ISSN 2616-5767 Vol.7, Issue 3: 12-16, DOI: 10.25236/AJETS.2024.070303

LPD-YOLO: A Lightweight License Plate Detection Method with Strong Generalization Ability

Junhua Xu[1], Jiawei Luo[1], Jintu Wei[1], Yiwen Luo[1], Minhua Ye[2], Junpeng Tang[1], Ruihan Chen[1], Xuewen Chen[1], Di Ning[3], Zhi Li[1,*]

[1]School of Mathematics and Computer, Guangdong Ocean University, Zhanjiang, 524088, China
[2]College of Ocean Engineering and Energy, Guangdong Ocean University, Zhanjiang, 524088, China
[3] School of Economics, Guangdong Ocean University, Zhanjiang, 524088, China
[*]Corresponding author

Abstract: *To detect common types of license plate, this paper proposes the LPD-YOLO algorithm based on YOLOv7 for license plate classification detection. The LPD -YOLO algorithm adopts the Content-Aware Reassembly of Features (CARAFE) upsampling operator to replace the nearest-neighbor interpolation method in YOLOv7, thereby improving the accuracy of object detection. Additionally, it introduces the Distribution Shifting Convolution (DSConv) module to replace some traditional convolutions in the YOLOv7 head network, achieving model lightweighting. Experimental results show that the LPD-YOLO algorithm achieves an mAP value of 91.72%, with a model computation of only 96G. This method features high accuracy and robustness, making it highly valuable in practical scenarios for license plate classification detection.*

Keywords: *Deep learning, License plate detection, YOLOv7, CARAFE, DSConv*

1. Introduction

With the continuous advancement of artificial intelligence technology, traditional domains are gradually transitioning towards intelligent directions[1]. Within the next decade, the total number of vehicles in China is projected to increase to around 400 million, posing significant burdens on urban infrastructure[2]. The localization and recognition technology of license plates is one of the most crucial technologies in the realm of smart transportation within smart cities, and the exploration of this technological field represents an eternal research direction. In practical engineering applications, the accuracy and speed of license plate recognition have always been conflicting factors; enhancing one aspect often leads to a certain degree of compromise in the other.

Currently, integrated license plate recognition barrier systems in the market employ highly complex algorithms to ensure high recognition rates[3]. Although these algorithms meet the requirements for recognition rates, they often suffer from high latency. Some high-end machines utilize high-performance processors to reduce latency, but this increases the overall cost of the equipment, and even with such enhancements, recognition times are difficult to reduce below 100 ms.

The emergence of the YOLO algorithm has changed this dilemma[4]. This algorithm integrates the advantages of current pattern recognition algorithms and has been widely applied since its inception. Considering these factors, this study proposes a license plate localization and recognition method based on YOLOv7, applying the YOLOv7 model in the process of license plate localization and recognition. This approach aims to enhance the efficiency of license plate recognition while ensuring high recognition rates.

2. Improved Detection Model Based on YOLOv5s

2.1. YOLOv7 Object Detection Algorithm

The YOLOv7 model primarily consists of three parts: Input, Backbone, and Head. Its Backbone is composed of modules such as CBS, ELAN, and MP-1, while the Head comprises modules including CBS, Conv, Upsample, SPPCSPC, MP-2, ELAN', and RepConv. The structure of the YOLOv7 model is illustrated in Figure 1. After a series of data preprocessing steps, the input images are resized to a fixed

ISSN 2616-5767 Vol.7, Issue 3: 12-16, DOI: 10.25236/AJETS.2024.070303

size through the Input module, then fed into the Backbone for feature extraction. The extracted features are subsequently passed to the Head for feature fusion, producing features of three different sizes: large, medium, and small. Finally, the fused features are sent to the detection head to output the detection results.

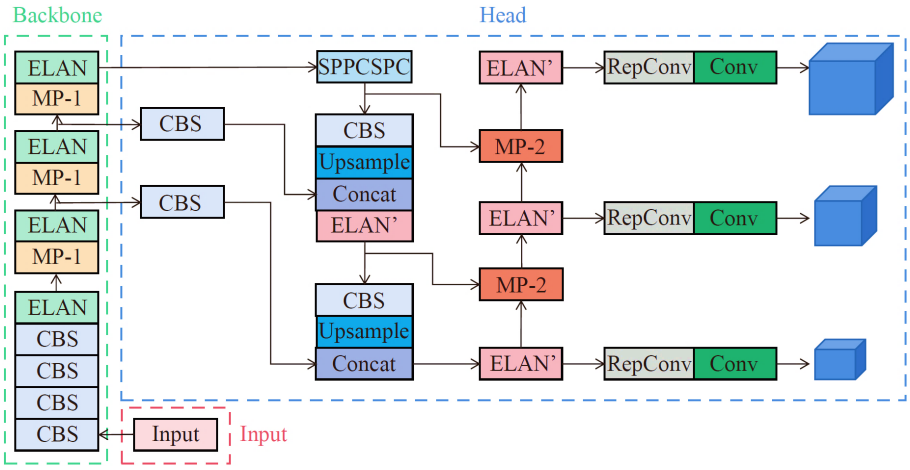

Figure 1: Structure diagram of YOLOv7

In the YOLOv7 model, the CBS module consists of convolution, batch normalization, and SiLU activation functions. The ELAN and ELAN' modules control the gradient paths to enable the model to learn more features. The SPPCSPC module increases the receptive field to adapt the algorithm to images of different resolutions. The MP-1 and MP-2 modules primarily perform downsampling. The Upsample module is responsible for upsampling, using the nearest-neighbor interpolation method. The RepConv module adjusts the channel numbers for features of different scales. The basic structure of some fundamental components of YOLOv7 is depicted in Figure 2.

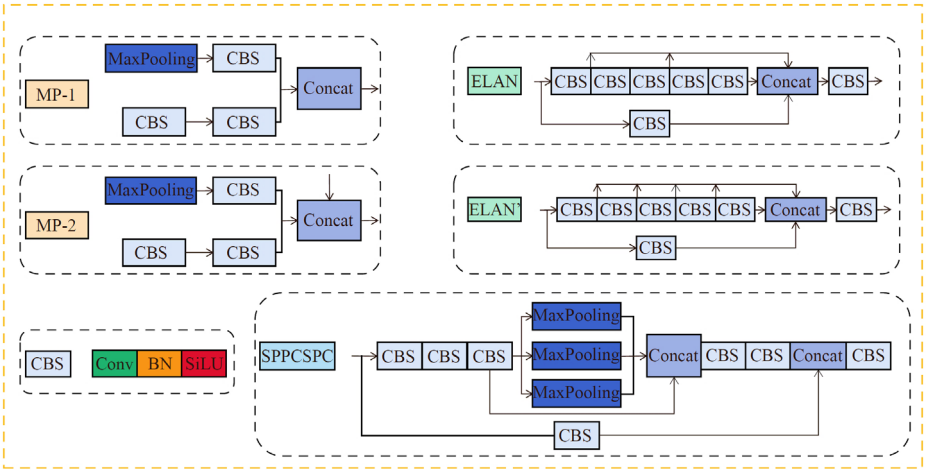

Figure 2: Basic component structure diagram of YOLOv7

2.2. A Improved YOLOv7

The upsampling module in YOLOv7 utilizes the nearest-neighbor interpolation algorithm, which assigns the grayscale value of the nearest pixel to the pixel being calculated, without requiring extensive computations. Although nearest-neighbor interpolation has advantages such as low computational cost and simplicity, it leads to significant image distortion due to the lack of consideration for the influence of surrounding pixels, resulting in a loss of image quality.

To address the influence of surrounding pixels and mitigate the loss of image quality, the CARAFE module is employed in place of the nearest-neighbor interpolation algorithm in the upsampling module of YOLOv7. The CARAFE module aggregates contextual information within a larger receptive field. Furthermore, to enhance model accuracy while maintaining lightweight characteristics, the DSConv module is introduced to optimize the Head part of YOLOv7, enabling reduced memory usage and improved computational speed through integer operations.

ISSN 2616-5767 Vol.7, Issue 3: 12-16, DOI: 10.25236/AJETS.2024.070303

The model architecture of LPD-YOLOv7 proposed in this study is illustrated in Figure 3, with the structure of the improved components depicted in Figure 4.

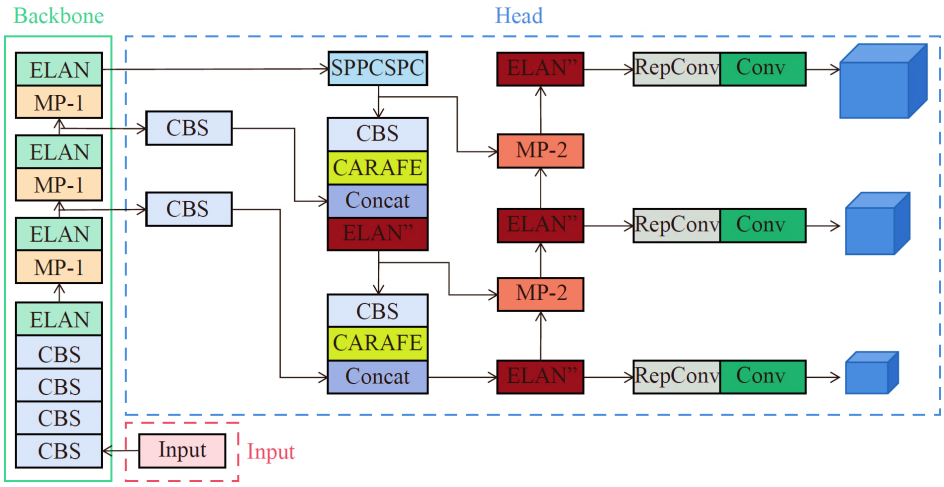

Figure 3: Structure diagram of LPD-YOLOv7

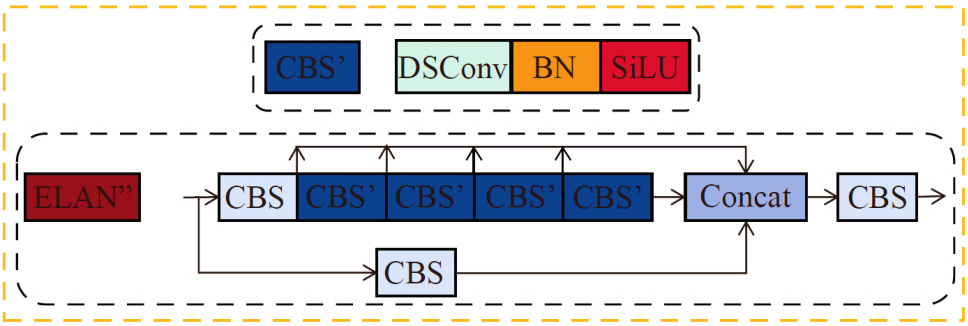

Figure 4: Improved component structure diagram

2.2.1. CARAFE is an upsampling operator

CARAFE is a lightweight upsampling operator that can have a larger receptive field during reassembly, allowing better utilization of surrounding information. It guides the reassembly process based on input features and does not introduce excessive parameters and computational complexity. Models using the CARAFE upsampling operator have shown significant improvements in detection performance in different tasks, while introducing only a small number of additional parameters and computational complexity.

2.2.2. DSConv stands for Distributed Shift Convolution

DSConv, known as depthwise separable convolution, aims to simulate the behavior of convolutional layers by using quantization and depthwise separability. Compared to traditional convolution, DSConv consumes less memory and computes faster. DSConv decomposes the traditional convolutional kernel into a variable quantization kernel (VQK) and depthwise offsets.

3. Experimental Results and Analysis

3.1. Experimental Environment

This study employed a computer system running Windows 10, equipped with 16 GB of memory operating at 3200 MHz, an Intel®Core™i7-11800H CPU@2.30GHz processor, and an NVIDIA GeForce RTX 3050 4GB graphics card. The deep learning framework utilized was PyTorch 1.12.1, with CUDA 10.2 and CUDNN 8.4.1, operating within a Python 3.9 environment. Parameter settings included a learning rate of 0.01, momentum parameter of 0.937, final learning rate of 0.1, batch size of 16, and 400 epochs. Additionally, a warm-up learning strategy with 3 epochs, a momentum parameter of 0.8, and a learning rate of 0.1 was employed. Confidence and IOU thresholds were set to 0.25 and 0.45, respectively, and image resolutions were uniformly adjusted to 640×640.

ISSN 2616-5767 Vol.7, Issue 3: 12-16, DOI: 10.25236/AJETS.2024.070303

3.2. Experimental Dataset

In this study, we utilized a sampling approach from the CCPD dataset, extracting 5000 images, with 4000 designated for training purposes and the remaining 1000 for testing. The CCPD dataset comprises images collected from the streets of Hefei, each image containing a single license plate. These images were captured under diverse weather conditions, locations, and angles, totaling 250,000 images, all standardized to dimensions of 720x1160x3. Furthermore, the dataset was acquired using mobile devices, introducing variations in license plate orientations.

3.3. Model Performance Evaluation Metrics

Single-label classification evaluation metrics include Accuracy (Acc), Precision (Prec), Recall (Rec), F1-score, Specificity (Spe), and confusion matrix. These evaluation metrics are calculated based on four fundamental indicators:

TP (True Positive): The number of images correctly classified as positive samples.

FN (False Negative): The number of images incorrectly classified as negative samples but are positive.

FP (False Positive): The number of images incorrectly classified as positive samples but are negative.

TN (True Negative): The number of images correctly classified as negative samples.

To objectively evaluate the model's performance in detecting tomato leaf diseases, this study employs Mean Average Precision (mAP), Recall (R), parameter count, detection rate (Frames Per Second, FPS), and model size as evaluation metrics. The formulas for mAP, R, and FPS are as follows (Equations (1) to (3)).

$$mAP = \frac{\sum AP}{N(Class)} \tag{1}$$

In the equations, "*AP*" represents Average Precision, and "*N (Class)*" denotes the number of images in the test dataset.

$$R = \frac{T_P}{T_P + F_N} \times 100\% \tag{2}$$

In the equations, "T_P" stands for True Positive, representing the number of samples correctly classified as positive by the model, and "F_N" denotes False Negative, indicating the number of samples incorrectly classified as negative by the model but are positive.

$$FPS = \frac{1000}{T_{pre} + T_i + T_N} \tag{3}$$

In the equations, "T_{pre}" refers to image preprocessing time, "T_i" represents inference time, which is the time taken for an image to go from input to output through the model after preprocessing, and "T_N" denotes post-processing time, which includes operations such as non-maximum suppression per image on the model output.

This study comprehensively evaluates the model from several aspects including mAP, R, FPS, and confusion matrix analysis.

3.4. Model Performance Experiments

To verify the performance of the improved model LPD-YOLOv7, comparative experiments were conducted on a license plate dataset, comparing with target detection network models such as Faster RCNN, SSD, RFBNet, RetinaNet, and YOLOv7. The experimental results are shown in Table 1.

From Table 1, it can be observed that LPD-YOLOv7 achieves the best detection performance with an mAP of 91.72%, which is 1.20%, 4.58%, 1.91%, 2.41%, and 1.7% higher than Faster R-CNN, SSD, RFBNet, RetinaNet, and YOLOv7, respectively.

In terms of detection speed, LPD-YOLOv7 achieves a detection speed of 50 frames per second, attributed to the utilization of the DSConv module to simplify traditional convolutional modules, making the network model more lightweight. LPD-YOLOv7 demonstrates faster detection speeds compared to SSD, RFBNet, and RetinaNet models by 117.4%, 100%, and 85.2%, respectively, indicating its advantage in real-time detection speed. Although LPD-YOLOv7's detection speed is 16.7% slower than the YOLOv7 network model, it still meets the requirements for real-time license plate detection tasks.

ISSN 2616-5767 Vol.7, Issue 3: 12-16, DOI: 10.25236/AJETS.2024.070303

Furthermore, in terms of lightweightness, LPD-YOLOv7 has only 35.6 million parameters, and the model's computational complexity is 96G. Compared to SSD, RFBNet, and YOLOv7, although the number of parameters is higher by 11.6M, 0.1M, and 0.1M, respectively, the model's complexity is lower by 178.5G, 221.3G, and 9.2G, respectively. Compared to Faster RCNN and RetinaNet, LPD-YOLOv7 has lower parameter counts and computational complexities by 101.2M, 0.8M, 305.8G, and 50.3G, respectively. In summary, LPD-YOLOv7 is lightweight enough to be deployed on most hardware devices.

Table 1: Performance comparison of different models

Models	mAP	Params/M	FLOPs/G	Speed/FPS
Faster R-CNN	90.52	136.8	401.8	10
SSD	87.14	24.0	274.5	23
RFBNet	88.81	35.5	317.3	25
RetinaNet	88.31	36.4	146.3	27
Yolov7	90.02	35.5	105.2	60
LPD-YOLOv7	91.72	35.6	96.0	50

4. Conclusions

This paper proposes a license plate detection model, LPD-YOLOv7, based on an improved YOLOv7. Compared to other object detection models, the improved model better balances the relationship between detection accuracy and model performance, maintaining model lightweightness while achieving high detection accuracy and speed. The CARAFE upsampling operator is introduced to replace the nearest neighbor interpolation operator in YOLOv7 for upsampling, which, although introduces a small amount of additional parameters and computational complexity, enhances the detection accuracy of the model. In the Head network part of YOLOv7, the DSConv module is introduced to replace some traditional convolutions. Due to the use of integer operations in the DSConv module, it consumes less memory and operates faster compared to floating-point operations in traditional convolutions. Tested on a license plate dataset, the mAP of the proposed license plate detection model, LPD-YOLOv7, reaches 91.72%, with a detection speed of up to 50 frames per second and computational complexity reduced to 96G, outperforming common object detection networks in comprehensive performance. The model can quickly and accurately detect common types of license plate, overcoming issues such as irregular shapes and large color differences, which lead to low accuracy and slow detection speed. It can provide technical support for improving license plate recycling and resource utilization.

Acknowledgments

This research was supported by a special grant from the Guangdong Provincial Science and Technology Innovation Strategy under Grant No. pdjh2023b0247, and Guangdong Ocean University Undergraduate Innovation Team Project under Grant No. CXTD2023014.

References

[1] Abeid E E S, Mosa A M, Tabakh E M A M, et al. Antifungal activity of copper oxide nanoparticles derived from Zizyphus spina leaf extract against Fusarium root rot disease in tomato plants.[J]. Journal of Nanobiotechnology, 2024, 22(1), 28.

[2] Ruihan C, Minhua Y, Zhi L, et al. Empirical assessment of carbon emissions in Guangdong Province within the framework of carbon peaking and carbon neutrality: a lasso-TPE-BP neural network approach. [J]. Environmental science and pollution research international, 2023, 30(58):121647-121665.

[3] Wen Y, Chen Y, Wang K, et al. A Review of Pest Detection Based on Machine Vision [J]. Journal of China Oils and Grains Society, 2022, 37(10): 271-279.

[4] Cai P, Yang D, Zou Y, et al. Reconsidering Multi-Branch Aggregation for Semantic Segmentation[J]. Electronics, 2023, 12(15).