Automatic Traffic E-challan Generation Using Computer Vision

Sinhgad College of Engineering, Vadgaon, Bk.

Manisha R. Dhage[1],Gaurav V Patil[2],Shrayank J Mistry[3] ,Prathamesh N Tambe[4], Parag H Nankar[5]

[1]Assistant Professor,Department of Computer Engineering,Sinhgad College of Engineering,Pune

[2,3,4,5]U.G Students,Department of Computer Engineering, Sinhgad College of Engineering, Vadgaon BK, Pune

Email-mrdhage.scoe@sinhgad.edu[1],ptgaurav10@gmail.com[2],shraymist@gmail.com[3],tpratham11198@gmail.com[4],paragnankar@gmail.com[5]

Abstract -: **The Automatic recognition of license plate will increase the accuracy of managing traffic, the automatic detection and localization of license plate is an important part.License plate detection involves segmentation and extraction of license plate region from the whole image which contains vehicle.A new deep learning network structure and training method based on CNN was designed, and designed network structure was used to detect and locate the license plate automatically.The system proposed by us involves automatic detection of vehicles that break the traffic rules at respective signals and registration number for every vehicle is recognized using CNN and deep learning. The vehicle number detected is searched in the database for type of vehicle and owner's information.This information is used to generate challan in the name of the person who owes the vehicle directly and instantly and send appropriate fine message to the owner.**

Keywords:**-E - Challan,CNN,otsu,contour, tesseract.**

I. Introduction

Automatic Traffic E-challan Generation using Computer Vision this system is to use to automate e-challan generation when vehicles cross zebra crossing during traffic signal. The system is based on detection of the vehicles that have broken the rule i.e the vehicles which stop in front of zebra line, license plate detection of the vehicle breaking rule and effective challan generation.We will have a database server it has information of all the vehicle registered.The rule violation on the streets of the city is on the rise and zebra crossing violation is one of the rule that is broken very often. People crossing the roads majorly rely on zebra crossing for to safely cross the roads. If vehicles are on the zebra crossing it is inconvenient and hazardous. Are system which can detect this violation instantly will help the people as well as vehicle owners to be safer. Above all, we hope to provide a smooth, easy and hassle-free system for the traffic authority.

II. Related-Works

In India, recently the working system is where two people (Traffic Police Officers) are involved in generation of E - challan.

One person looks at the video and takes images when signal is red. After that he checks for the vehicle which has broken the rule.The vehicles which are on Zebra crossing,number of those vehicles are taken by that person.He gives the number to the second person.

Second person takes the number and fills the form of violating traffic rule by that vehicle,by inserting that number in the form.And then the challan is generated by that system.

III. System-Implementation

The system being developed has two main features:

1) Detecting the vehicles that have violated the rule.

2) Generating e-challan for the vehicles that have broken the rule.

3.1) Image Preprocessing: Image is taken by the camera situated above the signal when it goes red. Image captured is full of noise and unwanted data. The image above the zebra crossing is cropped and noise is removed to make it ready for license plate detection.

3.2) License Plate detection: Image obtained after preprocessing has not one but Multiple License plates in them. Each license plate of the vehicles that have broken the rules will be detected by the system.

3.3) Feature Extraction: After all registration plates are detected the exact number of each vehicle needs to be recognized and extracted. The alphabets and numbers on the number plate are extracted and recognized.

3.4) Database Verification: The obtained registration number is searched in the database entire database until a match is found. Once a match has been found entire details of the user is obtained.

3.5) E-challan Generation: Using the user information obtained an e-challan will be generated in the name of the owner and notification will be sent to owner of vehicle that has violated the rule via SMS or mail.

IV. Methodology and Flowchart

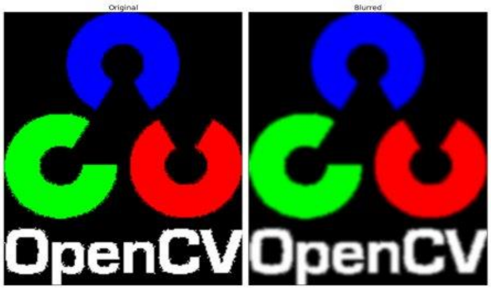
4.1) Pre-processing

Input Image

Crop Image

Gaussian blurring is highly effective in removing gaussian noise from the image.

ImgBlurred = cv2.GaussianBlur(img, (5,5), 0)



Original Blurred

Convert RGB to Gray Image -

RGB - Gray : Y <- 0.299 * R + 0.587 * G + 0.114 * B

Gray - RGB : R <- Y , G <- Y , B <- Y , A <- Max(Channel range)

This is done with the help of cvtColor() from cv.

Threshold - Sobel operator is discrete differentiation operator. It compute an approximation of the gradient of an image intensity function.

G = sqrt(Gx^2 + Gy^2)

4.2) Recognize License Plate

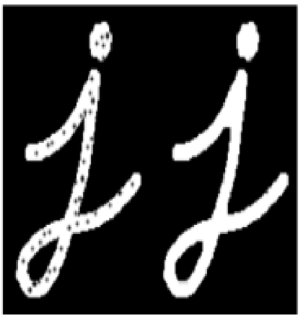
- *Get Structuring Element (Rectangle)*

element = cv2.getStructuringElement(shape=cv2.MORPH_RECT,ksize(17,3))

- *Closing*

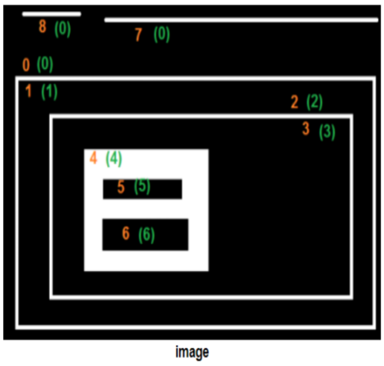
Used for closing small holes in object

cv2.morphologyEx(src=threshold_img, op=cv2.MORPH_CLOSE, kernel=element, dst=morph_img_threshold)



- *Find Contours*

Contours indicate the boundary pixels of objects



```
contours = cv2.findContours(morph_img_threshold,mode=cv2.RETR_EXTERNAL,method=cv2.CHAIN_APPROX_NONE)
```

1. First parameter is source image (morph_img_threshold)
2. Second parameter is contour retrieval mode mode=cv2.RETR_EXTERNAL, using this mode only outer edges are returned.
3. Third is contour approximation method.

- *Bounding Rectangle*

```
x,y,w,h = cv2.boundingRect(cnt)
```

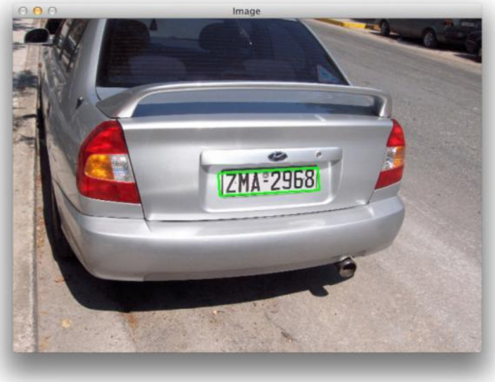
(x,y) top left corner co-ordinates of plate

w = width

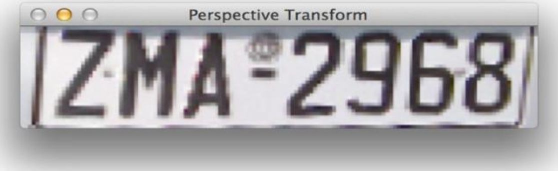
h = height

Plate image can be obtained by adding co-ordinates and width and height. Hence number plate is detected

4.3) Number Plate Detection



```
img = img.transform((new_width, height), Image.AFFINE,(1, m, -xshift if m > 0 else 0, 0, 1, 0), Image.BICUBIC)
```



- *Creating Threshold Image*

Extract the Value component from the HSV color space and apply otsu thresholding to reveal the characters on the license plate.

```
gaus = cv2.adaptiveThreshold(image,255,cv2.ADAPTIVE_THRESH_GAUSSIAN_C,cv2.THRESH_BINARY,115,1)
```

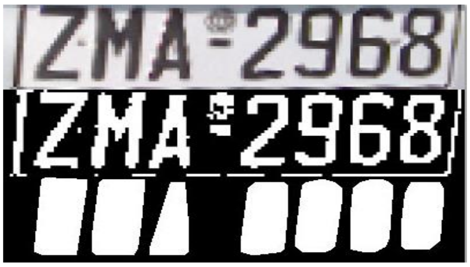
- *Creating Black and White Image*

```
ret, labels1 = cv2.connectedComponents(gaus)

# Map component labels to hue val

label_hue = np.uint8(179*labels1/np.max(labels1))

blank_ch = 255*np.ones_like(label_hue)

labeled_img = cv2.merge([label_hue, blank_ch, blank_ch])

# cvt to BGR for display

labeled_img = cv2.cvtColor(labeled_img, cv2.COLOR_HSV2BGR)

# set bg label to black

labeled_img[label_hue==0] = 255

labeled_img[label_hue!=0] = 0
```

- *Extracting numbers and alphabets*

```
,cnts,hierarchy= cv2.findContours(lthresh, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

for c in cnts:

     area = cv2.contourArea(c)

    x,y,w,h = cv2.boundingRect(c)

    rect_area = w*h

     extent = float(area)/rect_area

    if(rect_area > 1000 and rect_area < 9999):

        print(rect_area)
```

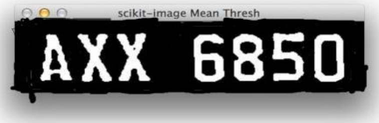
```
cv2.rectangle(labeled_img,(x,y),(x+w,y+h),255,0)
        pixelpoints = np.transpose(np.nonzero(mask))
cv2.drawContours(labeled_img,cnts,-1,(255,255,0),1)
```



Optical Character Recognition

Using Tesseract python library

```
        import Image
        from tesseract import image_to_string
        print image_to_string(Image.open('test.png'))
        print image_to_string(Image.open('test-english.jpg'), lang='eng')
```



Output: -

AXX 6850

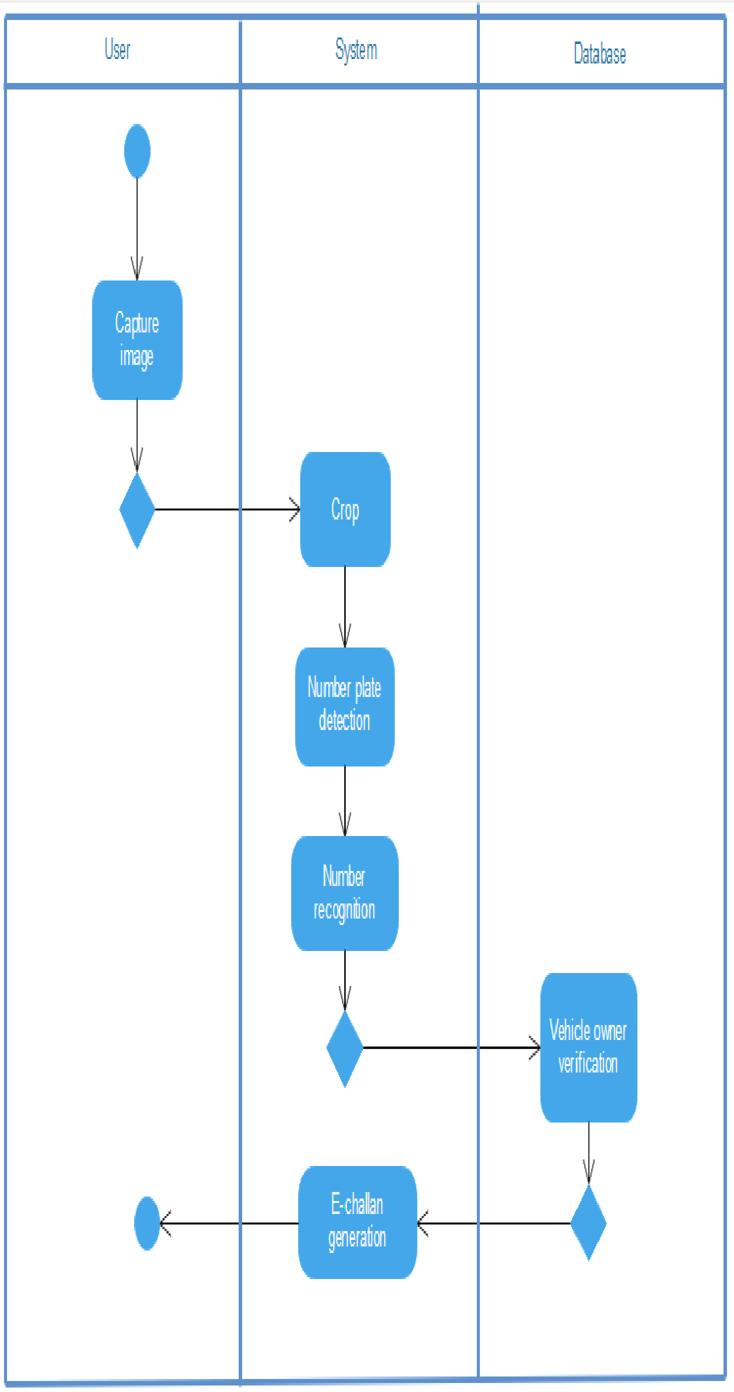
4.4) Database Verification:

Information regarding user details can be obtained from the api given below. Api for vehicle details : parivahan.gov.in/rcldstatus/

4.5) E-challan Generation:

Challan is generated by system using details obtained about user and SMS is send using api given below. Api for SMS sending : way2sms

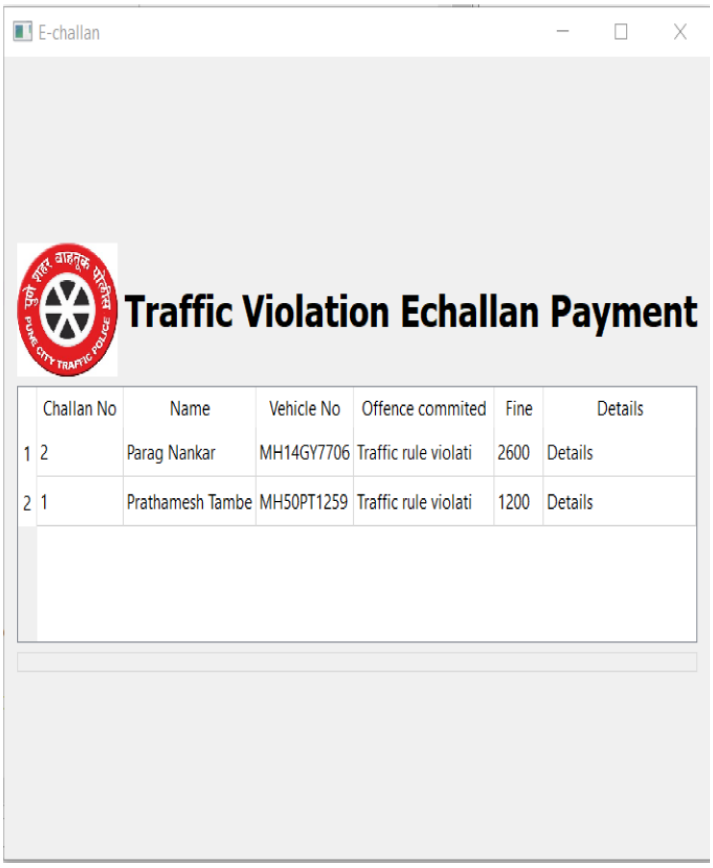
V. Activity Diagram



VI. Results And Discussions

6.1 Main GUI Snapshots

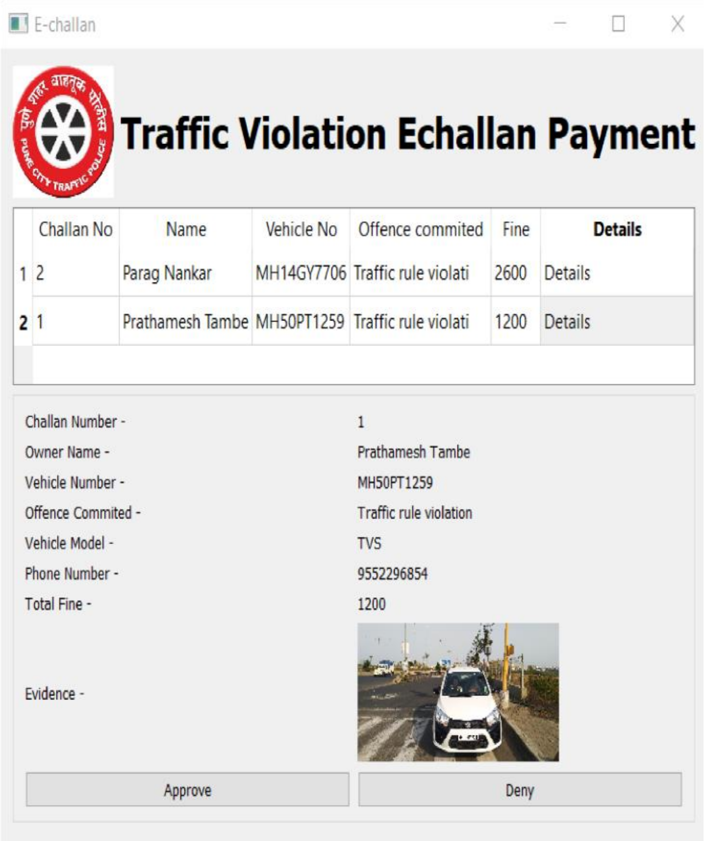
Initial Window



Traffic Violation Echallan Payment

	Challan No	Name	Vehicle No	Offence commited	Fine	Details
1	2	Parag Nankar	MH14GY7706	Traffic rule violati	2600	Details
2	1	Prathamesh Tambe	MH50PT1259	Traffic rule violati	1200	Details

After clicking on User 1

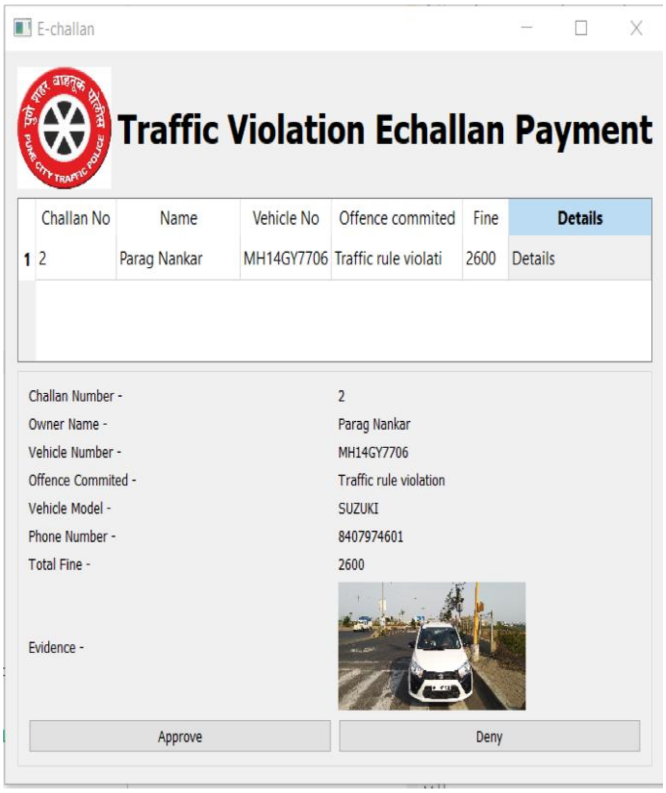


Traffic Violation Echallan Payment

	Challan No	Name	Vehicle No	Offence commited	Fine	Details
1	2	Parag Nankar	MH14GY7706	Traffic rule violati	2600	Details
2	1	Prathamesh Tambe	MH50PT1259	Traffic rule violati	1200	Details

Challan Number -	1
Owner Name -	Prathamesh Tambe
Vehicle Number -	MH50PT1259
Offence Commited -	Traffic rule violation
Vehicle Model -	TVS
Phone Number -	9552296854
Total Fine -	1200
Evidence -	

After clicking on User 2:



Traffic Violation Echallan Payment

	Challan No	Name	Vehicle No	Offence commited	Fine	Details
1	2	Parag Nankar	MH14GY7706	Traffic rule violati	2600	Details

Challan Number -	2
Owner Name -	Parag Nankar
Vehicle Number -	MH14GY7706
Offence Commited -	Traffic rule violation
Vehicle Model -	SUZUKI
Phone Number -	8407974601
Total Fine -	2600
Evidence -	

6.2 Discussions

Traffic E - Challan generation is totally python based application.It collects data from camera and detect the violated vehicle’s license number.On this application,first user will get list of violated incidents.Details of incidents is displayed on this application with all necessary information and image.After user approves, this application sends message to vehicle owner when vehicle breaks the rule.

VII. Conclusion

The objective of the Government to automate rule violation detection, recognise owner of vehicle, and generate e-challan can be obtained using the system proposed in this paper. The Automatic e-challan generation is a tool for the Government which can help her perform all duties that are currently done manually in an efficient and a smarter way and completely automatically. The system will make vehicle owner more aware of the rules as well as make it much safer for pedestrians.

- **References**

[1] TANG Yufeng, LUO Bin,TANG Jin, WU Fubu. New heuristic algorithm of license plate location based on statistical feature. [J] Computer Engineering and

Application, 2015,51(6): 154-158.

[2] Abolghasemi V, Ahmadyfard A. An edge-based color-aidedmethod for license plate detection[J]. Image & Vision Computing, 2009, 27(8):1134-1142.

[3] Comelli P,Ferragina P,Granieri M N. Optical Recognition of Motor Vehicle License Plates[J], Vehicular Technology, IEEE Transactions on, 1995,44: 790-799.

[4] SI Qing-ya. The design and implementation of License Plate Recognition System based on robust illumination[D]. Nanjing: Nanjing University of Posts and Telecommunications, 2013.

[5] Deneen K M V. An Algorithm for License Plate Recognition Applied to Intelligent Transportation System[J]. IEEE Transactions on Intelligent Transportation Systems, 2011, 12(3):830-845.