Improving Small License Plate Detection with Bidirectional Vehicle-Plate Relation

Songkang Dai[1], Song-Lu Chen[1], Qi Liu[1], Chao Zhu[1](✉), Yan Liu[2], Feng Chen[3], and Xu-Cheng Yin[1]

[1] School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, China
{songkangdai,qiliu7}@xs.ustb.edu.cn, {songluchen,chaozhu,xuchengyin}@ustb.edu.cn

[2] Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China
liuyan@ustb.edu.cn

[3] EEasy Technology Company Ltd., Zhuhai, China
cfeng@eeasytech.com

Abstract. License plate detection is a critical component of license plate recognition systems. A challenge in this domain is detecting small license plates captured at a considerable distance. Previous researchers have proved that pre-detecting the vehicle can enhance small license plate detection. However, this approach only utilizes the one-way relation that the presence of a vehicle can enhance license plate detection, potentially resulting in error accumulation if the vehicle fails to be detected. To address this issue, we propose a unified network that can simultaneously detect the vehicle and the license plate while establishing bidirectional relationships between them. The proposed network can utilize the vehicle to enhance small license plate detection and reduce error accumulation when the vehicle fails to be detected. Extensive experiments on the SSIG-SegPlate, AOLP, and CRPD datasets prove our method achieves state-of-the-art detection performance, achieving an average detection $AP_{0.5}$ of 99.5% on these three datasets, especially for small license plates. When incorporating a license plate recognizer that relies on character detection, we can achieve an average recognition accuracy of 95.9%, surpassing all comparative methods. Moreover, we have manually annotated the vehicles within the CRPD dataset and have made these annotations publicly available at https://github.com/kiki00007/CRPDV.

Keywords: License plate detection · License plate recognition · Small license plate · Bidirectional vehicle-plate relation

1 Introduction

Automatic license plate recognition (ALPR) has recently gained significant popularity in various applications, such as traffic enforcement, theft detection, and

S. Dai and S.-L. Chen—Equal contribution.

S. Rudinac et al. (Eds.): MMM 2024, LNCS 14555, pp. 253–266, 2024.
https://doi.org/10.1007/978-3-031-53308-2_19

automatic toll collection. The ALPR system typically consists of three stages: license plate detection, character detection, and character recognition [1]. Among these stages, license plate detection plays a pivotal role in determining the overall accuracy of the ALPR system. Specifically, detecting small license plates presents a significant challenge due to their size.

As shown in Fig. 1(a), many ALPR methods have been proposed to directly detect the license plate from the input image [6,28]. However, detecting the license plate directly can lead to missed detections, primarily due to its small size. To address this issue, Kim et al. [12,14] propose a two-step approach as depicted in Fig. 1(b), where the vehicle is first pre-detected, followed by license plate detection within the vehicle region. These methods reduce the search region and mitigate background noises, enhancing license plate detection. Nevertheless, these methods may encounter error accumulation if the vehicle fails to be detected, resulting in subsequent failures in license plate detection. To minimize error accumulation, Chen et al. [5] propose a fusion approach illustrated in Fig. 1(c), which combines direct license plate detection (Fig. 1(a)) and vehicle pre-detection (Fig. 1(b)), merging both detection branches to obtain the final results. However, this approach is time-consuming due to the involvement in multiple detection branches and the subsequent merge operation.

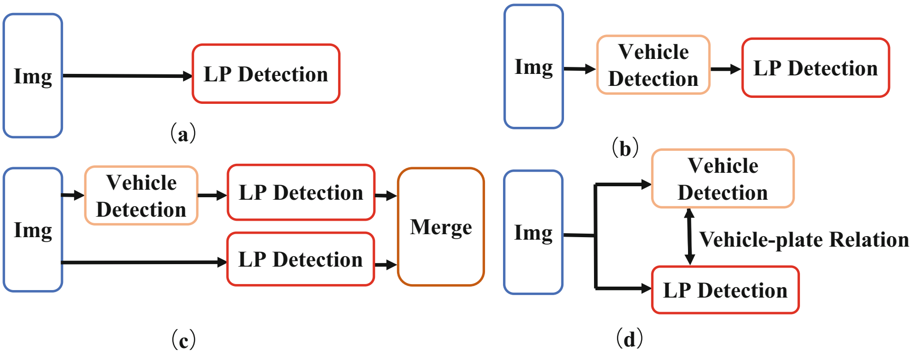

Fig. 1. (a) Direct license plate detection from the input image. (b) License plate detection based on vehicle pre-detection. (c) License plate detection by combining direct detection and vehicle pre-detection. (d) Our proposed method, using bidirectional vehicle-plate relationships to enhance license plate detection.

To address the challenges mentioned earlier, as depicted in Fig. 1(d), we propose simultaneous detection of both the vehicle and the license plate, leveraging their bidirectional relationship to enhance small license plate detection. This approach facilitates mutual reinforcement between vehicles and license plates due to their interdependency. In comparison to direct detection (Fig. 1(a)), our method utilizes the presence of the vehicle to improve license plate detection. Unlike the vehicle pre-detection approach (Fig. 1(b)), our method mitigates error accumulation arising from the one-way relationship between the vehicle and the license plate. Additionally, compared to the fusion approach (Fig. 1(c)), our method enhances inference speed through simultaneous detection and bidirectional relation mining. Extensive experiments on the SSIG-SegPlate [9], AOLP [11], and CRPD [32] datasets validate the effectiveness of our method, achieving an average detection $AP_{0.5}$ of 99.5%, particularly for small license plates. When combined with a YOLO-based character recognizer [15], our method outperforms other state-of-the-art techniques, achieving an average recognition accuracy of 95.9%. Notably, annotations for both vehicles and license plates are available for the SSIG-SegPlate and AOLP datasets within the community. However, for the CRPD dataset, only license plate annotations are provided. To support the community, we manually annotated vehicles in the CRPD dataset and made the annotations publicly available at https://github.com/kiki00007/CRPDV.

2 Related Work

2.1 Object Detection

Object detection is a task that involves locating the bounding box and predicting the category of an object. Previous object detectors can be broadly categorized into two types based on the detection stage: two-stage detectors [17,22] and one-stage detectors [8,18]. Additionally, they can be classified as anchor-based [18,22] or anchor-free [26,30] based on the matching mechanism. However, these methods typically involve complex post-processing and matching procedures. To reduce complexity, DETR-based methods [4,31] utilize the transformer [27] architecture and object queries to directly predict the class and bounding box of an object. However, the aforementioned methods ignore the relationships between different objects, which is suboptimal to small object detection. In this work, besides object detection, we propose to utilize the bidirectional relationships between vehicles and license plates to enhance small license plate detection.

2.2 License Plate Detection

There are two prevailing approaches for license plate detection: direct detection [6,28] and vehicle pre-detection [12,14]. The former involves directly detecting the license plate in the image. However, these methods may not work well for small license plates due to their small size. The latter approach, known as vehicle pre-detection, first detects the vehicle in the image and then locates the license plate within the vehicle region. This approach reduces the search region and mitigates background noises, enhancing small license plate detection. However, these vehicle pre-detection methods are prone to error accumulation because the absence of vehicle detection inevitably leads to the failure of license plate detection. To mitigate error accumulation, Chen et al. [5] introduce a method incorporating two detection branches. One branch focuses on pre-detecting the vehicle, and the other directly detects the license plate. The outputs from these

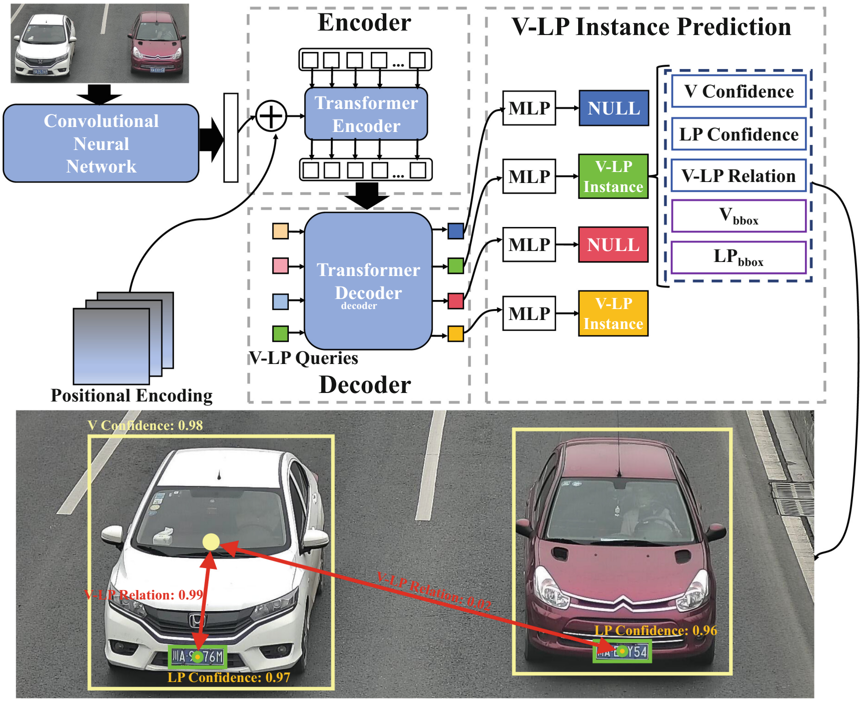

Fig. 2. Overall architecture. The network utilizes an encoder-decoder architecture, taking an image as input and generating predictions for the category, bounding box of vehicles (V) and license plates (LP), and the relationships between them.

branches are then fused to obtain the final results. However, this approach introduces significant computational overhead. In this work, we propose to simultaneously detect vehicles and license plates and leverage bidirectional relationships between them to enhance the effectiveness and efficiency of small license plate detection.

3 Method

As depicted in Fig. 2, our proposed network can simultaneously detect vehicles and license plates and generate their bidirectional relationships. When a license plate subordinates to a vehicle, their relation confidence is higher, and vice versa. This way, it can mutually enhance the detection of vehicles and license plates.

3.1 Network Architecture

The proposed network can be mainly divided into three parts: (I) A CNN backbone to extract visual features from the input image; (II) A transformer encoder-decoder to process visual features and generate global features; (III) A multi-layer perceptron layer (MLP) to generate predictions based on global features.

Backbone: We utilize ResNet-50 [10] to extract visual features from the input image into feature maps. The size of the input image and features maps is $[H_0, W_0, 3]$ and $[H, W, C]$, respectively, s.t., $H = H_0/32$ and $W = W_0/32$. Subsequently, a 1×1 convolutional layer is utilized to reduce the channel dimension from $C = 2048$ to $d = 256$. Since the subsequent encoder requires a sequence as input, we convert the reduced features into a sequence of length $H\times W$, where each step corresponds to a vector of size d. As a result, we obtain a flattened feature map with the dimension of $[H\times W, d]$.

Encoder: The encoder follows the vanilla transformer [27], incorporating six identical units. Each unit comprises an eight-head self-attention network and a two-layer feed-forward network (FFN) with the dimension of $d_{ff} = 2048$. The output dimension is set to $d_{model} = 512$. The Query, Key, and Value are all obtained by the sum of positional encodings and visual features from the CNN backbone to generate global features.

Decoder: The decoder also follows the vanilla transformer, incorporating six identical units. Each unit comprises an eight-head cross-attention network, an eight-head self-attention network, and a two-layer feed-forward network. Similar to the encoder, the FFN dimension is $d_{ff} = 2048$, and the output dimension is $d_{model} = 512$. The decoder takes three inputs, i.e., positional encodings, V-LP queries, and global features from the encoder, to generate $N = 100$ embeddings for predictions. In the cross-attention network, the Value is obtained directly from global features. The Key is the sum of global features and positional encodings, and the Query is the sum of positional encodings and V-LP queries.

Vehicle-Plate Instance Prediction: The output embeddings generated by the decoder are converted into vehicle-plate instances using MLPs. We define the vehicle-plate instance as a five-tuple consisting of vehicle confidence, vehicle-plate relation confidence, plate confidence, vehicle box, and plate box. Specifically, two three-layer MLPs are employed to predict the bounding box of the vehicle and the license plate. Additionally, three single-layer MLPs are utilized to estimate the confidence of the vehicle, the plate, and the vehicle-plate relation.

3.2 Training Objective

We treat the prediction of vehicle-plate instances as a problem of set prediction, involving a bipartite matching between the predicted instances and the ground truth. When presented with an input image, our model generates $N = 100$ predicted instances, where N represents the number of V-LP queries. The prediction set is represented as $P = p^i, i = 1, 2, ..., N$. The ground-truth set is represented as $G = g^i, i = 1, 2, ..., M, \phi, ..., \phi$, where ϕ denotes a null value for one-to-one matching between P and G, and M denotes the total number of ground-truth instances, s.t., $M \leq N$. The number of ϕ plus M equals N.

As demonstrated in Eq. (1), we use the Hungarian algorithm [13] to find the best bipartite matching $\widehat{\sigma}$ by minimizing the overall matching cost ζ_{cost}, which is composed of the matching cost of all N matching pairs.

$$\widehat{\sigma} = argmin\zeta_{cost}, \sigma \in \mathcal{O}_N \\ \zeta_{cost} = \sum_{i}^{N} \zeta_{match}(g^i, p^{\sigma(i)}) \tag{1}$$

where $\mathcal{O}_N$ represents the one-to-one matching solution space, and σ represents an injective function from the ground-truth set G to the prediction set P. $\zeta_{match}(g^i, p^{\sigma(i)})$ represents the matching cost between the i-th ground truth and $\sigma(i)$-th prediction, where $\sigma(i)$ represents the matching index of the prediction.

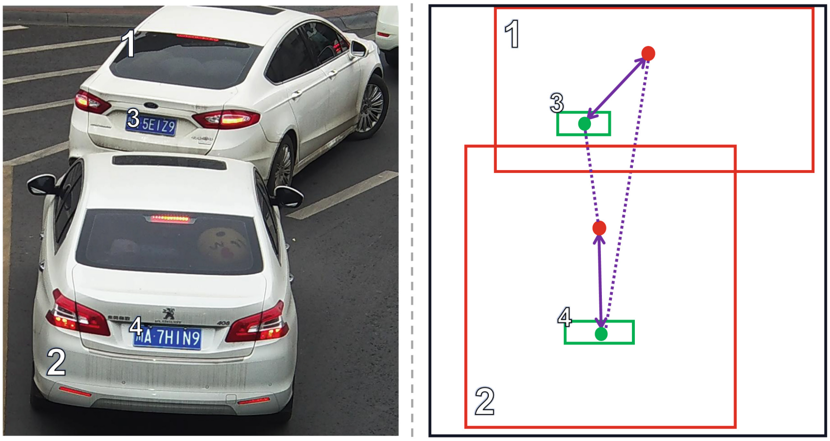

Fig. 3. Ground truth during training. The red and green boxes denote the ground-truth boxes of vehicles and license plates, respectively. The solid purple line represents the ground-truth V-LP relation, i.e., the positive relation sample used during training. The dotted purple line denotes no relation between the vehicle and the license plate, i.e., the negative relation sample, which is not used during training. (Color figure online)

As demonstrated in Eq. (2), the matching cost of each pair contains the classification loss ${\zeta^j}_{cls}$ and bounding box regression loss ${\zeta^k}_{box}$.

$$\zeta_{match}(g^i, p^{\sigma(i)}) = \beta_1 \sum_{j \in v,p,r} \alpha_j {\zeta^j}_{cls} + \beta_2 \sum_{k \in v,p} {\zeta^k}_{box} \tag{2}$$

where v, p, r represents the vehicle, license plate, and vehicle-plate relation, respectively. ${\zeta^j}_{cls}$ is calculated by the softmax cross-entropy loss. ${\zeta^k}_{box}$ is calculated by the weighted sum of L_1 loss and GIoU [23] loss. In this work, we emphasize classification by setting β_1 to 2 and β_2 to 1. Among classification, we emphasize vehicle-plate relation by setting α_r to 2, α_v to 1, and α_p to 1. The ground truth during training is illustrated in Fig. 3.

4 Experiments

4.1 Datasets

We utiliz three publicly available datasets: SSIG-SegPlate [9], AOLP [11], and CRPD [32]. SSIG-SegPlate and AOLP provide the annotations for the vehicle and the license plate, but CRPD only provides the annotations for the license plate. We manually annotated the vehicles in CRPD and made them available at https://github.com/kiki00007/CRPDV.

SSIG-SegPlate comprises 2,000 Brazilian license plates obtained from 101 vehicles. Following the official settings, we use 40% images for training, 20% for validation, and 40% for testing.

AOLP consists of three distinct subsets, each captured using different shooting methods. The AC subset focuses on static vehicles, while the LE subset captures vehicles violating traffic rules via roadside cameras. The RP subset captures images from various viewpoints and distances using cameras mounted on patrol vehicles. In total, the dataset includes 2,049 images containing Taiwanese license plates. When testing on one subset, the other two subsets are used for training and validation.

CRPD has 33,757 Chinese license plates captured by overpasses, which cover various vehicle models, such as cars, trucks, and buses. We follow the official split, i.e., 25,000 images for training, 6,250 for validation, and 2,300 for testing.

4.2 Training Settings

The backbone and transformer are initialized using the pre-trained DETR [4] model. During training, we utilize the Adam optimizer [21] to train the model for 50 epochs with the learning rate of 10^{-4} for the transformer and 10^{-5} for the backbone, weight decay to 10^{-4}, and batch size to 2. Moreover, data augmentation is adopted. First, we apply the image-level augmentation by adjusting the brightness and contrast with a probability of 0.5. Specifically, we randomly select a parameter from the range of [0.8, 1.2] for the brightness and contrast, slightly modifying the original image. Second, we perform scale augmentation by resizing the input image such that the shortest side ranges from 480 to 800 pixels, while the longest side is at most 1333 pixels. The input image is then scaled to the range of [0, 1] and normalized using channel mean and standard deviation. All the experiments are conducted on four NVIDIA 2080Ti GPUs.

4.3 Evaluation Protocols

We use Average Precision (AP) to evaluate license plate detection. Specifically, we utilize the computation method introduced in COCO [19] that calculates AP with different IoU (Intersection over Union) thresholds, i.e., ranging from 0.5 to 0.95 with an interval of 0.05. $AP_{0.5}$ refers to the average precision calculated at the IoU threshold of 0.5. We utilize Accuracy as the evaluation metric for license plate recognition, where all characters must be recognized accurately. We use Frame Per Second (FPS) to calculate the inference speed.

In addition, to verify the effectiveness of small license plate detection, we categorize license plates into three groups based on their height. License plates with a height of 25 pixels or less are categorized as small (S), those exceeding 25 pixels but not exceeding 50 pixels are categorized as medium (M), and license plates taller than 50 pixels are categorized as large (L).

4.4 Ablation Study

Table 1. Ablation study on SSIG-SegPlate. LP: license plate. V: vehicle.

Method	LP	V	Relation	Detection (V)		Detection (LP)		Recognition
				AP	$AP_{0.5}$	AP	$AP_{0.5}$	Accuracy
DETR	√			-	-	45.6%	96.3%	95.4%
	√	√		78.0%	99.2%	50.1%	97.5%	95.6%
Ours	√	√	√	**81.4%**	**100.0%**	**60.6%**	**100.0%**	**96.4%**

Table 2. Ablation study on AOLP. R: relation.

Method	LP	V	R	Detection (V)			Detection (LP)			Recognition		
				AP			AP			Accuracy		
				AC	LE	RP	AC	LE	RP	AC	LE	RP
DETR	√			-	-	-	53.2	52.2	43.6	96.1	94.3	95.3
	√	√		89.2	87.8	83.6	52.2	54.6	40.6	96.2	95.0	94.5
Ours	√	√	√	**93.9**	**90.8**	**91.5**	**65.2**	**60.8**	**58.2**	**98.1**	**98.0**	**97.6**

Table 3. Ablation study on CRPD.

Method	LP	V	Relation	Detection (V)		Detection (LP)		Recognition
				AP	$AP_{0.5}$	AP	$AP_{0.5}$	Accuracy
DETR	√			-	-	53.0%	96.3%	86.0%
	√	√		83.9%	98.5%	54.2%	96.2%	87.5%
Ours	√	√	√	**87.2%**	**98.6%**	**62.8%**	**98.6%**	**89.3%**

As presented in Table 1, Table 2, and Table 3, we investigate the impact of implicit and explicit relationships between vehicles and license plates on the SSIG-SegPlate, AOLP, and CRPD datasets, respectively. We conduct three ablation experiments: (I) direct license plate detection using the vanilla DETR model; (II) simultaneous vehicle and license plate detection using the vanilla DETR model, which implicitly captures the relation between vehicles and license plates;

(III) our proposed method, except for simultaneous vehicle and license plate detection, explicitly incorporating vehicle-plate relationships. After performing license plate detection, we employ the same YOLO-based character recognizer [15] for license plate recognition. Implicit vehicle-plate relationships have minimal impact on license plate detection and recognition. However, when incorporating explicit vehicle-plate relationships, our method substantially improves license plate detection and recognition. Additionally, our method enhances vehicle detection due to the bidirectional relationships between vehicles and license plates.

As shown in Fig. 4, we visualize the attention map of vehicle-plate relationships. The attention map highlights vehicles and their subordinated license plates, which means the relationships are constructed between them. Hence, the detection performance of vehicles and license plates are both enhanced.

Fig. 4. Visualization of vehicle-plate relationships.

4.5 Comparative Experiments

Table 4. Comparative experiments on SSIG-SegPlate.

Method	Detection			Recognition
	AP	$AP_{0.5}$	FPS	Accuracy
RARE [29]	-	-	-	93.7%
Rosetta [3]	-	-	-	94.3%
Direct Detection [4]	45.6%	96.3%	**13.0**	95.4%
Vehicle Pre-detection [6]	52.6%	97.5%	7.7	95.6%
STAR-Net [20]	-	-	-	96.1%
Two Branches [5]	53.8%	98.2%	5.4	96.2%
Ours	**60.6%**	**100.0%**	12.2	**96.4%**

As presented in Table 4, Table 5, and Table 6, we conduct comparative experiments on the SSIG-SegPlate, AOLP, and CRPD datasets, respectively. In all of these datasets, we compare three approaches: direct detection (Fig. 1(a)), vehicle

Table 5. Comparative experiments on AOLP.

Method	Detection						Recognition		
	AC		LE		RP		AC	LE	RP
	AP	$AP_{0.5}$	AP	$AP_{0.5}$	AP	$AP_{0.5}$	Accuracy		
RCLP [16]	-	98.5	-	97.8	-	95.3	94.8	94.2	88.4
DLS [24]	-	92.6	-	93.5	-	92.9	96.2	95.4	95.1
DELP [25]	-	99.3	-	**99.2**	-	99.0	97.8	97.4	96.3
Direct Detection [4]	53.2	98.2	52.2	96.1	43.6	97.8	96.1	94.3	95.3
Vehicle Pre-detection [6]	47.8	98.1	53.8	96.3	44.4	96.9	96.2	95.0	94.5
Two Branches [5]	58.4	96.4	57.8	93.5	48.8	98.2	94.7	92.2	96.2
Ours	**65.2**	**100.0**	**60.8**	99.0	**58.2**	**100.0**	**98.1**	**98.0**	**97.6**

Table 6. Comparative experiments on CRPD.

Method	Detection			Recognition
	AP	$AP_{0.5}$	FPS	Accuracy
SYOLOv4+CRNN [2]	-	-	-	71.0%
RCNN+CRNN [22]	-	-	-	73.7%
UCLP [32]	-	-	-	84.1%
Direct Detection [4]	53.0%	96.3%	**12.8**	86.0%
Vehicle Pre-detection [6]	57.4%	97.8%	7.4	86.2%
Two Branches [5]	58.8%	98.1%	4.8	87.5%
Ours	**62.9%**	**98.3%**	12.5	**89.3%**

pre-detection (Fig. 1(b)), and two branches combining direct detection and vehicle pre-detection (Fig. 1(c)). To ensure a fair comparison, all of these comparative methods utilize the same backbone and transformer as our proposed method. After performing license plate detection, both the comparative methods and our proposed method employ the same YOLO-based character recognizer [15] for license plate recognition. Our proposed method demonstrates superior detection and recognition performance on the SSIG-SegPlate and CRPD datasets while achieving the best performance on most subsets within the AOLP dataset. Concretely, our proposed method achieves an average $AP_{0.5}$ of 99.5% and an average recognition accuracy of 95.9% on the SSIG-SegPlate and CRPD datasets and three subsets of AOLP. However, for the LE subset of AOLP, our proposed method can not effectively handle some low-light images. In future work, we aim to enhance license plate detection under low-light conditions.

Moreover, the direct detection method [4] offers the fastest inference speed but suffers from the lowest detection and recognition performance due to its limited ability to detect small license plates. On the other hand, the vehicle pre-detection method [6] improves license plate detection at the cost of slower

Fig. 5. Visualization examples of license plate detection and recognition.

inference speed. By combining direct detection and vehicle pre-detection, the two branches method [5] further enhances license plate detection and recognition, albeit with the slowest inference speed. In contrast, our proposed method achieves the best detection and recognition performance while maintaining a comparable inference speed to the direct detection method.

Figure 5 demonstrates that our proposed method can accurately detect vehicles and license plates, and the YOLO-based character recognizer [15] can accurately recognize the detected license plates based on character detection.

4.6 Experiments on Multi-scale License Plates

Table 7. Comparative experiments on multi-scale license plates of the CRPD dataset.

Method	Detection (LP)						Recognition		
	S		M		L		S	M	L
	AP	$AP_{0.5}$	AP	$AP_{0.5}$	AP	$AP_{0.5}$	Accuracy		
Direction Detection [4]	45.3	92.4	56.7	96.5	62.6	96.9	82.2	86.1	86.8
Simultaneous Detection [7]	45.0	92.0	56.6	96.8	62.1	96.9	82.0	86.2	86.7
Vehicle Pre-detection [6]	48.7	93.5	59.0	97.3	62.4	98.0	83.5	87.4	87.6
Two Branches [5]	50.5	93.6	60.4	98.1	64.7	98.5	84.0	88.4	88.5
Ours	**55.0**	**95.6**	**62.5**	**98.4**	**67.3**	**99.2**	**85.1**	**89.2**	**89.9**

Table 7 presents comparative experiments involving multi-scale license plates on the CRPD dataset. Notably, we do not conduct multi-scale experiments on the SSIG-SegPlate and AOLP datasets because the size of license plates in these datasets is relatively consistent. In all of these sizes, we compare three approaches: direct detection (Fig. 1(a)), vehicle pre-detection (Fig. 1(b)), and two branches combining direct detection and vehicle pre-detection (Fig. 1(c)). Moreover, the simultaneous detection method denotes detecting vehicles and license plates simultaneously using the vanilla DETR model. To ensure a fair comparison, all of these comparative methods utilize the same backbone and transformer as our proposed method. After performing license plate detection, both the comparative methods and our proposed method employ the same YOLO-based character recognizer [15] for license plate recognition. Our proposed method demonstrates superior performance in both license plate detection and recognition across all sizes, especially for small license plate detection. Concretely, it achieves a 4.5% AP improvement in the detection performance of small license plates compared to the two branches method, with a 2.1% AP improvement for medium license plates and a 2.6% AP improvement for large license plates.

As depicted in Fig. 6, our method can effectively detect small license plates at a considerable distance. Our method can achieve comparative inference speed with the direct detection method, surpassing other comparative methods. Moreover, our method can detect vehicles truncated by image edges due to the bidirectional relationships between vehicles and license plates.

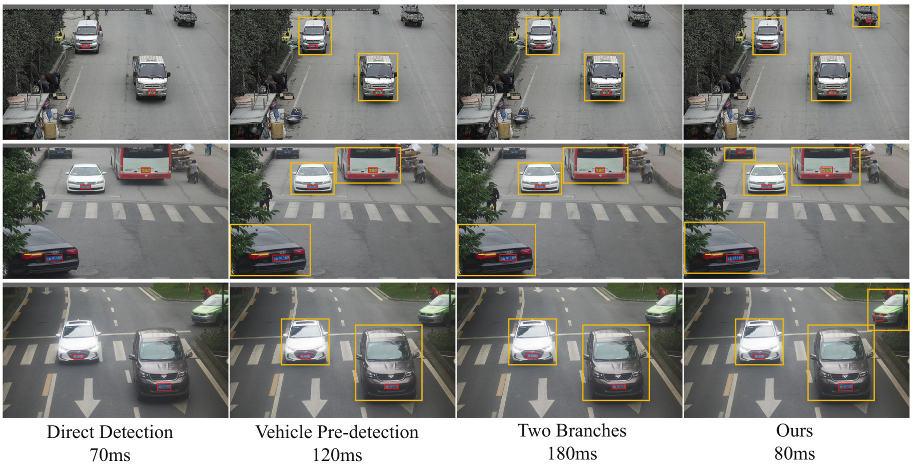

Fig. 6. Visualization examples. Under challenging conditions, our proposed method can accurately small license plates at a fast inference speed.

5 Conclusion

We propose to leverage bidirectional relationships between the vehicle and the license plate to enhance small license plate detection. Extensive experiments on the SSIG-SegPlate, AOLP, and CRPD datasets prove our method achieves state-of-the-art detection performance, especially for small license plates. When incorporating a character recognizer, our proposed method can surpass all comparative methods in license plate recognition. In the future, we aim to enhance license plate detection under severe low-light conditions, enabling it to handle more complex scenarios.

Acknowledgements. This work was partly supported by the National Key Research and Development Program of China under Grant 2020AAA0109700 and partly by the National Natural Science Foundation of China under Grant 62076024 and Grant 62006018.

References

1. Batra, P., Hussain, I., Ahad, M.A., Casalino, G., Alam, M.A., Khalique, A., Hassan, S.I.: A novel memory and time-efficient alpr system based on yolov5. Sensors **22**(14), 5283 (2022)
2. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
3. Borisyuk, F., Gordo, A., Sivakumar, V.: Rosetta: large scale system for text detection and recognition in images. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 71–79 (2018)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12346, pp. 213–229. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_13
5. Chen, S.L., Liu, Q., Ma, J.W., Yang, C.: Scale-invariant multidirectional license plate detection with the network combining indirect and direct branches. Sensors **21**(4), 1074 (2021)
6. Chen, S.L., et al.: End-to-end trainable network for degraded license plate detection via vehicle-plate relation mining. Neurocomputing **446**, 1–10 (2021)
7. Chen, S.L., Yang, C., Ma, J.W., Chen, F., Yin, X.C.: Simultaneous end-to-end vehicle and license plate detection with multi-branch attention neural network. IEEE Trans. Intell. Transp. Syst. **21**(9), 3686–3695 (2019)
8. Glenn Jocher, Alex Stoken, J.B.: yolov5. https://github.com/ultralytics/yolov5. (2020)
9. Gonçalves, G.R., da Silva, S.P.G., Menotti, D., Schwartz, W.R.: Benchmark for license plate character segmentation. J. Electron. Imaging **25**(5), 053034–053034 (2016)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
11. Hsu, G.S., Chen, J.C., Chung, Y.Z.: Application-oriented license plate recognition. IEEE T-VT **62**(2), 552–561 (2012)
12. Kim, S., Jeon, H., Koo, H.: Deep-learning-based license plate detection method using vehicle region extraction. Electron. Lett. **53**(15), 1034–1036 (2017)
13. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Res. Logistics Quarterly **2**(1–2), 83–97 (1955)
14. Laroca, R., et al.: A robust real-time automatic license plate recognition based on the yolo detector. In: 2018 International Joint Conference on Neural Networks (ijcnn), pp. 1–10. IEEE (2018)
15. Laroca, R., Zanlorensi, L.A., Gonçalves, G.R., Todt, E., Schwartz, W.R., Menotti, D.: An efficient and layout-independent automatic license plate recognition system based on the yolo detector. IET Intel. Transport Syst. **15**(4), 483–503 (2021)
16. Li, H., Shen, C.: Reading car license plates using deep convolutional neural networks and lstms. arXiv preprint arXiv:1601.05610 (2016)

17. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2117–2125 (2017)
18. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)
19. Lin, T.Y., et al.: Microsoft coco: Common objects in context (2014)
20. Liu, W., Chen, C., Wong, K.Y.K., Su, Z., Han, J.: Star-net: a spatial attention residue network for scene text recognition. In: BMVC, vol. 2, p. 7 (2016)
21. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
22. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. Advances in neural information processing systems 28 (2015)
23. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: a metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 658–666 (2019)
24. Selmi, Z., Halima, M.B., Alimi, A.M.: Deep learning system for automatic license plate detection and recognition. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, pp. 1132–1138. IEEE (2017)
25. Selmi, Z., Halima, M.B., Pal, U., Alimi, M.A.: Delp-dar system for license plate detection and recognition. Pattern Recogn. Lett. **129**, 213–223 (2020)
26. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: fully convolutional one-stage object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9627–9636 (2019)
27. Vaswani, A., et al.: Attention is all you need. Advances in neural information processing systems 30 (2017)
28. Xu, Z., Yang, W., Meng, A., Lu, N., Huang, H., Ying, C., Huang, L.: Towards end-to-end license plate detection and recognition: a large dataset and baseline. In: Proceedings of the European conference on computer vision (ECCV), pp. 255–271 (2018)
29. Zhang, H., Yao, Q., Yang, M., Xu, Y., Bai, X.: Autostr: efficient backbone search for scene text recognition. In: Computer Vision-ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16, pp. 751–767. Springer (2020)
30. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9759–9768 (2020)
31. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 (2020)
32. Zou, Y., Zhang, Y., Yan, J., Jiang, X., Huang, T., Fan, H., Cui, Z.: License plate detection and recognition based on yolov3 and ilprnet. SIViP **16**(2), 473–480 (2022)