correctly classified if there is a speeding violation or none.

TABLE II.   SPEED MEASUREMENT AND OVERSPEEDING DETECTION

Motorcycle	Speed Measurement (kph)	Overspeeding?	Is it Detected?	Remarks
1	30	YES	YES	👍
2	35	YES	YES	👍
3	19	NO	NO	👍
4	45	YES	YES	👍
5	20	YES	YES	👍
6	25	YES	YES	👍
7	30	YES	YES	👍
8	32	YES	YES	👍
9	16	NO	NO	👍
10	30	YES	YES	👍
11	40	YES	YES	👍
12	50	YES	YES	👍
13	75	YES	YES	👍
14	70	YES	YES	👍
15	60	YES	YES	👍
16	65	YES	YES	👍
17	62	YES	YES	👍
18	23	YES	YES	👍
19	18	NO	NO	👍
20	16	NO	NO	👍
21	28	YES	YES	👍
22	60	YES	YES	👍
23	79	YES	YES	👍
24	26	YES	YES	👍
25	22	YES	YES	👍

* videos during light traffic condition

👍 The system correctly determine if there is a speeding violation or none.

C. Plate Recognition

From the second stage of plate detection, where 46 out of 50 clear cropped images were detected, the proponent randomly picked 30 images to test for plate recognition. Out of these 30 images, 22 plates were recognized, which resulted in 73.33%.

D. Plate Matching and Sending of e-Mail Notification

The database is created using MySQL Connector-Python. It is used to determine the rider's e-mail address if there will be a detected violation.

To test the sending of e-mail notifications, we used a dummy e-mail account, jam.reybert@gmail.com, and assigned it to one of the plates recognized by the system. The account was stored in the database with an account name Reybert Tolentino. The image shown in fig. 9 is the sample

flow violation apprehension from the detection of violation to sending of e-mail notification.

Fig. 9. (a) Sidewalk Violation Detection (b) Plate Detection and Recognition (c) E-mail Notification

IV. CONCLUSION AND RECOMMENDATION

In this work, we presented a motorcycle apprehension system that can detect violations on the rider such as No Helmet, Overloading, Beating the Red-light, Over-speeding, and Wrong Lane with an accuracy of 93.09%, 94.37%, 94.12%, 94.63%, and 100%, respectively. The system recognizes the motorcycle traffic violator's plate and sends an e-mail notification to the registered rider. The system is effective and efficient; however, it is still subject to improvements and innovations, explicitly detecting and tracking a motorcycle even in a traffic situation.

ACKNOWLEDGMENT

REFERENCES

[1] D. L. Romero, D. M. De Barros, G. O. Belizario, and A. De Pádua Serafim, "Personality traits and risky behavior among motorcyclists: An exploratory study," *PLoS One*, vol. 14, no. 12, pp. 1–15, 2019.

[2] "No Contact Traffic Apprehension Policy." [Online]. Available: http://www.mmda.gov.ph/20-faq/2040-no-contact-traffic-apprehension-policy-11-things-you-need-to-know.html. [Accessed: 04-Jul-2019].

[3] G. Desai, V. Ambre, S. Jakharia, and S. Sherkhane, "Smart Road Surveillance Using Image Processing," *2018 Int. Conf. Smart City Emerg. Technol. ICSCET 2018*, pp. 1–5, 2018.

[4] R. Shreyas, B. V. P. Kumar, H. B. Adithya, B. Padmaja, and M. P. Sunil, "Dynamic traffic rule violation monitoring system using automatic number plate recognition with SMS feedback," *2nd Int. Conf. Telecommun. Networks, TEL-NET 2017*, vol. 2018-Janua, pp. 1–5, 2018.

[5] J. Spanhel, J. Sochor, and A. Makarov, "Detection of Traffic Violations of Road Users Based on Convolutional Neural Networks," *2018 14th Symp. Neural Networks Appl. NEUREL 2018*, 2018.

[6] A. C. P. Uy, R. A. Bedruz, A. R. Quiros, A. Bandala, and E. P. Dadios, "Machine Vision for Traffic Violation Detection System through Genetic Algorithm," no. January, 2016.

[7] W. Tan, "WHO PH: Over 90% of Motorcycle Deaths Didn't Wear Helmets." [Online]. Available: https://www.carmudi.com.ph/journal/philippines-90-motorcycle-deaths-didnt-wear-helmets/. [Accessed: 16-Mar-2019].

[8] J. Mistry, A. K. Misraa, M. Agarwal, A. Vyas, V. M. Chudasama, and K. P. Upla, "An automatic detection of helmeted and non-helmeted motorcyclist with license plate extraction using convolutional neural network," *Proc. 7th Int. Conf. Image Process. Theory, Tools Appl. IPTA 2017*, vol. 2018-Janua, pp. 1–6, 2018.

[9] K. Li, X. Zhao, J. Bian, and M. Tan, "Automatic Safety Helmet Wearing Detection," *2017 IEEE 7th Annu. Int. Conf. CYBER Technol. Autom. Control. Intell. Syst. CYBER 2017*, pp. 617–622, 2018.

[10] J. Li *et al.*, "Safety helmet wearing detection based on image processing and machine learning," *9th Int. Conf. Adv. Comput. Intell. ICACI 2017*, pp. 201–205, 2017.

[11] G. Zhang, L. Lv, L. I. Dan, and M. Zhu, "The Method for Recognizing Recognition Helmet Based On Color and Shape," vol. 126, no. 5th Int. Conf. on Machinery, Materials and Computing Tech ICMMCT, pp. 1219–1223, 2017.

[12] "How to train Tensorflow models - Towards Data Science." [Online]. Available: https://towardsdatascience.com/how-to-traine-tensorflow-models-79426dabd304. [Accessed: 19-Jun-2020].

[13] L. T. O. Phil., "Land Transportation and Traffic Code RA4136." 1964.

[14] Department of Transportation, "Procurement of Goods and Services for the Land Transportation Office Motor Vehicle License Plate Standardization Program," 2012.