SINGLE SHOT DETECTOR-EFFICIENTDET (SSD-ED) MODEL FOR REAL-TIME MALAYSIAN NUMBER PLATE DETECTION AND RECOGNITION

A.A. Sufian Chan[1,2], S.M. Mustam[2*], F.C. Seman[2], M.F.L. Abdullah[2], F.A. Po'ad[2], A. Joret[2] and H. Safdar[3]

[1]Penang - Robert Bosch Malaysia Sdn. Bhd.,
Bayan Lepas Free Industrial Zone Phase 1, 11900 Bayan Lepas, Pulau Pinang.

[2]Faculty of Electrical and Electronic Engineering,
Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Johor, Malaysia.

[3]Department of Electrical Engineering,
Federal Urdu University of Arts Science and Technology, Islamabad 44000, Pakistan.

*Corresponding Author's Email: saizal@uthm.edu.my

Article History: Received 16 December 2023; Revised 2 Jun 2024; Accepted 16 July 2024

ABSTRACT: Real-time number plate (NP) detection and recognition play a crucial role in intelligent transportation systems, enabling automated toll collection, smart parking systems, and traffic management. Despite advances in deep learning (DL) frameworks, various challenges persist in achieving robust performance across different scenarios, such as variations in languages, font types, colors, formatting regulations, illumination level, occlusion, display angle, and weather. This paper aims to propose a reliable and efficient DL framework model for real-time Malaysian NP detection and recognition. The single-shot detector-efficientdet (SSD-ED) DL-based model and the YouTube Malaysian NP dataset are used for NP detection and recognition. All SSD-ED variants are evaluated for their performance in NP detection and alphanumerical character recognition tasks. The results demonstrate the superiority of SSD-ED7 in accuracy, with 94.6% for NP detection and 93.4% for character recognition. However, it has longer processing times than other variants. SSD-ED4, on the other hand, shows balanced performance and speed, with accuracy rates of 91.6% and 90.6% for NP detection and character recognition, respectively. The higher the accuracy, the longer the processing time, making it less suitable for fast-response applications and more appropriate for accuracy-driven ones. Therefore, the SSD-ED4 model is well-suited for real-time applications, providing efficient and accurate NP detection and character recognition.

KEYWORDS: *Deep Learning; SSD-ED Model; Number Plate Detection; Alphanumerical Character Recognition; Automotive*

1.0 INTRODUCTION

An intelligent transport system (ITS) has revolutionized transportation sectors by merging advanced technologies for efficient operations. One significant application in ITS is number plate (NP) detection and character recognition, a technology that can automate the process of identifying NPs. The system integrates cutting-edge technology to offer a seamless and effective solution that enhances overall transportation efficiency and security, especially in law enforcement, toll collection, and parking management [1-2].

One of the key reasons why ITS has increased in popularity is the integration and establishment of deep learning (DL) technology [1-3]. Today, DL technology has become significantly robust and powerful, allowing it to become the industrial standard for various industries and applications. For example, DL based on a recurrent neural network can achieve an accuracy of 97.05% and 96.58% in the prediction of flank wear and surface roughness in manufacturing processes [4]. Therefore, by harnessing the power of DL technology [1] with computer vision, a robust system capable of detecting and recognizing NPs can be developed. The system will have the power to identify NPs in the blink of an eye, which can significantly improve the efficiency of toll and parking management. Furthermore, the system will aid in law enforcement, such as recognizing stolen vehicles and capturing traffic law offenders [2, 5].

Even though DL technology has established a strong foundation for NP detection and reorganization [2, 3, 5], it remains difficult and encounters various challenges. The challenges differ based on the country and the variability of the environment [2, 3, 5]. The first issue this system faces is the variation caused by countries. Each country may have a different numbering system and language for their NP. In addition, the countries might also dictate the use of different color plates, font types and sizes, and special characters, which might be difficult to comply with. This is especially true in Malaysia, where the NP is difficult to standardize. Many Malaysian owners have altered the NP fonts, sizes, and scales. As a result, NPs are more difficult to detect and recognize.

In general, NP detection and character recognition relied on traditional approaches [2], like color, haar, histogram of gradient (HOG), supervised machine learning (SVM), and edge detection [2, 5-9], for feature extraction. However, with the advent of DL frameworks like mask region-based convolutional neural network (Mask R-CNN) [10] and you only look once (YOLO) [6, 11, 12], DL has shown superior performance compared to traditional methods. Recent studies have demonstrated that DL-based methods outperform traditional ones in NP detection and recognition [6, 10-15, 17, 18].

In [14], the researchers introduced an NP detection algorithm that blends traditional methods with DL. Color segmentation, corner detection, and morphological operations were used to isolate the area of interest, reducing the impact of complex backgrounds. A convolutional neural network (CNN) model is then used to extract features and determine the NP region. This hybrid approach enhances the algorithm's performance and robustness for accurate NP detection, even in challenging backgrounds. Moreover, an NP detection and recognition system using Mask R-CNN was developed by Selmi et al. [10]. This system excels at recognizing NPs across different orientations and complex backgrounds. It employs a three-stage approach involving detection, recognition, and segmentation, enabling character recognition within the NP. The system was rigorously evaluated on a challenging database containing 610 Tunisian NPs under varying conditions, including orientation, weather, and complex backgrounds.

Furthermore, Jamtsho et al. [15] have proposed a real-time NP detection system for traffic law enforcement applications that recognizes non-helmeted motorcyclists using YOLO9000. The system operates by discerning motorcyclists without a helmet and detecting their NP. The YOLO9000 model introduced in [16] was trained to identify NPs on both helmeted and non-helmeted motorcyclists for better detection tasks. In addition to real-time NP detection, Silva et al. [17] have expanded the idea for both real-time NP detection and recognition tasks. The researchers presented an end-to-end automatic NP recognition method based on a hierarchical CNN. The main idea was to identify the vehicle and the NP using two passes on the same

CNN network. The first pass recognizes the vehicle and detects its NP. The second pass recognizes the characters on the NP [17]. In solving the issue of limited training data, the researchers have explored the use of synthetic and augmented data, which showed significant increases in the recognition rate. Furthermore, the work introduces a novel temporal coherence technique to stabilize the optical character recognition (OCR) output in video format [17].

Pham [18] has proposed a lightweight and effective deep convolutional neural network to tackle the NP detection and recognition issue. The proposed network differs from conventional networks as it does not use max-pooling modules. Instead, it consists of alternating convolutional layers and inception residual networks and uses different techniques to recognize the characters. It has achieved outstanding results in two public datasets, which are the Chinese city parking dataset (CCPD) [19] and the application-oriented license plate (AOLP) [20]. In general, the proposed network is designed to detect and recognize NPs at real-time speed despite using low-spec machines.

In summary, the review of the previous related works demonstrates the robustness and effectiveness of DL frameworks for detecting and recognizing NP in an unconstrained environment. The review also shows the limited study of NP detection and recognition in Malaysia, which might be attributed to two main factors. The first is a lack of local datasets to support the development of DL frameworks. It is undeniable that there are numerous NP datasets available [5, 8, 19-21], but the foreign dataset has significant differences from the local dataset. The differences include variations in languages, font types, colors, and formatting regulations. For example, the China NP in the CCPD [19] dataset uses different languages, color plates, and formatting compared to the local NP. This poses a challenge, as a model trained on a foreign dataset might not be able to accurately detect and recognize local NP. The next key issue is the lack of legal reinforcement for the standardization of Malaysian NPs. This leads to a wide array of customizations of the NP that include font types, sizes, and formatting regulations. In addition, the presence of commemorative NPs further complicates the current situation. This huge diversity poses a significant challenge in ensuring accurate detection and recognition of local NPs.

Furthermore, the unpredictability of the unconstrained environment of the real-world scenarios has posed a significant threat to the system's performance. The variability of the real-world environment, such as illumination level, occlusion, display angle, and weather conditions, can also severely impact the system's performance [2, 3, 5, 6]. In addition, image quality and resolution can severely impact system performance as well. However, despite these challenges, DL approaches still show remarkable potential in NP detection and recognition [6, 10-15, 17, 18].

Due to variations in NP, such as letters, size of letter, font, letter spacing, plate color, and others, a new model for Malaysian NP detection and recognition must be developed. Therefore, this paper proposes a reliable and efficient DL-based framework model for real-time NP detection and recognition, despite the challenges posed by NP variability and environmental challenges. The framework is developed for Malaysian NP detection and alphanumerical character recognition in an unconstrained environment, as well as for stationary and slow-moving cars. The integration of a single-shot detector-efficientdet (SSD-ED) DL-based framework model is considered for a robust and accurate DL architecture. The significance of this study extends beyond academic interest, as it has potential implications for various sectors reliant on efficient Malaysian transportation management. By enhancing the robustness and reliability of NP detection and recognition systems, this study can bolster the effectiveness of law enforcement, toll collection, and parking management, ultimately contributing to improved safety and efficiency within the transportation infrastructure in Malaysia.

2.0 METHODOLOGY

This section discusses the development of a new DL-based framework model using the SSD-ED to automatically detect Malaysian NP in real-time, recognize the alphanumerical characters, and sort them in the right order as printed on the car's NP. To cater to the issues of NP variability and environmental challenges, the acquisition of the YouTube Malaysian NP dataset is also discussed in this section.

The models were developed in Python by utilizing two DL libraries: TensorFlow and Keras. The experiments were conducted on hardware equipped with 32GB of RAM and a Nvidia RTX 3060 GPU. The evaluation process consists of two aspects. First, it assesses the NP module's detection capability for identifying NPs in unconstrained environments. The second aspect involves measuring the performance of alphanumeric character recognition.

The model's performance was evaluated in terms of accuracy and speed. Accuracy was calculated by using

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

where *TP* refers to true positive, *TN* is true negative, *FP* is false positive, and *FN* represents false negative. Meanwhile, the average processing time is measured to gauge how fast the model can detect an NP and recognize its characters. Furthermore, a comparative analysis was conducted with various EfficientDet model variations [22], spanning from D0 to D7. The assessment of different backbones in the proposed network provides insights into the impact of parameter count on system accuracy and the time required for NP detection and character recognition.

2.1 The SSD-ED DL-Based Model

This study introduces a new DL-based framework model for real-time Malaysian NP detection and recognition, employing a three-stage framework for the efficient detection and recognition of NPs and their corresponding alphanumeric values. Figure 1 illustrates the proposed framework's block diagram, which encompasses three key components: NP detection, alphanumerical recognition, and a sorting mechanism. The workflow initiates with image acquisition, followed by preprocessing to enhance data quality. Subsequently, the first stage focuses on NP detection within the image. Upon successful detection, the subsequent step involves alphanumerical recognition, where individual characters are identified. However, the recognized characters may not be in the correct order, prompting the sorting mechanism to rearrange them and generate an accurate reading of the NP.

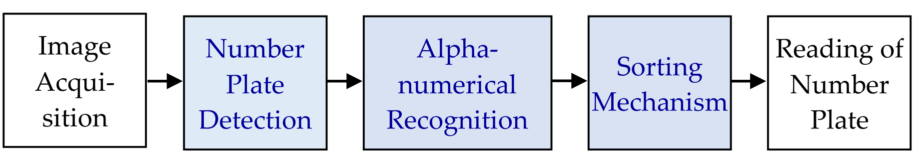

Figure 1: A block diagram of the three-stage DL-based framework model

Moving onto the development of NP detection and alphanumerical recognition using the proposed framework model. The DL architecture was used to detect and recognize NPs primarily due to its robustness in handling unconstrained environments. Both detection and recognition tasks used the same DL architecture but were trained separately using different sets of the YouTube Malaysian NP dataset. The DL architecture used EfficientDet [22] as the backbone, while single-shot detector (SSD) [23] was used as the detection network. By leveraging these two different networks, a robust and accurate DL architecture can be created.

EfficientDet was chosen as the backbone for three reasons. Firstly, it is a state-of-the-art object detection framework with seven different variations ranging from D0 to D7 [22]. These variations have different parameter sizes and performances, with D7 having the largest parameters and the best performance. The second reason is that EfficientDet excels at detecting objects of various sizes and scales, which can be attributed to the bi-directional feature network (BiFPN) and compound scaling methods. This feature is crucial for detecting the NP and alphanumerical characters, as in unconstrained environments, these objects can be in various sizes and scales where normal convolution layers might fail to detect them. The final reason is because of its ability to maintain high performance despite having far fewer parameters compared to its competitors. For example, the D1 variation outperforms RetinaNet-R50 and RetinaNet-R101 [22], despite having significantly fewer parameters. Since it uses fewer parameters, it allows the model to have a faster inference time.

The integration of EfficientDet into the SSD network [23] involves the process of replacing the SSD's backbone with different variations of EfficientDet [22]. SSD networks use the anchor box technique to detect multiple objects, while non-maximum suppression (NMS) removes redundant and low-confidence detections. The proposed architecture is called SSD-ED. Figure 2 shows a visualization of the proposed model architecture.

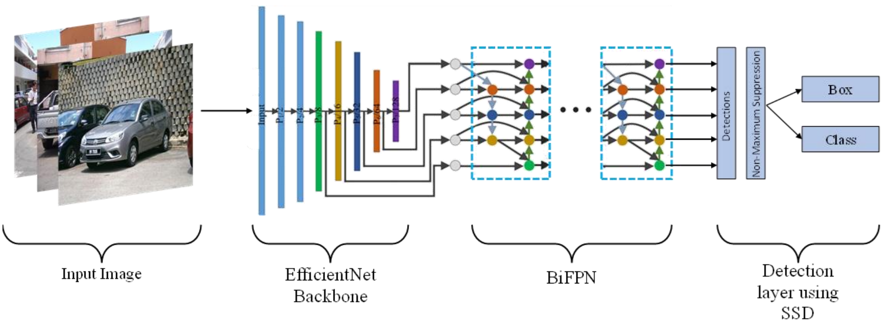

Figure 2: A flowchart of the proposed SSD-ED DL-based model architecture

The architecture was trained separately for two distinct tasks: NP detection and alphanumerical recognition, resulting in a dual-detector module with distinct responsibilities. The NP detection module was trained on the first set of the YouTube Malaysian NP dataset, comprising images annotated with NPs. Meanwhile, the alphanumerical recognition module was trained using the second set of the YouTube Malaysian NP dataset, featuring cropped NP images with alphanumerical annotations. Another critical component is the sorting mechanism, which leverages the center coordinates of detected alphanumerical bounding boxes to reorganize them on the x-axis, ultimately generating the accurate sequence of the NP value.

Figure 3 illustrates the novel framework's flowchart, which marks a significant advancement in NP recognition technology. The process begins with preprocessing, standardizing image dimensions for enhanced performance. Subsequently, the innovative NP detector module scans the image for NPs, while the alphanumeric detector module recognizes characters within cropped plate regions. Notably, the integration of EfficientDet into the SSD network stands out as a key innovation, enabling robust detection and recognition capabilities. Moreover, the dual-task training approach for NP detection and alphanumeric recognition contributes to the framework's versatility and accuracy. Lastly, the sorting mechanism ensures the correct sequencing of recognized characters, culminating in the precise extraction of NP values. This streamlined framework promises enhanced efficiency and accuracy in NP recognition tasks.

2.2 Acquisition of Dataset

Based on the literature review, no local car NP dataset is publicly available. The lack of a publicly available dataset can pose a significant obstacle to developing NP recognition systems in the country. Hence, one of the contributions of this research is the introduction of the YouTube Malaysian NP dataset.

In general, the dataset consists of two sets. The first set focuses on the presence of NP, while the second set focuses on the alphanumerical characters that exist in Malaysian NPs. The alphanumeric characters can be divided into three categories: alphabets, numbers, and special characters. The alphabet consists of capital letters from A to Z, except for "I" and "O" due to their similarities to the numbers "1" and "0". The number consists of 0 to 9, while the special character refers to "-", which is used on diplomatic NPs. Malaysia has a huge collection of commemorative NPs, and some of the examples are "XIIINAM", "BAMbee", and "Malaysia". However, this dataset will exclude the special characters and commemorative plates, as they are quite rare to be seen in public. The alphanumeric characters consist of 34 classes.

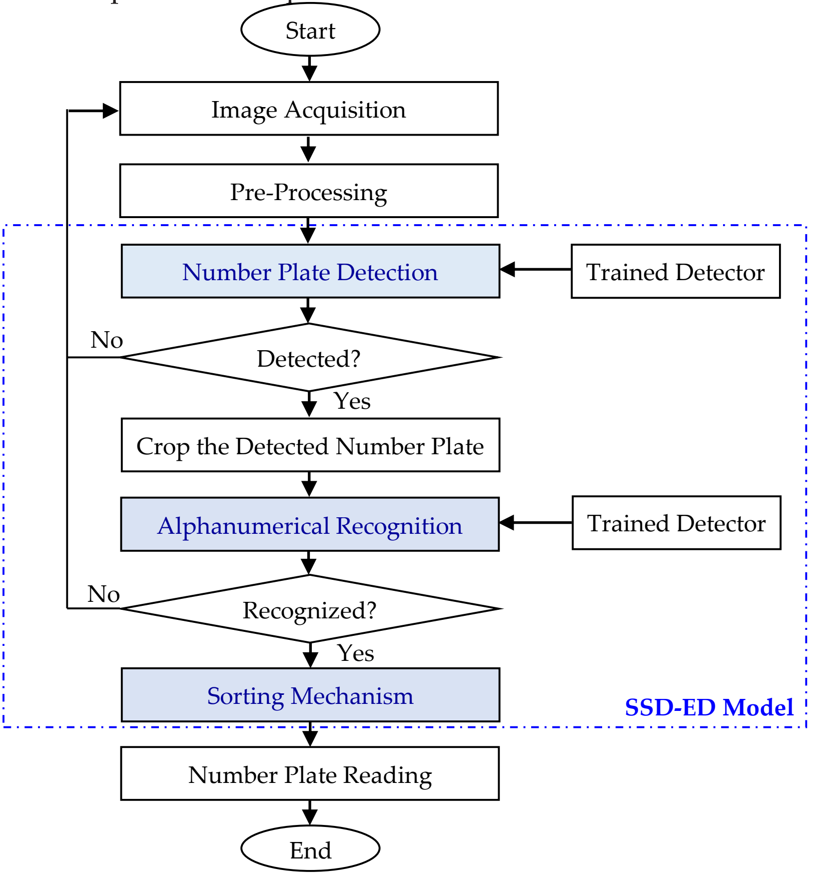

Figure 3: A flowchart of the proposed framework steps

In addition, the dataset was compiled from two primary sources. The first source is the images on various platforms on the Internet, such as Google Image, Flickr, and Wikimedia Commons. The second source is based on YouTube videos featuring cars in different regions of Malaysia. Most images are from the second source, as extracting images from YouTube videos is more straightforward. To ensure diversity in the dataset, similar images were systematically removed. A total of 3333 images were collected, and the images were taken in unconstrained environments.

Furthermore, the dataset annotation process was then implemented. It is divided into two parts. The first part focuses on annotating the car's

NP within the image, while the second part involves annotating the alphanumeric characters present on NPs. This annotation process is meticulous and time-consuming to ensure the accuracy of both the labels and the bounding boxes. A sample of annotated car plates is shown in Figure 4(a), while Figure 4(b) displays a sample of annotated alphanumeric characters. To evaluate the model's performance, each set was split into training and testing sets. The training set contains 80% of the dataset's images, while the testing set contains the remaining 20%. This helps to train the proposed models while at the same time providing the necessary data to evaluate the models' performance in detecting and recognizing the NP on a new, unseen image.

(a) (b)

Figure 4: Sample images featuring (a) NPs set and (b) annotated characters set

3.0 Result and Analysis

This section presents the model performance evaluation results and analysis of the proposed framework. First, it assesses the NP module's detection capability for identifying NPs in unconstrained environments. The second involves measuring the performance of alphanumeric character recognition.

Table 1 shows the NP detection results for different variations of SSD-ED, while Figure 5(a) depicts sample images of the detected NP. SSD-ED7 has the highest accuracy at 94.6% but requires the longest detection time. SSD-ED7's superior accuracy is attributed to its considerably larger parameter count, exceeding 52 million, which enhances detection precision but has a slower inference time due to increased complexity. Meanwhile, SSD-ED0 has the fastest detection time but has the lowest accuracy of 86.2%. SSD-ED4 strikes a balance with 91.6% accuracy and an average detection time of 152.95 ms. SSD-ED4 is ideal for real-time applications, offering upgraded accuracy from SSD-ED0 while maintaining fast detection.

Table 1: Performance evaluation of SSD-ED model variations for NP detection

Model	Accuracy (%)	Avg time taken (ms)
SSD-ED0	86.2	44.85
SSD-ED1	88.3	62.10
SSD-ED2	89.7	77.05
SSD-ED3	90.4	109.25
SSD-ED4	91.6	152.95
SSD-ED5	92.5	255.30
SSD-ED6	93.3	308.20
SSD-ED7	94.6	373.75

Table 2 displays the detailed results for alphanumerical recognition per character, whereas Figure 5(b) displays examples of recognized alphanumerical characters on NPs. A comparable comparative analysis was carried out with several EfficientDet model variations. The SSD-ED7 has the highest accuracy (93.4%), but it is also the slowest, taking roughly 399.75 ms, which may not be suitable for real-time applications where speed is important. Meanwhile, SSD-ED0 offers the quickest time average (47.97 ms) but has the lowest accuracy (83.4%). SSD-ED4 achieves 90.6% accuracy with an average duration of 163.59 ms. As a result, the SSD-ED4 is a great option for real-time applications as it provides good accuracy with fast recognition speed. The proposed model's recognition accuracy of SSD-ED3 to SSD-ED4 outperformed the range of accuracies in [24], which was 90.4%.

Table 2: The performance evaluation of alphanumerical recognition for different SSD-ED model variation

Model	Accuracy (%)	Avg time taken (ms)
SSD-ED0	83.4	47.97
SSD-ED1	85.6	66.42
SSD-ED2	87.3	82.41
SSD-ED3	89.4	116.85
SSD-ED4	90.6	163.59
SSD-ED5	91.8	273.06
SSD-ED6	92.4	329.64
SSD-ED7	93.4	399.75

(a)

(b)

Figure 5: Samples of (a) detected NP and (b) recognized characters

4.0 Conclusion

In conclusion, this study proposed a novel DL architecture that combines the SSD network with the EfficientDet network to detect and recognize Malaysian NP efficiently. Additionally, the YouTube Malaysian NP dataset was also introduced to provide the necessary data to effectively train the model for detecting and recognizing Malaysian NP in real-world environments. The DL model was trained using this dataset to address the challenges associated with detecting and recognizing Malaysian NPs, where it demonstrates robustness in handling various detection and recognition scenarios. SSD-ED7 managed to achieve the highest accuracy, but it is ideal for accuracy-related applications in border security or forensics due to its long processing time limit. However, the SSD-ED4 provided balanced performance and speed, making it well-suited for real-time applications like toll collection, where quick processing is essential. Hence, as a conclusion, the proposed three-stage framework of the SSD-ED model has addressed the research gap by enhancing the detection and recognition of Malaysian NPs.

ACKNOWLEDGMENTS

This research was supported by Universiti Tun Hussein Onn Malaysia (UTHM) through Tier 1 (Vot Q336) and RE-GG (Vot H884).

AUTHOR CONTRIBUTIONS

A.A. Sufian Chan and S.M Mustam: Conceptualization, Methodology, Coding Development and Simulation, Writing-Original Draft Preparation, Revision and Editing; M.F.L. Abdullah: Data Curation, Validation, Supervision; F.C. Seman, F.A. Po'ad and A. Joret: Coding Verification, Validation, Writing-Reviewing and Editing; H. Safdar: Revision, Writing-Reviewing and Editing.

CONFLICTS OF INTEREST

The manuscript has not been published elsewhere and is not under consideration by other journals. All authors have approved the review, agree with its submission, and declare no conflict of interest on the manuscript.

REFERENCES

[1] Y.Y. Lee, Z. Abdul Halim and M.N. Ab Wahab, "License plate detection using convolutional neural network–back to the basic with design of experiments," *IEEE Access*, vol. 10, pp. 22577-22585, 2022.

[2] J. Tang, L. Wan, J. Schooling, P. Zhao, J. Chen and S. Wei, "Automatic number plate recognition (ANPR) in smart cities: a systematic review on technological advancements and application cases," *Cities*, vol. 129, pp. 1-18, 2022.

[3] M.M. Khan, M.U. Ilyas, I.R. Khan, S.M. Alshomrani and S. Rahardja, "License plate recognition methods employing neural networks," *IEEE Access*, vol. 11, pp. 73613–73646, 2023.

[4] W.K. Lee, M.D. Abdullah, P. Ong, H. Abdullah and W.K. Teo, "Prediction of flank wear and surface roughness by recurrent neural network in turning process," *Journal of Advanced Manufacturing Technology (JAMT)*, vol. 15, no. 1, pp. 55–67, 2021.

[5] R. Zibani, F. Sebbak, M.E.Y. Boudaren, M. Mataoui, R.H. Aissa and Y.A. Benaissa, "Multi-attribute fusion-based approach for Algerian automatic license plate recognition," *Multimedia Tools Applications*, vol. 83, pp. 30233–30259, 2024.

[6] S. Arora, C. Prakash, P.S. Rana and K. Sood, "A comparative study of object detection techniques for automatic license plate recognition for Indian license plates," in 2023 5th International Conference on Energy, Power, and Environment: Towards Flexible Green Energy Technologies (ICEPE), Shillong, India, 2023, pp. 1-6.

[7] Z. Mahmood, K. Khan, U. Khan, S.H. Adil, S.S.A. Ali and M. Shahzad, "Towards automatic license plate detection," *Sensors*, vol. 22, no. 3, pp. 1245, 2022.

[8] S. Alghyaline, "Real-time Jordanian license plate recognition using deep learning," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, pp. 2601–2609, 2022.

[9] M.M. Khan, M.U. Ilyas, I.R. Khan, S.M. Alshomrani and S. Rahardja, "License plate recognition methods employing neural networks," *IEEE Access*, vol. 11, pp. 73613–73646, 2023.

[10] Z. Selmi, M.B. Halima, U. Pal and M.A. Alimi, "Delp-DAR system for license plate detection and recognition," *Pattern Recognition Letters*, vol. 129, pp. 213–223, 2020.

[11] Y. Cao, "Investigation of a convolutional neural network-based approach for license plate detection," *Journal of Optics (India)*, vol. 53, no. 1, pp. 697–703, 2024.

[12] H. Shi and D. Zhao, "License Plate Recognition System Based on Improved YOLOv5 and GRU," *IEEE Access*, vol. 11, pp. 10429–10439, 2023.

[13] X. Zhou, Y. Cheng, L. Jiang, B. Ning and Y. Wang, "FAFEnet: A fast and accurate model for automatic license plate detection and recognition,"

IET Image Processing, vol. 17, no. 3, pp. 807–818, 2022.

[14] X. Wu, J. Qiu and A. Qiu, "An efficient license plate location algorithm based on deep learning," in 2020 International Conference on Computer Engineering and Application (ICCEA), Guangzhou, China, 2020, pp. 543-546.

[15] Y. Jamtsho, P. Riyamongkol and R. Waranusast, "Real-time license plate detection for non-helmeted motorcyclist using Yolo," *ICT Express*, vol. 7, no. 1, pp. 104–109, 2021.

[16] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 6517-6525.

[17] S.M. Silva and C.R. Jung, "Real-time license plate detection and recognition using deep convolutional neural networks," *Journal of Visual Communication and Image Representation*, vol. 71, pp. 1-9, 2020.

[18] T.A. Pham, "Effective deep neural networks for license plate detection and recognition," *The Visual Computer*, vol. 39, no. 3, pp. 927–941, 2023.

[19] Z. Xu, W. Yang, A. Meng, N. Lu, H. Huang, C. Ying and L. Huang, "Towards end-to-end license plate detection and recognition: A large dataset and Baseline," in *Computer Vision - ECCV 2018*, *Lecture Notes in Computer Science*, vol. 11217, V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss, Eds., Springer, Cham, 2018, pp. 261–277.

[20] G.S. Hsu, J.C. Chen and Y.Z. Chung, "Application-oriented license plate recognition," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 2, pp. 552–561, 2013.

[21] A. Pattanaik and R.C. Balabantaray, "Enhancement of license plate recognition performance using Xception with Mish activation function," *Multimedia Tools Applications*, vol. 82, no. 11, pp. 16793–16815, 2023

[22] M. Tan, R. Pang and Q.V. Le, "EfficientDet: scalable and efficient object detection," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 10778-10787.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu and A.C. Berg "SSD: single shot multibox detector," in *Computer Vision - ECCV 2016*, *Lecture Notes in Computer Science*, vol. 9905, B. Leibe, J. Matas, N. Sebe and M. Welling, Eds., Springer, Cham, 2016, pp. 21–37.

[24] R. Al-batat, A. Angelopoulou, S. Premkumar, J. Hemanth and E. Kapetanios, "An end-to-end automated license plate recognition system using YOLO based vehicle and license plate detection with vehicle classification," *Sensors*, vol. 22, no. 23, pp. 9477, 2022.