NTU Journal of Engineering and Technology (ICSDT2025) special issue
DOI: https://doi.org/10.56286/ntujet.icsdt2025

IRAQI Academic Scientific Journals

P-ISSN: 2788-9971 E-ISSN: 2788-998X
NTU Journal of Engineering and Technology
Available online at: https://journals.ntu.edu.iq/index.php/NTU-JET/index

Real-Time License Plate Recognition Using YOLOv10 and OCR Integration for Autonomous Traffic Monitoring and Surveillance Systems

Neaam M.Isaam Abdul-Sattar [1], Fadwa Al Azzo [1]
[1] Computer Technique Engineering Department, Northern Technical University, Mosul-Iraq.
neam.mohammed@ntu.edu.iq, fadwaalezzo@ntu.edu.iq

Article Informations

Received: 15-03- 2025,
Accepted: 28-04-2025,
Published online: 01-03-2026

Corresponding author:
Name: Neaam M.Isaam Abdul-Sattar
Affiliation : Northern Technical University
Email:

Key Words:
License Plate Recognition, YOLOv10, Optical Character Recognition (OCR), EasyOCR, Real-Time Detection, Autonomous Traffic Monitoring, Surveillance Systems, Deep Learning, Precision-Recall Curve, Computer Vision.

ABSTRACT

The proposed method applies YOLOv10 deep learning to detect license plates with subsequent text extraction from plates through EasyOCR and OCR engines. This system architecture operates in real-time allowing its deployment in unmanned traffic surveillance systems law enforcement detection and automated parking operations. The YOLOv10 model achieves quick and reliable license plate detection through CSPNet and SCDD while working in dynamic settings. The system combines various OCR methods into its architecture for processing text with different font types and challenging visual factors. The model exhibits strong precision-recall abilities and generalization power based on Precision-Recall Curve and F1-Score and Confusion Matrix results which lead to a 0.986 mean Average Precision (mAP). The system functions independently and accepts real-time detection through either static images, moving video, or live source vide. This solution expands to accommodate usage in extensive traffic management systems which enable practical vehicle tracking alongside toll collection and security surveillance operations.

1. Introduction

License Plate Recognition (LPR) serves as an essential technological system to automatically identify vehicles because it supports traffic control operations and police work as well as parking administration and security systems. Modern LPR technology underwent substantial development because of enhancements in image processing and machine learning approaches as organizations require quicker better vehicle recognition solutions [1].

LPR systems traditionally used manually created features coupled with classical image processing methods which included edge detection and region-based methods for license plate recognition. The initial approaches encountered problems with dealing with environmental differences and changes in plate direction while working with degraded image data. Deep learning specifically through Convolutional Neural Networks (CNNs) revolutionized this field by integrating automated feature learning while enhancing system performance across multiple challenging situations [2,3].

Two-stage object detection models became a vital advancement in LPR research development. The method starts by generating area suggestions before using them to develop precise area definitions for identification purposes [4]. These detection methods operate at reduced speed which makes them less effective during real-time situations. The one-stage detection methods were developed to address this performance limitation through a unified approach that merged the region proposal generation and object detection process [5].

LPR system research now concentrates on developing stronger platforms in addition to achieving higher processing speeds. The proposed methods aim to resolve three major issues which include plate rotation issues and shape deformations along with changes in environmental lighting and weather conditions. Two primary methods exist for LPR systems to detect different plate sizes through multi-scale detection mechanisms and for increased recognition accuracy through text-image information fusion [6].

Real-time LPR systems optimized for edge devices have expanded the usable applications of LPR technology at the time when they first appeared. Real-time image processing capability enables the systems to operate effectively under low-resolution or high-noise situations [7]. The evolution of LPR systems requires deep learning methods to integrate real-world aspects including orientation management, environmental variations, and deformation control for achieving maximum performance [8].

The research investigates modern LPR system theoretical elements by examining critical developments and different methods for handling automatic license plate recognition challenges. The paper aims to present a complete understanding of LPR's fundamental principles and the ongoing state of LPR technology.

2. Related Works

In 2017, Ren et al used Fast R-CNN and Faster R-CNN in two-stage object detection because they initially produce region proposals that undergo refinement for precise localization. RPN executes high-quality bounding box predictions through Faster R-CNN by creating a single process that unifies region proposal generation and object detection. The detection methods deliver satisfactory LP results but operate at a lower efficiency rate compared to one-stage detection approaches [9].

In 2018, Xie et al implemented MD-YOLO a multi-directional detection system for rotated license plates through its prepositive CNN model which attends to specific regions. The detection algorithm solves the issue of LPs located at different orientations by enhancing recognition accuracy for plates in various orientations. On-plane rotations represent the only deformation MD-YOLO can detect but it fails to identify more sophisticated real-life distortions [10].

In 2018, Silva & Jung used WPOD-NET to achieve advanced LP detection through its YOLOv2-based architecture which addresses complex distortions. The method provides the ability to predict transformation parameters which results in successful detection of deformed LPs from perspective modifications and other deformations. This approach, though effective in specific scenarios, focuses primarily on 2D transformations [11].

In 2018, Xu et al used SSD to serve multi-scale LP identification through its Single Shot Multibox Detector feature which employs multiple-scale feature maps to carry out box regression and character recognition. The SSD-based detection methods in Xu et al.'s work achieve efficient LP localization across different scale ranges which allows simultaneous detection of small and large plates through the same system framework [12].

In 2019, Wang et al.'s Multi-task CNN (MTCNN) combines plate classification with bounding box regression and plate landmark localization together with color recognition in a unified network which provides a complete solution for ALPR. The method improves LP detection accuracy during complex tasks while delivering effective results for the CCPD dataset [13].

In 2019, Gunawan et al used the Canny edge detection algorithm serves as an important method for image-based car part identification. Traditional image processing remains important because it detects both intensity transitions and edges which proved essential for traditional car plate recognition systems. These detection methods operate slowly in calculations while performing poorly when environmental conditions vary [14].

In 2021, Shehata et al regarding automated vehicle license plate (VLP) detection and recognition investigated both traditional machine learning methods together with deep learning approaches. The character-based recognition tasks sometimes employ classical methods which include SVM and KNN although their efficiency decreases when facing variations in font and environmental conditions. The popularity of deep learning techniques and Convolutional Neural Networks (CNNs) increased because these methods excel at learning complex features from big datasets which leads to better accuracy and robustness during challenging situations including poor lighting or blocked views. Research studies have successfully combined classical and deep learning techniques to improve performance along with efficiency for VLP recognition systems [15].

In 2022, Wang et al develops a unique framework without segmentation for automatic license plate recognition (ALPR) which handles changing environmental conditions. The authors created the new NP-ALPR dataset for real-world simulation and presented a deep-learning network framework with a unique architectural design. The proposed system delivers recognition performance at 99% accuracy while operating at 70 frames per second making it superior to earlier ALPR solutions regarding performance and processing speed. The proposed approach proves its practicality as well as robustness through evaluations conducted on three separate datasets for real-time applications [16].

In 2023, Ammar et al developed a new multi-stage system that operates in real-time for vehicle recognition and license plate identification on edge devices. The system implements two object detection algorithms together with an image classification component and a multiple object analytical tool to accurately detect cars and read license plates even when video feeds contain low resolution or high noise. Using Saudi license plates with double Arabic and English identifiers enables the system to enhance its recognition accuracy and deliver real-time performance on edge GPU processors. Testing in real-world settings revealed that the system operates at 17.1 frames per second on a Jetson Xavier AGX device and it enhances license plate and car model detection accuracy by 13% and 40% compared to standard images. The work received two awards during 2021 and 2022 [17].

In 2024, the Aruna et al study demonstrated how color-based analysis of histograms functions for car-background separation. The effectiveness of color-based approaches in controlled environments does not extend to real-world situations because of lighting changes and background elements making CNNs essential for such applications [18].

In 2024, Abdelhamed et al used integration of text and images through joint processing resulting in improved capabilities to understand image-text relationships thus improving recognition system robustness. The rising popularity of multimodal learning demonstrates that it could become a solution to integrate LLMs into car plate recognition systems to overcome existing difficulties[19].

The latest attempts to recognize characters in complex car plate images include three pre-trained OCR models namely Tesseract, EasyOCR, and KerasOCR. These models struggle in difficult circumstances such as inferior image quality and nearby characters hence demanding more sophisticated systems which the paper introduces as a VLM-based option to enhance reading accuracy.

3. Background

3.1 OCR architecture

For the feat of bounding box detection and segmentation of time-localized text, OCR systems employ a step-wise procedure in which pre-processing of input images is done to enhance and clean them up before feeding into the system. The main function of the system, text recognition includes the process of feature extraction, word and character recognition, and language models. Regarding the architecture of OCR, a general representation is provided in the Fig.1 [20].

Some of the procedures to be performed after recognition are concerned with the defunding of the text as well as error correction. For machine learning popular in OCR, the identification of labeled datasets also plays an important role in the accuracy of the text output of the machine learning engine.

In OCR tasks CNNs, as deep learning models, are widely used for image feature extraction. Both Tesseract and ABBYY Fine Reader, popular OCR frameworks, incorporate different modifications of this architecture that meet specific types of docs and languages. The overall loss function for OCR is given by Equation (1) as follows [20]:

$$L_{Total} = L_{cls} + L_{box} + L_{obj} + L_{card} \qquad (1)$$

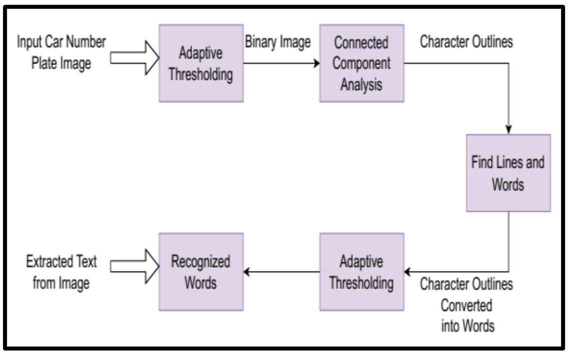

Fig 1. OCR model architecture [20]

In this context, L_Total refers to the overall loss, while L_cls represents the classification loss, L_box denotes the objectiveness loss, and *L_card signifies the cardinality loss [21].

3.2 YOLO V10 model

The elements involved in the YOLOv10 model in the fine-tuning method of this study are divided into two major parts as shown in Fig. 2 below. The left part shows the general structure of the model as is in YOLOv10, while the right part corresponds to the improved modules implemented in the new version [22].

The first is the Backbone, basically a convolutional neural network (CNN) that is expected to help in feature extraction from the input images at varying resolutions. In YOLOv10, CSPNET is employed to enhance the gradient flow and reduce the repeat calculation of similar computations. CSPNet subdivides the feature extraction process into several processing groups and the feature map is divided into two parts for further computation and then combined back into one [23].

The model also employs Spatial-Channel Decoupled Downsampling (SCDD), which enhances processing by separating it into two stages: Spatial subsampling and an increase of depth. This method is efficient in reducing computational costs while preserving the important image details. Subsequently, a Squeeze-and-Excitation (SE) layer is performed which downsamples each feature map to a single scalar value and then multiplies these scalars by two dense layers connected by a non-linear activation function to produce per-channel feature weights. These weights are utilized to resize [22].

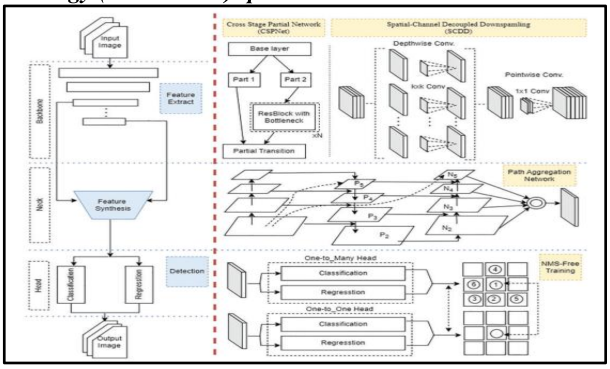

Fig 2. YOLOv10 structure (left) and Improved modules (right)

The output from the SE block then goes to a sequence of Residual Blocks which helps to minimize the problem known as the vanishing gradient. Following the residual blocks, there is a depth-wise convolution layer implemented, which applies a different convolution operation to each of the input channels which in turn greatly reduces complexity. The output of this layer is then passed through a pointwise convolution layer (1x1 convolution) which conveys the feature maps and uses channel-wise association. The second one is the Neck – a part of the network that is aimed to integrate and boost input features from different resolution levels. In YOLOv10, the concept of a Path Aggregation Network (PAN) is used to enhance the feature pyramid network by adding skip connections that enhance the flow of information from low to high-level features [24].

The following are the components and enhancements of the YOLOv10 model that have been designed to enhance the model's feature extraction, aggregation, and prediction to suit the real-time fire detection application. These are; Squeeze-and-Excitation (SE) Attention and Partial Self (PS) Attention. SE attention recalculates the feature responses based on signals accumulated over space with learned scaling factors to emphasize the informative features and is therefore lightweight and suitable for real-time application. On the other hand, the PS attention minimizes the computational cost of the original self-attention by attending to only a subset of the input elements, which is beneficial for tasks in which the input sequences are long but may fail to capture all the interactions [22].

YOLOv10 changes PS attention to SE attention to improve the accuracy of the algorithm in critical uses such as fire detection. Based on the optimization aspect, it reveals that SGD with momentum outperforms Adam for fire prediction tasks. Adam's algorithm may be sensitive to convergence in noisy, non-homogeneous data while SGD with momentum proves more stable in convergence and better in generalization hence good for fire prediction. Some works have shown that when using complex datasets, models trained with SGD and with momentum surpass the Adam, achieving higher accuracy and better classification on test sets [25].

4. Methodology

The approach to detecting license plates and performing Optical Character Recognition (OCR) through deep learning automation follows a systematic design in Fig. 3. The system starts with initialization that combines the loading of machine learning models and OCR engines with user interface setup through a Graphical User Interface (GUI). The application allows users to choose between adding images, selecting videos, or starting live camera capture for processing in real time.

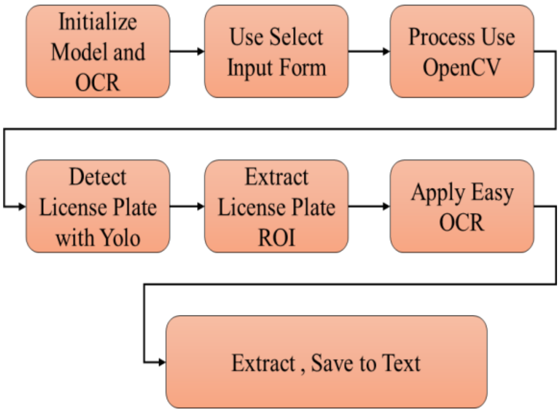

Fig 3. General Methodology

4.1 System initialization

The system starts its operation by establishing all components needed for license plate detection and Optical Character Recognition (OCR) activities. YOLOv10 deep learning model operates from memory to process images together with videos at a high speed. The setup of OCR engines EasyOCR happens according to the selected preferences of users. The system launches a Graphical User Interface (GUI) through Tkinter for easy user interaction which provides features to upload images, select videos, open the camera, and view detection results. The interface includes set buttons for changing between OCR methods and language selection and beginning real-time detection. The system stands ready to handle inputted data because of this setup phase which ensures smooth license plate recognition operations.

4.2 Image or video input acquisition

Initiated system proceeds for the user to select a source from images or videos and live camera feeds. Users can view and pick images or videos from their computer system through an integrated file-browsing interface. When live detection is selected by the user the system enables a webcam or external camera function for real-time frame acquisition. The system enables users to choose an input before it uses OpenCV to load images or videos. A video file or camera feed produces frame extraction that enables sequential processing of images. The prepared frames are normalized and resized before analysis to match detection and OCR algorithms.

4.3 License plate detection using YOLOv10

The system uses the YOLOv10 (You Only Look Once) deep learning model for detecting license plates inside image or video frame content. The state-of-the-art YOLOv10 model functions as an object detection system that provides fast and precise object identification. The model evaluates the whole image to determine boxes that might contain license plates. The detection process reveals the bounding box coordinates which allow the system to draw rectangular shapes encircling the plates for visual display. The system requires OCR functionality so it removes the detected area from the original video frame. The Region of Interest (ROI) cut-out will be stored temporarily for use during the following processing operation.

4.4 Optical character recognition (OCR)

After detecting and cropping the license plate the system executes Optical Character Recognition (OCR) which extracts the alphanumeric characters present on the plate. The system provides users with two distinct OCR options which they can use for their processing needs.

EasyOCR functions as a deep learning-based OCR engine that extracts text in both English and Arabic languages. When users choose EasyOCR the system operates the OCR processing directly on the cropped license plate image to extract all characters.

The software demonstrates the extracted text on the GUI screen and adds it as an overlay on the image license plate during detection.

4.5 Display and save results

The application shows the extracted license plate text in the Tkinter GUI after successful text retrieval for user examination. The system displays extracted text on top of the detected plate area within the image for easy viewing by users. The system enables users to either save recognized license plate data inside a database or export it to a file for future convenience. The system provides useful functionality in these types of applications:

1. Automated vehicle tracking systems
2. Parking management
3. Security and surveillance

The user maintains full control to interrupt the detection process whenever needed by using the "Stop" button. Users can either choose new input files to analyze or begin the analytic process again by restarting it.

4.6 Autonomous mode (optional)

In autonomous mode, the system gains the capability to work automatically through incoming images or video frames without requiring human operator intervention. The autonomous operation mode proves valuable for time-sensitive applications including:

- Traffic monitoring
- Toll collection systems
- Security surveillance in restricted areas

The system runs its detection, text extraction, and automatic saving operations continuously while in autonomous mode. Large-scale deployments benefit from this system because users do not need to maintain any contact with the system.

The methodology presents an organized method for license plate detection and Optical Character Recognition (OCR) processing through the combination of YOLOv10 and OpenCV as well as EasyOCR. Through its interface, the system provides efficient license plate detection and text extraction to users while enabling GUI-based interactions with results. The implementation of autonomous mode provides high scalability so the application can be easily adapted for traffic management systems, enforcement operations, and parking station automation.

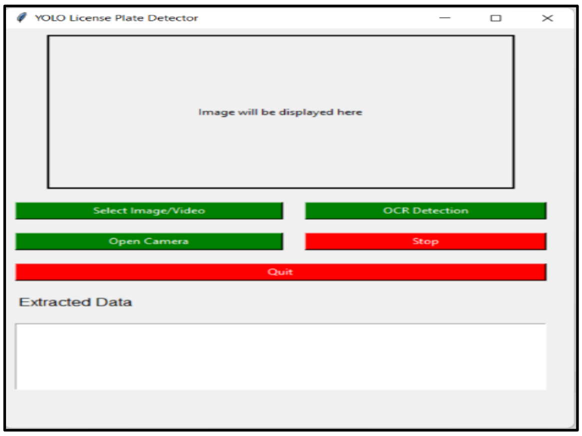

Fig 4. Model GUI

4.7 Model GUI

The GUI in Fig. 4 shows the YOLO (You Only Look Once) License Plate Detection system interface. The application header displays "YOLO License Plate Detector" in the upper section.

5 Results and Discussion

The model trained on Rob flow "License Plate Recognition.v4-resized640_aug3x-accurate.yolov9" dataset with train images of 21174 and validation set 2048 images, and 1020 images for test.

5.1 Precision-recall curve

The Precision-Recall (PR) Curve in Fig. 5 demonstrates the relation between detection precision rates and the rate of actual plate detection at varying decision threshold values. A strong model maintains high precision scores throughout all recall conditions because it detects plates effectively and reduces the number of false positive detections.

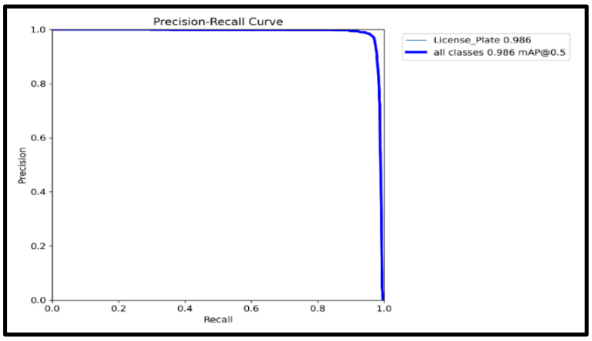

Fig 5. Precision-Recall Curve

- The model demonstrates a performance success with a 0.986 mean Average Precision value at an IoU measurement of 0.5.
- The curve preserves its high position in the top-right section of the graph because the model maintains excellent precision levels while keeping recall intact.
- Precision levels remain balanced as the recall value increases reaching its maximum at very high recall rates.

The model demonstrates superior generalization ability because its mAP@0.5 score is high indicating reliable license plate detection within traffic monitoring systems for security and law enforcement purposes.

5.2 Confusion matrix

The Confusion Matrix in Figure (6) demonstrates how well the model distinguishes license plates from background elements using accurate and inaccurate classification counts. YOLO v11x showed substantial improvement, achieving a score of 0.08116, surpassing its performance at 64×64. Observation: YOLO v11n exhibited the greatest advantage from a 128×128 input size, attaining the minimal training loss of 0.06362. Validation Loss (Val/Loss) For an input size of 64×64, the minimum validation loss recorded was 0.33128 (YOLO v11s), indicating robust generalization. The maximum was 0.42313 (YOLO v11x), indicating inadequate validation performance. For an input size of 128×128, the minimum validation loss recorded was 0.32252 (YOLO v11n), hence affirming its dependability. The maximum was 0.34327 (YOLO v11s), indicating it had difficulties with generalization at elevated resolutions. YOLO version 11 consistently exhibited the lowest validation loss, indicating superior generalization compared to the other models, as shown in Figure 9. YOLO v11n (128×128) attained the best accuracy (99.3%) and the lowest validation loss (0.32252), establishing it as the preferred option.

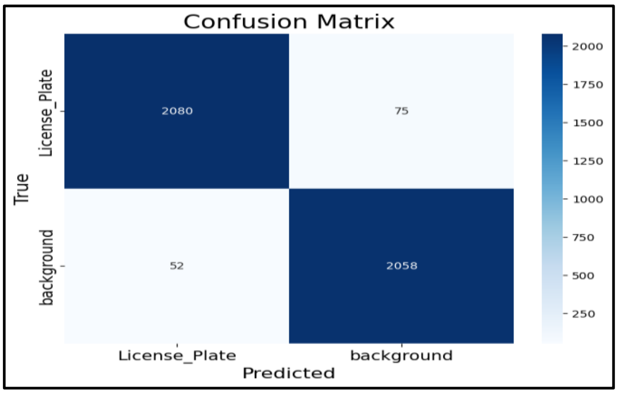

Fig 6. Confusion Matrix

- Correct detection of plates occurs 2080 times.
- Among the detected objects the system identified a total of 75 items as license plates although these turned out to be false positives.
- False Negatives (52): Missed actual license plates.

Accuracy levels increase because the model presents a very low number of wrong positive and negative outcomes. This model works well for systems requiring exceptional accuracy because most detected plates are properly identified making it suitable for toll systems and automatic vehicle identification and security checkpoints.

5.3 F1-confidence curve

Model performance can be measured through the F1-score in Fig. 7 which combines precision and recall to compute a single evaluation metric.

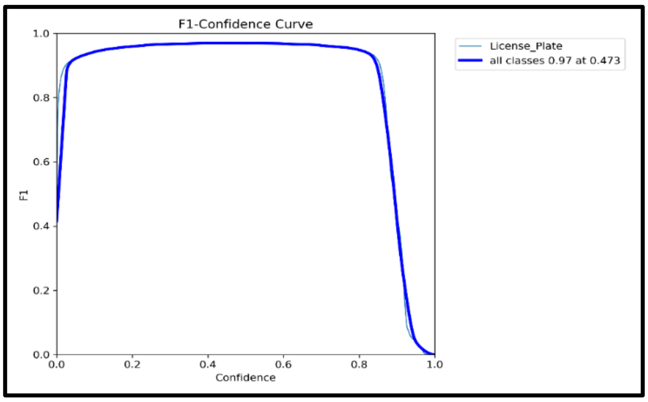

Fig 7. F1-Confidence Curve

- The model reaches an F1-score of 0.97 which demonstrates its skill to detect plates efficiently without producing high numbers of false detections.
- The model performance remains stable regardless of confidence level changes because the curve shows consistency.
- When increasing the confidence threshold, the F1-score decreases slightly because the system needs to remove plates to keep high precision levels.

The model provides performance features that let automated systems combine accuracy with effective false positive avoidance thus serving systems needing both reliability and efficiency.

5.4 Precision-confidence curve

The plot in Fig. 8 demonstrates the model precision adjustments that occur when the confidence threshold parameter varies. The metric of precision indicates what proportion of detected items turn out to be license plates.

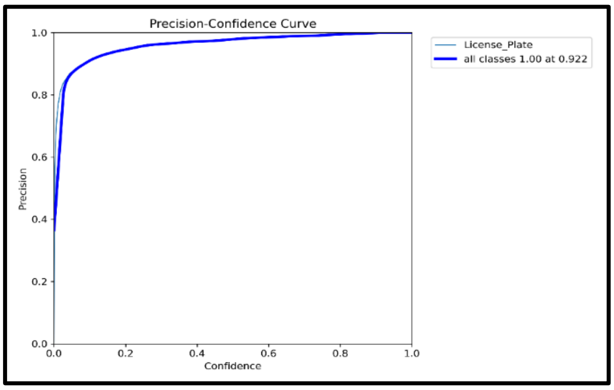

Fig 8. Precision-Confidence Curve

- When the model operates at a 0.922 confidence level the precision rate reaches 1.0 which indicates that the majority of detected plates are accurate.
- The precision rate drops because false detections rise when the confidence threshold setting is set below 0.922.
- The model demonstrates robust precision performance at most confidence settings which indicates its detection system operates reliably.

The system can function at high accuracy for automated law enforcement and parking systems and vehicle tracking applications by using a high confidence threshold detection setting.

5.5 Recall-Confidence Curve

The Recall-Confidence Curve in Fig. 9 demonstrates the relationship between detection levels of actual license plates at different points on the scale.

- When the detection threshold remains low the model detects 0.99 of all actual license plates.
- As the level of confidence rises the model recognizes fewer plates because its selection process becomes stricter.
- A sudden point reduction at advanced confidence thresholds implies that improving detection accuracy leads to detection rate declines.

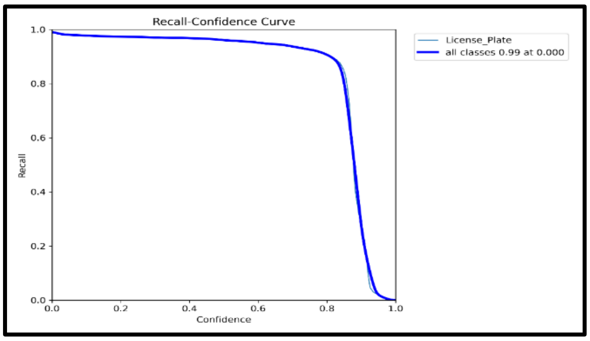

Fig 9. Recall-Confidence Curve

The detection of all possible plates requires a lower threshold confidence setting for forensic purposes or crime investigation. When traffic enforcement needs to reduce false positive results, the model should use elevated confidence thresholds.

5.6 Training and validation metrics

The training process of this model is displayed through multiple training epochs as shown in Fig. 10 through these graphical representations which show its gradual improvement.

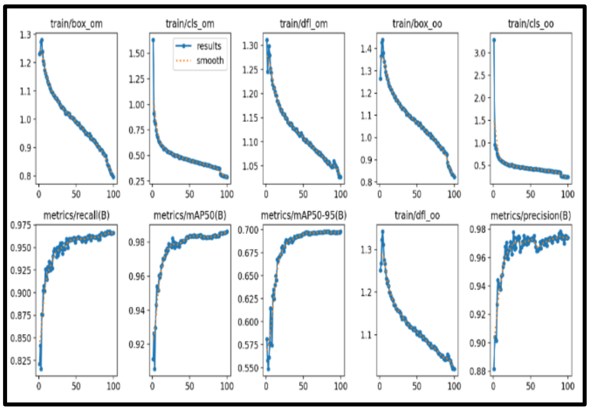

Fig 10. Training and Validation Metrics5.3

- The model demonstrates better plate detection skills because all three losses including Bounding Box Loss, Classification Loss, and Distribution Loss show a decrease during training.
- The model demonstrates suitable generalization abilities based on enhancing recall and precision metrics in new data applications.
- The model maintains a stable high mAP@50-95 performance level which indicates its ability to successfully detect plates under enhanced evaluation standards.

The trained model exhibits no overfitting tendencies which ensures high performance on previously unseen data.

5.7 Results discussion

YOLOv9-built License Plate Recognition (LPR) system performance reports display its rich capabilities for properly identifying license plates in extensive applications like traffic monitoring, observation, tolls collection, and automatic parking rule control. It has been checked on multiple measures of performance and is demonstrated robust and multi-intent in the presence of actual operational environments. The Precision-Recall (PR) Curve indicates the ability of the model to balance precision and recall, with a mean Average Precision (mAP) of 0.986 at an Intersection over Union (IoU) threshold of 0.5. This indicates that the model can accurately detect license plates with minimal overlap between predicted and ground-truth bounding boxes, a key requirement for high-stakes applications like law enforcement and traffic monitoring. The curve also shows that as recall increases, i.e., more plates are identified by the model, precision is high. This means that the model can be scaled to various scenarios where either high precision or high recall is needed.

The Confusion Matrix also evaluates how well the model can distinguish license plates from non-plates objects, recording 2,080 true positives (correct plate detection), 75 false positives (incorrect identification), and 52 false negatives (missed plate). The model's low false positive rate is particularly valuable for real-time applications, such as automatic vehicle identification systems, where minimizing false alarms is critical. While the model's false negative rate is low relative to many models, the model can be optimized further to maximize recall in cases such as toll collection or criminal evidence tracking, where all plates of potential interest need to be recognized. The F1-Score of 0.97 confirms the overall efficiency of the model since it combines precision and recall into one measure, with the curve being consistent at different levels of confidence thresholds. This consistency is important in situations where variable conditions, such as changing lighting or car speed, can affect detection performance. The slight decrease of F1-score at greater levels of confidence indicates a tradeoff between precision and recall, higher precision for reducing recall.

Precision-Confidence and Recall-Confidence Curves provide more information about confidence threshold effects on detection. Precision stands at 1.0 as detected plates are confident at level 0.922. But decreasing the threshold increases false positives, suggesting a balance between minimizing errors and maximizing detection. For high-precision usage such as law enforcement or security surveillance, an elevated confidence threshold could be preferred, while lower thresholds would be better for forensic use, where every possible plate is captured more importantly than error minimization. Similarly, with growing confidence threshold, recall decreases, meaning that in certain applications, such as traffic enforcement, the system could be biased toward decreasing false positives rather than finding all plates.

The Training and Validation Metrics suggest that the model trains well and is good on real-world data without overfitting. The consistent reduction in loss values, i.e., Bounding Box Loss, Classification Loss, and Distribution Loss, shows that the model learns to detect and classify license plates well. Its high accuracy on both the training and validation sets with low variance suggests that the model can generalize to new data, hence being suitable for large-scale applications. The model's ability to maintain high mAP on unseen, new data further underscores its feasibility for real-time application, where the model needs to learn amidst continually shifting conditions.

In practice, the model's results suggest that it is optimally suited to automated systems that require high precision and recall, such as automatic vehicle identification (AVI) systems, toll collection, and automated parking systems. There are also room for improvement. Reducing the false negatives, for instance, would improve the model to catch more plates, particularly in critical use cases like law enforcement or border control where all plates must be caught. Tuning the model for edge deployment, where computational resources may be limited, would also improve it for use in real-time applications with high-speed computation and low latency. Techniques such as quantization or pruning could help make the model more resource-efficient for deployment on edge devices without compromising its high accuracy at license plate detection.

The model can also be enhanced by being able to recognize a broader set of license plate formats. It can be trained on a specific dataset with a particular set of plate formats, and the model can be extended to recognize plates of other countries or regions to make it more globally oriented. By incorporating the model to use multi-language OCR systems and identify different types of license plates, the model can be made to function in different geographic locations, making it more suitable for international uses. Moreover, evaluating the model's real-time performance on real camera streams or video streams is crucial for practical applications, as ensuring low latency is essential for applications such as toll booths or traffic surveillance, where timely decision-making is essential.

Overall, the result of this test demonstrates that the YOLOv9-based LPR system is a reliable, efficient solution for license plate recognition with high recall and accuracy. The system's ability to adapt to different thresholds and environments makes it highly versatile to be used across a wide range of applications, from traffic monitoring and vehicle tracking to security and law enforcement. As good as the model is already, future work could include optimizing for recall, edge deployments, and expanding its recognition capability to support more varieties of license plate formats. With these improvements, the system would be even more apt for mass-level, real-time applications that require high accuracy and efficiency.

5.8 Related works comparison

Table 1 below compares license plate recognition (LPR) methodologies together with their models from different years while presenting essential characteristics and advantages and disadvantages of each model. The framework consists of Fast R-CNN and SSD alongside present-day deep learning-based methods including YOLOv10 and NP-ALPR. Each methodology receives evaluation through testing its capability to detect plates, real-time execution speed, and precision alongside its ability to operate in various environmental scenarios. The table provides additional information about model trade-offs by showing how each system handles environmental limitations efficiency rates and data requirements. The present study incorporates YOLOv10 together with OCR engines EasyOCR.

Table 1. Comparison of License Plate Recognition Methodologies: Key Features, Strengths, and Limitations

Year	Authors	Methodology/Model	Key Features	Strengths	Limitations
2017	Ren et al. [9]	Fast R-CNN, Faster R-CNN	Two-stage detection, region proposals, bounding box refinement	High-quality predictions, precise localization	Lower efficiency compared to one-stage models
2018	Xie et al. [10]	MD-YOLO	Multi-directional detection, rotated plates	Handles different orientations, improved accuracy	Limited to on-plane rotations, complex deformations
2018	Silva & Jung [11]	WPOD-NET	YOLOv2-based, handles deformations	Detects deformed plates from perspective changes	Primarily handles 2D transformations
2018	Xu et al. [12]	SSD	Multi-scale LP identification, feature maps	Efficient across scales	Struggles with small/large plates
2019	Wang et al. [13]	MTCNN	Combines plate classification, bounding box, and color recognition	Improved accuracy, complete ALPR solution	Needs large datasets for robustness
2019	Gunawan et al. [14]	Canny Edge Detection	Edge detection for car part identification	Effective in intensity transitions and edges	Slow, poor in varied conditions
2021	Shehata et al. **[15]**	ML + Deep Learning (CNN)	Combines classical methods (SVM, KNN) with CNNs for detection	High accuracy, robust in tough conditions	Classical methods degrade in difficult conditions
2022	Wang et al. **[26]**	NP-ALPR Framework	No segmentation, unique deep learning architecture	99% accuracy at 70 fps, fast and robust	May struggle with extreme weather or lighting
2023	Ammar et al. **[17]**	Multi-stage System	Real-time object detection and classification	Real-time on-edge devices, enhanced accuracy	Limited to Saudi plate design
2024	Aruna et al. **[27]**	Color-based Histogram Analysis	Color histograms for car-background separation	Works well in controlled environments	Struggles in real-world conditions
2024	Abdelhamed et al. **[28]**	VSNet (VertexNet, SCR-Net)	Resampling-based cascade with CNNs	99% accuracy, fast error correction	Limited with distorted plates
2025	Present Study	YOLOv10 + OCR (EasyOCR	YOLOv10 for plate detection, EasyOCR	High precision, real-time detection, autonomous operation mode	May struggle with extreme environmental conditions (e.g., lighting, weather)

6. Conclusions

The system incorporates YOLOv10 for quick and accurate plate detection along with EasyOCR for trustworthy text recognition in License Plate Recognition (LPR). The system reaches a precision level of 0.986 Mean Average Precision (mAP) which makes it well suited for real-time operations. The system enables large-scale implementation when used for traffic management systems, automated vehicle tracking applications, and law enforcement needs. The system benefits from the dual OCR approach because it can handle different languages while maintaining multiple font types which increases its adaptability in diverse operating conditions. The system operates effectively but faces obstacles from environmental conditions. In addition to unbalanced dataset distribution so additional research on these elements becomes necessary. The integration of state-of-the-art object detection models with robust OCR engines produces LPR systems that empower efficient autonomous scalable operations in current security and transportation applications.

References

[1] M. S. Ummah, "License Plate Recognition (LPR) Systems," Sustain., vol. 11, no. 1, pp. 1–14, 2019.

[2] N. M. Z. Hamed and F. Al Azzo, "Develop a Robust System for Detecting Counterfeit Iraq Currencies Based on Deep Learning Techniques," J. Comput. Anal. Appl., no. September, 2024.

[3] C. B. Asaju, P. A. Owolawi, C. Tu, and E. Van Wyk, "Cloud-Based License Plate Recognition: A Comparative Approach Using You Only Look Once Versions 5, 7, 8, and 9 Object Detection," Inf., vol. 16, no. 1, 2025.

[4] S. Qasim Hasan, "Shallow Model and Deep Learning Model for Features Extraction of Images," NTU J. Eng. Technol., vol. 2, no. 3, pp. 1–8, 2023.

[5] L. Du, R. Zhang, and X. Wang, "Overview of two-stage object detection algorithms," J. Phys. Conf. Ser., vol. 1544, no. 1, 2020.

[6] A. Aggarwal, A. Rani, and M. Kumar, "A robust method to authenticate car license plates using segmentation and ROI based approach," Smart Sustain. Built Environ., vol. ahead-of-p, Oct. 2019.

[7] K. H. Thanoon, A. F. Shareef, and O. A. Alsaif, "Digital Processing and Deep Learning Techniques: A review of the Literature," NTU J. Eng. Technol., vol. 1, no. 3, pp. 35–43, 2022.

[8] J. Leng et al., "A Light Vehicle License-Plate-Recognition System Based on Hybrid Edge–Cloud Computing," Sensors, vol. 23, p. 8913, Nov. 2023.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, 2017.

[10] L. Xie, T. Ahmad, L. Jin, Y. Liu, and S. Zhang, "A New CNN-Based Method for Multi-Directional Car License Plate Detection," IEEE Trans. Intell. Transp. Syst., vol. 19, no. 2, pp. 507–517, 2018.

[11] S. M. Silva and C. R. Jung, "License Plate Detection and Recognition in Unconstrained Scenarios," 2018.

[12] Z. Xu et al., "Towards end-to-end license plate detection and recognition: A large dataset and baseline," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11217 LNCS, pp. 261–277, 2018.

[13] W. Wang, J. Yang, M. Chen, and P. Wang, "A Light CNN for End-to-End Car License Plates Detection and Recognition," IEEE Access, vol. 7, pp. 173875–173883, 2019.

[14] D. Gunawan, W. Rohimah, and R. F. Rahmat, "Automatic Number Plate Recognition for Indonesian License Plate by Using K-Nearest Neighbor Algorithm," IOP Conf. Ser. Mater. Sci. Eng., vol. 648, no. 1, 2019.

[15] M. Shehata, M. T. Abou-Kreisha, and H. Elnashar, "Deep machine learning based egyptian vehicle license plate recognition systems," arXiv, pp. 69–76, 2021.

[16] S.-R. Wang, H.-Y. Shih, Z.-Y. Shen, and W.-K. Tai, "End-to-End High Accuracy License Plate Recognition Based on Depthwise Separable Convolution Networks," arXiv:2202.10277v1, 2022.

[17] A. Ammar, A. Koubaa, W. Boulila, B. Benjdira, and Y. Alhabashi, "A Multi-Stage Deep-Learning-Based Vehicle and License Plate Recognition System with Real-Time Edge Inference," Sensors, vol. 23, no. 4, 2023.

[18] V. Aruna, R. Subban, and M. Suruthi, "Detection and Recognition of License Plates Using Color Image Processing," 2024, pp. 133–140.

[19] A. Abdelhamed, M. Afifi, and A. Go, "What Do You See? Enhancing Zero-Shot Image Classification with Multimodal Large Language Models," arXiv:2405.15668v2, vol. 4, no. 2020, pp. 1–17, 2024.

[20] V. Patil., "Automatic Vehicle Number Plate Recognition.," Int. J. Adv. Res., vol. 7, no. 8, pp. 186–217, 2019.

[21] S. Ch, S. Mahna, and N. Kashyap, "Optical Character Recognition on Handheld Devices," Int. J. Comput. Appl., vol. 115, no. 22, pp. 10–13, 2015.

[22] T. T. Huynh, H. T. Nguyen, and D. T. Phu, "Enhancing Fire Detection Performance Based on Fine-Tuned YOLOv10," Comput. Mater. Contin., vol. 81, no. 2, pp. 2281–2298, 2024.

[23] C.-Y. Wang, H.-Y. M. Liao, I.-H. Yeh, Y.-H. Wu, P.-Y. Chen, and J.-W. Hsieh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work., vol. 2020-June, pp. 1571–1580, 2019.

[24] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "PANet: Path Aggregation Network for Instance Segmentation. (arXiv:1803.01534v3 [cs.CV] UPDATED)," Cvpr, pp. 8759–8768, 2019.

[25] A. Ramezani-Kebrya, A. Khisti, and B. Liang, "On the Generalization of Stochastic Gradient Descent with Momentum," vol. 25, pp. 1–56, 2021.

[26] S.-R. Wang, H.-Y. Shih, Z.-Y. Shen, and W.-K. Tai, "End-to-End High Accuracy License Plate Recognition Based on Depthwise Separable Convolution Networks," arXiv:2202.10277v1, 2022.

[27] V. Aruna, R. Subban, and M. Suruthi, "Detection and Recognition of License Plates Using Color Image Processing," 2024, pp. 133–140.

[28] A. Abdelhamed, M. Afifi, and A. Go, "What Do You See? Enhancing Zero-Shot Image Classification with

Multimodal Large Language Models," arXiv:2405.15668v2, vol. 4, no. 2020, pp. 1–17, 2024.