Vehicle Detection and License Plate Detection

Omar ElSedawy
OmarElSeadawy@aucegypt.edu

AUC

Introduction

Improving traffic management systems by building a framework that is capable of automatically detecting illegal offences in the streets and being able to identify the cars responsible for such offences would be a great feat. In this paper, I design and experiment with the first two phases of the framework that are responsible for vehicle detection, traffic line detection and license plate detection.

Those models were built using deep learning approaches using the YOLOv5 pre-trained models and extending those models with extra training using two new data sets. The first data set specifically designed for vehicle detection and traffic line detection while the other data set is customized for the License plate recognition part of the system. I was capable of designing both models and reaching around 0.8 mAP in the vehicle and traffic line detection, and almost 1 mAP in the license plate recognition.

Datasets

In this project, I used two different data sets to build two different models. The first data set was obtained for detecting vehicles and identifying their types in addition to identifying traffic lines in the street and this data set had around 340 images, while the other UFPR-ALPR data set was obtained from [1] and is only released for academic research purposes and has around 4000 images that were annotated.

I tried combining both of the datasets in order to build a more generalized detector but it was not very effective as they had different class annotations; however, I do believe finding a bigger combined dataset would definitely improve the performance of these models. I did some data preprocessing by adjusting the annotations to suit the YOLO model and discard any addition unneeded information.

Future Work

Future work for this project, would be to combine both detectors using a common much bigger dataset to improve the speed and accuracy of detections, build a framework for detecting the offences based on the output of the detectors and finally extend the LPR to be capable of extracting characters and identifying the license plate numbers automatically.

References

[1] L. A. Zanlorensi L. S. Oliveira G. R. Gonçalves W. R. Schwartz R. Laroca, E. Severo and D. Menotti. A robust real-time automatic license plate recognition based on the yolo detector. International Joint Conference on Neural Networks (IJCNN), July 2018.

[2] A.; Borovec J.; Changyu L.; Hogan A.; Diaconu L.; Ingham F.; Poznanski J.; Fang J.; Yu L.; et al. Jocher, G.; Stoken. Yolov5. 2021.

[3] Y.; Li L.; Wang Z.; Zhou Y. Lian, J.; Yin. Small object detection in traffic scenes based on attention feature fusion. 2021.

[4] Ruiz-del-Solar Javier Verschae Rodrigo. Object detection: Current and future directions. Frontiers in Robotics and AI.

Vehicle and Traffic Line Detection

The specs of the laptop to train both models: Intel i7-8750H CPU @ 2.2 GHZ, 16 GB RAM and NVIDIA GTX 1060. The image size used was 620, the batch number of images were 6. The first experiment was to test the vehicle detection and traffic line detection using the first dataset of 340 images using both YOLOv3 and YOLOv5 and comparing results.

As we can from Table 1, using YOLOv3 or YOLOv5 didn't matter much in terms of mean average precision, the results were satisfactory; however, the main problem lies in the dataset as it is very small (340 images) compared to other researchers who use datasets with over 10000 images and that is why I was not able to achieve the same results as them. The figure also shows some output examples from the model predictions.

Figure 1: Predictions of vehicle type and traffic lines

Experiments	Precision/Recall	mAP@0.5
50 Epoch YOLOv3l	0.8844/0.7712	0.8333
100 Epoch YOLOv5l	0.9411/0.746	0.8361

Table 1: Results

License plate detection

The data set for this part of the system was much bigger and thus was much slower when it came to training the model. I was only capable of running the model for 25 epochs over the provided 4000+ images and came up with very satisfying results in very short time using the YOLOv5l model as well.

Experiments	Precision/Recall	mAP@0.5
YOLOv5l	0.963/0.9528	0.9864
ALPR[1]	1.0/1.0	N/A

Table 2: Results

The table shows I was able to achieve very high results and close to the best results from people in the literature. It could've required an increase in the number of epochs to be able to reach the performance of the ALPR proposed by [1]. Below the figure also shows the difference between the labeled test batch and the predicted test batch output from the model.

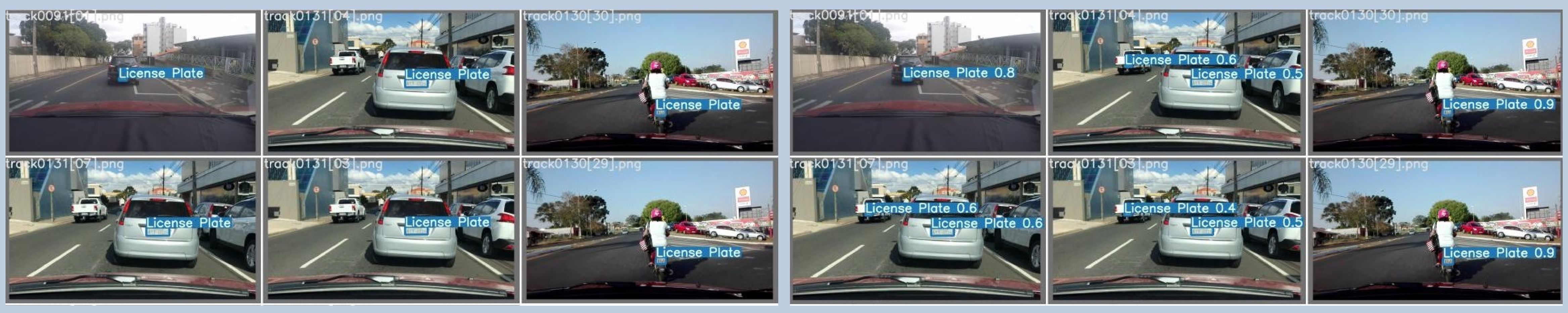

Figure 2: Labeled data (left); Predicted Output (right)

We can see from the figures on the right that the model was capable of identifying all the license plates that were labeled in the testing data and even identify extra plates that existed in the image but were not annotated which shows the effectiveness of this model.

Extension of this LPD is to extract the bounded boxes from these images and convert the model into LPR by doing character segmentation on the license plate numbers/letters and use temporal redundancy across multiple frames to improve the quality and be able to correctly extract the information from any moving license plate.