

License Plate Recognition System for Taiwanese Vehicles Using Cascade of YOLOv4 Detectors

Chun-Cheng Lin, Mao-Huan Hsu, and Cheng-Yu Yeh*

Department of Electrical Engineering, National Chin-Yi University of Technology,
57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung 411030, Taiwan

(Received January 12, 2023; accepted May 31, 2023)

Keywords: license plate recognition (LPR), You Only Look Once (YOLO), object detection, deep learning

In this paper, we present a study of the license plate recognition (LPR) system for Taiwanese vehicles using a cascade of You Only Look Once version 4 (YOLOv4) detectors. The LPR system is composed of a vehicle detection model, a license plate (LP) detection model, an LP corner prediction model, and an LPR model. Herein, the pretrained YOLOv4 model was directly applied to vehicle detection. The YOLOv4 framework was adopted in the LP detection and LP recognition models, performing transfer learning on each model. Furthermore, to enhance the accuracy of the LPR system, an LP corner prediction model was developed in this study to predict the four corner positions of an LP to perform a perspective transformation on the plate for alignment purposes. The experimental results show that our LPR system achieves an accuracy of 98.88% when tested on 2049 images of the application-oriented LP dataset, outperforming most LPR systems reported in the literature.

1. Introduction

Deep learning is one of the hottest research topics today and has been widely used in various research fields. In addition, it has fruitful research results in computer vision and image recognition.^(1–18) For example, the most common techniques of deep learning in image recognition applications are image classification,^(4–6) object detection,^(7–10) semantic segmentation,^(11,12) face recognition,^(13,14) and license plate recognition (LPR).^(15–18)

An object detection task requires the detection and recognition of specific objects in an image or a video. Today, object detection models that are commonly used involve EfficientDet⁽⁸⁾ and You Only Look Once (YOLO).^(9,10) The Common Object in Context (COCO) dataset⁽¹⁹⁾ can be used to train an object detection model to recognize up to 80 types of object, including people, cars, cats, and dogs. Thus, such techniques can be widely applied to different fields, such as intelligent transportation, intelligent image analysis and retrieval, smart home, and smart security. However, the well-established object detection techniques have a limitation in that they can only detect objects in a pretrained model, such as the 80 types of object in the COCO

*Corresponding author: e-mail: cy.yeh@ncut.edu.tw
<https://doi.org/10.18494/SAM4328>

dataset. However, transfer learning must be performed to detect objects that are not contained in the COCO dataset. In this manner, object detection techniques can be applied to various disciplines.

However, LPR technology is also combined with deep learning in the recent literature^(3,15–18) to provide higher recognition efficiency and robustness. The traditional LPR system generally includes three main steps: vehicle detection, license plate (LP) location, and LP character recognition. Nowadays, an end-to-end model is used in more research studies to simultaneously integrate the above three processes to create the LPR system. Typical LPR applications include the parking lot's automatic management and charging system, the automatic billing system on the roadside, and so forth. Moreover, LPR is often combined with the intelligent transportation system,^(20–22) which integrates the management of people, roads, and vehicles, and provides real-time information to improve the efficiency and safety of the transportation system. Its application field is broad.

Given this, we propose a cascade of You Only Look Once version 4 (YOLOv4) detectors for the Taiwan LPR system. That is, the model architecture of YOLOv4⁽⁹⁾ is adopted for the three models of vehicle detection, LP detection, and LP character recognition in the system. Nevertheless, transfer learning will be required for the LP detection and LP character recognition models. In addition, an LP corner prediction model is also developed to find the four corners of the LP and to perform a perspective transformation on the tilted LP image for LP alignment. This function is enabled after LP detection but before LP character recognition because the corrected LP image will help improve the accuracy of the LPR system.

In the following, the datasets used in this study and the annotations for the datasets are described in Sect. 2. An LPR system for Taiwanese license plates is presented in Sect. 3. The experimental results are provided in Sect. 4 and the conclusions of this study are given in Sect. 5.

2. Materials and Annotations

2.1 Materials

In this study, two datasets were used. The first is the application-oriented license plate (AOLP) dataset.^(23,24) It contains 2049 images representing various locations, times, and traffic and weather conditions. The AOLP dataset is categorized into three subsets, namely, access control (AC), law enforcement (LE), and road patrol (RP), to provide image samples for the three major applications. The second is a dataset that we collected, with a total of 11652 images. Then, we provided these two datasets to the three models of LP detection, LP corner prediction, and LP character recognition for training and testing the models. The detailed information is presented in Table 1.

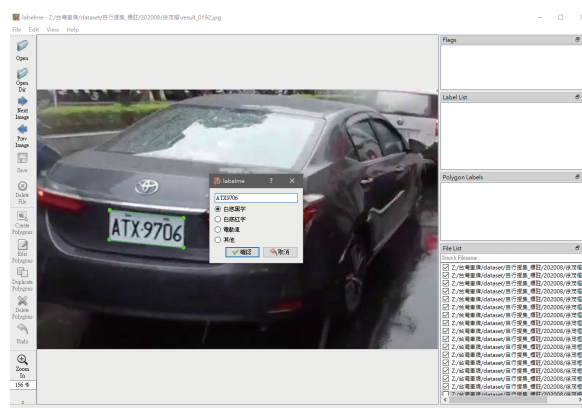
Table 1
Materials used for LP detection, LP corner prediction, and LP character recognition models.

Item	Our collection	AOLP	Total
Training set	8156	0	8156
Test set	3496	2049	5545

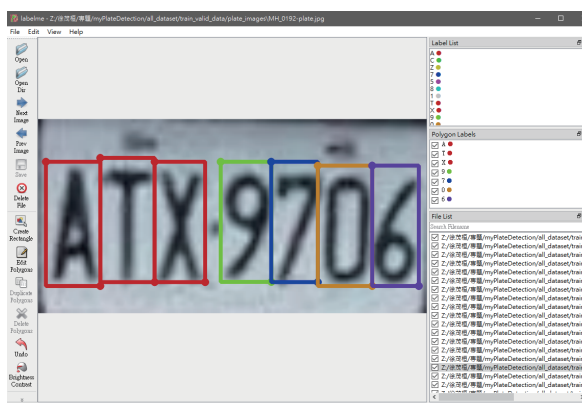
2.2 Annotations

As it is necessary to label our dataset, a set of tailor-made labeling software for LP images was developed, as shown in Fig. 1. First, Fig. 1(a) shows the first stage of labeling. When an LP image is opened in the software, the labeler marks the four corners of the LP with this software in a clockwise order from the upper left, upper right, lower right, and lower left. When the four corner points are marked, a small window will pop up for the labeler to enter the number and type of LP: black characters on white background, red characters on white background, electric vehicles, and others. In such manner, we can obtain four lines of labeled information of this LP: (i) the four corners of the LP, (ii) the LP number, (iii) the LP type, and (iv) the bounding box of the LP, of which (iv) is calculated on the basis of (i).

Among the four lines of labeled information, the information in (iii) and (iv) will be used to train the LP detection model; the information in (i) will be used when training with the LP corner prediction model. Next, the second labeling stage is for the LPR model, as shown in Fig. 1(b). The LP image is corrected first on the basis of the information in (i) and (ii), and then the labeler labels the bounding box of each character.



(a)



(b)

Fig. 1. (Color online) Schematic of developed annotation tool: (a) labeling for LP corners, characters, and type; (b) labeling for LP characters' bounding boxes.

3. Proposed LPR System

Figure 2 shows the flow chart of the LPR system. The input image will be processed successively by the four models of the YOLOv4 detector for vehicle detection, LP detection, LP corner prediction for LP correction, and LP character recognition, which is the entire LPR process. Among them, the three models of vehicle detection, LP detection, and LP character recognition are of YOLOv4 architecture. Transfer learning is required for the latter two to perform the corresponding tasks. An LP corner prediction model is developed in this study to correct the LP image. The four models are then introduced sequentially.

In this paper, the pretrained YOLOv4 model is directly applied in the vehicle detection task because it is one of the state-of-the-art models in today's object detection. In the LP detection task, transfer learning is required for the YOLOv4 model. Therefore, Table 1 shows the data used for model training and testing. Furthermore, in addition to performing the LP detection, this model is also trained to recognize the LP types, including black characters on white background (Type I), red characters on white background (Type II), electric vehicle LPs (Type III), and other LPs (Type IV), as shown in Table 2. In other words, this model has two functions: LP detection and LP recognition.

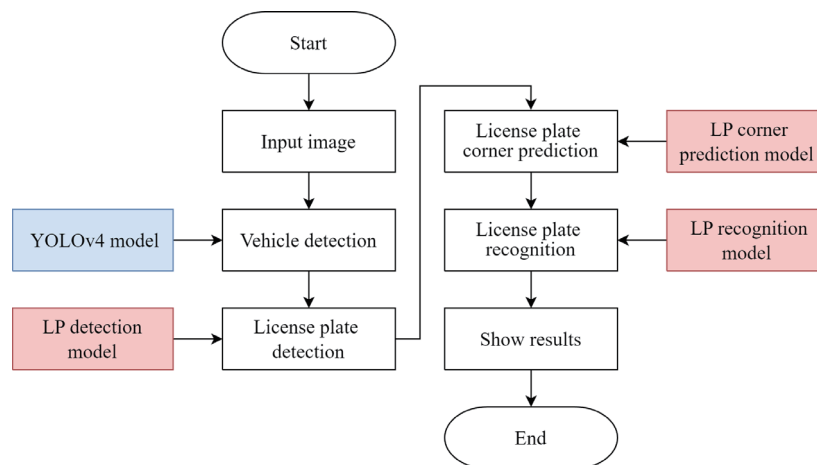


Fig. 2. (Color online) Flowchart of presented LPR system.

Table 2
Collected LP types used for LP detection model.

License plate type	Training set	Test set
Type I	7527	5261
Type II	547	243
Type III	55	24
Type IV	27	17
Total	8156	5545

The framework of the LP corner prediction model is shown in Fig. 3, and all sizes are expressed in the form of $W \times H \times C$. The size of the input images is $128 \times 64 \times 3$; then, two convolutional layers using kernel size = 3×3 , filters = 32, and stride = 1 are connected. Conv_BR (3×3 , 32, 1) means that this block contains Convolutional layer + Batch Normalization + ReLU activation three-layer connection. The MP_Dropout (2×2 , 0.25) in the figure means that the block contains 2×2 Max Pooling and Dropout of 0.25 two-layer connection. A total of nine convolutional layers are used in this model. Sigmoid activation is adopted in the final output layer to output four corner coordinates of the LP. The mean squared error (MSE) loss function and Adam optimizer are adopted during the training of this model, and training is performed at the parameter configuration of batch size = 1024, and epoch = 300; then, the model with the lowest loss is saved. After the four corner coordinates of the LP are obtained with this model, perspective transformation can be performed to correct the image of the LP for subsequent LP character recognition.

Finally, in the LP character recognition task, each character in the LP is considered as an object and then the YOLOv4 is adopted for transfer learning, so this model needs to recognize 0–9 and A–Z (excluding O), a total of 35 character objects. The images used in this model training are shown in Table 1, including a total of 52599 character objects.

In the inference phase of the LP character recognition model, after all the character objects in the LP image are detected, the LP number of the LP image can be obtained by sorting these detected character objects from left to right on the basis of the position of each object's bounding box to complete the LPR task.

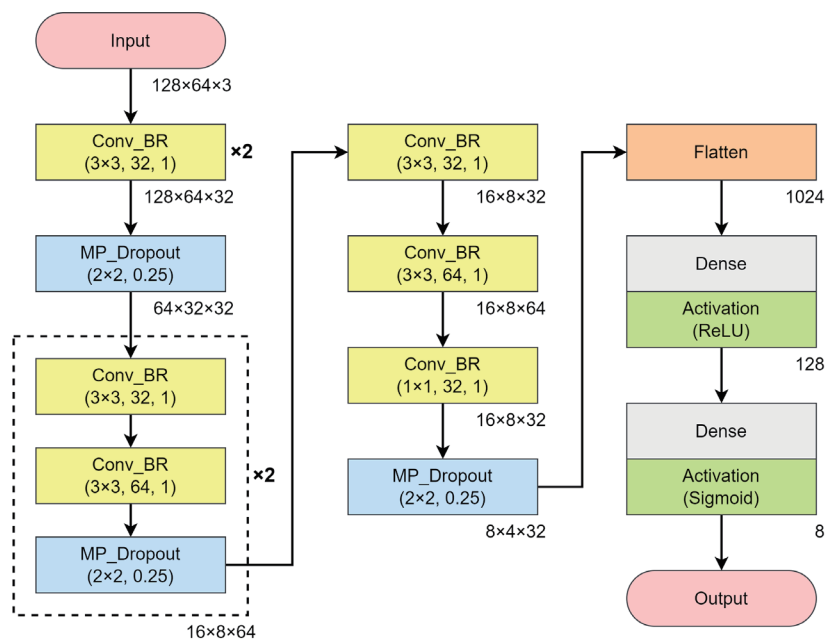


Fig. 3. (Color online) Framework of presented LP corner prediction model.

4. Experimental Results

The overall workflow of our LPR system is shown in Fig. 4. After the image is input, vehicle detection, LP detection, LP corner prediction, and LP recognition will be performed consecutively to obtain the LP number and type. The experiments in this study include determining the performance of LP detection, the accuracy of LP recognition, and the execution time of the entire system. The data used in the test is shown in Table 1 and the development environment is shown in Table 3.

The performance test of LP detection includes the following: The first is determining the LP detection rate, which can accurately detect the proportion of the number of LPs in all LP images. The formula is shown in Eq. (1), where the number in the denominator is 5545, as shown in Table 2. The second is the LP type accuracy, which can correctly determine the proportion of the number of LP types in all LP images. The formula is shown in Eq. (2) and the test results are shown in Table 4. An overall LP detection rate of 98.81% and an LP type accuracy rate of 98.59% can be obtained by the method mentioned in this paper.

$$LP\ detection\ rate = \frac{\text{number of detected license plates}}{\text{number of all test license plates}} \quad (1)$$

$$LP\ type\ accuracy = \frac{\text{number of correct type recognition of license plates}}{\text{number of all test license plates}} \quad (2)$$

Then, among the 5479 LPs successfully detected, the confusion matrix of four LP types and various true positive rates (*TPRs*) and precision values were further analyzed, as shown in Table 5. Overall, except for the poor *TPR* of Type IV, the results of the other three categories are excellent, with *TPR* above 99% and precision above 93%.

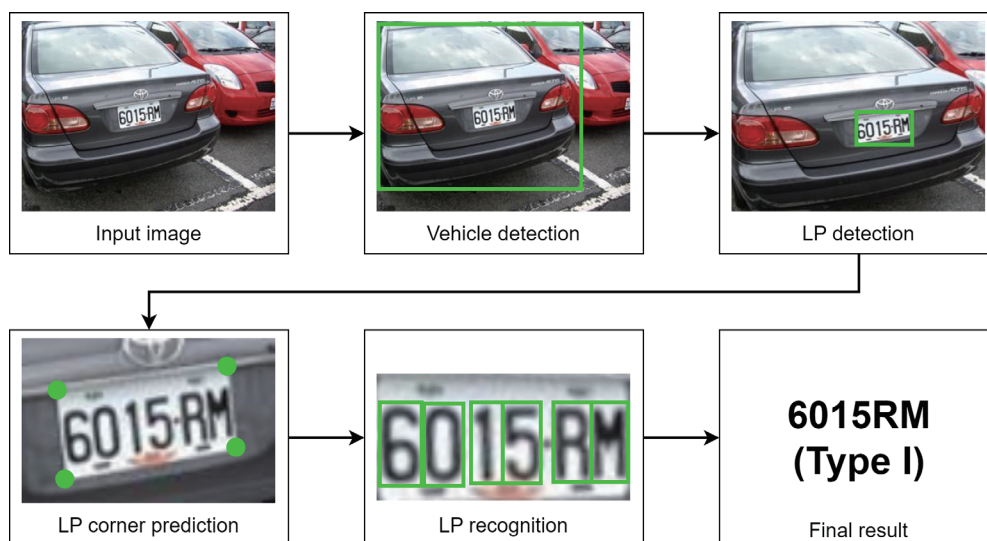


Fig. 4. (Color online) Schematic diagram of proposed LPR system.

Table 3
Development environment for LPR system.

Programming language	Python
Library	CUDA, PyTorch, Numpy, OpenCV
Hardware	OS: Windows 11 (64-bit)
	CPU: AMD Ryzen 5 3600 (3.6 GHz)
	RAM: DDR4-3200 (16 GB)
	GPU: NVIDIA GeForce RTX 3070 (8 GB)

Table 4
Performance of LP detection rate and LP type accuracy.

Item	LP detection rate (%)	LP type accuracy (%)
AOLP	99.71	99.61
Our collection	98.28	98.00
Overall	98.81	98.59

Table 5
Confusion matrix for LP type recognition.

Real case	Predicted case				TPR (%)	Precision (%)
	Type I	Type II	Type III	Type IV		
Type I	5210	5	1	0	99.88	99.92
Type II	1	235	0	1	99.16	97.51
Type III	0	0	14	0	100.00	93.33
Type IV	3	1	0	8	66.67	88.89

In the LPR performance test, two experiments were conducted. The first is performing LPR directly without employing the proposed LP corner prediction. The second is performing LP corner prediction prior to the LPR stage. Table 6 gives the test results, which show that the accuracies obtained without and with LP correction are 79.53 and 98.47%, respectively. This result confirms that the accuracy obtained with LP correction is considerably improved and the effect is marked.

The last test deals with the operating time of the LPR system. In this test, a total of 3313 images were used for testing and obtaining the average operating time in each sub-item and the overall system, as shown in Table 7. The experimental results show that the average operating time of the system mentioned in this paper is 0.113 s when executing the overall LPR system, and the real-time work is allowed.

5. Discussion

Table 8 gives an accuracy comparison for LPR among this study and four recently published counterparts. Ideally, models must be tested using the same dataset for performance comparison. Therefore, Refs. 15–18 were employed as comparison counterparts because they used the open-access AOLP dataset as the test data. As previously mentioned, the AOLP dataset is composed of three subsets: AC, LE, and RP, in which 681, 757, and 611 images are given, respectively.

Table 6
LPR accuracy analyses.

Item	LPR accuracy with/without performing LP alignment	
	Without (%)	With (%)
AOLP	83.75	98.88
Our collection	77.06	98.23
Overall	79.53	98.47

Table 7
Processing time in every model and entire system.

Item	Time spent (s)	Input image size (W × H)
Vehicle detection	0.040	608 × 608
LP detection	0.036	608 × 608
LP corner prediction	0.003	128 × 64
LP recognition	0.034	320 × 320
Overall	0.113	–

Table 8
Accuracy comparison among this work and counterparts in AOLP dataset.

Method	Accuracy (%)			
	AC	LE	RP	Overall
Proposed	98.68	99.60	98.20	98.88
[15] ¹	97.60	99.50	99.00	–
[16]	93.75	98.91	90.42	94.66
[17]	97.80	97.40	96.30	97.20
[18]	96.91	98.01	98.20	97.70

¹Only 1891 images were used for the test.

As shown in Table 8, it is experimentally validated that the proposed model provides accuracies of 98.68, 99.60, and 98.20% in the AC, LE, and RP test data, respectively. The accuracy of this study shows that the AC and LE test data rank first and the RP test data ranks second, in comparison with their Refs. 15–18 counterparts. Moreover, this work substantially outperforms Ref. 16 on the AC and RP test data.

In terms of overall accuracy, the proposed model achieved an accuracy as high as 98.88%, which outperformed Ref. 16 (94.66%), Ref. 17 (97.20%), and Ref. 18 (97.70%). It must be pointed out that only 1891 images out of the AOLP dataset (2049 images) were used for the performance test in Ref. 15; thus, the overall accuracy in Ref. 15 was not provided in Table 8. Finally, in this study, the excellent performance of the presented LPR system was experimentally demonstrated.

6. Conclusions

In this paper, we proposed using multi level YOLOv4 detectors to develop the Taiwan LPR system. The test data showed that this method performed well. Moreover, the presented LP corner prediction model was integrated into the system to correct the LP image before the LPR.

As a result, the accuracy was improved significantly. Its performance was more excellent and leading compared with those of the related methods described in the literature. Furthermore, the LP type recognition was appended in this system and its accuracy reached 98.59%.

In the future, we plan to develop an end-to-end LPR model to combine all the functions distributed over this work into a single model, which is expected to provide higher performance than this study.

Acknowledgments

This research was financially supported by the Ministry of Economic Affairs under grant number 110-EC-17-A-02-S5-008, Taiwan.

References

- 1 L. Jing and Y. Tian: IEEE Trans. Pattern Anal. Mach. Intell. **43** (2021) 4037.
- 2 M. M. Adnan, M. S. M. Rahim, A. Rehman, Z. Mehmood, T. Saba, and R. A. Naqvi: IEEE Access **9** (2021) 50253.
- 3 Lubna, N. Mufti, and S. A. A. Shah: Sensors **21** (2021) 3028.
- 4 K. He, X. Zhang, S. Ren, and J. Sun: Proc. IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2016) 770. <https://doi.org/10.1109/CVPR.2016.90>
- 5 P. Tang, X. Wang, B. Shi, X. Bai, W. Liu, and Z. Tu: IEEE Trans. Neural Netw. Learn. Syst. **30** (2019) 2244.
- 6 X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer: Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (IEEE, 2022) 12104. <https://doi.org/10.1109/CVPR52688.2022.01179>
- 7 S. Ren, K. He, R. Girshick, and J. Sun: IEEE Trans. Pattern Anal. Mach. Intell. **39** (2017) 1137.
- 8 M. Tan, R. Pang, and Q. V. Le: Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (IEEE, 2020) 10778. <https://doi.org/10.1109/CVPR42600.2020.01079>
- 9 A. Bochkovskiy, C. Y. Wang, and H. Y. Liao: CoRR. (2020) <https://arxiv.org/abs/2004.10934> (accessed May 2020).
- 10 C. Y. Wang, A. Bochkovskiy, and H. Y. Liao: CoRR. (2020) <https://arxiv.org/abs/2011.08036> (accessed December 2020).
- 11 K. He, G. Gkioxari, P. Dollár, and R. Girshick: Proc. IEEE Int. Conf. Computer Vision (IEEE, 2017) 2980. <https://doi.org/10.1109/ICCV.2017.322>
- 12 O. Ronneberger, P. Fischer, and T. Brox: CoRR. (2015) <https://arxiv.org/abs/1505.04597> (accessed May 2020).
- 13 F. Schroff, D. Kalenichenko, and J. Philbin: Proc. IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2015) 815. <https://doi.org/10.1109/CVPR.2015.7298682>
- 14 J. Deng, J. Guo, N. Xue, and S. Zafeiriou: Proc. IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2019) 4685. <https://doi.org/10.1109/CVPR.2019.00482>
- 15 T. Björklund, A. Fiandrotti, M. Annarumma, G. Francini, and E. Magli: Pattern Recognit. **93** (2019) 134.
- 16 Y. Kessentini, M. D. Besbes, S. Ammar, and A. Chabbouh: Expert Syst. Appl. **136** (2019) 159.
- 17 Z. Selmia, M. B. Halimab, U. Pal, and M. A. Alimi: Pattern Recognit. Lett. **129** (2020) 213.
- 18 C. L. Tung, C. H. Wang, and B. S. Peng: Math. Probl. Eng. **2021** (2021) 3723715.
- 19 COCO dataset: <http://cocodataset.org/> (accessed June 2020).
- 20 L. Hu, X. Q. Zhou, X. Zhang, F. Wang, Q. Q. Li, and W. G. Wu: IET Intell. Transp. Syst. **15** (2021) 1093.
- 21 K. Lu, J. Liu, X. Zhou, and B. Han: IEEE Trans. Intell. Transp. Syst. **22** (2021) 2535.
- 22 S. H. Wan, X. L. Xu, T. Wang, and Z. H. Gu: IEEE Trans. Intell. Transp. Syst. **22** (2021) 4487.
- 23 G. S. Hsu, J. C. Chen, and Y. Z. Chung: IEEE Trans. Veh. Technol. **62** (2013) 552.
- 24 AOLP dataset: <https://github.com/HaoRecog/AOLP> (accessed July 2021).