International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

Plate-Plus Tracker

Mrs. Mohini Avatade[1], Sanket Thorat[2], Shravani Kadlag[3], Aditya Sawant[4], Ansh Badagandi[5]
[1]Associate Professor, Dept. of Computer Engineering, DYPIEMR, Pune, India
[2,3,4,5]UG Scholar, Dept. of Computer Engineering, DYPIEMR, Pune, India
Emails: monu13.engg@gmail.com[1], thoratsanket2222@gmail.com[2], kadlagshravani2004@gmail.com[3], adityasawant3010@gmail.com[4], anshbadagandi540@gmail.com[5]

Abstract
In today's fast-moving world, keeping our roads safe and managing traffic effectively has become really important. This research paper, titled "Plate-Plus Tracker," aims to transform how we track vehicles by combining video analysis with smart image processing technology. The system records video of vehicles traveling along specific roads and stores all footage safely in a database. Using computer vision and image recognition methods, the system can automatically identify vehicles on the road and pull out license plate information from the recorded videos. On top of that, it tracks the path a vehicle has taken using Google Maps. Users can search for any vehicle using the stored data and quickly find out whether that vehicle appears in any footage. This paper demonstrates how automation can help build a more reliable and safer vehicle tracking solution. Index Terms—Plate-Plus Tracker, Deep Learning, YOLOv5, Computer Vision, OCR, Vehicle Surveillance, Intelligent Transport Systems

Keywords: *Plate-Plus Tracker, Deep Learning, YOLOv5, Computer Vision, OCR, Vehicle Surveillance, Intelligent Trans-portation Systems*

1. Introduction

With the continuous growth of our cities, such problems as car theft, vehicle misuse and the inability to find your vehicle after an accident or emergency are growing more common. This is something most of us experience: the anger about not knowing where our car is, when it was stolen, when it was hurt or when we just lost track of it by simply leaving it somewhere we cannot recall. In the mean time, police can hardly keep an eye on suspicious cars when checked manually since it will take far too much time and effort to check everything manually. Such real-life issues emphasize the true necessity of a system, which would allow locating, identifying and monitoring vehicles quickly and precisely. The example of automatic license plate recognition (ALPR) is one of the essential innovations of intelligent traffic control, police activity, and urban security. The conventional approaches used in these kinds of ALPR systems, which are mainly based on image processing, are not always need. They do not perform well in variable lighting conditions, incorporating different perspectives and in various weather conditions. Another factor is the huge variety of license plates. Other problems are the ones we find moving vehicles as motion blur and camera obstructions. All this contributes to the higher load of complications for the accuracy of these systems to be judged, which stresses the need for more robust and intelligent systems. Deep learning techniques are to address them. have been popularized in this region. The current ALPR systems are based on convolutional neural networks. YOLO-based methods provide a very nice tradeoff among. delays and accuracy to recognize plates in real-time. The majority of the existing studies are inclined to focus on making. individual parts of an ALPR system better rather than to work at bettering the whole system collectively. A considerable blank exists as regards to construction. a complete ALPR solution where all the advanced pieces work together seamlessly. This paper demonstrates an end-to-end vehicle surveillance system that was constructed to bridge the existent gaps between high-performing models and an application that is ready to deploy. Our deep learning model allows us to develop a system, which encourages the monitoring of vehicles. The system will be in charge of video footage, scan vehicles and their plate number, OCR the characters and display

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

everything in a convenient web interface. The paper is structured as follows: Section II covers the literature review; Section III describes how we built it; Section shares our results and how well it performed; and Section wraps things up with thoughts on where this research could go next.

2. Literature Review

With deep learning, computer vision, and OCR, work on ALPR has greatly advanced. Laroca and coworkers [1] presented a YOLO-based real-time ALPR with CNN-driven OCR, which achieves good accuracy and good performance on popular datasets, such as SSIG and UFPR. To go a step further, Li et al. [2] designed a single pipeline, which performs plate detections and recognition simultaneously, eliminating modular dependencies and enhancing performance in smart city applications. Additional enhancements have narrowed down to real-time elasticity. A YOLOv5 and EasyOCR-based system demon-strated by Salsabila and Sriani [3] in their study was able to achieve 100 percent detection rate and 74.7 percent character accuracy on live 4K video demonstrating what a hybrid system can achieve. Equally, Gaqiwo [4] combined YOLO with Mask R-CNN to read plates, which are tilted and angled and read plates successfully with the help of segmentation. To this, Wang et al. [5] developed a segmentation-free CNN, which achieved 97 percent recognition and high FPS, rendering it feasible to use phones and cars-mounted systems. Several studies have been conducted on the enhancement of images to read plates with low resolutions or in harsh environments. Al-Halawani et al. [6] used diffusion-based super-resolution to scale up plate images, getting better recognition on lower-quality inputs. Thalpiyal et al. [7] further improved OCR performance through the detection of YOLOv5 and the input through Tesseract and EasyOCR with detection and recognition accuracy of 85.8 percent and 84.7 percent, respectively. Similarly, Shi and Zhao [8] proposed a channel-attention YOLOv5+GRU+CTC, with an accuracy of 98.94 percent on varying datasets formats. Full CNN pipelines have also been found to be reliable in case of various lighting, fonts and backgrounds. Masood et al. [9] constructed such a model to achieve a constant recognition between different conditions though a separate group of researchers headed by Zhang et al. [10] achieved 96.9 percent accuracy in terms of detecting and subsequently sorting layouts in eight datasets. More recent diffusion-based algorithms such as DiffPlate by Nascimento et al. [11] have achieved 2637 percent improvement in PSNR on plates of low and has an interface in the form of a map, which enables us to view the position of the cars and their movements in a very easy to read and understand interactive format.

3. Methodology

The approach to our system is based on the design of the system and the way it works. We begin by exploring the system background: a three-layer architecture, where the user interface, logic and data storage are separated to allow the system to be more scaled and to provide easier maintenance of the services. Next we describe the System Workflow, containing all the steps of the pipeline of processing the data, starting with the video input and finishing with final display of the result, and how data passes around and is processed in this architecture.

3.1.System Architecture

The Plate-Plus Tracker runs on three connected layers — one that users interact with, one that handles all the processing logic, and one that stores the data. Every action a user takes on the front end gets handled by a specific manager working behind the scenes. The figure below shows this design and how the different tiers communicate. The React-based Presentation Tier contains user-facing elements, such as Signup, Login, About, Search, Result, Past History and Map, and connects to the corresponding managers in the Application Tier via API calls. The Application Tier is Python based and deals with all the business logic and business processing via specific managers. The Session Manager processes the user requests, establishes sessions, and delegates big jobs such as YOLOv5 frame detection and OCR to a background queue to ensure all the processes run well. A Background Worker executes the detection pipeline by processing the frames sequentially, saving the in-progress output in the plates collection, and launching post-processing modules that clean, verify and eliminate duplicates to the final output, which is then saved to the plate detections collection. New user info is stored in the users collection by

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

Signup Manager and verified by the Login Manager during a sign-in. Search History Manager retrieves past searches of a user in the database. When a human being enters a vehicle registration number, the Search Manager searches the indexed plate detections collection to locate resembling information. This information is then formatted into an easy to read form by the Result Manager and the location information processed by the Map Manager in a manner that it can be plotted correctly on the interactive map.As shown in Figure 1 Tier Architecture Diagram of Plate-Plus Tracker.

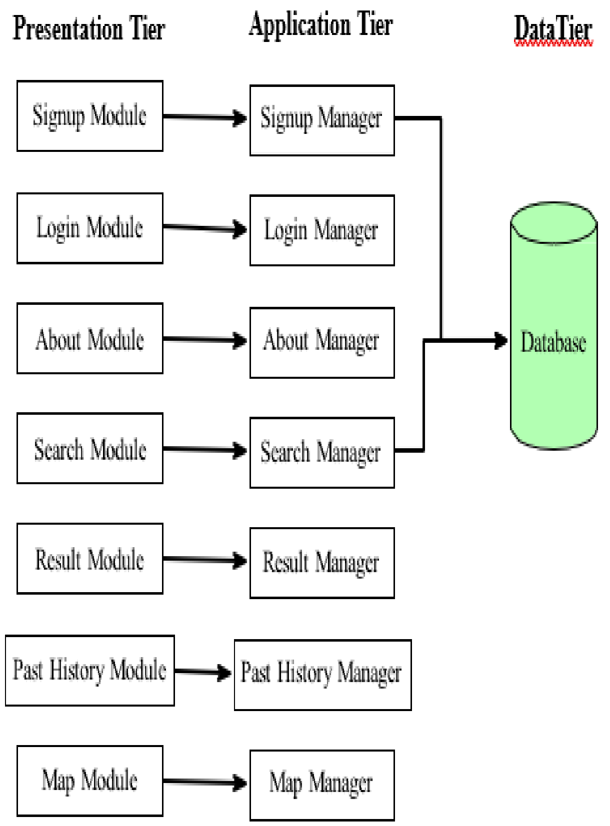

Figure 1 Tier Architecture Diagram of Plate-Plus Tracker

Layered System Architecture showing Presentation, Application, and Data Layers

3.2.System Workflow

The proposed Plate-Plus Tracker is designed to work through a modular process with six stages: (1) Video Fetching and Session Management, (2) Frame Processing (3) Plate Detection, (4) Plate Post-Processing and Deduplication, (5) Data Storage, (6) Search and Result Visualization.

- Video Fetching and Session Management: In this step, the system selects a stored video of a surveillance in the internal database, automatically. Metadata (e.g., camera ID, timestamp, video number) is associated with the surveillance video, and these are stored within the database, which makes it possible to access the video record with this ID. The system generates a Session ID of the chosen surveillance video that will be used to ensure the connection between the video record and its metadata, and any outcome to be produced during the surveillance process.
- Frame Processing: Once a session is established the video is passed to the frame processing plug-in. It processes frames in the video. The YOLOv5 model (already trained) searches for license plates and determines their location in each frame [6]. A region of interest (ROI) of the size of the frame (plate) is extracted. This image is then given to a hybrid optical character recognizer (OCR) for recognition. All the calculations are done in a separate thread to make sure the UI won't be blocked when the calculation is running.
- Plate Detection: In plate detection, each frame is passed on to the pre-trained YOLOv5 network which will detect the presence of license plates. They are cropped and delivered

to a hybrid Optical Character Recognition (OCR) system to successfully read the plate as text (with letters and digits). This step will output the plate image (cropped), recognised text and the metadata of the surrounding frames.As shown in Figure 2 System Workflow from Video Fetching to Visualization, Figure 3. Per-frame Processing and Recognition Pipeline

IRJAEH
International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

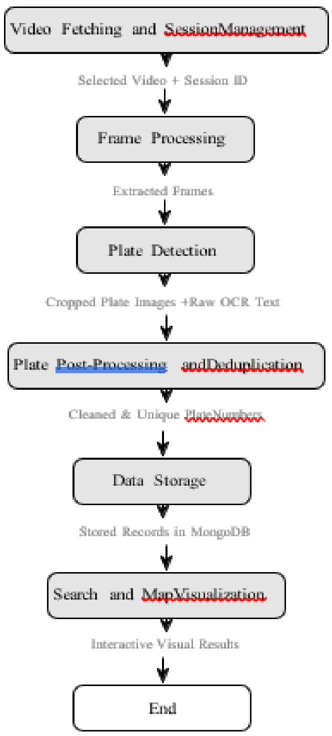

Figure 2 **System Workflow from Video Fetching to Visualization**

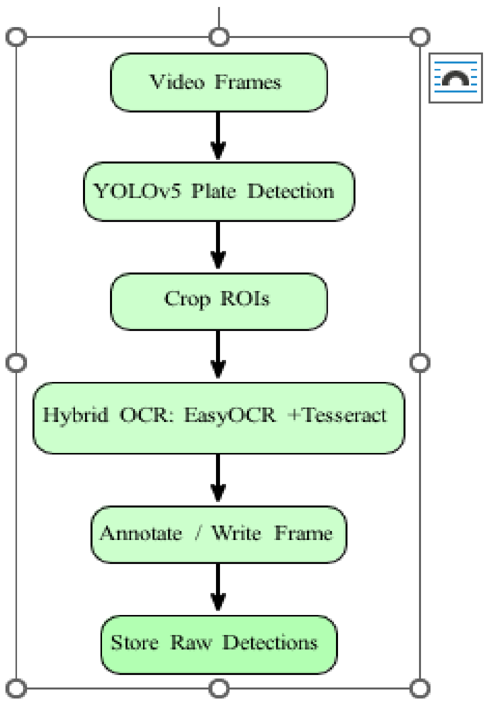

Figure 3 **Per-frame Processing and Recognition Pipeline**

- Cleaning up/Improving OCR results: and Post-processing of OCR results: this step. Images of the cropped plates are enhanced (also known as image pre-processing) to improve the text using grayscale transformation, CLAHE, and noise removal method. The text based on the above method is normalised to allow OCR mistakes, such as, O to 0 or I to 1. Then the text is tested against the desired format of the licence plates using regular expressions. Levenshtein distance is used to remove duplicates and select the best detection to be able to process the same number plate in the following frames.As shown in Figure 4 Cleaning, Validation, and Deduplication Flowchart

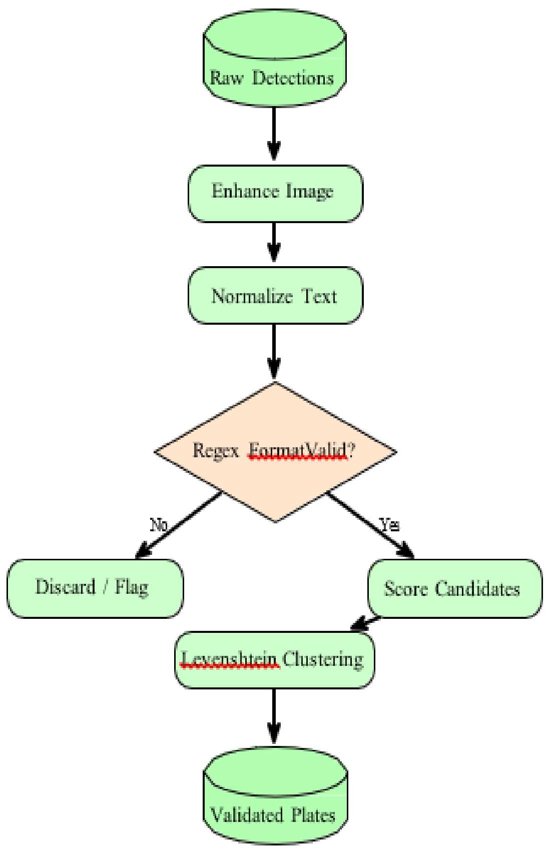

Figure 4 **Cleaning, Validation, and Deduplication Flowchart**

- Data Storage: All the processed information will enter a MongoDB data base during Data Storage. Each session has four collections;

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

video metadata, raw OCR products, plate detects and error detects as shown in Figure 5 MongoDB Collections and Relations

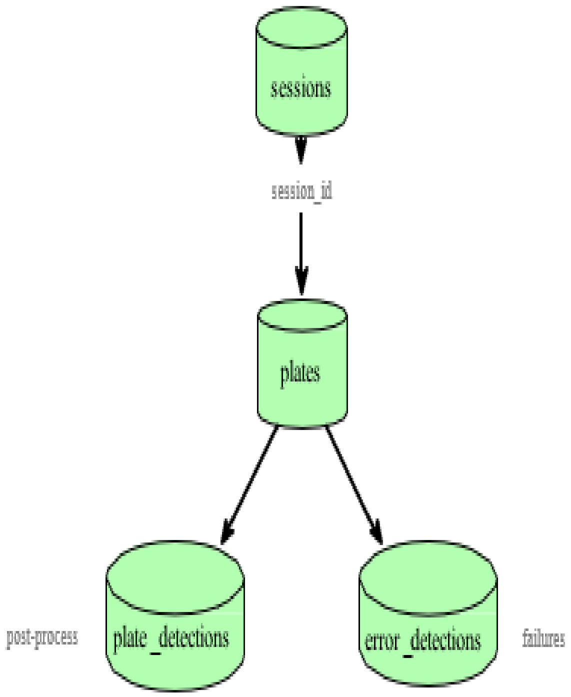

Figure 5 **MongoDB Collections and Relations**

Result Visualization: The last stage allows users to search and view the saurified detections stored in an interactive dashboard. The search engine will transform the search query of the user and then it will return all the corresponding entries from the database that best match the given information. The dashboard will present the results in several ways: a Map View that tries to show the geographical location of the detections, a Statistics View where details of the trends are represented over time and location, an Image View which shows cropped number plate images and an automatically created PDF Report that provides a summary of the session. Figure 6 Frontend Components and Backend APIs, Figure 7 High-level Architecture from Ingestion to Analysis.

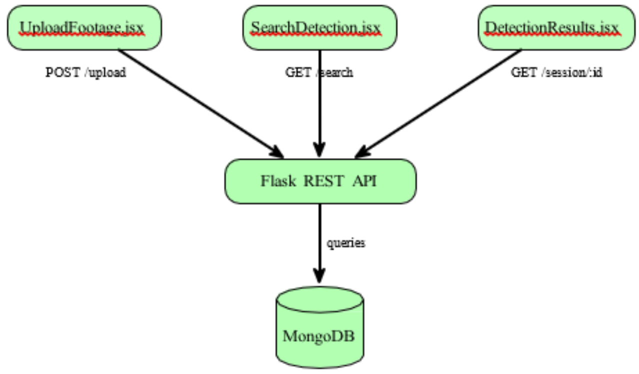

Figure 6 **Frontend Components and Backend APIs**

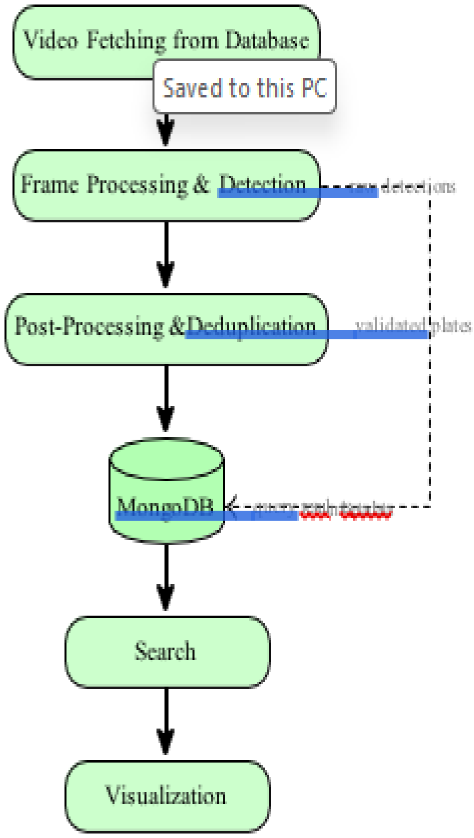

Figure 7 **High-level Architecture from Ingestion to Analysis**

4. Comparative Analysis

To test the level of our system performance, we conducted a set of experiments based on a set of traffic surveillance video of real traffic. The data

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

collection was done in different places such as the streets and highways in the city.

4.1.Detection and Recognition Accuracy

The accuracy of our system over various states of pipeline was measured and compared to hand-labeled ground truth. A summary of the main performance figures is presented in table The ”Raw OCR” column displays the accuracy with just the YOLOv5 detector and the raw OCR output and the second line, Full Pipeline, gives the results after running our entire post-processing and validation pipeline.As shown in Table 1 Detection and Recognition Accuracy Comparison.

Table 1 Detection and Recognition Accuracy Comparison

Metric	Raw YOLOv5	Raw OCR	VechileTracke
Plate Detection Rate	94.2%	-	94.2%
Character Rec. Rate	-	82.7%	91.5%
Full Plate Rec.	-	76.3%	88.9%
False Positive Rate	3.8%	8.2%	2.1%

The results clearly show the big impact of our post-processing pipeline. It raised the character recognition rate by 8.8 percentage points (from 82.7% to 91.5%) and the critical full plate recognition rate by 12.6 points (from 76.3% to 88.9%). Additionally, the pipeline cut the false positive rate by over 74%, dropping it from 8.2% to a much more acceptable 2.1%. This improvement is shown in Figure 9 Improvement in recognition accuracy after applying the full post-processing pipeline

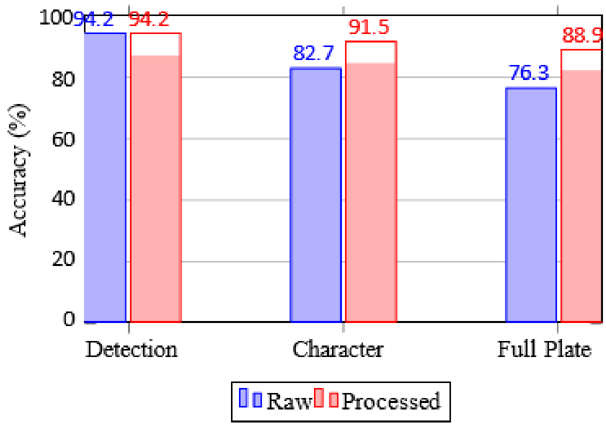

Figure 9 Improvement in recognition accuracy after applying the full post-processing pipeline

4.2.Processing Performance

We also measured the processing speed of the system on various hardware configurations to determine its suitability in deploying the system in different situations. The average processing time per frame (in milliseconds) of the main computational tasks is presented in Table II. We tested them on a regular multi-core CPU, a middle-end NVIDIA RTX 2080 graphics card, and a top-end NVIDIA RTX 3090 graphics card.As shown in Table 2 Processing Performance (Ms/Frame).

Table 2 Processing Performance (Ms/Frame)

Component	CPU Only	GPU (RTX 2080)	GPU (RTX 3090)
YOLOv5 Detection	125.3	18.7	9.2
OCR (per plate)	342.1	78.5	42.3
Post-processing	56.8	56.8	56.8
Total (avg.)	524.2	154.0	108.3

As you’d expect, GPU acceleration makes a huge difference in performance. With an RTX 3090, the system can handle a frame in roughly 108.3 ms, which translates to over 9 frames per second (FPS). On an RTX 2080, performance sits around 6.5 FPS. This speed is enough for near-real-time processing of

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

pre-recorded footage, which is the main use case we designed

4.3.Comparison with Existing Systems

Please note that to contextualise our work, we compare our framework to the tasks and techniques often seen in the literature that is referenced. While many other research provides effective solutions to some of the ALPR problems, our system focuses on combining these features into a single, integrated system. A qualitative comparison is presented in table 3 Figure 10 Comparison of processing performance across different hardware configurations.

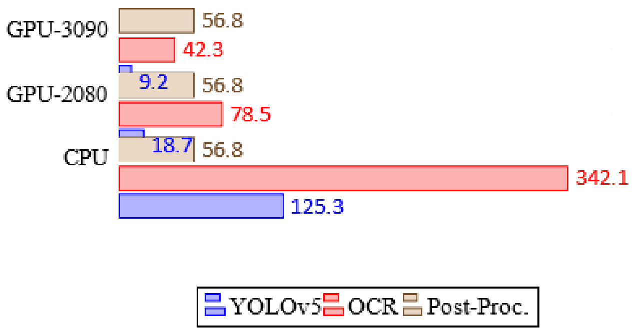

Figure 10 Comparison of processing performance across different hardware configurations

Table 3 Comparison of Proposed System with Existing Works

Feature	Common Approaches in Literature	Our Proposed System
Detection	Traditional CV [3] or specific YOLO ver-sions like YOLOv3/v4 [4], [5].	Uses a modern YOLOv5 model [6] for a solid balance of speed and accuracy.
OCR	Often relies on a single engine like Tesseract [8] or a custom-trained CNN [7], [23].	Uses a hybrid OCR en-gine (EasyOCR + Tesseract) with multiple decoding strate-gies to maximize recognition chances.
Post-processing	Usually focuses on specific enhancements like character normal-ization [10] or format validation.	Implements a thorough multi-stage pipeline including image enhancement (CLAHE), text normalization, format valida-tion, and Levenshtein-based deduplication.
End-to-End	Many works focus on algorithm tweaks for detection/recognition	Provides a complete, user-ready framework from video upload and processing to intersis

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

Visualization	Not a focus results are usually shown in table or static imstage [2],[5],[21] ages.	**Features an interactive web dashboard with map-based tracking, route calculation, stactive data display and analystical charts and image galleries.**
Low-Light	Addressed with spe-cialized models like URetinex-Net as a sep-arate enhancement step [14].	Builds CLAHE directly into the post-processing pipeline to improve performance on low-contrast footage.

We see that the strength of our system does not lie simply in the quality of its components, but also in their ability to cooperatively interact. While there are experts for low-light improvement [14], particular OCRs [23] and so forth, we provide a robust allrounder solution. The paper provides the bridge from academia to those deployable surveillance systems that are accessible to everyone because it provides a fast detector, incorporates a hybrid OCR system and a powerful verification system into a simple to use web interface.

Conclusion

In this paper, we presented a complete end-to-end Plate-Plus Tracker System with automatic plate recognition and tracking. By integrating a state-of-the-art detector YOLOv5 with a hybrid OCR system and a robust post-processing, our system can be accurate in practice. Our test results confirm that our method has the desired effect and the complete pipeline boosts plate detection rates to 88.9 percent and a ten-fold reduction in false positives. Front-end interface and modularity of the system turn it into a tangible and a useful traffic control tool for the police. [15], [22].

While the current system works really well, there are several directions for future work that could make it even better and more reliable:

- Enhanced Low-Light Performance: We plan to extend the system to include sophisticated deep learning-based image enhancement methods such as URetinex-Net [14] to specifically address and enhance recognition performance in low-light conditions, including nighttime.
- Live Video Processing: The current system is designed for batch processing of video, but we will develop the capability to process live video streams from internet protocol (IP) cameras, for example, to support real-time video monitoring.
- Edge Deployment of Light Models: We will experiment with reducing and deploying smaller, more lightweight YOLO models like YOLOv8-s [12] or YOLOv11 [25] models on edge devices to allow for processing video data locally and reduce network lag.
- Vehicle Make, Model, and Colour Recognition: The system could be further enhanced to also recognise vehicle make, model and colour for more information for surveillance and crime scene investigations.
- Privacy-Preserving Mechanisms: Given the increasing ethical and privacy concerns, we

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

will investigate and incorporate techniques to anonymise data and protect privacy, such as face blurring and encryption techniques. Through these directions for research, we hope to continue to advance the state of the art of automatic surveillance to support safe and efficient smart cities.

References

[1]. S. Du, M. Ibrahim, M. Shehata, and W. Badawy, ”Automatic license plate recognition (ALPR): A state-of-the-art review,” IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 2, pp. 311–325, 2013.

[2]. Z. Xu et al., ”Towards end-to-end license plate detection and recognition: A large dataset and baseline,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 255–271.

[3]. C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and

[4]. E. Kayafas, ”A license plate-recognition algorithm for intelligent transportation system applications,” IEEE Trans. Intell. Transp. Syst., vol. 7, no. 3, pp. 377–392, 2006.

[5]. R. Laroca et al., ”A robust real-time automatic license plate recognition based on the YOLO detector,” in Proc. Int. Joint Conf. Neural Netw. (IJCNN), 2018, pp. 1–10.

[6]. H. Li, P. Wang, and C. Shen, ”Toward end-to-end car license plate detection and recognition with deep neural networks,” IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 3, pp. 1126–1136, 2019.

[7]. G. Jocher et al., ”ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference,” 2022. [Online]. Available: https://github.com/ultralytics/yolov5

[8]. S. M. Silva and C. R. Jung, ”License plate detection and recognition in unconstrained scenarios,” in Proc. European Conference on Computer Vision (ECCV), 2018, pp. 580–596.

[9]. R. Smith, ”An overview of the Tesseract OCR engine,” in Proc. Ninth International Conference on Document Analysis and Recognition (ICDAR), 2007, pp. 629–633.

[10]. J. Zhuang, S. Hou, Z. Wang, and Z.-J. Zha, ”Towards human-level license plate recognition,” in Proc. European Conference on Computer Vision (ECCV), 2018, pp. 306–321.

[11]. S. Qiao, Y. Zhu, X. Li, T. Liu, and B. Zhang, ”Research on improving the accuracy of license plate recognition based on deep learning,” in Proc. IEEE International Conference on Information and Automation (ICIA), 2020, pp. 1601–1606.

[12]. A. Sugiharto et al., ”Enhanced Automatic License Plate Detection and Recognition using CLAHE and YOLOv11 for Seat Belt Compliance Detection,” Engineering, Technology & Applied Science Research, vol. 15, no. 1, 2025.

[13]. B. Satya et al., ”Optimized YOLOv8 for Automatic License Plate Recognition on Resource Constrained Devices,” Engineering, Technology & Applied Science Research, 2025.

[14]. K. Hefnawy, A. Lila, E. Hemayed, and M. Elshenawy, ”A Robust License Plate Detection and Recognition Framework for Arabic Plates with Severe Tilt Angles,” International Journal of Advanced Computer Science and Applications, vol. 15, no. 2, 2024.

[15]. V. W. Saputra et al., ”Low Light Image Enhancement in License Plate Recognition Using URetinex-Net,” Procedia Computer Science, 2024.

[16]. P. Agrawal et al., ”Automatic License Plate Detection and Recognition us-ing Deep Learning and Image Processing,” CEUR Workshop Proceedings, vol. 3682, 2024.

[17]. C. Ma et al., ”A review of vehicle detection methods based on computer vision and deep learning,” Journal of Intelligent and Connected Vehicles, 2024.

[18]. ”Automatic Number Plate Recognition Using Advanced Imaging Technol-ogy and Machine Learning,” International Journal of Scientific Research in Science and Technology, vol. 11, no. 5, pp. 439–448, 2024.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 06 June 2026
Page No: 4539 - 4548
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0596

[19]. K. S. Ravichandran, ”Estimation of Automatic License Plate Recognition Using Deep Learning Algorithms,” Spectrum of Decision Making and Applications, vol. 2, no. 1, pp. 100–119, 2024.
[20]. ”Image Enhancement Technique Utilizing YOLO Model for Automatic Number Plate Recognition,” International Journal of Transport Develop-ment and Integration, 2024.
[21]. ”Superior Use of YOLOv8 to Enhance Car License Plates Recognition,” Revue d’Intelligence Artificielle, vol. 38, no. 1, 2023.
[22]. ”A Deep Learning Framework for Accurate Number Plate Recognition System,” Journal of Intelligent Systems and Internet of Things, 2025.
[23]. ”Real-Time Traffic Surveillance and Vehicle Speed Estimation Using Computer Vision,” Journal of Intelligent Systems and Engineering Management, 2025.
[24]. P. Patchigulla et al., ”License Plate Detection And Recognition System Using YOLOv8 And LPRNet,” International Journal of Creative Research Thoughts (IJCRT), vol. 13, no. 4, 2025.
[25]. ”Traffic Violation Detection Using Computer Vision Techniques,” Journal of Informatics and Vocational Education, 2024.
[26]. ”YOLO11 for ANPR: License Plate Recognition,” Ultralytics Blog, 2025.