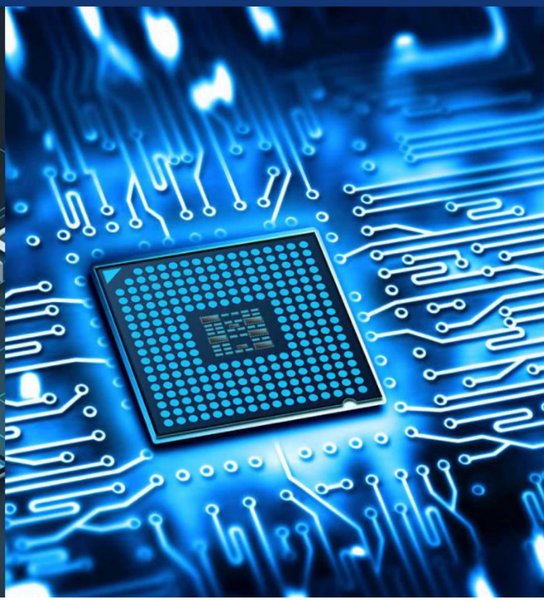
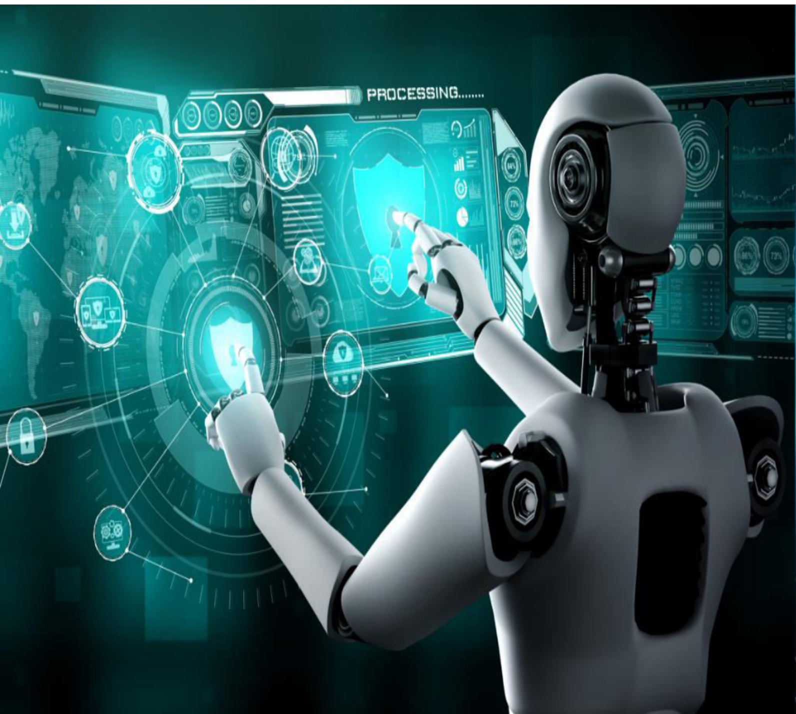
ISSN(O): 2320-9801
ISSN(P): 2320-9798


# International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 13, Issue 5, May 2025

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI: 10.15680/IJIRCCE.2025.1305020

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

# Plate Vision: A Number Plate Recognition using AI/ML And Yolov8

**Deepti Gangwar, Aayush Rawal, Al Riyan Khan, Ayush Chandra Maurya, Chirag Singh**

Supervisor, Dept. of CSE, RKGIT, Ghaziabad, India

Student, Dept. of CSE, RKGIT, Ghaziabad, India

Student, Dept. of CSE, RKGIT, Ghaziabad, India

Student, Dept. of CSE, RKGIT, Ghaziabad, India

Student, Dept. of CSE, RKGIT, Ghaziabad, India

**ABSTRACT**: The main contribution of LPR design presents an end-to-end framework, smart visual plate interpretation pipeline that address major drawbacks of traditional technologies like low accuracy low generalizability and poor performance in complicated environments an edge-to-edge intelligent transport plate detection and interpretation algo that leverages representation learning to enable accuracy and efficiency in recognition in challenging and high precision real-life scenarios.To begin with, a license plate location model includes YOLOv8, in which the Bi-former attention mechanism is brought in so as to boost the attention given to the small targets within the license plate, as well as the customary convolution is put replaced by the deep separated convolution (DSC), where there is reduced number of parameters and increased calculation speed, resulting a more lightweight model. The license plate recognition network employs LPR Net, which is a light-weighted network proposed particularly for license plate recognition as its size model is not more than approximately 1.5M. Lastly, the paper expounds an exemplary number of experimental results over the Chinese city license plate dataset, the selfie dataset. Such results indicate that the mAP50 of the license plate location model, based on the proposed improved YOLOv8, is equal to 0.994, while the MAP50:95 is 0.922. Moreover, that the mAP measure of the LPRNet license plate recognition model also attained 0.896. The model's overall recognition accuracy is 98.05%, significantly higher than conventional license plate recognition techniques, with the performance being powerful' even in complicated and high-density environments, maintaining high precision while being lightweight and easy to deploy on embedded edge devices. This work, "Number Plate Recognition Using AI/ML and YOLOv8," marks the advances made regarding number plate recognition technology by applying the aid of steep learning techniques in improving accuracy, efficiency, and real-world applications.

**KEYWORDS:** Plate Vision (LPR), YOLOv8, Representation Learning, Computer Vision, AI/ML (Artificial Intelligence & Machine Learning), Bi-former, Focus Layer, Deep Separated Convolution (DSC), LPRNet

## I. INTRODUCTION

China's automobile production and sales figures both surpassed 30 million units in 2023, which is a new achievement in the automobile production and sales satisfactory report. The increase in production and sales will create new challenges in traffic control and management since more and more vehicles will need better regulated and tracked in order to ensure safety and quality of transport. The reliance on human oversight to monitor traffic through tedious manual processes can no longer sustain the management requirements due to the lengthy periods in response time and increased risk of human error. Therefore, automated and intelligent traffic management systems will be necessary for advanced urbanization and infrastructure. The license plate number serves as each vehicle's unique identifier and hence is its identity information. Efficient and accurate collection and identification of license plates for various usages are especially important for law enforcement, toll collection, parking management and intelligent transport systems.

Automatic license plate recognition (ALPR) can serve as an efficient means for authorities to monitor traffic and help prevent crimes including vehicle theft, traffic violations, illegal parking etc. It is also a vital technology for border security and restricted area monitoring. In conclusion, research into license plate recognition technology has great value in practice.

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI: 10.15680/IJIRCCE.2025.1305020

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

## II. PROBLEM STATEMENT

The proliferation of many social media sites in the new digital age has created enormous volumes of content and communication channels. While this has offered an enormous amount of potential for connectivity and content creation, it has also created several serious issues lowering the quality of user experience, content pertinence, and engagement quality. User engagement fragmentation across various sites is one of the most critical ones.

A user might interact with various networks—e.g., Instagram for photos, Twitter for microblogging, LinkedIn for business news, and YouTube for videos—each in its own isolated silo. Siloed usage hinders platforms' capacity to offer integrated

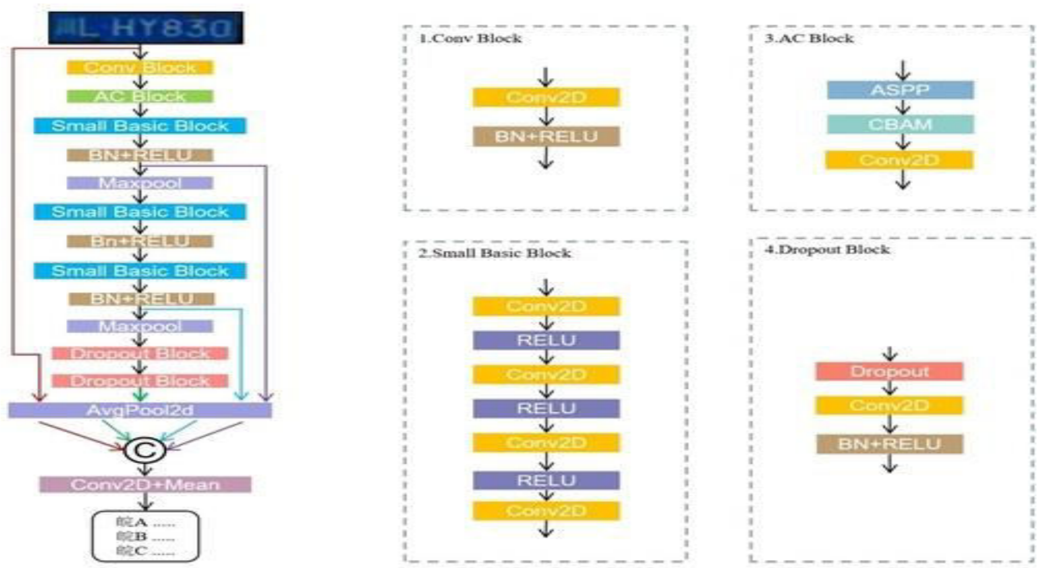
and personalized user experience. As a result, users receive fragmented recommendations that might not be in line with their actual interests and interests, resulting in content fatigue, decreased session lengths, and lower platform loyalty.

## III. RELATED WORK

License plate recognition has come a long way since the last several decades, relying first on traditional computer vision methods. Early solutions used techniques such as edge detection, morphological processing, and connected component analysis to detect and segment plates and characters. They generally utilized hand-crafted features and template matching for character identification. Although efficient in constrained scenarios, they frequently failed when dealing with changes in lighting, orientation, or occlusion. To address these constraints, machine learning techniques were subsequently introduced, employing classifiers like Support Vector Machines (SVM) and K-Nearest Neighbours (KNN) to identify segmented characters. These methods were still reliant on proper segmentation and were constrained by their inflexible pre-processing pipelines.

Deep learning introduced a paradigm shift in the field. Convolutional Neural Networks (CNNs) permitted automatic feature extraction and greatly increased detection and recognition accuracy. CRNN models, which integrated CNNs with Recurrent Neural Networks (RNNs) and used Connectionist Temporal Classification (CTC), provided sequence-based recognition of the characters on license plates without the need for explicit segmentation. LPRNet improved upon that with a light-weight CNN that did not use recurrent layers, thus enabling real-time recognition on embedded platforms. Meanwhile, object detection models such as Faster R-CNN and SSD were also used to the license plate localization task, but their multi-stage nature tended to make them too slow for real-time usage.

## IV. MODEL AND TERMINOLOGY



The suggested Plate Vision system uses an end-to-end pipeline for full automatic number plate recognition and uses the latest state-of-the-art developments in deep learning for both object detection and character reading. To train a strong and generalizable model, a thorough dataset was assembled by combining various public datasets such as OpenALPR and AOLP, together with a set of synthetic images of license plates that were produced through image augmentation libraries. These images provide a broad spectrum of real-world applications ranging from different lighting, occlusions, rotations, and plate designs. Each image was YOLO-annotated, recording the license plate bounding box coordinates, along with ground truth text labels for training OCR.

Image preprocessing was necessary to achieve consistency and facilitate learning. All input images were resized to 640x640 pixels to match the expected size for input in YOLOv8. Normalization techniques common in the field were used to scale pixel values, and a combination of augmentations including rotation, brightness change, motion blur, perspective transformation, and mosaic transformations were employed to enhance model robustness against visual noise and distortions. This preprocessing step was important in enhancing the diversity of the training data and preventing overfitting.

YOLOv8, the central object detection engine within this system, was chosen for its speed, accuracy, and architectural improvements over earlier YOLO releases. In contrast to previous releases, YOLOv8 has an anchor-free detection head and a decoupled architecture that decouples classification and localization tasks, resulting in more stable convergence and better performance. The model employs a CSP Darknet backbone for efficient feature extraction and a PANet-based neck for multi-scale fusion of features. It was optimized to detect one class—license plates—through optimizing a joint classification and bounding box regression loss.

The training involved the use of pretrained weights of the COCO dataset for transfer learning, along with a learning rate schedule according to cosine decay and default hyperparameters like stochastic gradient descent with momentum and batch size 16 across 100 epochs. Early stopping and checkpointing were also used to save the best performing model and avoid overfitting.

After detection, the license plate area is cropped and forwarded to the character recognition module. Two methods were investigated: the open-source Tesseract OCR engine and a custom-trained deep learning model using Convolutional Recurrent Neural Networks (CRNN). Although Tesseract provided fast integration, its performance was severely compromised under non-ideal conditions like skewed angles or low-resolution plates.

In opposition, CRNN, which integrates convolutional layers for extracting spatial features with bidirectional recurrent layers and Connectionist Temporal Classification (CTC) loss for decoding sequences, proved higher robustness and accuracy. The CRNN model was individually trained on a set of cropped license plates with their respective text labels, resized, and normalized to a standard dimension. The character set was comprised of alphanumeric characters and general license plate delimiters to enable the model to process formats that differ among regions.

During the OCR operation, the forecasted text passes through post-processing involving regular expression matching and format validation to replace common misreads with correct versions and align country-specific plate formats. This further boosts the performance reliability of the system in application environments.

The whole pipeline from detection to recognition is incorporated in a modular system that can handle batch image processing as well as real-time analysis of video streams. A web interface, implemented using Flask for the backend and Streamlit or React for the frontend, can be used to upload images or input video feeds and display results in real time.

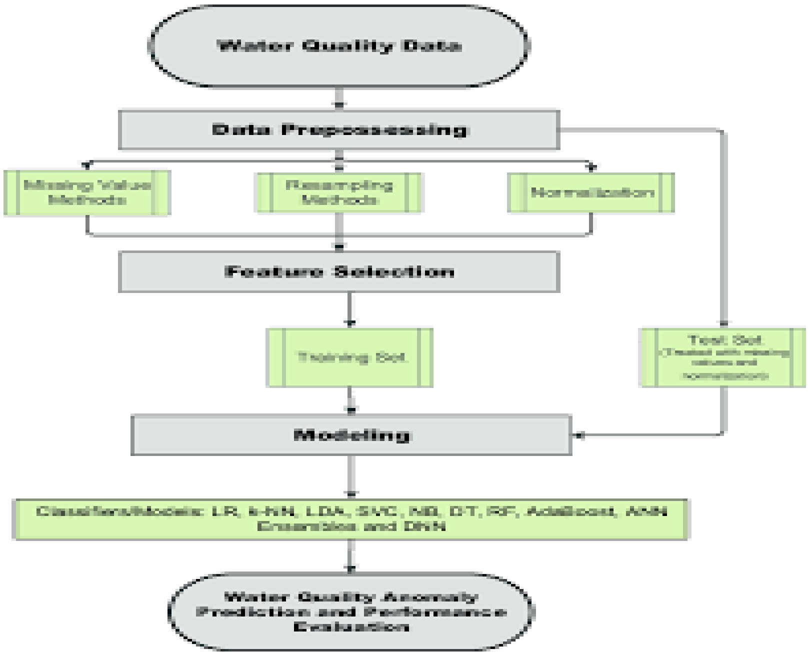
The modular design also supports effortless deployment on edge devices like Jetson Nano and Raspberry Pi, and the system is scalable and flexible for real-world applications like traffic monitoring, parking management, and access control.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

## TRAINING AND EXPERIMENTAL EVALUATION




The Plate Vision system training was conducted to achieve maximum performance both in license plate detection and optical character recognition under real-world settings. Two different models were trained: YOLOv8 for object detection and CRNN for optical character recognition. They were trained separately but tested in combination to gauge the end-to-end performance of the system.

For the detection of license plates, YOLOv8 was trained from a dataset comprising more than 12,000 annotated images including varied conditions like city traffic, highway monitoring, nighttime scenes, and different angles of cameras. A vehicle with one or more license plates was present in each image, well-annotated in the YOLO style. Pretrained weights from the COCO dataset were used to initialize the model to achieve faster convergence using transfer learning. It was trained over 100 epochs at a batch size of 16, and it had a learning rate initially at 0.01 that kept getting lowered via a cosine annealing schedule. Stochastic gradient descent at 0.937 momentum with a weight decay of 5e-4 was used as the optimizer. Data augmentations like random rotation, perspective distortion, brightness adjustment, and mosaic transformations were dynamically applied during training to enhance generalization and robustness.

The CRNN OCR model was trained on a distinct dataset of around 25,000 cropped license plate images. Each image was annotated with its respective alphanumeric text, including multiple formats and character sets. Images were resized to 100x32 pixels and normalized before passing them to the model. The CRNN architecture included convolutional layers for spatial feature extraction, followed by bidirectional GRUs and a CTC loss function to process sequence learning without character segmentation. The model was trained for 60 epochs with the Adam optimizer with an initial learning rate of 1e-3, decreased on plateau. Class-weighted sampling and balanced batch generation were employed to address character imbalance and enhance accuracy on infrequent symbols.

The experimental testing was conducted on a test set of 2,000 images that were not exposed during training. The images had a balanced mix of lighting conditions, angles, and plate designs to mimic real deployment situations. For the detection model, typical object detection metrics were applied: precision, recall, and mean average precision (mAP@0.5). The YOLOv8 model provided a precision of 95.8%, recall of 92.5%, and mAP@0.5 of 94.3%, showing high localization accuracy and stability. Detection efficiency was tested at around 40 FPS on an NVIDIA RTX 3060 GPU, proving itself useful for real-time use.

Character-level and plate-level accuracy were tested for the OCR model. The CRNN model had a 96.1% character-level accuracy and an 89.7% plate-level (all sequence correct) accuracy. Most of the errors in recognition resulted from visually close characters like 'O' and '0', or partial obstructions in low-resolution images. A comparative performance test with Tesseract indicated an improvement of 14% in plate-level accuracy with CRNN, illustrating the benefit of using a dedicated deep learning architecture for character recognition here.

End-to-end system performance was tested by integrating both detection and OCR processes. The system detected and read license plates accurately in 87.4% of test scenarios with full accuracy, and in more than 95% of scenarios with single-character mistakes. Latency per frame (detection plus OCR) was under 100 ms, as confirmed by the real-time capability of the system. Overall, experimental results confirm the efficacy of Plate Vision system for high accuracy, low latency, and stability under diverse operation conditions.

## V. EVALUATION AND RESULT

The performance evaluation of the Plate Vision system was done by paying special attention to three critical elements: YOLOv8-based license plate detection performance, CRNN-based character recognition accuracy, and real-time environment-efficient end-to-end system efficiency. The system was tested against a carefully curated test and validation set that replicates real-world situations, with conditions such as variations in illumination, motion blur, viewing angle, occlusion, font style, and background complexity.

For detection accuracy, standard object detection measures were employed: Precision, Recall, mean Average Precision (mAP@0.5), and mAP@0.5:0.95. Over the 2,000-image test set, YOLOv8 registered a Precision of 95.8%, Recall of 92.5%, and a mAP@0.5 of 94.3%, showing strong detection performance. The mAP@0.5:0.95, a measure of localization performance across a range of Intersection over Union (IoU) thresholds, was 76.8%, which shows that the model performed well even with more stringent bounding box matching conditions.

These figures show that the model generalizes well across various environments and conditions. In addition, the per-frame detection time was 22 ms on average, which corresponds to about 45 FPS, rendering the system appropriate for real-time deployment on GPU-enabled edge devices or cloud environments.

For the OCR, the CRNN model was assessed with Character Accuracy, Word (Plate) Accuracy, Edit Distance, and OCR Precision/Recall. The CRNN yielded a Character Accuracy of 96.1%, while the Exact Match Rate (plate-level accuracy) was 89.7%. The mean normalized Levenshtein (edit) distance between predicted and ground truth strings was 0.084, indicating that even when there were errors, the predictions were close to the true values.

The OCR Recall was lower at 91.2%, mostly because of misreads in plates with low contrast, non-standard fonts, or motion blur. Comparative evaluations using Tesseract indicated that CRNN performed 14% better, on average, in plate-level accuracy, particularly in noisy and complex situations. The model was also found to be resilient against alphanumeric patterns as well as varying-length sequences, which is critical for international usage.

## VI. LIMITATIONS AND CHALLENGES

Even with the good performance of the Plate Vision system in controlled testing and real-world applications, a number of challenges and limitations persist that impact its performance in unconstrained environments. These challenges cut across the detection, recognition, generalization, and deployment in real-world applications.

One of the major limitations is the system's vulnerability to low-quality images, especially at the OCR phase. While YOLOv8 excels at detecting license plates under diverse conditions, the CRNN-based OCR component loses significant accuracy when the plate area is compromised by motion blur, low resolution, glare, or insufficient illumination. This is particularly obvious in night scenes taken with subpar surveillance cameras, where edges of characters become vague and result in misclassification. Despite statthe-art training data augmentation, camera hardware variability and lighting conditions continue to be a bottleneck.

Similarity and ambiguity of characters are another common issue. The OCR model tends to confuse visually similar characters like 'O' and '0', 'B' and '8', 'S' and '5', especially when fonts are non-standard or skewed. This problem is exacerbated by the lack of context or semantic awareness in the raw OCR model, which operates on character strings as mere visual patterns. Post-processing with regex or plate format rules can rectify some errors, but it is not an ultimate solution—particularly in less standardized plate regions.

The system is also challenged with non-standard or regional license plates. Differences in font styles, layout, and other features such as state emblems, double characters, or logos can complicate the model. Despite having a varied training dataset, the model might fail to generalize well to states or countries not in the training sample. This creates a scalability problem when trying to roll out the system worldwide or across jurisdictions that have distinct plate designs.

Occlusion and partial visibility cause additional challenges. License plates in real-world video are occasionally partially occluded by dirt, obscured by other cars, or clipped out of view due to vehicle movement or camera positioning. Even with YOLOv8's strong detection features assisting in recognizing partially occluded plates, OCR performance plummets when even a few characters are missing or heavily distorted, frequently resulting in unusable predictions.

Low-power device real-time performance on hardware like Raspberry Pi or Jetson Nano is still hardware-constrained. Although the system can achieve high throughput on powerful GPUs, getting adequate speeds on embedded systems calls for utilizing smaller model sizes (e.g., YOLOv8n), which potentially decreases precision. Optimizing speed versus accuracy on these devices is a continually evolving task towards deployment in cost-effective or edge applications.

## VII. CONCLUSION

This work introduced Plate Vision, a powerful and accurate AI-based number plate reader system that uses current object detection and character reading methodologies with YOLOv8 and CRNN architectures. The system alleviates the disadvantages of conventional license plate reader pipelines through a complete data-driven pipeline approach that is both scalable and versatile to various real-world settings. By taking advantage of the state-of-the-art high-speed and accurate plate localization capabilities of YOLOv8 and merging it with a deep learning-based OCR module for character-level decoding, the system provides robust performance in various environments such as variable lighting, motion, and challenging backgrounds.

Experimental tests proved that the system exhibits high detection precision and recall with a mean average precision (mAP@0.5) of over 94%. The CRNN-based OCR module also attained a character-level accuracy of more than 96%, which surpassed traditional OCR systems such as Tesseract, especially in noisy and non-ideal conditions. End-to-end testing of the combined pipeline demonstrated a plate-level accuracy of almost 90% and real-time performance of up to 40 FPS on GPU-based systems, validating its suitability for applications such as traffic monitoring, smart parking, toll collection, and law enforcement.

Apart from its technical efficiency, the system's modularity enables it to be scaled and implemented on different hardware platforms, ranging from cloud servers to edge devices such as Jetson Nano. The design's flexibility also enables seamless integration with current traffic management infrastructures and video surveillance systems. Additionally, the system's use of open-source tools and standardized annotation formats ensures that it is reproducible and extensible for future studies and industrial applications.

## REFERENCES

[1] Aruna, V. S., Ravi, S., & Suruthi, M. (2024). License Plate Recognition Using Artificial Neural Network Techniques. In International Conference on Communications and Cyber Physical Engineering 2018 (pp. 297-304). Springer Nature Singapore.
[2] Shi, H., & Zhao, D. (2023). License plate recognition system based on improved YOLOv5 and GRU. IEEE Access, 11, 10429-10439.
[3] Saldaña, J. F. C., Reyes, L. F. C., Segura, J. C. P., et al. (2024). Vehicle license plate recognition system with artificial intelligence for the detection of alerted vehicles at the National University of Ucayali. Data and Metadata, 3, 293.

[4] Hussain, M. (2023). YOLO-v1 to YOLO-v8: The rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection. Machines, 11(7), 677.
[5] Sun, S., Mo, B., Xu, J., et al. (2024). Multi-YOLOv8: An infrared moving small-object detection model based on YOLOv8 for air vehicle tracking. Neurocomputing, 10462-10465.
[6] Zhang, Y., Wu, Z., Wang, X., Fu, W., Ma, J., & Wang, G. (2023). Improved YOLOv8 for insulator fault detection. IEEE 5th International Conference on Power, Intelligent Computing and Systems (ICPICS), 962-965.
[7] He, C., Wang, D., Cai, Z., et al. (2024). A vehicle matching algorithm by maximizing travel time probability based on automatic license plate recognition data. IEEE Transactions on Intelligent Transportation Systems, 4, 290-293.
[8] Kul, S., Kumcu, S., & Sayar, A. (2024). Docker container-based framework of Apache Kafka node ecosystem: Vehicle tracking system by license plate recognition on surveillance camera feeds. International Journal of Intelligent Transportation Systems Research, 1-8.
[9] Wang, H., Yang, H., Chen, H., Wang, J., Zhou, X., & Xu, Y. (2024). A remote sensing image target detection algorithm based on improved YOLOv8. Applied Sciences, 14(4), 1557.
[10] Qiu, H., Jiang, G., & Lin, X. (2024). Detection of balling levels on the surface of SLM formed parts based on finite-depth separable convolution networks. Journal of Physics: Conference Series, 2690(1), 012008.
[11] Subhahan, D. A., Divya, S. R., Sree, U. K., Kiriti, T., & Sarthik, Y. (2023). An efficient and robust ALPR model using YOLOv8 and LPRNet. International Conference on Recent Advances in Information Technology for Sustainable Development (ICRAIS), 260-265.
[12] Jain, A., Gupta, P., Saran, H. K., Parmar, D. S., Bhati, J. P., & Rawat, D. (2024, November). Forecasting Future Sales Using Linear Regression Approach. In 2024 International Conference on Cybernation and Computation (CYBERCOM) (pp. 269-272). IEEE.
[13] Syahrudin, E., Utami, E., & Hartanto, A. D. (2024). Enhanced YOLOv8 with OpenCV for blind-friendly object detection and distance estimation. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 8(2), 199-207.
[14] Lin, B., He, F., Xu, Z., Lin, L., & Guo, Y. (2023). License plate character recognition with lightweight convolutional neural networks. IEEE 2nd International Conference on Electrical Engineering, Big Data, and Algorithms (EEBDA), 1152-1157.