Vehicular Odd-Even compliance and E-Challan generation

Rishank Pratik
19BCE1606
Vellore Institute of Technology, Chennai

Wilson Vidyut Doloy
19BCE1603
Vellore Institute of Technology, Chennai

Prashant Verma
19BPS1023
Vellore Institute of Technology, Chennai

Abstract — Pollution in India is a definite offshoot of many other environmental problems - be it air, water, land, or noise. One major source of environmental emissions are cars. India needs smarter transportation options as well, and this is where startups can pitch in apart from other ways to tackle pollution. This script talks about designing a system based on an Odd-Even formula to control traffic. Odd- Even formula stated that vehicle with odd numbers license plate will ply on odd dates and those with even numbers license plate will ply on even dates.

Keywords— Odd-Even, License plate, OCR, Challan, QR Code, Traffic, E-challan, API, Tesseract.

I. INTRODUCTION

The odd-even rule is an emergency measure to control air pollution in the city by means of restricting the use of private vehicles. As per the rule, only even numbered vehicles (0, 2, 4, 6, 8) will be allowed on city roads. On odd dates, only vehicles having registration numbers ending with 1, 3, 5, 7 and 9 will be allowed.

This was first introduced in 2016, in Delhi, the odd-even scheme is a car rationing system. Under the scheme, only vehicles with even registration numbers will be allowed on roads today.

Implemented by Arvind Kejriwal-led Aam Aadmi Party (AAP) government, the odd-even scheme will apply to all non-transport four-wheeled vehicles plying on roads in Delhi.

Twenty-nine categories of vehicles, including those of President, Prime Minister, emergency and enforcement vehicles, cars carrying school children in uniform have been exempted from the odd-even scheme.

A challan is a piece of paper with an official text inscribed on it that is given to a certain individual. In India and adjacent countries, a challan is frequently issued for breaking traffic rules and regulations. If a challan is issued in your name, you must pay the fine specified under The Motor Vehicles Act, 1988. A traffic police officer has the authority to issue a challan in the name of anyone who violates one or more traffic laws. An E-Challan is a challan that is generated electronically using the Electronic Challan System. Physical pieces of paper have recently been replaced with E-Challan. Several phases of the procedure have been digitised in India, from a person breaching a traffic regulation to paying the appropriate fee, thanks to the advent of the e-challan check. We are generating E-challan on one click, for reducing time work. The management of E-challan record is done on online database. So, there is no need of any paper work. Therefore, we are reducing the minimal paper work and this concept is also useful for making corruption free and digital India. First the license plate is detected either in a photo or video input, then the number is extracted and last digit is checked for odd or even. If odd digit license plate is detected on even date or even digit license plate is detected on odd day, the license plate number will be sent to the web application. This web application which generates challan includes current challan amount, previous violation charges(if any) and taxes.

II. RELATED WORKS

Controlling visitors is a major concern in cities across the sector. Prepared traffic offence system is a reliable cellular-based software that tracks all traffic offences committed across the city. The application aids the traffic police in maintaining accurate records of all road offences committed, as well as the databases of the motive force and vehicle information.

In Mumbai, an automatic challan machine can be used to issue a price tag to visitors who violate the law. The facility will include a hand-held device for quality collection on the spot and a layout printed receipt. The goal of this facility's back end (connecting the e-challan system to CCTV cameras) is to relieve the burden on traffic police and help us go paperless while ensuring that no offender walks free.

The technique used in [1] served as a base for us. The author in [2] suggested to place camera at airport entrances or random angles, at angles of right, left ~ 30° or straight. That being not an ideal position.

The segmentation module as in [3] extracts the license plate in the detected car image using neural networks as filters for analyzing the color and texture properties of the license plate. The recognition module then reads the characters on the detected license plate with a support vector machine (SVM)-based character recognizer.

Technique presented in [4] was a comprehensive review of

the state-of-the-art techniques for ALPR. They categorize different ALPR techniques according to the features they used for each stage, and compare them in terms of pros, cons, recognition accuracy, and processing speed.

In [5] a new algorithm for vehicle license plate identification is proposed, on the basis of a novel adaptive image segmentation technique (sliding concentric windows) and connected component analysis in conjunction with a character recognition neural network.

The proposed system in [6] detects the number plate of a vehicle from video input and then performs the super resolution technique. Applying the Optical Character Recognition Technique it acquires the text from the super resolution image of vehicle number plate by means it compares with the RTO database and then it display the details of the vehicle such as owners name, vehicle registration etc.

The proposed system in [7] performs knowledge generation through data clustering mechanism. Further, the hidden information pattern within the database of detected number plates is used to provide insight into the vehicle information towards decision making and analysis.

The impact of odd-even rule on Delhi's environment before and during the implementation was studied in [8]. The concentrations of various pollutants at different locations were measured and their impact was evaluated quantitatively.

Assessment of the related vehicle presence on the road before and during odd-even implementation and their impact was provided in [9].

The proposed idea in [10] is the efficient e-challan generation using generation technique using OCR(Optical Character Recognition) in which the challan is created using an android application. It detects the variety plate and fetches facts from the database and generates E-challan.

A new technique about measuring distance between vehicles was presented in [11]. The camera in front of vehicles can capture images of the next vehicles. The process by Fast Fourier Transform(FTT) is there. The systems can check the position of the plate car with the properties of the picture. The systems can be compared with data stored in the database as reference again, thus giving distance between vehicles.

A new error measure which quantifies the performance of an image segmentation algorithm for identifying multiple objects in an image was defined in [12]. This error measure is based on object-by-object comparisons of a segmented image and a ground-truth (reference) image.

The authors of [13] had mainly explored the related issue of vehicle licence tax payment and four different techniques for classifying payment on taxation, including related studies of vehicle licence tax, Delphi method, rough set theory and its algorithm, Decision trees C4.5, artificial neural networks – Multilayer perceptron and Naive bayes.

Challan system based on Raspberry modules such as gas sensor in [14] It has a challan system but being placed at the exhaust pipe of the vehicle, it is removable/modifiable by anyone.

The method used in [15] was OpenCV along with Radial Basis Function Neural Network in a Mobile App to detect Number plates. Mobile phones cannot be installed in a static position, proper camera has to be used.

The method used in [16] captures images of motorcycle riders without helmets and click the chassis number of vehicles. Though it is not feasible to click the chassis number of vehicles even if it is not moving.

The authors of [17] surveyed the work done in Automatic License Plate Detection and E-Challan generation systems for the past 5 years. For object detection, edge detection algorithms are most preferred, especially the Canny edge detection method. This is followed by OCR. e-challan generation is commonly hosted on websites and android applications and notification is sent via SMS.

The system proposed by the authors of [18] consists of integration of algorithms like: 'feature-based number plate localization' for locating the number plate, 'image scissoring' for character segmentation and statistical feature extraction for character recognition; which are specifically designed for Indian number plates.

The authors of [19] explored and analyzed the existing approaches and techniques used in ALPR solutions in recent literature. The single-stage deep learning-based solutions showed high performances with diverse datasets. The multi-stage object detection based deep learning solutions can be pre-trained on large datasets, how-ever, they showed less computational efficiency and accuracy than single-stage approaches.

III.PROPOSED METHODOLOGY

Automatic Number Plate Recognition (ANPR)-based E-challan generation is a potential future technology for effective traffic rule infraction control. Authorities are finding it difficult to control and monitor traffic offences and apply appropriate sanctions as the population grows and more vehicles are added to the road.

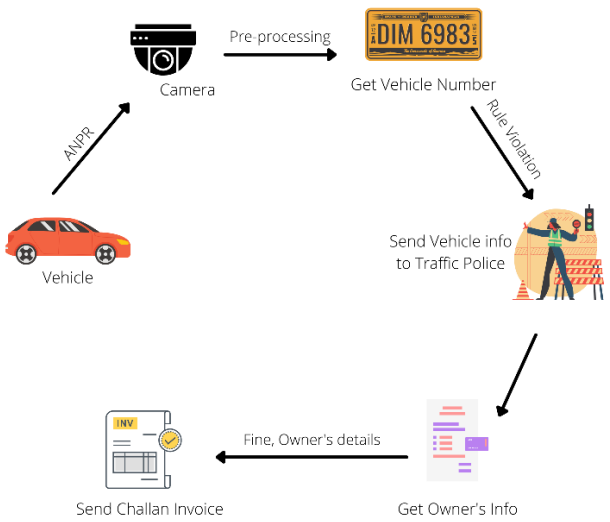

Fig. 1. Process of generating E-Challan

Traditional techniques for manually inspecting and confirming vehicle documentation, as well as monitoring and detecting violations, not only consume people and time, but also slow down traffic. The goal of this work is to propose a way for automatically issuing E challans for traffic violators utilizing an ANPR system that uses image processing and optical character recognition (OCR) to read the vehicle number plate using installed cameras and CCTV systems, as shown in Fig. 1.

These systems may work 24 hours a day, seven days a week, resulting in effective and simple detection of traffic rule violations and E challan handling, assisting officials with minimal user participation.

The existing system makes it impossible to keep track of traffic violations, which adds to the paperwork. As a result, the proposed approach can solve the aforementioned problem.

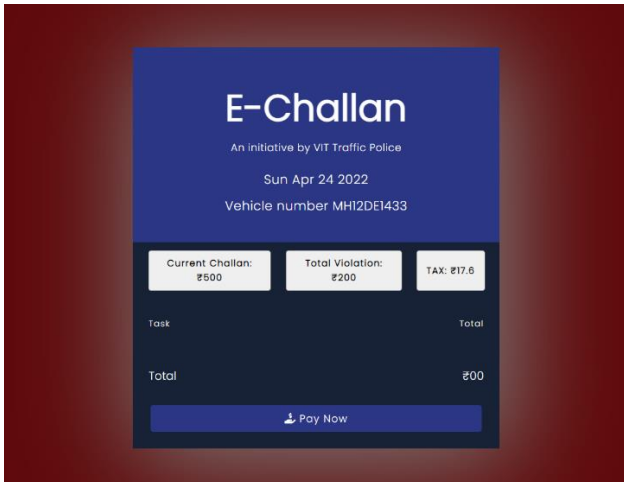

Fig. 2- E-Challan Web Application

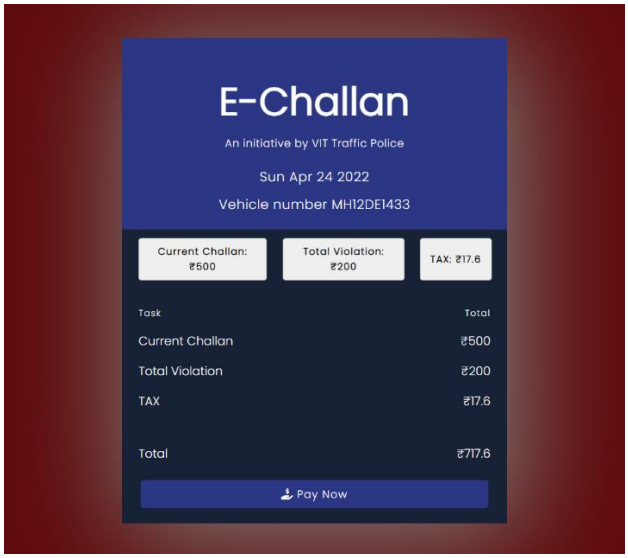

Fig. 3 - Web App Tabs and payment breakdown

In this script, the development of s web application is as shown in Fig. 2 and Fig. 3 which consists of tabs. Tab 1 will show current challan fees, Tab 2 will show previous challan fees (if any) and Tab 3 will show taxes.

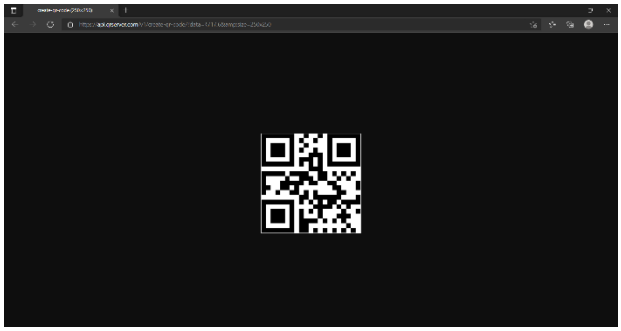

Fig. 4 – Generated QR Code for payment

After submitting the challan, the two messages is generated using API. First message contains the e- challan breakdown and second message contains the QR code as shown in Fig. 4. which is the challan payment method, which directly goes to the traffic police.

Fig. 5- Extraction of license plate number

IV. IMPLEMENTATION

There are three basic steps that software goes through in order to identify and recognize a license plate as shown in Fig. 5.

1) Using an image of a car as input - The application uses the image of the car as input to detect the license plate.

2) Processing the input - The image used as the input is processed to identify the section of the vehicle that contains the license plate.

3) Recognizing the license plate - From the number plate image, the values of the detected license plate are extracted.

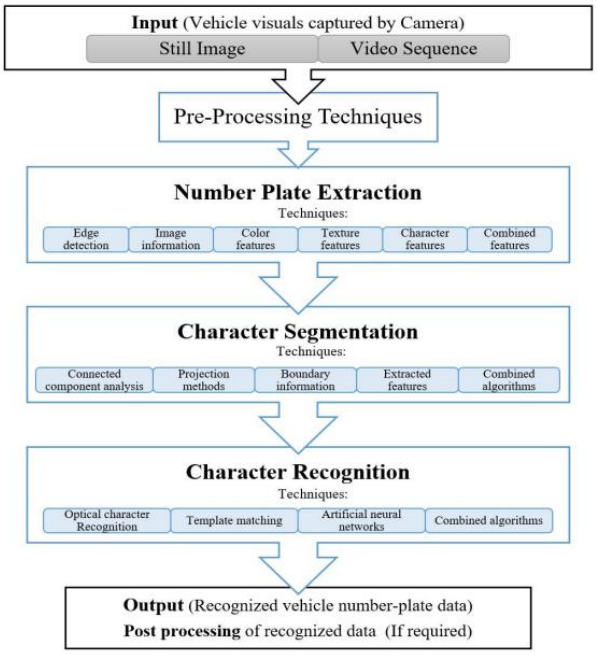

Fig. 6- Flowchart of ANPR Algorithm

The pseudo-code is as follows:

1. Importing the libraries that we need
2. Specifying the path to which tesseract is installed
3. Taking in our image input and resizing its width to 300 pixels
4. Converting the input image to greyscale
5. Reducing the noise in the greyscale image
6. Detecting the edges of the smoothened image
7. Finding the contours from the edged image
8. Sorting the identified contours
9. Finding the contour with four sides
10. Cropping the rectangular part identified as license plate
11. Drawing the selected contour on the original image
12. Extracting text from the image of the cropped license plate, as shown in Fig. 6

The libraries needed

import cv2
import imutils
import pytesseract

Fig. 7- test.jpg (Sample license plate)

We're going to use image = *cv2.imread('test.jpg')* as our input. The image's name is test.jpg whch is our Fig. 7.

We're scaling our image with *imutils.resize (image, width=300)*. The image we used as input is image. We're adjusting the width of that image to 300 pixels with width=300.

cv2.imshow("original picture", image): The image is displayed when its width is set to 300 pixels. The name of the window that displays the image is original image. Give it your name if you want.

cv2.waitKey(0): To continue running the code that follows, we are waiting for any key on the keyboard to be pressed.

cv2.cvtColor(image, cv2.COLOR BGR2GRAY): gray image = cv2.cvtColor(image, cv2.COLOR BGR2GRAY): We're going to make a variable called gray image. The image is then passed to cv2.cvtColor, where cv2.COLOR BGR2GRAY specifies that the image should be converted to grey.

cv2.imshow("greyed image", gray image): cv2.imshow("greyed image", gray image): We're showing an image that has already been converted to greyscale.

gray image = cv2.bilateralFilter(gray image, 11, 17, 17): We're smoothing the grey image by lowering the noise.

We're presenting the already smoothed image with *cv2.imshow("smoothened image", gray image)*.

edged = cv2.Canny(gray image, 30, 200): The edged variable is being created. After that, we use cv2.canny to find the edges in our smoothed image.

cv2.imshow("edged picture", edged): The image with the identified edges is displayed.

cnts: The contours are represented by this.

RETR LIST: This function retrieves all contours without establishing a parent-child relationship.

CHAIN APPROX SIMPLE: Removes all unnecessary points on the observed contours.

We're making a copy of the original input image with *image1=image.copy()*. This is due to the fact that we do not wish to alter the original image.

We are painting the specified contours on our picture with *cv2.drawContours(image1,cnts,-1,(0,255,0),3)*. Fill in the values exactly as they are.

cv2.imshow("contours",image1): The image is displayed with the contours recognized around it.

cnts = sorted(cnts, key = cv2.contourArea, reverse = True)

[:30]: We are sorting contours based on the minimum area 30 and ignoring the ones below that.

Cnt = None: Stores the number plate contour.

cv2.drawContours(image2,cnts,-1,(0,255,0),3): Draws the sorted contours on the image.

cv2.imshow("Top 30 contours",image2): Displays the image which contains the top 30 contours drawn around it.

for c in cnts: : We are creating a for loop over the contours we did sort. This is to find the best contour of our expected number plate.

perimeter = cv2.arcLength(c, True): Perimeter is also referred to as arclength. We are using the arclength function to find it.

*cv2.approxPolyDP(c, 0.018 * perimeter, True)*: Approx

PolyDP approximates the curve of polygon with precision.

x,y,w,h = cv2.boundingRect(c) : This finds the coordinates of the part identified as the license plate.

cv2.imwrite('./'+str(i)+'.png',new_img): Stores the new image of the cropped number plate.

break: breaks the for loop.

cv2.drawContours(image, [screenCnt], -1, (0, 255, 0), 3): This creates the contour on our initial image that was picked to represent the number plate.

cv2.imshow("image with recognised licence plate", picture): Showing the final image with the number plate contoured.

Cropped loc = './7.png': This is the file name for the licence plate cropped image.

cv2.imread(Cropped loc), cv2.imshow("cropped"): The image of the clipped licence plate portion is being shown.

We're sending the image of the cropped part of the licence plate to *pytesseract.image_to_string(Cropped loc, lang='eng')*. The text on the image is then extracted using pytesseract.

cv2.destroyAllWindows(): All of the open windows are being closed.

There are different ways in which you can run the application that you created in the previous section. One of the most straightforward ways to launch a Flask app for local development.

By default, Flask will run the application you defined in app.py on port 5000. While the application is running, go to http://localhost:5000 using your web browser.

First the picture of the vehicle is taken from front or back. Then using Optical Character Recognition (OCR), the license plate number is extracted. Then whether it is odd or even is calculated, by observing the last digit. The license plate number is sent to website which shows the challan along with the date as shown in Fig. 8.

If the license plate number is odd and date recorded is even, then accordingly some amount in the current challan is shown. Same for vice versa. Else 0 will be shown if date and number both are even or odd.

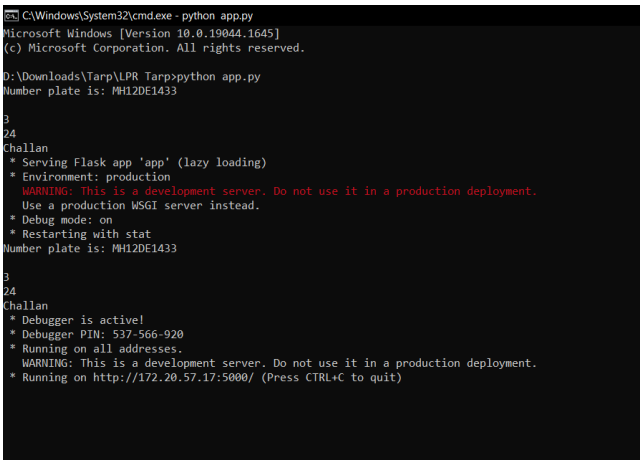

Fig. 8- Sending license plate number to Web App.

If the user has any previous challan pending, then accordingly fees will add up. Finally, along with taxes, the total amount will be displayed. The user will then click on pay now button.

This will redirect them to a QR Code as shown in Fig. 8 with the specified amount. This is done using an API Provided by Google. The user will use any of the UPI Apps such as PhonePe, Google Pay, Paytm etc. to scan and pay the fees. The money will directly go the Traffic Police of that zone.

V. RESULT

It can be seen that the license plate number was recognized and odd and even was calculated.

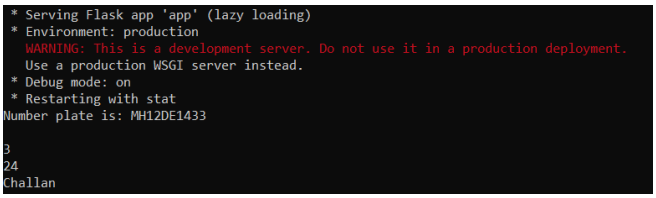

Fig. 9 – Calculation of odd and even

We can see the terminal screen in Fig. 9 for reference. Here, 3 denotes last digit of license plate which is odd, and 24 denotes date, which is even. Hence challan has to be collected.

Fig. 10– Calculation of challan

The user will be able to see his license number and date of challan as shown in Fig. 10. He will also be able to see any previous violations, if any and tax.

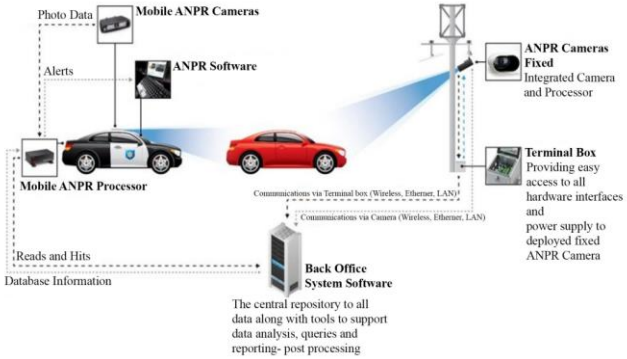

Fig. 11– Alternative method for capturing license plate image.

In Fig. 11 we described alternatives positions for placing the camera. Either on poles or on a police car.

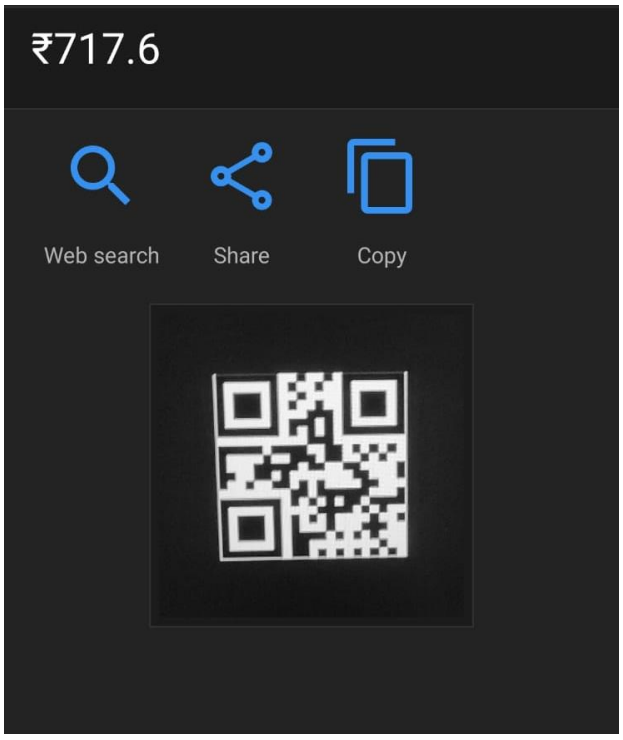

Fig. 12– QR Code showing total fees to pay

The user will then proceed to pay through UPI as per the QR Code that is generated. A sample QR code generated has been shown in Fig. 12. Always a new QR code will be generated as per the total fees calculated, done using a google API.

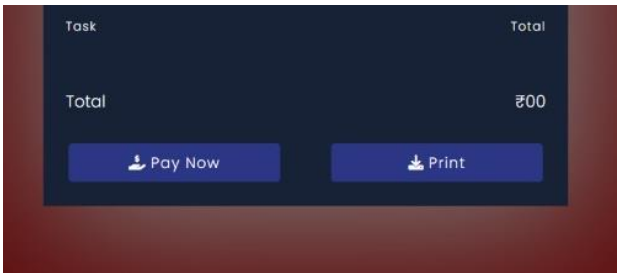

Fig. 13– Print option

Furthermore, print option is also added in case the user wants a hardcopy of the receipt, as shown in Fig. 13.

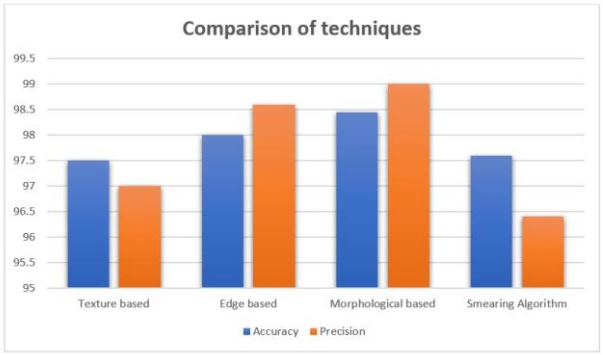

Fig. 14– Comparison of techniques

As per the data obtained from [20], [21] and [22] we plotted a graph, that is Fig. 14. We observe that Morphological based approach, that is the base for OCR technique, has the highest Accuracy and Precision.

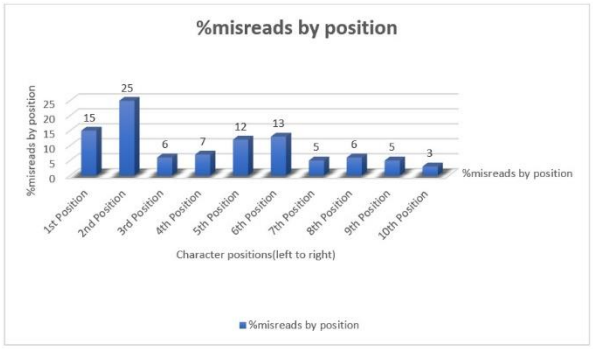

Fig. 15 – Percentage misreads by positions in ANPR Algorithm.

As shown Fig. 15, as the characters increase, misreads decrease, as the algorithm knows what characters are supposed to be next.

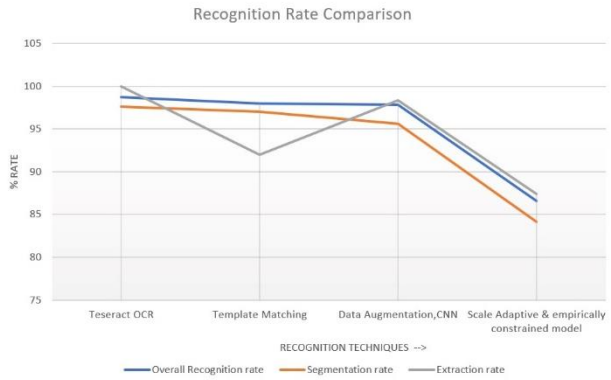

Fig. 16 – Recognition rate comparisons of different algorithms

As depicted in Fig. 16, the overall recognition rate, segmentation rate and extraction rate is the highest for Tesseract OCR. That is why we choose to adopt this algorithm.

Some methods that can be considered is Razor Pay, PayU, PayPal etc, but then amount will have to be manually entered then.

VI. FUTURE WORK

The QR Code Generated requires a UPI Recipient, that is, the UPI of some traffic police officer of that zone. The QR code here is automatically generated as per the amount but this is not a payment code.

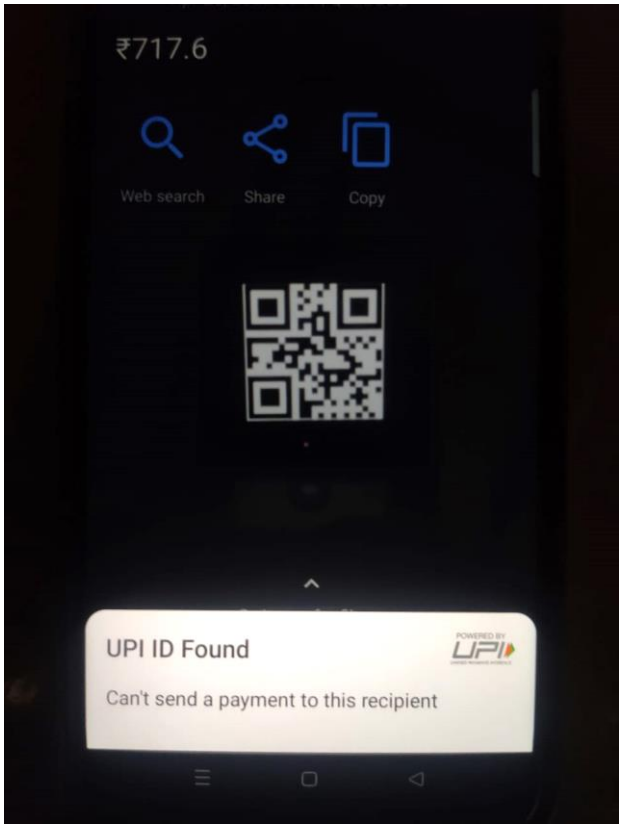

Fig. 17 – UPI of traffic police officer is needed

The QR Code has to be first linked with traffic police officer, as shown in Fig. 17 and for which license number it is being paid has to integrated.

Also, we plan to create a login module so that user can see his or her own details only. This is an enormous work as vehicle owners all over India will have to create their login and upload their vehicle information.

Fig. 18 – Various types of license plates

There are various kinds of license plate shapes, color, font size, font style and other modifications. Common everyday examples are in Fig. 18. Although algorithm can be trained to be accurate but sometimes there is additional text in the license plate that may still interfere with the recognition. There is no absolute solution to this except to make modifications illegal and to adopt a standard license plate rule.

VII. FUTURE SCOPE

As air pollution levels are increasing day by day, especially in metro cities, this will serve an excellent way to aware people and minimize manual work and labour. If the payment method and user login are completed by all vehicle owners, definitely it will be the best method of challan collection and may expand to other violation types.

Also, Automatic Number Plate Recognition (ANPR) systems have evolved from difficult to set up, expensive fixed-based applications to simple mobile applications that may be used with the "point and shoot" method. This is made possible by the development of software that could run on less expensive PCs and non-specialized hardware, eliminating the need to specify the direction, angles, speed, and size of the plate as it passes through the camera's field of vision. Smaller cameras that could scan licence plates quickly, as well as smaller, more durable computers that could fit in police vehicles, allowing officers to patrol on a regular basis with the benefit of real-time number plate identification.

Currently it is only based for Indian license plates and cannot work on other countries license plates, as plate size and format change. Separate methods needed to be adopted for other countries.

This system can be used for other violations also such as speeding, rash driving, wrong lane etc. but accordingly some changes would have to be made in the algorithm.

VIII. CONCLUSION

The operation of an automated vehicle license plate recognition system was analyzed in this paper in terms of software aspects. Its operation is divided in two image processing phases: the phase of license plate segmentation and the phase of license plate processing and character recognition. After the extraction of license plate number, the last digit is compared with the current date. If the last digit is odd or even and date is even or odd respectively, then challan is generated. We also proposed a QR code system for payment.

REFERENCES

[1] License Plate Welekar, Amit R., et al. "Analysis of Rules Violation & Efficient E-Challan Generation Using OCR In Real Time Traffic." *International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET)* 4.4 (2018): 339-343.

[2] Duan, T. D., Du, T. H., Phuoc, T. V., & Hoang, N. V. (2005, February). Building an automatic vehicle license plate recognition system. In Proc. Int. Conf. Comput. Sci. RIVF (Vol. 1, pp. 59-63).

[3] Kim, K. K., Kim, K. I., Kim, J. B., & Kim, H. J. (2000, December). Learning-based approach for license plate recognition. In Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop (Cat. No. 00TH8501) (Vol. 2, pp. 614-623). IEEE.

[4] Du, S., Ibrahim, M., Shehata, M., & Badawy, W. (2012). Automatic license plate recognition (ALPR): A state-of-the-art review. IEEE Transactions on circuits and systems for video technology, 23(2), 311-325.

[5] Anagnostopoulos, C. N. E., Anagnostopoulos, I. E., Loumos, V., & Kayafas, E. (2006). A license plate-recognition algorithm for intelligent transportation system applications. IEEE Transactions on Intelligent transportation systems, 7(3), 377-392.

[6] Balamurugan, G., Punniakodi, S., Rajeswari, K., & Arulalan, V. (2015, February). Automatic number plate recognition system using super-resolution technique. In 2015 International Conference on Computing and Communications Technologies (ICCCT) (pp. 273-277). IEEE.

[7] Agrawal, R., Agarwal, M., & Krishnamurthi, R. (2020, March). Cognitive number plate recognition using machine learning and data visualization techniques. In 2020 6th International Conference on Signal Processing and Communication (ICSC) (pp. 101-107). IEEE.

[8] Mehta, M. N., & Sharma, A. A Preliminary Study on Effect of Odd-Even Rule on Air Pollution Scenarios of Delhi. Journal of Water Resource & Pollution Studies, 2(2).

[9] BARIK, K. (2016). Road Rationing and Economic Gains: Assessing Delhi's Odd—Even Formula. *Economic and Political Weekly*, 12-14.

[10] Welekar, A. R., Rajesh, S. D., Shubham, M. B., Tanvee, B., & Wawre, R. (2018). Analysis of Rules Violation & Efficient E-Challan Generation Using OCR In Real Time Traffic. International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), 4(4), 339-343.

[11] Phelawan, J., Kittisut, P., & Pornsuwancharoen, N. (2012). A new technique for distance measurement of between vehicles to vehicles by plate car using image processing. Procedia Engineering, 32, 348-353.

[12] Polak, M., Zhang, H., & Pi, M. (2009). An evaluation metric for image segmentation of multiple objects. Image and Vision Computing, 27(8), 1223-1227.

[13] Chen, Y. S., & Cheng, C. H. (2010). A Delphi-based rough sets fusion model for extracting payment rules of vehicle license tax in the government sector. Expert Systems with Applications, 37(3), 2161-2174.

[14] Gautam, Aayushi, et al. "Vehicle pollution monitoring, control and challan system using MQ2 sensor based on internet of things." Wireless Personal Communications 116.2 (2021): 1071-1085.

[15] Ali, Farheen, Himanshu Rathor, and Wasim Akram. "License plate recognition system." 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). IEEE, 2021.

[16] Srinath, Raghunandan, et al. "Smart Vehicle Recognition And E-Challan Generation System." 2020 International Conference for Emerging Technology (INCET). IEEE, 2020.

[17] Barreto, Fabian, et al. "Object Detection and e#Challan Generation Systems for Traffic Violation: A Review." International Journal of Research in Engineering, Science and Management 4.5 (2021): 54-57

[18] Babu, R. Naren, V. Sowmya, and K. P. Soman. "Indian car number plate recognition using deep learning." 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT). Vol. 1. IEEE, 2019.

[19] Shashirangana, Jithmi, et al. "Automated license plate recognition: a survey on methods and techniques." IEEE Access 9 (2020): 11203-11225.

[20] Critical Study of License plate Detection Method in LPR System Satellite Conference ICST SD 2016 International Conference on Science and Technology for Sustainable Development, Kuala Lumpur, MALAYSIA, May 24-26, 2016.

[21] Sanap, P.R. and Narote, S.P., 2010, November. License Plate Recognition System-Survey. In AIP

Conference Proceedings (Vol. 1324, No. 1, pp. 255-260). American Institute of Physics.

[22] Sharma, G., 2018. Performance analysis of vehicle number plate recognition system using template matching techniques. Journal of Information Technology & Software Engineering, 8(2), pp.1-9.

[23] Angeline, L., Teo, K. and Wong, F., 2009, July. Smearing algorithm for vehicle parking management system. In *2nd Seminar on Engineering and Information Technology* (pp. 331-337).