Journal of Real-Time Image Processing (2024) 21:168
https://doi.org/10.1007/s11554-024-01537-2

RESEARCH

Integrating YOLOv8 and CSPBottleneck based CNN for enhanced license plate character recognition

Sahil Khokhar[1] · Deepak Kedia[1]

Received: 1 May 2024 / Accepted: 7 August 2024

Abstract

The paper introduces an integrated methodology for license plate character recognition, combining YOLOv8 for segmentation and a CSPBottleneck-based CNN classifier for character recognition. The proposed approach incorporates pre-processing techniques to enhance the recognition of partial plates and augmentation methods to address challenges arising from colour diversity. Performance analysis demonstrates YOLOv8's high segmentation accuracy and fast processing time, complemented by precise character recognition and efficient processing by the CNN classifier. The integrated system achieves an overall accuracy of 99.02% with a total processing time of 9.9 ms, offering a robust solution for automated license plate recognition (ALPR) systems. The integrated approach presented in the paper holds promise for the practical implementation of ALPR technology and further development in the field of license plate recognition systems.

Keywords Computer vision · Optical character recognition · Deep learning · Artificial intelligence · ALPR

1 Introduction

Optical character recognition (OCR) technology has revolutionised the way textual information is extracted from images, enabling automated data processing in various domains. By leveraging machine learning and computer vision techniques, OCR systems can accurately interpret text from scanned documents, images, and other visual sources. The applications of OCR extend across industries, facilitating efficient data entry, document digitisation, and text analysis.

One prominent application area of OCR technology is automated license plate recognition (ALPR) systems, where recognising alphanumeric characters from license plates plays a crucial role. ALPR systems utilise OCR algorithms to automatically capture, analyse, and interpret license plate information from images or video streams captured by cameras installed in different environments. These systems find wide-ranging applications in law enforcement, traffic management, parking enforcement, toll collection, and security operations.

In this context, accurately recognising license plate characters forms the cornerstone of ALPR systems' functionality. Traditionally, ALPR systems relied on manual feature extraction techniques and rule-based algorithms to perform license plate recognition tasks. However, the limitations of these conventional approaches, such as susceptibility to noise, varying lighting conditions, and complex backgrounds, necessitated exploring more robust solutions.

This paper proposes an integrated approach combining cutting-edge object detection techniques for character segmentation and deep learning-based character recognition to address these challenges. The proposed work harnesses the power of YOLOv8, a state-of-the-art object detection algorithm, for precise character segmentation while cascading it with a CSPBottleneck-based Convolutional Neural Network (CNN) classifier for accurate character identification. This synergistic approach aims to achieve high accuracy and real-time processing capabilities essential for practical ALPR applications. Comprehensive experiments conducted on a diverse dataset encompassing various plate designs, colours, and environmental conditions evaluate the effectiveness of the proposed methodology. Key performance metrics include segmentation accuracy, recognition accuracy and processing time, providing insights into the system's performance and efficiency.

✉ Sahil Khokhar
Khokhar.sahil0809@gmail.com

[1] Department of ECE, GJUS &T, Hisar, India

Published online: 10 September 2024

The findings presented in this paper contribute to advancing the capabilities of ALPR systems, addressing critical challenges in license plate recognition, and paving the way for enhanced real-time solutions. Moreover, the research outcomes have broader implications for intelligent transportation systems, public safety, and security applications, underscoring the significance of OCR technology in modern-day information processing and automation endeavours. The proposed methodology achieves a remarkable accuracy rate of 99.02% coupled with an impressive processing speed of 9.9 milliseconds. To evaluate the effectiveness of this approach, the AOLP dataset, a benchmark dataset widely recognised in the field of license plate recognition, has been employed.

The accomplishments of this research are multiface- ted:

- Optimising the YOLOv8 algorithm to prioritise character detection while enhancing processing speed without compromising accuracy.
- A novel pre-processing technique was introduced to enhance partial plate recognition and overcome limitations associated with the resizing constraints of YOLOv8.
- Implementing an innovative augmentation method involving colour augmentation and intensity adjustments to address the diversity of license plate colours, ensuring robust recognition performance across varied types of license plates.
- Implementation of popular image classifiers, namely, VGG16, ResNet, Inception, DenseNet, EfficientNet, MobileNet, NASNet and Xception algorithms for character recognition.
- Implementation of a CSPBottleneck-based CNN classifier for character recognition, enhancing the accuracy and efficiency of the overall system.

In summary, this paper showcases a significant advancement in License Plate Recognition, focusing on character recognition. A robust and efficient methodology that exceeds the limitations of existing approaches is proposed.

The subsequent sections of this paper are organised as follows. In Sect. 2, we briefly review the related ALPR methods; in Sect. 3, we introduce the technique proposed in detail; in Sect. 4, we elaborate on the experimental results and compare them with existing techniques; and finally, in Sect. 5, a summary of the research work and the scope for future work is provided.

2 Related work

The research on ALPR systems can be mainly categorised into two areas: license plate location and license plate recognition.

2.1 License plate detection

Identifying license plates accurately is essential, especially when pinpointing their exact location within everyday environments such as streets or parking areas. In earlier research, traditional methods like morphological operations [5] and connected component analysis (CCA) were used to capitalize on the unique features, especially the rectangular form of license plates. Additional characteristics like colour and placement relative to vehicle rear lights also played a role in improving the accuracy of the detection. However, as machine learning and artificial intelligence have advanced, there has been a significant shift towards these technologies in research. Techniques such as genetic algorithms [17], support vector machines (SVMs) [16], and neural networks [19] have become fundamental in the task of locating license plates. The Histogram of Gradients (HoG) [2] became a preferred feature for detecting plates effectively. With the advent of deep learning, algorithms gained the ability to independently optimise feature extraction, leading to the adaptation and refinement of general-purpose object detectors [14] such as YOLO [12], SSD, RetinaNet, Masked RCNN [20] and RNN [29]. These adjustments have enabled these systems to achieve exceptional results in the complex process of detecting license plates.

2.2 License plate recognition

License plate recognition is a sophisticated process that involves identifying and deciphering the characters on a license plate. Within this process, two main approaches have emerged, each with its unique advantages. The segmented character approach, foundational in the evolution of license plate recognition technologies, meticulously segments each character for individual recognition. This method has historically been the cornerstone of recognition systems, employing techniques such as horizontal and vertical histogram analysis and connected component analysis to isolate characters. Following segmentation, various character recognition techniques have been successfully applied, including template matching and advanced machine learning algorithms like Neural Network based classifiers [4], genetic algorithms, and SVMs [13]. Table 1 summarises the most popular CNN-based image classifier architectures that can be used for character recognition.

Parallel to this, the non-segmented character approach offers an alternative strategy by recognising characters without prior segmentation, treating the entire license plate as a single input. This method benefits from integrated models such as YOLO and Faster RCNN, which

Table 1 Comparison of the most popular image classifiers

Classifier	Year	Main contribution	Depth	Parameters	Key architectural features	Use cases
VGG16 [23]	2014	Simplicity and depth	16 layers	~138 M	Sequential convolutions	General-purpose, simpler tasks
ResNet [6]	2015	Residual connections to enable deeper networks	50 layers	~25 M	Skip connections and bottleneck layers	General-purpose
Inception [25]	2014	Parallel convolutions with varying kernel sizes	48 layers	~23 M	Inception modules, hybrid convolution sizes	Efficient processing of different scales
DenseNet [9]	2017	Feature reuse through dense connections	121 layers	~8 M	Dense connectivity, feature reuse	Dense feature extraction and reuse
EfficientNet [26]	2019	Compound scaling (width, depth, resolution)	237 layers	~66 M	Efficient use of resources, dynamic scaling	Mobile and edge devices
MobileNet [7]	2017	Lightweight architecture with depthwise separable convolutions	28 layers	~3.2M	Depthwise and pointwise convolutions, small model size	Real-time applications on mobile devices
NASNet [30]	2017	Architecture search for performance optimisation	Over 1000 layers	~88 M	Cell-based architecture, reinforcement learning design	High-performance tasks
Xception [1]	2016	Depthwise separable convolutions	71 layers	~22 M	Modified depthwise separable convolutions	General-purpose

have been fine-tuned to deliver enhanced performance in recognising license plates in their entirety.

This comprehensive overview underscores the dynamic evolution from traditional methods to sophisticated ML and AI techniques in license plate detection and recognition, demonstrating the continual progress in ALPR research.

3 Proposed method

This section outlines the proposed method for improving character recognition in license plates using a combination of optimised YOLOv8x algorithm with a CSPBottleneck integrated CNN classifier. The research aims to enhance recognition accuracy while maintaining a high processing speed. A comprehensive framework has been developed that encompasses several key components, including accuracy and speed optimisation, pre-processing, and augmentation techniques.

3.1 Optimised YOLOv8 for character segmentation

The You Only Look Once version 8 (YOLOv8) [28] algorithm is a state-of-the-art object detection system known for its efficiency and accuracy. Figure 1 shows the detailed architecture of the YOLOv8x algorithm. YOLO [21] algorithms operate by dividing the input image into a grid and simultaneously predicting bounding boxes and class probabilities for each grid cell. This approach enables real-time object detection by eliminating the need for multiple passes through the neural network.

3.1.1 Optimising YOLOv8 for speed

The YOLOv8x algorithm was adopted for character segmentation within license plates. While the standard YOLOv8 is used for general object detection, its architecture and training process were specifically optimised for segmenting license plate characters. Focusing solely on one class, license plate characters, allowed further optimisation of the algorithm for speed. This optimisation was feasible due to the simplicity of license plate images compared to the complex images on which YOLOv8x was originally trained. As illustrated in Fig. 2, the complexity of the backbone and neck of the algorithm was reduced. Additionally, the detection head for small objects was eliminated since the characters were of large and medium sizes relative to the license plate image. Layers used for identifying object classes were removed because there was only one class of objects. These changes reduced computational overhead and enhanced processing time, which is crucial for real-time applications. These optimisations not only improved detection accuracy but also significantly increased the system's speed, making it highly effective for deployment in real-world scenarios where rapid character recognition is essential.

3.1.2 Pre-processing techniques for partial plate recognition

One key challenge in license plate recognition is dealing with partial plates, which may contain only a subset of characters. To address this issue, we introduced a pre-processing technique that adjusts the aspect ratio of partial plates to match that of complete plates. Specifically, if a plate is

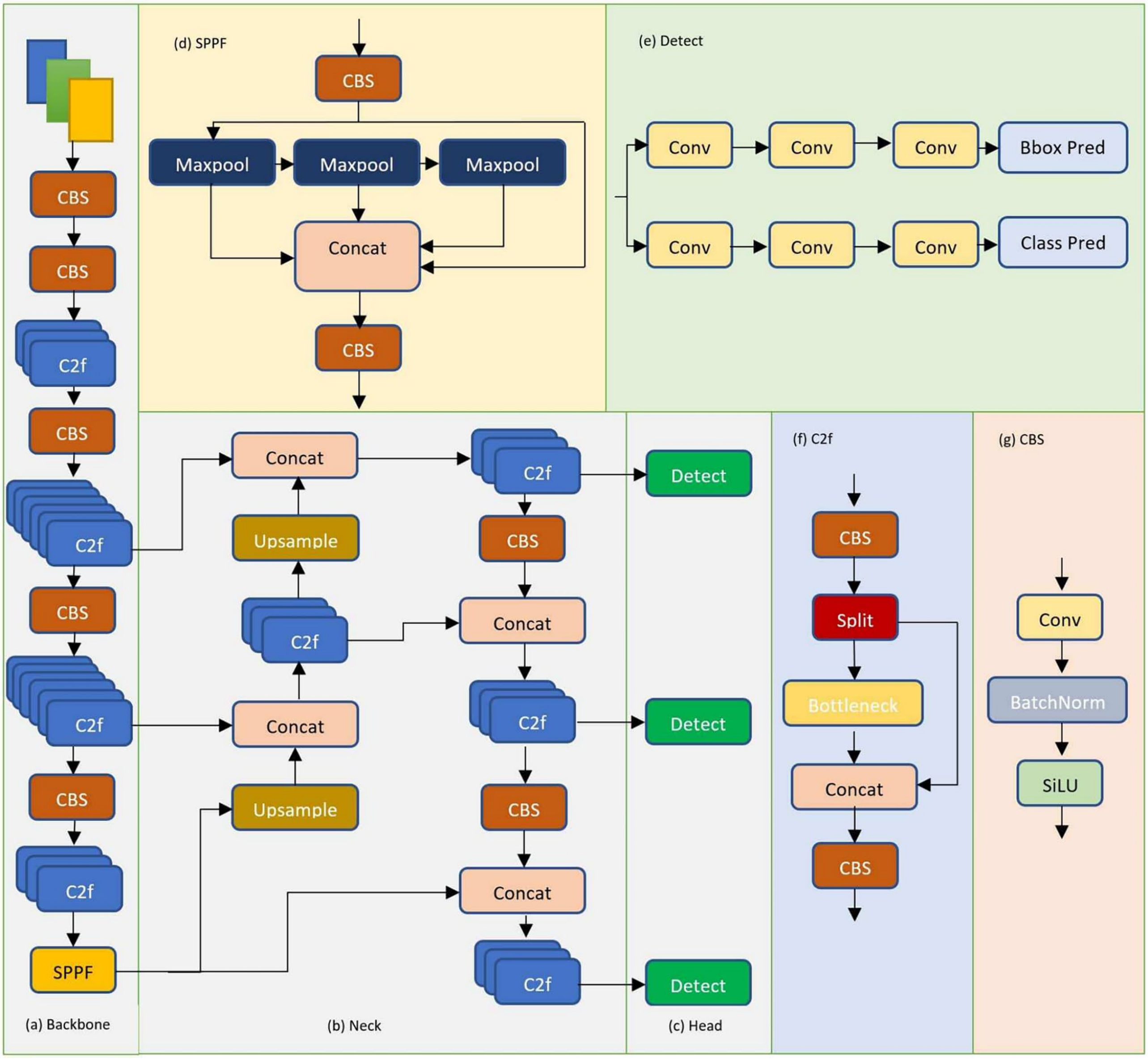


Fig. 1 Architecture of YOLOv8

detected with an aspect ratio greater than 1:2, we employ a method to strategically add dead pixels to the image. These dead pixels are inserted in such a way that the resulting image achieves an aspect ratio of 2:7, which is more consistent with the dimensions of complete license plates. This adjustment ensures that characters in partial plates undergo similar processing as those in complete plates, thereby improving detection accuracy. Figure 3 depicts the proposed pre-processing technique.

3.1.3 Augmentation techniques for colour diversity

License plates exhibit significant variability in colour, which can pose challenges for traditional recognition methods. To address this variability, we implemented augmentation techniques to enhance colour diversity in the training dataset. Specifically, we employed methods for colour inversion and colour alteration. Colour inversion involves flipping the black and white colours within the license plate image, while colour alteration replaces white pixels with colours such as yellow, green, and blue. These augmentation techniques help the model generalise better to different colour variations encountered in real-world scenarios. Figure 4 illustrates the effect of the proposed data augmentation technique.

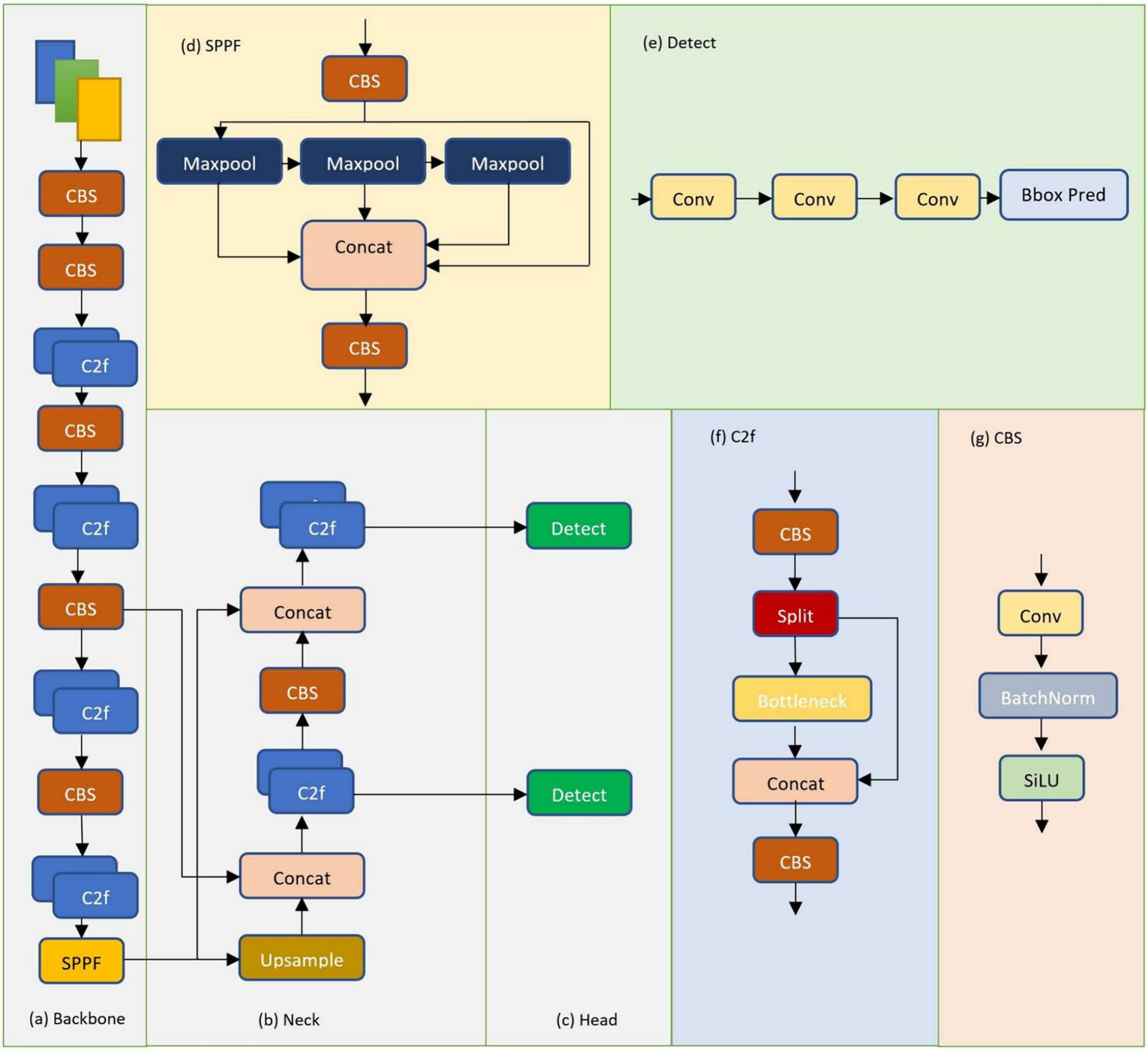
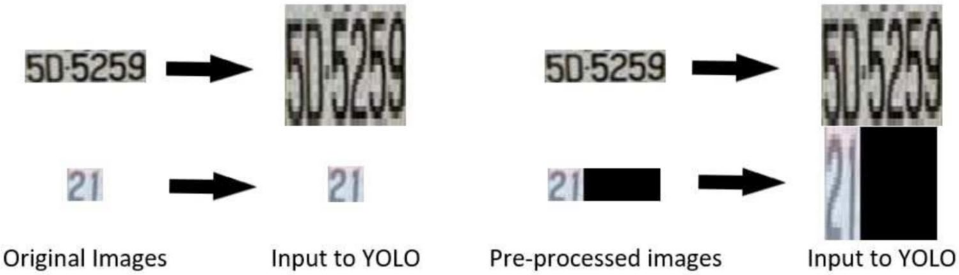


Fig. 2 Architecture of optimised YOLOv8 for character segmentation

Fig. 3 The proposed pre-processing technique for improved character segmentation



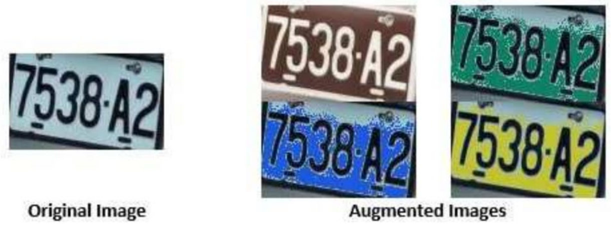


Fig. 4 Augmentation for diverse plate colours

3.2 CSPBottleneck-based CNN Classifier for Character Recognition

The CSPBottleneck-based CNN classifier is a crucial component of our proposed method for license plate recognition. Unlike traditional CNN architectures, it integrates bottleneck structures with cross-stage feature aggregation to enhance feature representation and learning capability. The bottleneck structures efficiently utilise computational resources by reducing the number of parameters while maintaining representational capacity. Cross-stage feature aggregation facilitates the propagation of information across different stages of the network, enabling the fusion of high-level and low-level features for improved discrimination.

3.2.1 Data augmentation techniques

Various data augmentation techniques were employed during the training phase to enhance the robustness and generalisation of the CNN classifier. Figure 5 illustrates the effect of various data augmentation techniques in training the CSPBottleneck-based classifier. These techniques include:

- Image Rotation: Randomly rotating the input images by a certain degree to expose the classifier to variations in orientation commonly observed in license plates.
- Shear Transformation: Shear transformations are applied to the input images to simulate perspective distortions, enabling the classifier to recognize characters from different viewing angles.
- Zooming: Randomly zoom in on specific regions of the input images to facilitate the detection of characters at varying scales.
- Adjusting Width and Height: Altering the width and height of the input images to expose the classifier to variations in aspect ratio commonly encountered in license plates.
- Brightness Range Adjustment: Adjusting the brightness range of input images helps the classifier accurately interpret license plates under various lighting conditions, enhancing its robustness and versatility in real-world scenarios.

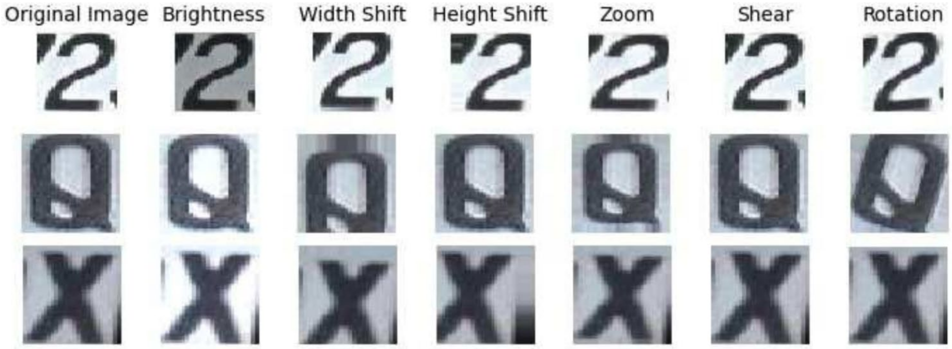
By augmenting the training data with these transformations, the CNN classifier becomes more resilient to variations in perspective and scale, thereby improving its ability to accurately recognise characters under diverse conditions.

3.2.2 Architecture of the classifier

The CSPBottleneck-based CNN classifier comprises over 4.7 million trainable parameters with a depth of 60 layers. These parameters collectively define the network architecture and enable the representation of complex relationships within the input data. These parameters include weights and biases associated with the convolutional layers, fully connected layers, and other network components. The overall architecture is shown in Fig. 6.

The Parametric Rectified Linear Unit (PReLU) was employed as the activation function. The Parametric Rectified Linear Unit (PReLU) activation function enhances neural network learning by allowing a small, learnable coefficient for negative inputs. For positive inputs, it behaves like the standard ReLU, passing the input directly. For negative inputs, it outputs a small value proportional to the input, preventing neurons from becoming inactive. This learnable parameter helps maintain gradient flow during backpropagation, improving model performance and flexibility.

Fig. 5 Data augmentation techniques used for classifier



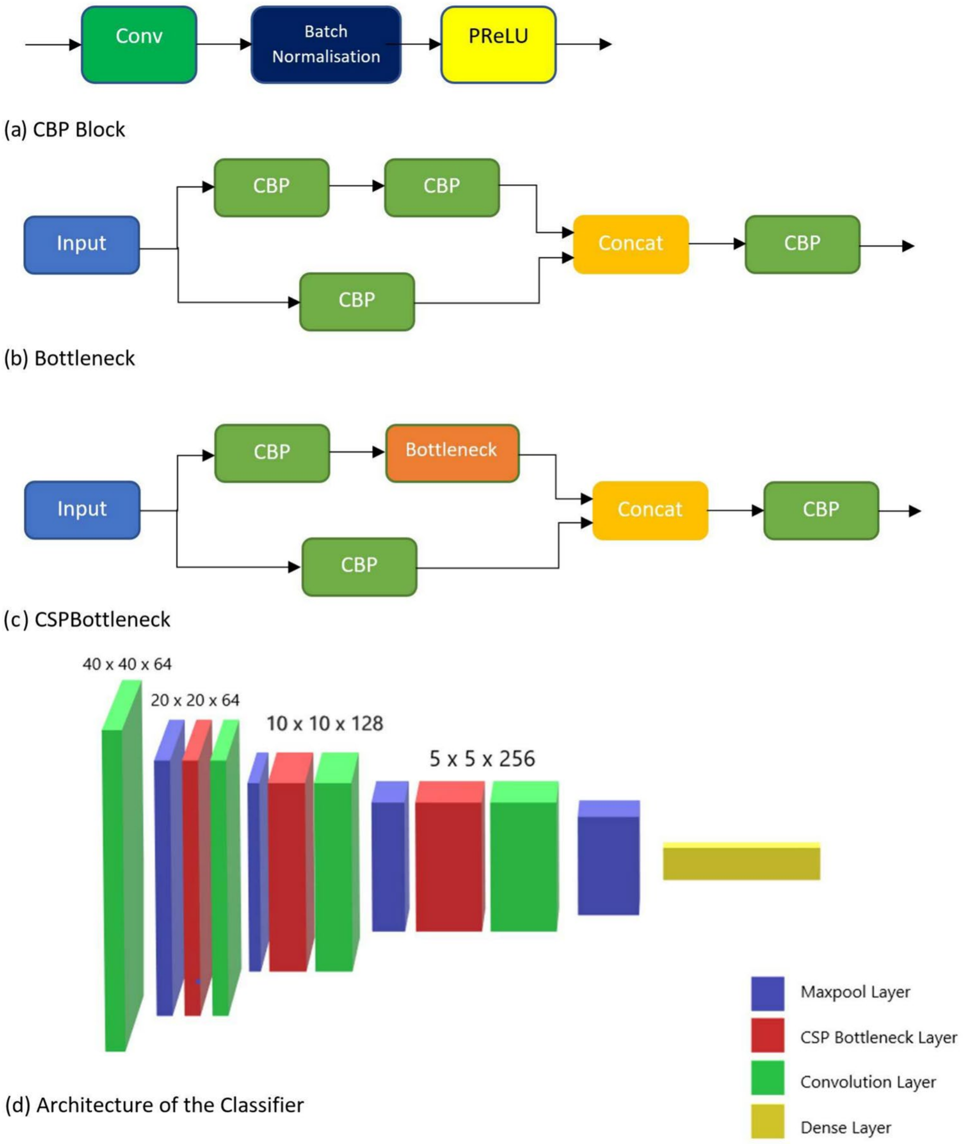


Fig. 6 **a** Architecture of CBP Block. **b** Architecture of Bottleneck block. **c** Architecture of CSPBottleneck Block **d** The architecture of the CSPBottleneck-based CNN classifier

This flexibility allows PReLU to adjust dynamically to the training data, potentially improving learning outcomes and enhancing model performance, especially in deep learning tasks with complex or noisy data sets. The Parametric Rectified Linear Unit (PReLU) activation function is defined as:

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha x & \text{if } x \leq 0 \end{cases} \tag{1}$$

where x is the input to the neuron and α is a learnable parameter.

3.2.3 Loss

As seen from Fig. 7, a significant imbalance in the class distribution of various characters can significantly skew the model's performance towards majority classes, resulting in biased predictions and poor generalisation of minority classes. This leads to issues such as overfitting to frequent classes, decreased sensitivity to rarer ones, and reduced overall training efficiency. The Focal Cross Entropy Loss function was employed to address this class imbalance issue.

The Focal loss function is particularly adept at enhancing the model's focus on difficult-to-classify examples, often overshadowed by the majority class in traditional training methods. By introducing a focusing parameter, γ, the Focal Loss adjusts the contribution of each sample to the loss based on the ease of classification, reducing the influence of easy examples and amplifying that of hard ones. This approach not only accelerated the convergence of our model during training but also significantly improved the robustness and accuracy of the classifier across diverse and skewed datasets. The adaptive nature of Focal Loss proved crucial in our experiments, especially in scenarios where

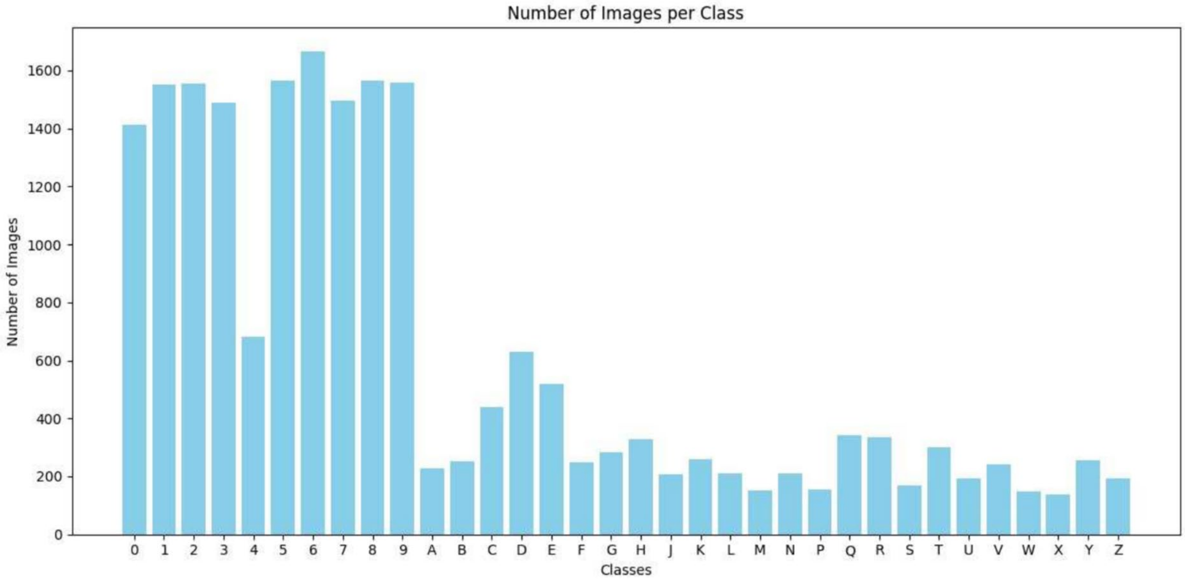


Fig. 7 Class distribution of the dataset

discriminative learning from imbalanced data was critical for achieving high performance.

$$\text{Focal Loss}(p_t) = -\alpha \times (1 - p_t)^{\gamma} \times \log(p_t) \tag{2}$$

Where:

- p_t: The model's estimated probability for each class being the true class.
- α: A weighting factor for each class.
- γ: A focusing parameter to adjust the rate at which easy examples are down-weighted.

3.2.4 Optimiser

The role of an optimiser in machine learning models is primarily to minimise the loss function that quantifies the error between predicted outcomes and actual targets. Optimisers achieve this by iteratively adjusting the model parameters, such as weights and biases, enhancing accuracy and performance on the designated tasks. The efficacy of these optimisers not only influences the speed of convergence during training but also affects the overall stability and performance of the model. As such, selecting an appropriate optimiser is crucial for ensuring efficient learning and managing the trade-off between convergence rate and the likelihood of reaching a satisfactory local minimum on complex loss landscapes.

Among various optimisers, AdamW stands out as a refined variant of the widely used Adam optimiser, integrating adjustments specifically in handling weight decay, which aids in better generalisation capabilities and prevents overfitting. Unlike standard Adam, which may mishandle weight decay leading to suboptimal regularisation, AdamW decouples weight decay from the adaptive learning rate updates, thereby ensuring more effective regularisation. This property of AdamW not only supports faster convergence but also facilitates more robust learning across different parameter scales and complex model architectures. Consequently, AdamW's adaptive learning rate mechanism and enhanced handling of weight decay made it a superior choice for training the network.

3.3 Evaluation metrics and results

The accuracy and processing time were used to evaluate the performance of the optimised YOLOv8 and the CSPBottleneck-based CNN classifier. These metrics were computed based on the predictions compared to ground truth labels of the corrected dataset.

4 Experimental results

Extensive experiments using the AOLP dataset [8] were conducted. The AOLP (Application-Oriented License Plate) database is a publicly available dataset comprising 2,049 images of Taiwanese license plates. It is categorised into three subsets: access control (AC), law enforcement (LE), and road patrol (RP), containing 681, 757, and 611 images, respectively. Each subset showcases diverse scenarios, including variations in plate tilt, width ratio, distance, etc. Subset AC features images of vehicles passing through fixed checkpoints, like toll stations. Subset LE encompasses

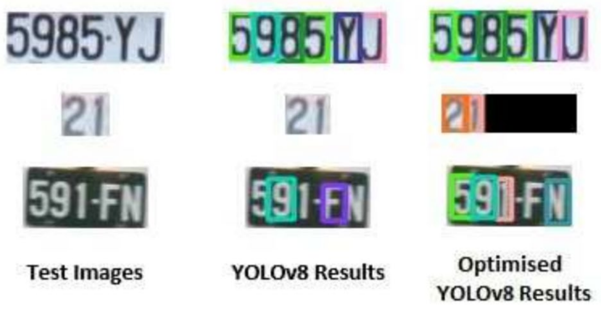


Fig. 8 Comparison of segmentation results of YOLOv8 algorithm and optimised YOLOv8 algorithm

images from roadside cameras used to enforce traffic laws. Lastly, subset RP comprises images taken by handheld cameras employed in tasks such as detecting parking violations and locating lost vehicles. The dataset was split in 70:20:10 for training, validation and testing purposes. The experiments were conducted using a system with an i7-11700K processor, 16GB RAM and RTX 3060Ti.

The evaluation of the models is based on two primary metrics, accuracy and the processing time taken by the model. These metrics are crucial for determining the overall effectiveness and efficiency of the models in practical applications. It's essential to note that the character segmentation algorithm processes cropped license plate images and the character recognition algorithm utilises these segmented images. It segments and recognises characters within these images. A result is deemed correct only if the characters are segmented correctly and recognised accurately. Conversely, if certain characters are missed or extra characters are segmented, it will be considered incorrect segmentation, and if the character is misrecognised, it is considered incorrect recognition. The accuracy of the model is calculated as:

$$\text{Accuracy (Segmentation)} = \frac{\text{Correct Segmentation}}{\text{Correct Seg} + \text{Incorrect Seg}} \tag{3}$$

$$\text{Accuracy (Recognition)} = \frac{\text{Correct Recognition}}{\text{Correct Rec} + \text{Incorrect Rec}} \tag{4}$$

Figure 8 shows the segmentation results of YOLOv8 and the optimised YOLOv8 algorithm. The figure illustrates that the proposed method outperforms the base YOLOv8 model, particularly in cases of coloured or partial plates. Table 2 presents the individual effects of optimising the YOLOv8 algorithm, the proposed pre-processing technique and the proposed data augmentation technique. Optimising the YOLOv8 algorithm results in a marginal improvement in accuracy and a significant improvement in speed. Additionally, the proposed pre-processing and data augmentation techniques contribute significantly to the increase in the algorithm's accuracy.

Table 2 Effect of pre-processing and data augmentation techniques on character segmentation

Algorithm used	Pre-processing technique used	Data augmentation technique used	Accuracy	computation time (ms)
Improved YOLOv8			98.9	**7.8**
Improved YOLOv8	✓		99.3	**7.8**
Improved YOLOv8		✓	99.2	**7.8**
Improved YOLOv8	✓	✓	**99.6**	**7.8**

Best results are highlighted in bold

Table 3 Effect of data augmentation technique on character recognition using CSPBottleneck classifier

Data augmentation technique used						Accuracy	Computation time (ms)
Brightness	Width Shift	Height Shift	Zoom	Shear	Rotation		
						98.97	2.1
✓						99.14	2.1
	✓					99.06	2.1
		✓				99.01	2.1
			✓			99.04	2.1
				✓		99.09	2.1
					✓	99.12	2.1
✓	✓	✓	✓	✓	✓	99.42	2.1

Table 3 shows the effect of various data augmentation techniques used on the accuracy of character recognition using CSPBottleneck classifier.

To evaluate the effectiveness of our newly proposed classifier, we conducted a comparative analysis with other widely recognized classifiers, all implemented, trained and tested under uniform conditions using the same dataset. As illustrated in Table 4, the proposed classifier achieved the highest accuracy rate at 99.42%, while maintaining an impressive computation speed of just 2.1 milliseconds per image. While MobileNet presented the fastest processing time at 1.1 milliseconds per image, its accuracy was noticeably lower at 81.5%. This demonstrates that, although MobileNet processes images faster, it compromises significantly on accuracy, unlike our CSPBottleneck based CNN classifier, which effectively balances both speed and precision.

The YOLOv8-based algorithm achieved a segmentation accuracy of 99.6% with a processing time of 7.8ms. This indicates the high precision of the algorithm in accurately identifying and segmenting characters within license plate images. The CSPBottleneck-based CNN classifier demonstrated exceptional character recognition accuracy, achieving an average accuracy of 99.42% and a processing time of 2.1 ms per image. This highlights the effectiveness of the classifier in accurately recognizing individual characters under various conditions.

The total processing time of the integrated license plate recognition system was measured at 9.9 ms per image. This includes the segmentation and character recognition stages, indicating the system's efficiency in real-time applications. The proposed method achieved an impressive accuracy rate of 99.02%, showcasing its superior recognition capabilities. Moreover, the model operates at a remarkable speed of 9.9 milliseconds, making it suitable for real-time applications. Table 5 compares the performance of the proposed algorithm with that of existing techniques. As can be seen in the table, not only does the proposed algorithm achieve high accuracy, but it also takes the least processing time.

Table 4 Performance comparison of various image classifiers with the proposed classifier

Recognition algorithm	Proposed by	Accuracy (%)	Computation time (ms)	Dataset used
VGG16	[23]	98.8	4.3	AOLP
ResNet	[6]	97.8	4.5	AOLP
Inception	[25]	95.9	3.5	AOLP
DenseNet	[9]	96.8	4.3	AOLP
EfficientNet	[26]	98.2	16	AOLP
MobileNet	[7]	81.5	**1.1**	AOLP
NASNet	[30]	96.7	14.8	AOLP
Xception	[1]	98	5.1	AOLP
LPRNet	[31]	97.5	–	AOLP
CRNN	[27]	97.7	15.9	AOLP
CNN	[3]	96	–	AOLP
CSPBottleneck-based CNN	us	**99.42**	2.1	AOLP

Best results are highlighted in bold

Table 5 Comparison of proposed algorithm with existing techniques

Method	Proposed by	Total accuracy	Total processing time (ms)	Dataset
CNN	[22]	95.5	–	AOLP
Bidirectional RNN	[15]	93	41	AOLP
MobileNet + Bi-LSTM	[11]	91.27	370	AOLP
ResNet +FPN	[10]	95.12	–	AOLP
Inception-v3	[24]	97.46	–	AOLP
Xception	[18]	99	–	AOLP
YOLOv8	Us	98.4%	13.8	AOLP
Proposed Method	Us	**99.02**%	**9.9**	AOLP

Best results are highlighted in bold

5 Conclusion and future work

In summary, our research has presented a novel approach to license plate recognition that addresses key challenges faced by existing systems. Through the implementation of YOLOv8-based character segmentation and CSPBottleneck-based CNN classifiers, significant improvements have been demonstrated in accuracy and speed. Our findings underscore the importance of our proposed method in overcoming limitations inherent in traditional license plate recognition systems. By optimising algorithms, introducing pre-processing techniques for partial plate recognition, and augmenting colour diversity, we have achieved notable advancements in character segmentation and recognition.

While our study has yielded promising results, it is not without limitations. Future research endeavours should address these constraints to further refine and enhance the proposed method.

Moving forward, several avenues for future research warrant exploration:

- Complex challenges: While our ALPR system excels in many real-world scenarios, there are still challenges, such as extreme lighting conditions and novel license plate designs.
- Dataset expansion: Expanding the dataset to include more countries and diverse license plate variations should be considered to enhance the system's adaptability. This would broaden the system's applicability to a wider range of scenarios.
- Multi-language support: In a globalized world, extending the system's capabilities to recognize license plates in multiple languages would cater to the needs of multinational urban environments.

Author Contributions S.K. and D.K. conducted the experiments and implemented the proposed technique. S.K. wrote the main manuscript text and prepared figures. All authors reviewed the manuscript.

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare that they have no Conflict of interest.

References

1. Chollet, F.: Xception: Deep Learning with Depthwise Separable Convolutions (2017)
2. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference On Computer Vision And Pattern Recognition (CVPR'05). **1**(1), 886–893 (2005)
3. Fahd, S., Khan, K., Shah, Y., Shahzad, M., Khan, U.: Towards automatic license plate recognition in challenging conditions. Appl. Sci. **13**, 3956 (2023)
4. Fan, X., Zhao, W.: Improving robustness of license plates automatic recognition in natural scenes. IEEE Trans. Intell. Transp. Syst. **23**, 18845–18854 (2022)
5. Faradji, F., Rezaie, A., Ziaratban, M.: A Morphological-Based License Plate Location. In: 2007 IEEE International Conference on Image Processing. **1**(I), 57–60 (2007)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition (2015)
7. Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Efficient Convolutional Neural Networks for Mobile Vision Applications, MobileNets (2017)
8. Hsu, G., Chen, J., Chung, Y.: Application-oriented license plate recognition. IEEE Trans. Veh. Technol. **62**, 552–561 (2012)
9. Huang, G., Liu, Z., Maaten, L., Weinberger, K.: Densely Connected Convolutional Networks (2018)
10. Huang, Q., Cai, Z., Lan, T.: A single neural network for mixed style license plate detection and recognition. IEEE Access **9**, 21777–85 (2021)
11. Huynh, A., Tan, L., Huynh, K.: License Plate Detection and Recognition using CRAFT and LSTM (2022)
12. Kedia, D., Khokhar, S.: Enhanced licence plate detection using YOLO framework in challenging environments. Int. J. Comput. Vis. Robot. (2024)
13. Khokhar, S., Dahiya, P.: Character Recognition for ALPR Systems: A New Perspective (2020)
14. Khokhar, S., Kedia, D., Dahiya, P.: License Plate Detection Techniques: Conventional Methods to Deep Learning. In: ICT with Intelligent Applications. pp. 729–734 (2023)
15. Li, H., Wang, P., Shen, C.: Toward end-to-end car license plate detection and recognition with deep neural networks. IEEE Trans. Intell. Transp. Syst. **20**, 1126–1136 (2019)
16. Liu, W., Lin, C.: A hierarchical license plate recognition system using supervised K-means and Support Vector Machine. In: 2017 International Conference on Applied System Innovation (ICASI). pp. 1622–1625 (2017)
17. Muhammad, J., Altun, H.: Improved license plate detection using HOG-based features and genetic algorithm. In: 2016 24th Signal Processing And Communication Application Conference (SIU). pp. 1269–1272 (2016)
18. Pattanaik, A., Balabantaray, R.: Enhancement of license plate recognition performance using Xception with Mish activation function. Multimed. Tools Appl. **82**, 16793–16815 (2023). https://doi.org/10.1007/s11042-022-13922-9
19. Pustokhina, I., Pustokhin, D., Rodrigues, J., Gupta, D., Khanna, A., Shankar, K., Seo, C., Joshi, G.: Automatic vehicle license plate recognition using optimal K-means with convolutional neural network for intelligent transportation systems. IEEE Access **8**, 92907–92917 (2020)
20. Rattanawong, S., Hsu, G., Chung, S.: Thailand License Plate Detection and Recognition. In: 2021 25th International Computer Science and Engineering Conference (ICSEC). pp. 116–121 (2021)

21. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision And Pattern Recognition. pp. 779–788 (2016)
22. Selmi, Z., Ben Halima, M., Alimi, A.: Deep Learning System for Automatic License Plate Detection and Recognition (2017)
23. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition (2015)
24. Slimani, I., Zaarane, A., Okaishi, W., Issam, A., Hamdoun, A.: An automated license plate detection and recognition system based on wavelet decomposition and CNN. Array **8**, 100040 (2020)
25. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going Deeper with Convolutions (2014)
26. Tan, M., Le, Q.: Rethinking Model Scaling for Convolutional Neural Networks, EfficientNet (2020)
27. Tung, C., Wang, C., Peng, B.: A deep learning model of dual-stage license plate recognition applicable to the data processing industry. Math. Probl. Eng. **2021**, 3723715 (2021)
28. Ultralytics YOLOv8 - docs.ultralytics.com. https://docs.ultralytics.com/models/yolov8/,0. Accessed 08 Apr 2024
29. Yu, H., Wang, X., Shao, Y., Qin, F., Chen, B., Gong, S.: Research on license plate location and recognition in complex environment. J. Real-Time Image Process. **19**, 823–837 (2022)
30. Zoph, B., Le, Q.: Neural Architecture Search with Reinforcement Learning (2017)
31. Zou, Y., Zhang, Y., Yan, J., Jiang, X., Huang, T., Fan, H., Cui, Z.: License plate detection and recognition based on YOLOv3 and ILPRNET. Signal, Image Video Process. **16**(2), 473–80 (2022)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.