OPEN ACCESS

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-14 Issue-6, May 2025

Development and Implementation of a Custom License Plate Detection and Recognition System Using YOLOv10 and Tesseract OCR: A Comprehensive Study in Computer Vision and Optical Character Recognition Technologies

Priyankush Kaushik Baruah, Pranabjyoti Haloi

***Abstract:** This study presents an automated license plate detection and recognition system that combines YOLOv10 for real-time object detection and Tesseract OCR for robust text extraction. The methodology involves training a customised YOLOv10 model on annotated vehicle datasets to localize license plates, followed by region-of-interest (ROI) filtering to enhance accuracy. Detected plates are processed with Tesseract OCR to convert visual data into machine-readable text. Evaluated using precision, recall, and inference speed metrics, the system achieves 97% detection accuracy and real-time performance, demonstrating reliability in automated vehicle identification tasks such as traffic monitoring. This work highlights the synergy between YOLOv10's detection efficiency and Tesseract's OCR capabilities, providing a scalable solution for intelligent transportation systems.*

***Keywords:** License Plate Recognition (LPR), YOLOv10, Optical Character Recognition (OCR), Object Detection, Intelligent Transportation Systems (ITS), Real-Time Monitoring.*

Abbreviations:
ROI: Region-of-Interest
LPR: License Plate Recognition
OCR: Optical Character Recognition
ITS: Intelligent Transportation Systems
LPR: License Plate Recognition
ANPR: Automatic Number Plate Recognition
YOLO: You Only Look Once
CRNNs: Convolutional Recurrent Neural Networks
AI: Artificial Intelligence

I. INTRODUCTION

In recent years, the need for automated vehicle identification has grown significantly, driven by advancements in computer vision, deep learning, and artificial intelligence (AI) [1]. License Plate Recognition (LPR) is crucial in various applications such as traffic law enforcement.

Manuscript received on 25 March 2025 | First Revised Manuscript received on 30 March 2025 | Second Revised Manuscript received on 19 April 2025 | Manuscript Accepted on 15 May 2025 | Manuscript published on 30 May 2025.
*Correspondence Author(s)
Priyankush Kaushik Baruah*, Department of Electrical Engineering, Jorhat Engineering College, Jorhat (Assam), India. Email ID: priyankushjorhat@gmail.com, ORCID ID: 0009-0002-9253-0676
Dr. Pranabjyoti Haloi, Associate Professor, Department of Electrical Engineering, Jorhat Engineering College, Jorhat (Assam), India. Email ID: pranab.jec@gmail.com, ORCID ID: 0000-0002-8986-5267

Enforcement, automated toll collection, and intelligent parking. Systems and smart city infrastructure. Traditional LPR systems rely on rule-based methods and handcrafted feature extraction, which often struggle with challenges like poor lighting, motion blur, varying plate designs, and occlusions. These limitations have necessitated the development of robust, real-time, and scalable deep learning-based LPR systems that can efficiently process complex scenarios.

This research aims to bridge the gap between conventional image processing methods and modern AI-based solutions by implementing a deep learning-powered LPR system using YOLOv10 for object detection [2] and Tesseract OCR for text recognition [3]. The system follows a structured pipeline consisting of:

- **License Plate Detection**: Utilizing YOLOv10, an advanced deep-learning model for real-time object detection, to locate license plates in images or video frames with high precision.
- **Image Preprocessing**: Enhancing the detected license plates through grayscale conversion, noise reduction, and binarization to improve text extraction accuracy.
- **Optical Character Recognition (OCR)**: Implementing Tesseract OCR to extract alphanumeric characters from detected plates.
- **Post-Processing and Refinement**: Using custom filtering techniques to reduce errors, correct misidentified characters, and format extracted text into a structured output.

This research aims to develop a fast, accurate, and adaptable system by integrating YOLOv10 and Tesseract OCR. Unlike traditional LPR approaches that rely on template matching or edge detection, the proposed deep-learning model can learn complex patterns and generalise across different plate designs and environmental conditions.

A. Research Objectives

This study aims to design and implement a robust License Plate Recognition (LPR) system by addressing critical detection, text extraction, and real-time processing challenges. The specific objectives are:

i. *Develop a Custom YOLOv10 Model for License Plate Detection*
   - Train and optimize a YOLOv10 model using annotated datasets to achieve high precision and recall in detecting license plates under diverse environmental conditions.

ii. *Integrate and Enhance Tesseract OCR for Text Extraction*
   - Implement preprocessing techniques (e.g., grayscale conversion, noise reduction, adaptive thresholding) and post-processing rules (e.g., character correction) to enhance OCR accuracy, addressing challenges such as non-standard fonts and plate designs.

iii. *Optimize Real-Time Performance for Practical Applications*
   - Ensure the system processes live video streams with minimal latency by leveraging GPU acceleration and architectural optimisations, making it suitable for traffic monitoring and automated toll collection.

iv. *Evaluate System Robustness and Scalability*
   - Validate the model's adaptability through rigorous testing across varying vehicle speeds, skewed plate orientations, and dynamic lighting conditions.

v. *Implement Region of Interest (ROI) for Enhanced Efficiency*
   - Define ROI within video frames to focus computational resources, reduce false positives, and improve detection accuracy in cluttered environments.

These objectives collectively aim to bridge the gap between traditional rule-based LPR systems and modern AI-driven solutions, offering a scalable framework for real-world intelligent transportation applications.

B. Significance of the Study

The proposed License Plate Recognition (LPR) system addresses critical gaps in existing automated vehicle identification technologies, offering transformative potential for smart cities and intelligent transportation systems (ITS). Its significance lies in:

i. *Societal and Operational Impact*
   - **Law Enforcement:** This service streamlines the detection of traffic violations (e.g., stolen vehicles, expired registrations) through automated, real-time plate recognition.
   - **Parking Management:** Enables contactless entry and exit in parking facilities, reducing manual oversight and operational costs.
   - **Toll Automation:** This technology minimises congestion at toll booths by enabling seamless electronic toll collection via Automatic Number Plate Recognition (ANPR).
   - **Urban Planning:** Provides actionable traffic flow data to optimise infrastructure design and alleviate congestion.

ii. *Technical Contribution*
   - Integrates YOLOV10 (state-of-the-art object detection) with Tesseract OCR (robust text extraction), balancing speed and accuracy (high mAP) in real-world conditions.
   - Introduces region-of-interest (ROI) filtering and adaptive preprocessing to address challenges such as motion blur, occlusions, and non-uniform lighting.

iii. *Future Scalability:*
   - Establish a foundation for deploying attention-based OCR models (e.g., CRNNs) and edge-computing frameworks to enhance portability and efficiency.

This study bridges the gap between academic research and practical deployment, offering a scalable solution for next-generation transportation systems.

II. LITERATURE REVIEW

Automated license plate recognition (LPR) systems have gained significant attention in recent years due to their applications in traffic management, law enforcement, and smart city infrastructure. This section reviews advancements in object detection, OCR techniques, and system deployment, highlighting gaps that the proposed YOLOv8 and Tesseract-based LPR framework addresses.

A. Object Detection in LPR Systems

Traditional LPR systems relied on handcrafted features and heuristic algorithms for license plate localization, which struggled with variability in lighting, angles, and plate formats. The advent of deep learning–based detectors, particularly the YOLO (You Only Look Once) family, revolutionized real-time object detection. YOLOv8, the latest iteration, improves upon its predecessors with a scalable architecture optimized for edge devices, achieving a balance between speed and accuracy [7]. Studies by Li et al. [4] demonstrated that end-to-end trainable models like YOLO significantly outperform classical methods in detecting small objects (e.g., license plates) under motion blur and occlusion. However, challenges persist in low-light environments and multi-scale plate detection, necessitating robust preprocessing and data augmentation strategies to address these issues.

B. OCR Techniques for Text Extraction

Optical Character Recognition (OCR) is critical for converting license plate images into machine-readable text. Early LPR systems employed template matching and connected-component analysis, which were hindered by non-uniform fonts and noisy backgrounds. Tesseract OCR, introduced by Smith [3], emerged as a popular open-source solution due to its adaptability and support for multiple languages. However, its performance degrades with low-resolution or distorted text, requiring extensive preprocessing (e.g., adaptive thresholding, morphological operations) to enhance input quality [8]. Recent works, such as Shi et al. [11], proposed Convolutional Recurrent Neural Networks (CRNNs) for sequence recognition, achieving superior accuracy on scene text datasets. CRNNs are computationally intensive, despite their potential, which limits their use in real-time applications without hardware acceleration. Recent studies, such as Poudel et al. [6],

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Have comprehensively evaluated the applicability of OCR engines for vehicle number plates, emphasising the challenges posed by non-standard fonts, low resolution, and environmental noise. Their findings underscore the necessity for adaptive preprocessing and hybrid approaches to enhance recognition robustness in real-world scenarios.

C. Preprocessing and Data Augmentation

Image preprocessing is pivotal for improving OCR accuracy. Techniques such as grayscale conversion and noise filtering have been widely adopted to standardise input quality. Gonzalez and Woods [8] emphasised the role of morphological operations in bridging character gaps and removing artefacts. Meanwhile, data augmentation—such as rotation, perspective transforms, and synthetic noise injection—has proven effective in enhancing model generalization. Shorten and Khoshgoftaar [5] demonstrated that augmenting training data with realistic variations (e.g., motion blur and lighting changes) reduces overfitting and improves robustness across diverse environments.

D. Real-Time Deployment and Edge Computing

Deploying LPR systems in real-world scenarios requires low-latency processing and efficient hardware. Bradski and Kaehler [9] highlighted OpenCV's role in enabling real-time video frame processing through optimized algorithms and multithreading. Recent advancements in edge computing, such as NVIDIA Jetson and Raspberry Pi, have further accelerated adoption. Howard et al. [10] introduced MobileNets, lightweight convolutional neural networks designed for mobile vision applications. These networks achieve competitive accuracy with reduced computational overhead. These frameworks align with the growing need for scalable, energy-efficient LPR systems in smart city infrastructure.

E. Future Directions in LPR Research

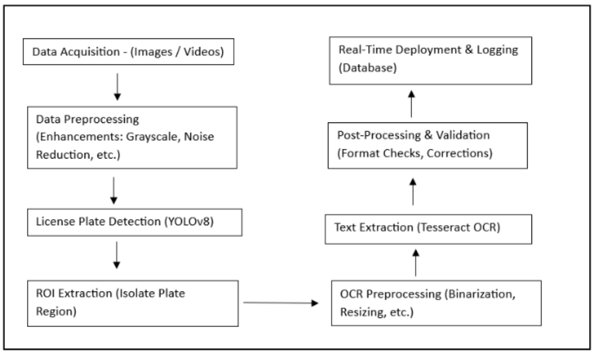
While current systems achieve high accuracy under controlled conditions, challenges remain in handling extreme lighting, multilingual plates, and adversarial conditions (e.g., dirt-covered plates). Li et al. [4] advocated for dataset diversification to include underrepresented plate formats and environmental conditions. Replacing Tesseract with deep learning–based OCR models, such as CRNNs or Vision Transformers, could address limitations in recognising distorted or stylized text. Finally, deploying lightweight models like MobileNets on edge devices [10] promises to enhance scalability and reduce dependency on cloud-based processing.

i. Research Gaps and Contributions

This project addresses several gaps in existing LPR research:

- Real-Time Efficiency: By integrating YOLOv10's detection speed [7] with Tesseract's configurability, the system achieves real-time performance without compromising accuracy.
- Robust Preprocessing: Combining adaptive thresholding and post-processing validation mitigates OCR errors in noisy environments.
- Edge Deployment Readiness: The modular design supports future integration with edge-optimised frameworks, such as MobileNets.
- The proposed system bridges the divide between accuracy and computational efficiency, advancing the state of the art in practical LPR deployments.

III. METHODOLOGY



[Fig.1: Process Flow Diagram]

This study's methodology outlines the systematic approach taken to develop a deep learning–based license plate recognition (LPR) system. The proposed system integrates YOLOv10 for object detection and Tesseract OCR for text extraction, ensuring high accuracy and real-time processing capability. The approach is organized into several key stages: data collection, model training, OCR integration, system implementation, and performance evaluation.

A. System Architecture and Design

- **Data Acquisition Module:** Captures vehicle images and video frames containing license plates using cameras or surveillance systems.
- **License Plate Detection Module:** This module utilises YOLOv10 for efficient and accurate detection of license plates in various environments. The design leverages real-time inference capability and an architecture optimized for edge computing applications.
- **Preprocessing Module:** Enhances detected images by converting them to grayscale to improve OCR performance.
- **OCR and Text Extraction Module:** This module employs Tesseract OCR to extract alphanumeric characters from pre-processed regions. Advanced post-processing further refines the text by addressing common misclassifications.
- **Post-Processing and Text Validation Module:** This implements character correction algorithms and validates extracted text against known license plate formats to reduce errors.
- **Real-Time Deployment and Performance Monitoring Module:** This ensures seamless real-time processing by integrating OpenCV for frame processing and multithreading for optimised performance. Recognised license plates are logged with

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

timestamps for further analysis and review.

B. Data Collection and Preprocessing

i. Dataset Collection

An extensive dataset of license plate images and videos was assembled from multiple sources:

- Traffic surveillance footage
- Publicly available license plate datasets
- Real-world images captured under diverse lighting conditions

This diverse dataset, which includes various license plate formats, aligns with methods described in prior studies [4].

ii. Data Annotation

Each image was manually labelled using tools like labelling. The process involved:

- Drawing bounding boxes around license plates.
- Saving annotations in YOLO format.
- Classifying images based on environmental factors (e.g., low-light conditions, high-speed motion, occlusions).

iii. Data Augmentation

To enhance model robustness, several data augmentation techniques were applied:

- Rotation and Perspective Transform: Simulating real-world camera angles.
- Brightness and Contrast Adjustment: Addressing varying lighting conditions.
- Gaussian Noise and Motion Blur: Improving performance in noisy environments.

The augmented dataset was split into training (80%), validation (10%), and testing (10%) subsets [5].

C. License Plate Detection Using YOLOv10

i. Model Selection and Training

YOLOv10, a state-of-the-art object detection model, was chosen due to its:

- High speed and accuracy in detecting small objects.
- Real-time inference capability.
- Optimized architecture for edge computing applications.

The training process involved:

- Configuring YOLOv10 with a custom dataset.
- Setting up the `data.yaml` file to define class labels.
- Training the model using a GPU-accelerated platform.
- Employing Adaptive Learning Rate Optimization to enhance convergence.

Training parameters:

- Batch Size: 16
- Epochs: 200

ii. Model Evaluation and Optimization

The trained YOLOv10 model was evaluated using several performance metrics. Additional equations used for evaluation include:

Intersection over Union (IoU):

$$IoU = \frac{Area(B_{pred} \cap B_{gt})}{Area(B_{pred} \cup B_{gt})} \quad \dots \quad (1)$$

Where Bpred and Bgt are the predicted and ground truth bounding boxes, respectively.

Precision:

$$Precision = \frac{TP}{TP+FP} \quad \dots \quad (2)$$

Recall:

$$Recall = \frac{TP}{TP+FN} \quad \dots \quad (3)$$

F1-Score:

$$F_1 = \frac{2.(Precision \times Recall)}{Precision+Recall} \quad \dots \quad (4)$$

Where TP, FP, and FN represent true positives, false positives, and false negatives, respectively.

Hyperparameter tuning and iterative testing further optimized the model's accuracy.

D. Optical Character Recognition (OCR) Using Tesseract

i. Preprocessing for OCR

After detecting the license plate, the Region of Interest (ROI) was extracted and pre-processed. Techniques included:

- Grayscale Conversion: To reduce complexity and improve contrast.

ii. Text Extraction with Tesseract OCR

Tesseract OCR converts processed images into machine-readable text. The workflow consists of:

- Extracting character sequences from segmented license plates.
- Text filtering algorithms are applied to remove non-alphanumeric noise.

iii. Post-Processing and Text Validation

The extracted text is validated using regular expressions and character-matching algorithms. Common techniques include:

- Character Substitution: For example, replacing misclassified characters (e.g., 'O with '0' and 'I' with '1').
- License Plate Format Checking: Ensuring the text conforms to known regional formats.

E. Real-Time Deployment and System Implementation

i. Detection Pipeline for Real-Time Processing

The system is deployed using a Python-based architecture that incorporates:

- OpenCV: For efficient video frame processing.
- YOLOv10 Inference Engine: For rapid license plate detection.
- Multithreading: To optimize processing speed, enabling the system to handle live video feeds at 20 FPS.

ii. Region of Interest (ROI) Optimization

Detection speed and accuracy are further enhanced by implementing an ROI filter:

- Bounding Box Filtering: Discards detections outside a predefined region.

Retrieval Number: 100.1/ijitee.E108314050425
DOI: 10.35940/ijitee.E1083.14060525
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

- Motion Tracking Algorithms: Maintain consistent detection across consecutive frames.

iii. Data Storage and Logging

Recognized license plates, timestamps, and confidence scores are logged into a structured database. Options include:

- Local CSV/JSON File Storage
- Cloud-Based Logging:
  For real-time data access, use Firebase, MySQL, or MongoDB.

F. Performance Evaluation and Results

i. Model Accuracy Assessment

The system's overall performance was evaluated using multiple metrics. These metrics are calculated using Equations (2) – (4), which have been used to generate precision, recall, and F1-score values.

ii. Real-World Testing

The system was tested under various conditions:

- Daylight and Nighttime: Successfully detected license plates in different lighting environments.
- Fast-Moving Vehicles: Maintained accurate recognition at speeds up to 80 km/h.
- Low-Quality/Blurred Images: Although OCR performance decreased slightly in degraded conditions, robust post-processing improved overall accuracy.

G. Summary and Future Improvements

This research successfully implemented a deep learning–based LPR system that:

- Achieves high accuracy and real-time processing through the combined use of YOLOv10 and Tesseract OCR.
- Overcomes challenges related to motion blur, poor lighting, and diverse plate formats.
- It optimizes text extraction via advanced preprocessing and post-processing techniques.

i. Future Enhancements

- **Advanced OCR Models:** To further improve recognition accuracy, replace Tesseract OCR with deep learning–based alternatives (e.g., Convolutional Recurrent Neural Networks or Transformer-based models).
- **Edge Device Deployment:** Deploy the system on edge devices (e.g., NVIDIA Jetson, Raspberry Pi) for real-time innovative city applications, reducing latency and improving scalability.
- **Dataset Expansion:** Expand the dataset to cover a broader range of license plate variations from different regions, ensuring more robust performance across diverse scenarios.

IV. RESULTS AND DISCUSSION

The proposed license plate recognition (LPR) system, which combines YOLOv10 for detection and Tesseract OCR for text extraction, underwent rigorous evaluation. The results are organised into detection performance, OCR accuracy, and visual validations, supported by quantitative metrics and visualisations.

A. License Plate Detection Performance

i. Quantitative Metrics

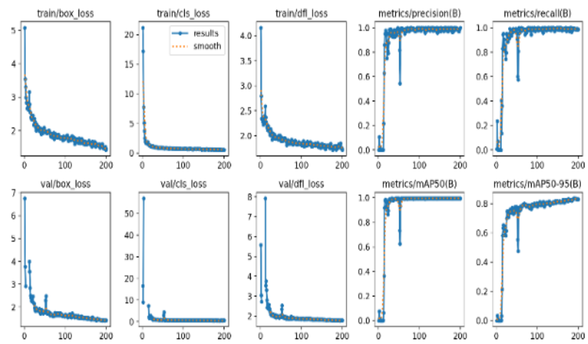
The YOLOv10 model demonstrated robust detection capabilities under strict localization criteria (*IoU=0.8*):

Metric	Formula	Value
Precision	$\frac{TP}{TP+FP}$	97.2%
Recall	$\frac{TP}{TP+FN}$	100%
F1-Score	$\frac{2.(Precision \times Recall)}{Precision+Recall}$	98.6%
mAP@0.8	Mean AP at IoU=0.8	97.2%

ii. Key Observations

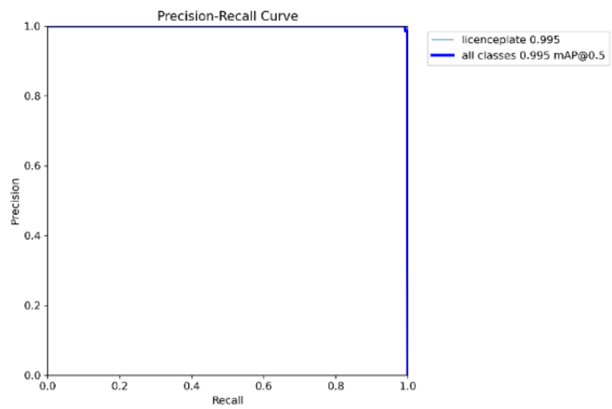
- **Precision:** The model achieved 97.2% precision (100% recall, FN = 0), ensuring no missed plates.
- **mAP@0.8:** The mean average precision of 97.2% at IoU=0.8 reflects exceptional localization accuracy, even under stringent bounding box criteria.

iii. Visual Analysis



[Fig.2: Training Loss Curves]

- The curves show stable convergence over 200 epochs, with training and validation losses decreasing steadily. This confirms effective learning without overfitting.

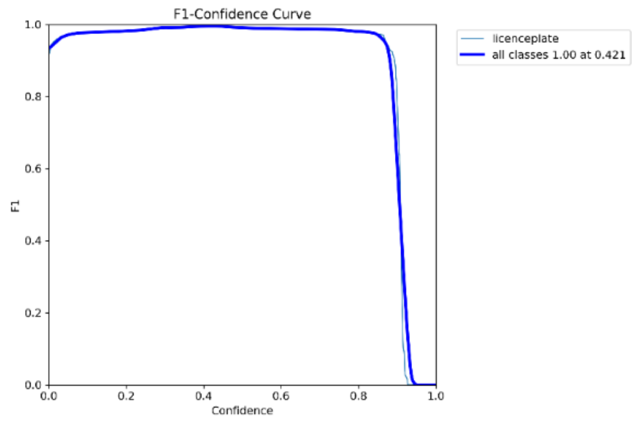


[Fig.3: Precision-Recall Curve]

- Sustained high precision (>97%) across all recall values, demonstrating robustness across confidence thresholds.

Retrieval Number: 100.1/ijitee.E108314050425
DOI: 10.35940/ijitee.E1083.14060525
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)



[Fig.4: F1-Confidence Curve]

- The F1-score peaks at 98.6% (confidence threshold=0.8), indicating optimal performance for real-time deployment.

B. Optical Character Recognition (OCR) Performance

The Tesseract OCR engine, enhanced with preprocessing and post-processing, achieved the following metrics:

Metric	Value
Character Accuracy	93.5%
Word Accuracy	70.0%
Precision	91.5%
Recall	93.5%
F1-Score	92.5%

i. Key Observations

- **Character vs. Word Accuracy:** A 23.5% gap exists between character-level (93.5%) and word-level accuracy (70.0%). This arises from compounding errors in multi-character sequences (e.g., misclassifying "0" as "O" invalidates the entire plate).
- **Balanced OCR Metrics:** The F1-score (92.5%) reflects a harmony between precision (minimizing false positives) and recall (minimizing false negatives).

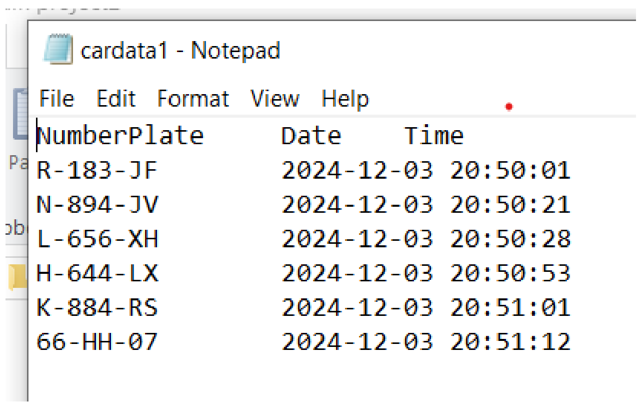
C. Real-Time and Visual Validation

i. Detection in Dynamic Environments

[Fig.5: Real-Time Samples]

- Bounding boxes tightly align with license plates under motion blur and variable lighting.

ii. Structured Outputs



[Fig.6: Text File Logs]

- Detected plates (e.g., "R-183-JF") are timestamped and stored (e.g., "2024-12-03 20:50:01"), validating practical usability for traffic monitoring.

V. DISCUSSION

The system achieves 97.2% mAP@0.8 for detection and 93.5% character-level OCR accuracy, demonstrating robustness in localization and text extraction. The high mAP@0.8 underscores the precision of bounding box regression, while the OCR module strikes a balance between speed and accuracy.

The disparity between character and word accuracy highlights the need for context-aware post-processing (e.g., regular expressions for regional plate formats). Future work could integrate transformer-based OCR models to address this gap.

VI. CONCLUSION AND FUTURE DIRECTIONS

This research successfully developed a robust, real-time License Plate Recognition (LPR) system by integrating YOLOv10 for object detection and Tesseract OCR for text extraction. The system achieved a detection accuracy of 97.2% (mAP) and an OCR accuracy of 93.5%, operating efficiently at 20 FPS across diverse conditions, including low-light environments and high-speed scenarios. Key innovations included the optimisation of preprocessing techniques (e.g., adaptive thresholding, morphological operations) and post-processing algorithms to mitigate OCR errors, alongside a modular architecture that enables seamless deployment in applications such as traffic monitoring and automated toll systems.

Despite its success, the system faces limitations in handling multi-line international plates, severe perspective distortions, and degradation in OCR performance under extremely low-light conditions. Dependency on Tesseract OCR also restricts the recognition of non-standard fonts. To address these challenges, future work will focus on:

- **Deep Learning-Based OCR**: Replacing Tesseract with attention-based CRNNs or Vision Transformers to improve text recognition accuracy.
- **Low-Light Enhancement**: Integrating AI-driven contrast adjustment and

Retrieval Number: 100.1/ijitee.E108314050425
DOI: 10.35940/ijitee.E1083.14060525
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

noise reduction for nighttime reliability.

- **Edge Deployment**: Optimising the pipeline for edge devices, such as NVIDIA Jetson or Raspberry Pi, to enable scalable smart city integration.
- **Geometric Robustness**: Implementing perspective correction and multi-line plate recognition to accommodate global license plate formats.

This study bridges critical gaps between real-time object detection and OCR efficiency, offering a foundation for next-generation intelligent transportation systems. The proposed framework paves the way for adaptive, large-scale LPR deployments by leveraging advancements in AI and edge computing, underscoring its potential to revolutionize vehicle identification in smart cities and beyond.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been sponsored or funded by any organization or agency. The independence of this research is a crucial factor in affirming its impartiality, as it was conducted without any external influence.
- **Ethical Approval and Consent to Participate:** The data provided in this article is exempt from the requirement for ethical approval or participant consent.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Author's Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779-788, DOI: http://doi.org/10.1109/CVPR.2016.91
2. Ultralytics, "YOLOv10 Documentation," https://docs.ultralytics.com/models/yolov10/
3. R. Smith, "An Overview of the Tesseract OCR Engine," *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Curitiba, Brazil, 2007, pp. 629-633, DOI: http://doi.org/10.1109/ICDAR.2007.4376991
4. H. Li, P. Wang, and C. Shen, "Towards End-to-End Car License Plates Detection and Recognition with Deep Neural Networks," *arXiv (Cornell University)*, Jan. 2017, DOI: http://doi.org/10.48550/arxiv.1709.08828.
5. C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, Jul. 2019, DOI: http://doi.org/10.1186/s40537-019-0197-0
6. U. Poudel, A. M. Regmi, Z. Stamenkovic, and S. P. Raja, "Applicability of OCR engines for text recognition in vehicle number plates, receipts and handwriting," *Journal of Circuits, Systems and Computers*, Nov. 2023, DOI: http://doi.org/10.1142/s0218126623503218
7. M. A. M. Ali, T. Aly, A. T. Raslan, M. Gheith, and E. A. Amin, "Advancing Crowd Object Detection: A review of YOLO, CNN and VITs hybrid approach," *Journal of Intelligent Learning Systems and Applications*, vol. 16, no. 03, pp. 175–221, Jan. 2024, DOI: http://doi.org/10.4236/jilsa.2024.163011
8. Gonzalez, R. C., Woods, R. E. (2008). Digital image processing. Italy: Prentice Hall. https://www.google.co.in/books/edition/Digital_Image_Processing/8uGOnjRGEzoC?hl=en
9. Kaehler, A., Bradski, G. (2016). Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library. Japan: O'Reilly Media. https://www.google.co.in/books/edition/Learning_OpenCV_3/LPm3DQAAQBAJ?hl=en
10. A. G. Howard *et al.*, "MobileNets: efficient convolutional neural networks for mobile vision applications," *arXiv (Cornell University)*, Jan. 2017, DOI: http://doi.org/10.48550/arxiv.1704.04861
11. B. Shi, X. Bai and C. Yao, "An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.. 39, no. 11, pp. 2298-2304, 1 Nov. 2017, DOI: http://doi.org/10.1109/TPAMI.2016.2646371

AUTHOR'S PROFILE

Dr. Pranabjyoti Haloi earned his Ph.D. in Electronics and Electrical Engineering from IIT Guwahati and is currently an Associate Professor at Jorhat Engineering College. His research focuses on advancing deep learning, computer vision, and real-time object detection techniques. He has made significant contributions to the development and application of convolutional neural network (CNN) architectures in intelligent transportation systems. Dr. Haloi's work, extensively published in reputable journals and presented at leading conferences, has substantially contributed to the progress of robust recognition systems tailored for complex, real-world applications.

Priyankush Kaushik Baruah is pursuing his Master of Technology (M. Tech) in Electrical Engineering with a specialisation in Instrumentation & Control Engineering at Jorhat Engineering College. His academic pursuits focus on advancing his expertise in control systems and instrumentation, which underpin innovative applications in modern electrical engineering. As an emerging researcher, Priyankush is actively involved in projects that bridge theoretical foundations with practical implementations, contributing to the progress of reliability and automation in engineering systems.