Automatic Indonesian License Plate Recognition with YOLO As Object Detector

IBNAZHIFI. NEVINDRA[1], MUSLIMAH. SEKAR LARASATI[2], BLESSINDA. KIMBERLEY[3]
[1] *Faculty of Computer Science and Information System, Gunadarma University, Jakarta-Indonesia*
[2] *School of Electrical Engineering and Informatics, Bandung Institute of Technology, Bandung-Indonesia*
[3] *Faculty of Technology Information and Science, Parahyangan Catholic University, Bandung-Indonesia*

Abstract- This paper created a license plate and vehicle number detection system to facilitate parking management and regularity. This system aims to change the traditional way of doing manual recording into computational automation of vehicle numbers. In this study, the Indonesian License Plate Recognition System consists of 2 processes, namely Object Plate Detection and Digit Number Recognition. The construction of the Object Plate Detector is carried out using the YOLOv4 transfer learning model. Meanwhile, Digit Number Recognition is built using the YOLOv5 transfer learning model. In this study, the YOLO model is used which is the current state-of-the-art object detector. With an accuracy of 89% for localization and detection of car plates and an accuracy of 87% for classifying characters on vehicle number plates.

Indexed Terms- License Plate Recognition, Object Plate Detection, Digit Number Recognition, YOLOv4, YOLOv5

I. INTRODUCTION

One of the issues that the government must address in order to establish an effective and efficient transportation system in Indonesia is the growing number of private motorized vehicles. In terms of both the physical infrastructure and the information system that underpins it. Only in the capital city of Jakarta, the number of new private motorized vehicle submissions per day has reached 4,000 units per day in 2019, with 3,000 units of two-wheeled motorized vehicles and 1,000 units of four-wheeled motorized vehicles. Several years ago, between 2012 and 2016, the annual rise in the number of passenger automobiles was 6.48 percent.

These figures highlight the urgent need for smart transportation systems that can sustain future increases in the number of motorized vehicles. One of these may be fulfilled by using technology into the provision of effective parking services. Today's technological advancements are one of the most important factors in increasing efficiency in people's everyday lives, especially in terms of transportation convenience. Artificial Intelligence (AI), sometimes known as AI, makes it easier for people to automate practically every action they do, including smart parking services, which may be used to construct a smart transportation system.

Smart parking services take advantage of AI's role in creating a safe and comfortable transportation system, particularly for Indonesian motorists. From 2020 to 2027, the rise in market share of smart parking services is expected to reach 12.6 percent on a global scale. The CAGR (Compound Annual Growth Rate) ratio, or average annual growth rate, is smaller on the Asia Pacific scale.

II. RELATED WORKS

A. License Plate Recognition System

License Plate Recognition System is a system to recognize each plate on a vehicle and then read all the digits listed on the number plate. Automation of plate reading and digitizing the digits on the plate will facilitate many traditional jobs that are often distracted by human error. There are several papers that have discussed this with a tendency to use two different neural networks for efficiency and accuracy problems. In paper [1], the License Plate Recognition System is executed using YOLOv2 to help the system identify vehicle plates. After detecting the presence of a vehicle plate in the image, then paper [1] uses ResNet to digitize the digits. In addition, paper [2] uses the same systematic steps to execute the License Plate Recognition System but uses a different neural network. Paper [2], uses YOLOv4 to perform both stages, both detecting the presence of vehicle plates and digitizing digits.

Therefore, for better accuracy in this paper, we created a license plate detection system consisting of 2 stages. In the first stage our system will detect the location of the license plate which will be detected with YOLOv4. After the number plate is detected and cut, the second stage is carried out, namely the detection of each number plate digit using YOLOv5.

B. You Only Look Once (YOLO)

You Only Look Once or YOLO is an advanced real-time object detection system using artificial neural networks and a full convolution model [4]. Objects detected by YOLO can be helpful in classifying objects, giving identity to humans and in this paper helping us to classify a plate. For this study, we used YOLOv4 for license plate detection in an image.

C. Object Detection with YOLOv4

YOLOv4 is a fast-operating neural network for object detection. This will improve production system performance and optimization in parallel. Yolov4 operates in real-time on a conventional GPU with 8–16 GB of VRAM, and training on Yolov4 requires only one conventional GPU. Yolov4 works efficiently and excels in terms of speed (FPS) and accuracy [3].

D. Object Detection with YOLOv5

YOLOv5 is a family of compound-scaled object detection models trained on the COCO dataset. It includes simple functionality for Test Time Augmentation (TTA), model ensembling, hyperparameter evolution, and export to ONNX, CoreML, and TFLite. YOLOv5 is available in four models, namely s, m, l, and x, each one of them offering different detection accuracy and performance. The implementation of YOLOv5 can be seen in https://github.com/ultralytics/yolov5. In this work, YOLOv5 is used to classify every digit character in a vehicle plate number.

III. DATASET

We used data of vehicle images with license plates from Open Images V6. Open Images is a platform that provides various image datasets. In this work, we used an image dataset of Vehicle Plate Number class. The total data we used to train the license plate detection model with YOLOv4 was 3368. A total of 3000 images were used as train data and 368 images as test/validation data.

Figure 1. Examples of Dataset

We arbitrarily collected 441 images of car and motorcycle license plates in a parking area in Jakarta, Indonesia with varying but incomplete lighting conditions. It should be noted that the data do not include all types of additional license plate accessories, for example acrylic cases, and all possible camera orientations, e.g., slope and angle. Standard license plates in Indonesia begin with one or two alphabetic characters indicating the region where the vehicle is registered, followed by a one-to-four-digit number and then one to three alphabetic characters. Data as many as 441 number plate images that have been collected are then labeled for each character object contained in each image. Image labeling is done by using the LabelImg library.

IV. DEEP LEARNING ARCHITECTURE

YOLOv4 is a one-stage object detection model. The general architecture of one-stage object detector can be seen in the figure 2. A one-stage model is capable of detecting objects without the need for a preliminary step such as region of importance detection. The advantage of a one-stage detection model is the fast prediction speed that allows real time use. The architecture of YOLOv4 is consists of three parts.

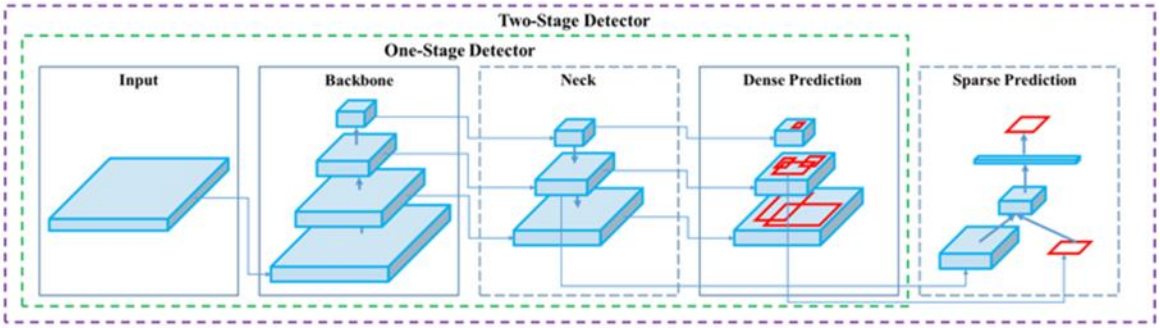

Figure 2. Object Detector in YOLOv4

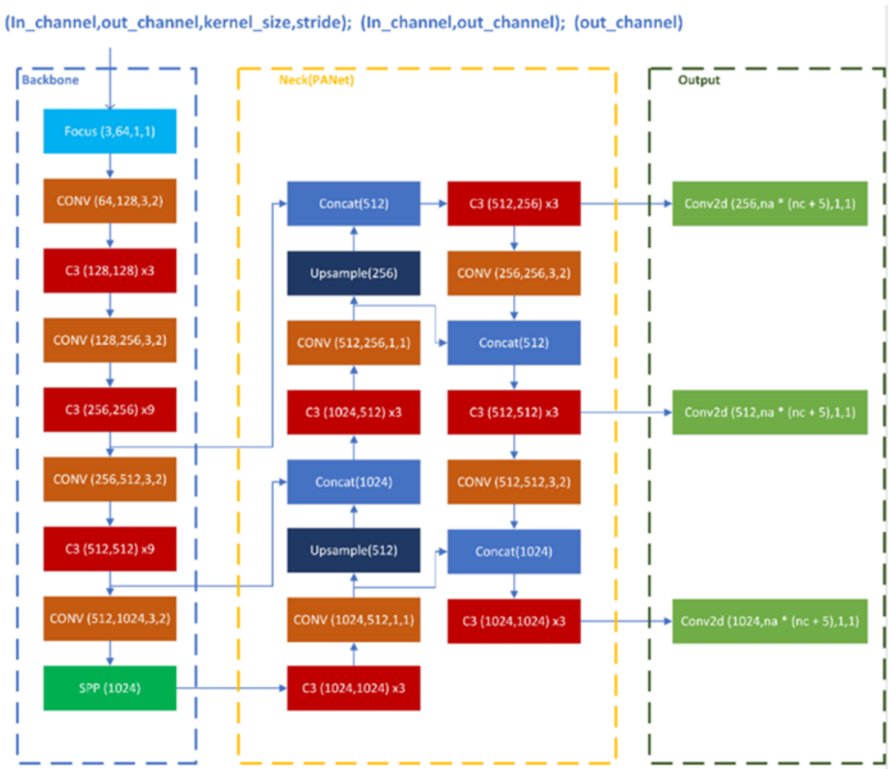

Figure 3. Model Structure Overview of YOLOv5

1. Backbone

The main objective of the backbone is to extract the essential features of the input images. YOLOv4 used CSPDarknet53 as the backbone. The Cross Stage Partial (CSP) [5] architecture is derived from the DenseNet [8] architecture which uses the previous input and concatenates it with the current input before moving into the dense layer.

2. Neck

The key role of the neck is to collect feature maps from different stages. YOLOv4 used SPP (Spatial Pyramid Pooling) block to generate fixed size features whatever the size of out feature maps [6] and PaNet to aggregate parameter different backbone levels [7].

3. Head

In the case of a one-stage detector, the key role of the head is to perform dense prediction. The dense prediction is the final prediction which is composed of a vector containing the coordinates of the predicted bounding box, the confidence score, and the label of the prediction.

As shown in the figure 3, YOLOv5 has a similar model architecture to YOLOv4. YOLOv5 almost resembles the YOLOv4 except the framework and configuration.

YOLOv5 is based on the PyTorch framework and uses a .yaml file for configuration.

V. TRAINING

A. Training and Evaluation of YOLOv4

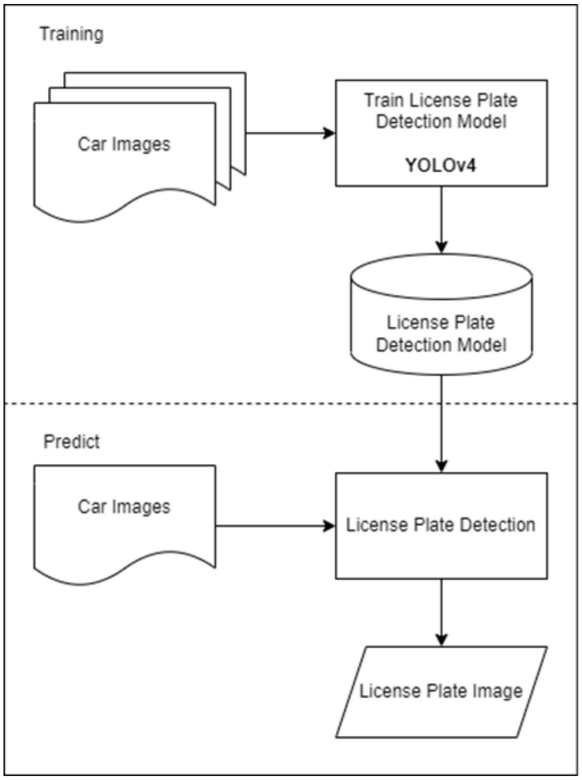

Figure 4. Flow Training of YOLOv4

For the License Plate Detection model, we trained the YOLOv4 model by using the transfer learning method. We trained the model to detect license plate objects in the image. We trained the model on 3000 vehicles with registration plate images and then validated the trained model in 368 images to calculate the mAP value.

B. Training and Evaluation of YOLOv5

For the Digits Detection model, we trained the YOLOv5 model by using the transfer learning method. We trained the model to classify each character object in the license plate image. We trained the model on 330 license plate images and then validated the trained model in 111 images to calculate the mAP value.

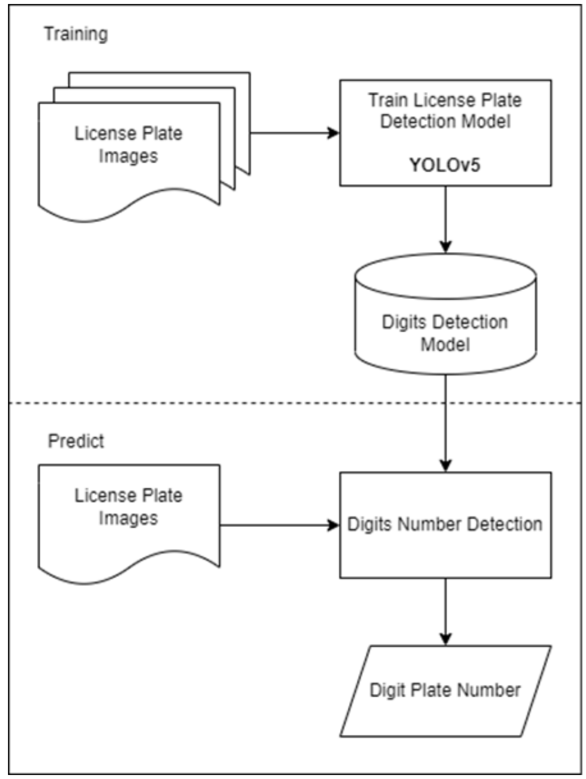

Figure 5. Flow Training of YOLOv5

C. Inference

As shown in the figure 6, we use the License Plate Detection model and the Digit Recognition model that previously trained in our License Plate Recognition System as follows:

1. First, the system will receive a vehicle image as an input.
2. Then, the image will be processed by the License Plate Detection model. The model detected and cropped the plate number object.
3. The cropped image will be processed by the Digit Recognition model. The model classified all the characters in the cropped image and returned a sequence of digit plate numbers.

Figure 6. Inference after Training

VI. RESULT AND ANALYSIS

A. License Plate Detection with YOLOv4

It can be seen from Table 1 that using YOLOv4, our model trained to detect vehicle plates has the ability to make True Positive (TP) comparisons with the number of positive predicted data of 92% which is denoted as precision and has the ability to make comparisons True Positive (TP) with the number of data that is actually positive by 86% which is denoted as recall. Because of the dilemma between the use of precision and recall, these two parameters are the 2 basic components in producing the F_1-score which can be written mathematically as follows

$$F_1 = 2 \cdot \frac{precision \ \cdot recall}{\text{precision} + recall}. \qquad (1)$$

Table 1. Experiment Result of License Plate Detection

Detector	TP	FP	Precision	Recall	F1-score	mAP@0.50	Total Time Detection
YOLOv4	348	31	0.92	0.86	0.89	88.95%	6 seconds

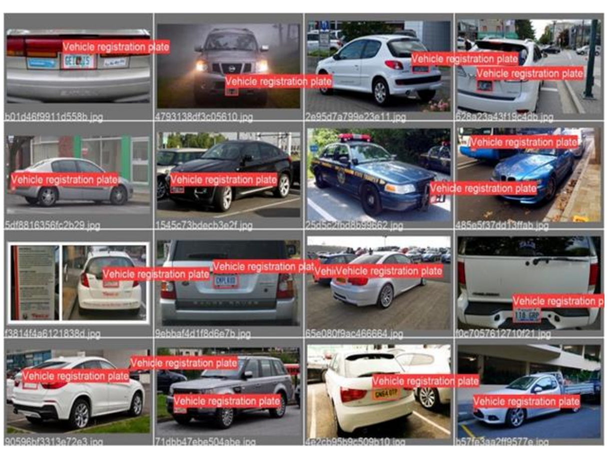

Figure 7. Samples of License Plate Detection Result

B. Digit Recognition with YOLOv5

As shown in the Table 2, the digit recognition model as a result of the YOLOv5 transfer learning model using the existing dataset can produce a F1-Score value of 84% with a precision value of 84,11% and a recall of 83,97%. In addition, this model can achieve a mAP value of 87,4% for an IoU threshold value of 50%. The total detection time required for this model is 13.5 ms. The digit recognition model with YOLOv5 shows satisfactory results even with a small number of datasets.

Table 2. Experiment Result of Digit Recognition

Detector	F1-Score	Precision	Recall	mAP@50
YOLOv5	0.8726	0.9081	0.8397	0.8740

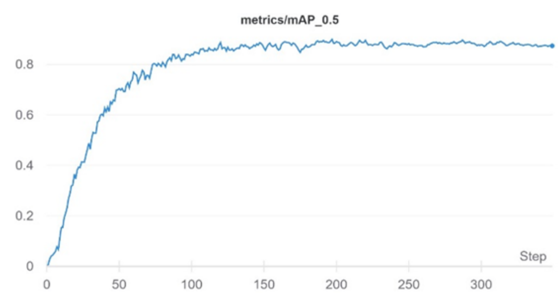

Figure 8. mAP Graph of Digit Recognition

Figure 9. shows several examples of the digit recognition results with YOLOv5. Every character on the license plate is well recognized. However, there are some misclassified characters such as the letter B which is not perfect and not in straight orientation, it is often classified as 3, E, G, or P. In addition, the letter D is also often misclassified as the letter Q. The letter X is often recognized as K.

Figure 9. Samples of Digit Recognition Result

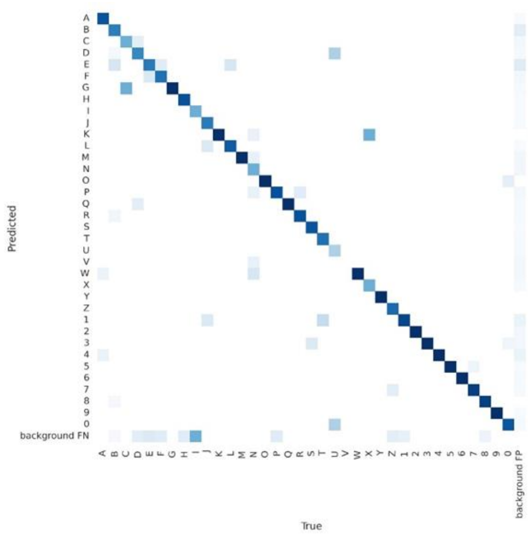

Figure 10. Correlation between Actual and Predicted Digits

CONCLUSION

In this work, we create a system to help the automatization of an Indonesian parking system. Our system focused on how to automatically recognize a vehicle plate number. Our license plate recognition system consists of two processes. Each process is equipped with a machine learning model.

The first process is Object License Plate Detection. We used transfer learning on YOLOv4 to build the Object License Plate Detector. The process takes a vehicle image as an input and returns the detected and cropped license plate object image as a result. Our model achieved a mAP score of 88.95% with detection time 6ms.

The second process is Digit Recognition. We used transfer learning on YOLOv5 to build the Digit Recognition model. The process takes the cropped image as an input and returns a sequence of digit plate numbers as a result. Our model achieved a mAP score of 87.40% with detection time 13.5ms.

There are several misclassified characters in the Digit Recognition model as explained in the experiment and result points. So, for the future works, it would be better to use much more data with a balanced distribution of the number of characters. On the other hand, it would also be better to have a dataset with variations in image orientation and lighting to simulate the real-world license plate recognition.

ACKNOWLEDGMENT

This work was carried under the financial assistance of Bangkit 2021 program funded by Google and Kedaireka (Indonesian Directorate General of Higher Education, Ministry of Education, Culture, Research, and Technology). Also, this work was carried out under the supervision and incubation program of Gunadarma University.

REFERENCES

[1] Hendryli, J., & Herwindiati, D. E. (2020, August). Automatic License Plate Recognition for Parking System using Convolutional Neural Networks. In 2020 International Conference on Information Management and Technology (ICIMTech) (pp. 71-74). IEEE.

[2] Sung, J. Y., & Yu, S. B. (2020, November). Real-time Automatic License Plate Recognition System using YOLOv4. In 2020 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia) (pp. 1-3). IEEE.

[3] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

[4] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once:

Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

[5] Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 390-391).

[6] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE transactions on pattern analysis and machine intelligence, 37(9), 1904-1916.

[7] Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path aggregation network for instance segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 8759-8768).

[8] Huang, G., Liu, Z., Pleiss, G., Van Der Maaten, L., & Weinberger, K. (2019). Convolutional networks with dense connectivity. IEEE transactions on pattern analysis and machine intelligence.