Number Plate Detection, recognition and velocity estimation from surveillance videos

Internship Project Report

by

Namburi GNVV Satya Sai Srinath

Under the guidance of

Dr.Deepak Mishra

Department Of Avionics

Indian Institute Of Space Science and Technology

Thiruvananthapuram

December, 2018

Declaration by author

This is to declare that this report has been written by Namburi GNVV Satya Sai Srinath. No part of the report is plagiarised from other sources. All information included from other sources have been duly acknowledged. I aver that if any part of the report is found to be plagiarised I shall take the full responsibility of it

By
Namburi GNVV Satya Sai Srinath
27/12/2018

Acknowledgements

I would like to thank Dr.Deepak Mishra for guiding me throughout the internship period. I would also like to thank Aswathy and Sowmya for helping with lab system and clarifying doubts. My fellow interns helped me a lot during this period and I am very grateful to them. Finally, I would like to thank my parents for allowing me for this internship despite vacation. Finally, I am grateful to The Almighty for giving me patience and courage to undertake Internship

Contents

1 Introduction **6**

2 Approach Used **7**
2.1 Localisation of number plate 7
2.2 Recognition of characters on localised number plate . . 8
2.3 Estimating velocity 8
2.4 Designed approach 8
2.5 Procedure 9
2.5.1 Finalization of dataset 9
2.5.2 Detecting characters 11
2.5.3 Estimating velocity 14

3 Results and Discussions **16**

4 Conclusions and Recommendations **17**

5 Appendix **21**

List of Figures

2.1 Flowchart design of the implemented process to localise and detect number plate from video footage 8
2.2 Flowchart design of the implemented process to estimate velocity from video footage 9
2.3 Example images from BRNO Dataset 9
2.4 Results obtained after training.
Left: Couldn't detect due to wrong dataset. Right: Detected but varying bounding boxes 10
2.5 Example images from UFPR-ALPR dataset. 10
2.6 Extracting number plates from test images 11
2.7 Left: A video frame. Right: Number plate extracted by algorithm 11
2.8 TopLeft: Denoised image. TopRight Erosion operation BottomLeft: Dilation operation BottomRight: Edges removed by bordering and other preprocessing steps . . . 12
2.9 Improper Segmentation by Watershed Algorithm. . . . 12
2.10 Cropped images by segmentation using histogram approach 13
2.11 Histogram of image along the width as X-axis. Look for the peaks to get location of characters. An abnormal peak at the end is because of abruptness (cropped) . . 13
2.12 Initial Architecture of MLP(MultiLayer Perceptron) designed to detect digits in MNIST 14
2.13 Example images used for testing MLP architecture. Left failed due to borders. Right failed as training and testing images has different background(MNIST has black background with white numbers) 14
2.14 Details of EMNIST Dataset. 15
2.15 Left: Accuracy vs epochs of training images
Right: Loss vs epochs of testing images 15
2.16 Estimating velocity of a vehicle using Kalman filters and optical flow 15

5.1 An architecture proposed to detect number plate by three steps. 21
5.2 An architecture proposed to detect number plate by two steps. 21

Chapter 1

Introduction

India has been suffering from traffic problems over the past century. Many new innovations in this field has been bought up during the last two decades. One common problem faced by the traffic community in India is to drive slowly so that accidents can be avoided which has not been solved to the fullest yet. But technological shift bought us a possibility from manual checking of velocity and then imposing a fine to automatically detect velocity and number plate from a surveillance camera and then fine accordingly.

My project is to develop an end-end system which can

1. Localisation of Number Plate
2. Recognise the characters on Number Plate
3. Detect velocity of the vehicle

As the part of internship, I looked into the problem as three different tasks and tried to solve it.

Chapter 2 discusses briefly about the traditional approaches used so far to tackle the problem and the approach designed along with technical content.

Chapter 3 discusses about the results obtained during internship and interpreting them along with the problems faced.

Chapter 4 discusses the conclusions drawn so far and future work, developments on the project

Chapter 2

Approach Used

So, the problem statement has been subdivided into three parts

2.1 Localisation of number plate

Given a traffic/dashboard video there will be an area of interest where our number plate is located. Detecting a number plate is of atmost importance so that imposing fine to the correct person can be achieved.

In [1], the entire process of number plate localisation and character recognition has been reviewed and discussed very thouroughly. From it, combining of two or more features seems to be a promising technique which has good accuracy and its complexity and computational expensive can be overcome by the usage of good GPU for processing.

NVIDIA AI City Challenge 2018 focuses mainly on velocity detection and tracking of vehicles where [2] has used number plate detection as a step to track vehicles in different video streams.

The concept of CNN to detect number plate has also been put forward in [3],[4] which has localised and detected number plates from images.

In [5], a more robust technique YOLO[] has been applied to detect vehicles in the first stage, and then applying YOLO(which is trained on number plates) on the images to detect number plates and then apply YOLO which is trained on characters to segment them accordingly.

2.2 Recognition of characters on localised number plate

After detecting number plate, characters are to be extracted so that it can be used further. [5] uses YOLO again which is pretrained to detect characters. OCR(Optical Character Recognition) is an old but an effective technique which can be applied to number plates to extract characters. Template matching is an old technique whose accuracy may fall low due to noise and other factors[6]

2.3 Estimating velocity

Optical flow is a technique which is used to estimate the points in the next couple of frames and kalman filters can be used[7] to update and have a track on the vehicle. [2] has estimated velocity using a set of parameters and analysing the loss functions.

[8] uses Mean Shift Algorithm to track vehicles across different frames thus estimating the velocity

2.4 Designed approach

Figure 2.1: Flowchart design of the implemented process to localise and detect number plate from video footage

Several architecture designs has been made to implement and analyse the following parameters

1. Processing speed
2. Complexity of Algorithm
3. Accuracy achieved

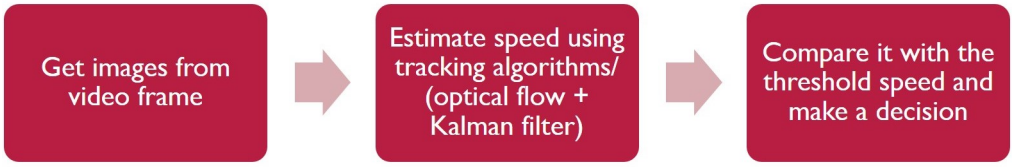

Figure 2.2: Flowchart design of the implemented process to estimate velocity from video footage

2.5 Procedure

2.5.1 Finalization of dataset

As hardware setup is not realisable due to shortage of time, it has been thought to use the existing datasets. NVIDIA AI Challenge 2018 works on the same problem and serves as an excellent dataset. But the challenge has been completed and the dataset has been closed for usage. So, as a first step BRNO dataset of Low Quality License Plates[12] has been suggested and the initial work has been proceeded with this dataset.

Figure 2.3: Example images from BRNO Dataset

YOLO(You Only Look Once)[9] is an existing, highly reliable and robust architecture which can be used for object recognition. So, BRNO dataset has been trained with YOLO model by changing few parameters accordingly (Filters = 30, classes = 1 from formulas)

But when a test frame is given to YOLO, it performed badly to detect number plate.

Figure 2.4: Results obtained after training.
Left: Couldn't detect due to wrong dataset. Right: Detected but varying bounding boxes

After thourough understanding of algorithm, the problem has been identified as choosing an inappropriate dataset for this purpose i.e the train images has only number plates while the test images has number plates along with car and other background.

So, UFPR-ALPR dataset[13] has been requested and further work has been done on this dataset. When a test image is send to the architec-

Figure 2.5: Example images from UFPR-ALPR dataset.

ture, it is able to properly detect number plate from the background. Accuracy is very decent when compared to the previous one. 1500 images (900 positive examples and 600 negative examples) are given as training dataset and 400 images (300 positive and 100 negative examples) given as validation dataset and trained for 1300 epochs. Text files for negative examples are empty files with name of image. By observing the results, it is guaranteed and promising that if the network is trained with a larger dataset and for several epochs than this one, the accuracy can be pretty good. As the process is obvious, this step has been paused for a while and looked to detect characters

Figure 2.6: Extracting number plates from test images

2.5.2 Detecting characters

After localising number plate, the characters are to ber extracted. Initially, OCR(Optical Character Recognition) using Tesseract has been applied on image but the accuracy is very bad. The reasons may be

Figure 2.7: Left: A video frame. Right: Number plate extracted by algorithm

1. The bounding boxes generated by the localised algorithm in the previous step may also include some edges which becomes very important to distract OCR

2. Noise/Very low resolution of the number plates as it has been extracted from a video sequences

So, some preprocessing steps has been done but still the output of OCR is not satisfying.
Adaptive Gaussian filter, Adaptive mean filtering, thresholding, erosion, dilation are a few preprocessing steps applied on the image.

Figure 2.8: TopLeft: Denoised image. TopRight Erosion operation BottomLeft: Dilation operation BottomRight: Edges removed by bordering and other preprocessing steps

So, it has been decided to segment characters and then pass it to a neural network so that it can classify the character.

Segmentation of characters

Segmentation of characters using the preexisting "Watershed Algorithm" didn't give a proper output of characters. Reason has to be explored in greater detail.

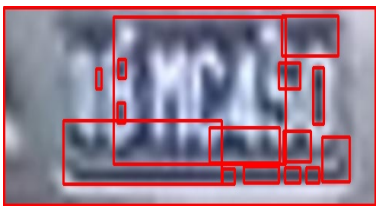

Figure 2.9: Improper Segmentation by Watershed Algorithm.

Histogram Approach

So, the histogram of preprocessed number plate has been plotted with x-axis having the dimension of width. The area under which the character is present has values different from that of background. This property has been used to extract number plate characters.

Figure 2.10: Cropped images by segmentation using histogram approach

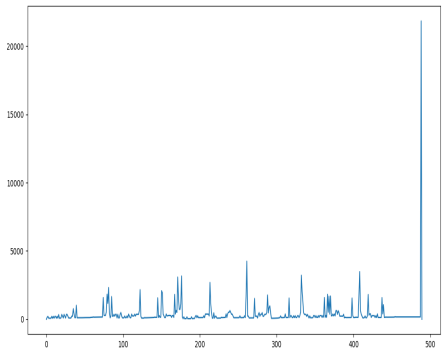

Figure 2.11: Histogram of image along the width as X-axis. Look for the peaks to get location of characters. An abnormal peak at the end is because of abruptness (cropped)

Neural networks for classification

As a subproblem, to test whether the extracted number plate characters are good enough to be recognised by neural network, a simple MLP(Multi Layer Perceptron) has been trained with MNIST dataset[10] and when the test images are given, it didn't give satisfactory results. The problems has been explored and are

1. The edges of the images makes huge difference while detecting number
2. MNIST has images which is trained with black background and white digits. But the images that was given to test has the opposite.

So, to solve these problems, CNN was implemented instead of MLP

Layer (type)	Output Shape	Param #
flatten (Flatten)	multiple	0
dense (Dense)	multiple	401920
dropout (Dropout)	multiple	0
dense_1 (Dense)	multiple	5130

Total params: 407,050
Trainable params: 407,050
Non-trainable params: 0

Figure 2.12: Initial Architecture of MLP(MultiLayer Perceptron) designed to detect digits in MNIST

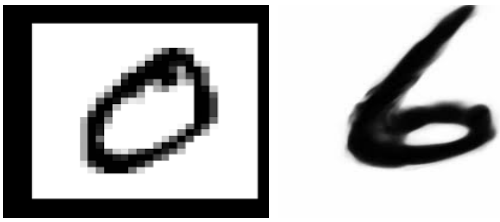

Figure 2.13: Example images used for testing MLP architecture. Left failed due to borders. Right failed as training and testing images has different background(MNIST has black background with white numbers)

Convolutional Neural Networks

While the basic CNN's are able to recognise the digits correctly, it failed with background variation. So, to work on with alphabets, EMNIST dataset[11] has been choosen which is much more challenging than MNIST dataset.EMNIST dataset comes with matlab files rather than direct images unlike MNIST dataset. So, data downloading and processing steps has been changed a bit to access them. "EMNIST Balanced Class" has been choosen as it has almost same number of example images for all the images thus avoiding overfitting/biasing problems while training. **results of emnist dataset recognising digits and characters. say about number of epochs and all architecture summary** So, it is promising that, by following this procedure with more data and better preprocessing, we can achieve higher accuracy. So, for time being, this step has been stopped

2.5.3 Estimating velocity

Basic idea to estimate velocity is by generating contours for each vehicle and observe it's motion in subsequent frames so that relative velocity can be estimated. In this regard, only little work has been done

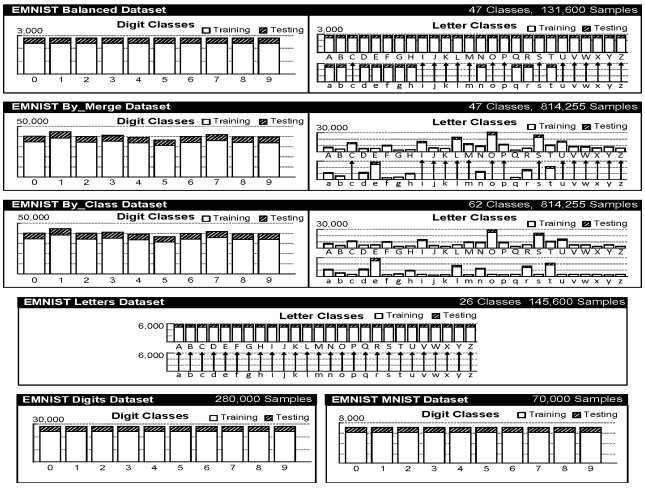

Figure 2.14: Details of EMNIST Dataset.

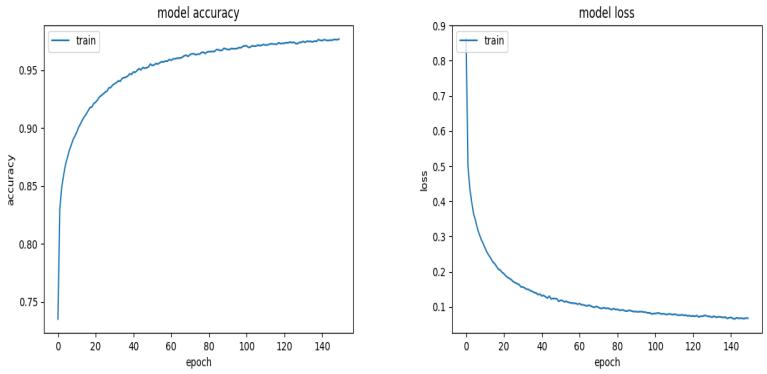

Figure 2.15: Left: Accuracy vs epochs of training images
Right: Loss vs epochs of testing images

Figure 2.16: Estimating velocity of a vehicle using Kalman filters and optical flow

Chapter 3

Results and Discussions

The following works has been completed

1. Convert traffic footage into image frames
2. Localise number plate from image frames
3. Crop number plate from this using coordinates
4. Preprocess image so that it can be useful for character segmentation
5. Segment characters using histogram approach
6. Processed characters given to MLP, CNN to get number plates which has been trained with MNIST, EMNIST

Problems faced are

1. Availability of dataset and choosing proper dataset.
2. Non availability of ground truths for number plates of a video sequence. Given a video footage, if we have number plates data of all the vehicles in video it is possible to get squared mean error. But there are no videos which has number plate data of vehicles to validate the results obtained

Chapter 4

Conclusions and Recommendations

So, extraction of number plate and character segmentation has been done with satisfactory results. Further improvements can be made as

1. A local dataset for Indian Road scenarios can be made because Autos are not extensively researched and there is lack of such dataset. If produced/taken from traffic police community as a tie-up, it can be very useful to develop our own algorithms and dataset by labelling
2. There is no dataset which has ground truth of number plates of ll the vehicles in a video thus calculation of squared means error becomes invalid. So, when making dataset, data related to this field can also be marked
3. Estimation of velocity has not been extensively covered due to time constraints and it can be made as an extension
4. Number plates after cropping has very low resolution. Super-resolution techniques can also be used to improve the resolution which brings another novel approach in number plate detection
5. A comparison study of these three architectures (one implemented, two others given in appendix) can be made in aspects of accuracy, processing speed, complexity, memory allocation etc;
6. CNN which are robust to background color has to be implemented so that if the segmentation part is not proper, the more complex CNN needs to manage and predict proper output
7. CNN + RNN can be implemented directly to get number plate instead of histogram segmentation + CNN. So, as one is already

implemented, if the other one(CNN + RNN) is done a comparison study regarding speed, accuracy, complexity can be understood in great detail

8. Publications describing number plate detection, character segmentation and velocity estimation all together are very few in number

Bibliography

[1] Shan Du, et.al, "Automatic License Plate Recognition (ALPR): A State-of-the-Art Review" in IEEE Transactions on Circuits and Systems for Video Technology

[2] Zheng Tang, et.al, "Single-camera and inter-camera vehicle tracking and 3D speed estimation based on fusion of visual and semantic features" in IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) 2018

[3] Francisco Delmar Kurpiel, et.al, "Convolutional Neural Networks for License Plate Detection in Images" in International Conference on Image Processing (ICIP), 2017

[4] Sergio Montazzolli, Claudio Jung "Real-Time Brazilian License Plate Detection and Recognition Using Deep Convolutional Neural Networks" in SIBGRAPI Conference on Graphics, Patterns and Images, 2017

[5] Rayson Laroca, et.al, "A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector" in Computer Vision and Pattern Recognition(CVPR), 2018

[6] N. H. C. Yung, et.al, "Recognition of Vehicle Registration Mark on Moving Vehicles in an Outdoor Environment" in IEEE Conference On Intelligent Transportation Systems, Proceedings, Itsc, 1999, p. 418-422

[7] Shuai Hua, Manika Kapoor, David C. Anastasiu, "Vehicle Tracking and Speed Estimation from Traffic Videos" in IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) 2018

[8] Aparna Shivhare, Vinita Choudhary "Object Tracking in Video Using Mean Shift Algorithm: A Review" in International Journal of Computer Science and Information Technologies, 2015

[9] Joseph Redmon, et.al. “You Only Look Once” in Computer Vision and Pattern Recognition(CVPRW), 2015

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. “Gradient-based learning applied to document recognition.” Proceedings of the IEEE, 86(11):2278-2324, November 1998.

[11] Gregory Cohen, et.al. “EMNIST: an extension of MNIST to handwritten letters” in Computer Vision and Pattern Recognition, 2017

[12] Jakub Sochor et.al, “Holistic recognition of low quality license plates by CNN using track annotated data” in 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2017

[13] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonçalves, W. R. Schwartz, and D. Menotti, “A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector” in 2018 International Joint Conference on Neural Networks (IJCNN), July 2018, pp. 1–10.

Chapter 5

Appendix

Figure 5.1: An architecture proposed to detect number plate by three steps.

Figure 5.2: An architecture proposed to detect number plate by two steps.