Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.062826

ARTICLE

Mitigating Fuel Station Drive-Offs Using AI: YOLOv8 OCR and MOT History API for Detecting Fake and Altered Plates

Milinda Priyankara Bandara Gamawelagedara[1], Mian Usman Sattar[1] and Raza Hasan[2,*]

[1]College of Science and Engineering, University of Derby, Derby, DE22 1GB, UK
[2]Department of Science and Engineering, Southampton Solent University, Southampton, SO14 0YN, UK

*Corresponding Author: Raza Hasan. Email: raza.hasan@solent.ac.uk

Received: 28 December 2024; Accepted: 24 March 2025; Published: 19 May 2025

ABSTRACT: Fuel station drive-offs, wherein the drivers simply drive off without paying, are a major issue in the UK (United Kingdom) due to rising fuel costs and financial hardships. The phenomenon has increased greatly over the last few years, with reports indicating a substantial increase in such events in the major cities. Traditional prevention measures such as Avutec and Driveoffalert rely primarily on expensive infrastructure and blacklisted databases. Such systems typically involve costly camera installation and maintenance and are consequently out of the budget of small fuel stations. These conventional approaches also fall short regarding real-time recognition, particularly regarding first-time impostors using fictitious plates, which represent an increasingly significant proportion of such forgery. This research presents an AI (Artificial Intelligence)-driven detection system using the MOT (Ministry of Transport) History API (Application Programming Interface) to scan in real-time at gas stations to recognize and prevent such fraud. The system integrates various state-of-the-art technologies to offer a foolproof system. Using the latest YOLO (You Only Look Once) model to recognize number plates and EasyOCR (Optical Character Recognition) to recognize characters, the system correctly reads license plates in various environmental conditions like lighting, viewpoint, and weather conditions. This approach minimizes the utilization of expensive camera systems and employs cheaper ANPR (Automatic Number Plate Recognition) gear, availing existing installed surveillance cameras on filling stations. The system operates with a basic web-based application to notify operators of stolen vehicles in real-time, enabling them to react immediately. Real-world testing achieves 84% success with CCTV (Closed-Circuit Television) images, depicting its real-world applicability. The results indicate that the AI-driven solution offers a monumental leap compared to current practices, giving fuel stations a cost-effective and efficient means of reducing financial loss from drive-off incidents.

KEYWORDS: EasyOCR; MOT; ANPR; YOLO model; CNNs

1 Introduction

Detecting drivers who flee without paying is costly and time-consuming, creating ongoing challenges for fuel station owners and law enforcement. Conventional monitoring systems, such as 'Avutec' and 'Driveoffalert,' are blacklist-based and depend on costly infrastructure. These systems tend to identify only previously known offenders and do not support real-time detection. Additionally, hashed blacklist records prevent conventional cameras from detecting fake or altered plates, rendering these systems ineffective against first-time offenders [1,2].

The aim of this research is to introduce an original AI-based detection system, which utilizes the MOT History API called VES (Vehicle Enquiry Service) United Kingdom for real-time verification of vehicles at fuel stations. The developed system seeks to help identify and prevent fraudulent activities associated with forged plates. To ensure that, we incorporated the latest AI technologies, including the YOLO model for number plate detection and EasyOCR for character recognition, which were included in the developed systems. The design of the system paid considerable attention to ensuring that it is accurate and cost-effective in a variety of environmental scenarios typical for a fuel station. Notably, the system was developed with a user-friendly web interface that allows operators to respond in real-time to vehicles engaged in fraudulent activities [3–5].

1.1 Background of the Study

Fuel station drive-offs are a persistent issue in the UK, straining law enforcement resources and causing financial losses. Notably, increasing fuel prices exacerbate the impact on the economic state due to the attractive pecuniary advantage that can be gained by fuel theft. A report by the RAC Foundation stated that petrol and diesel thefts from garage forecourts increased by 77% in one year. As a result, the financial losses to fuel station owners are dramatic, frequently up to thousands of pounds per year. Resources invested by law enforcement are substantial, with almost a million hours being dedicated to fuel theft annually [6].

Traditional prevention systems are highly infrastructure-based. For instance, some systems use high-tech cameras and blacklisted databases to identify repeat offenders. To achieve this, the system regularly cross-references all the number plates captured by the camera with a number plate list in a central database. However, these systems are not effective since they are not real-time and are always reactive. In any case, such systems are ineffective against first-time offenders who usually have fake number plates for such purposes. Many offenders have adopted the use of non-existent or different plates, which are fake or altered [7].

Currently, such systems as "Avutec" or "Driveoffalert" rely on expensive infrastructure and databases of blacklisted customers and license plates. Therefore, they are not efficient in terms of detecting first-time offenders who commit fraud with the use of fake, substituted, or altered number plates. Moreover, these processes are not implemented in real-time, thus letting the fraudsters carry out their crimes and cause the station thousands of dollars in damages. Finally, the implementation of current solutions may vary depending not only on the quality of hardware but also on environmental parameters, including light and quality of the image. As a result, developing a system of cost-efficient, real-time, and environmentally resilient number plate recognition solutions is crucial in the context of the raised gaps [8–10].

1.2 Introduction Summary

Fuel station drive-offs in the UK, exacerbated by rising fuel prices and economic challenges, pose significant financial and operational burdens for station owners and law enforcement. Traditional systems like "Avutec" and "Driveoffalert" rely on costly infrastructure and blacklisted databases, failing to detect first-time offenders using forged plates in real time. This research proposes an AI-based solution leveraging the YOLOv8 model for number plate detection and EasyOCR for character recognition, integrated with the MOT History API for real-time vehicle verification. Tested in real-world conditions, the system achieved 84% accuracy in detecting fraudulent vehicles, offering a cost-effective, real-time, and environmentally resilient alternative to traditional methods. This approach aims to reduce economic losses and improve law enforcement efficiency, addressing the limitations of current systems.

2 Literature Review

2.1 Overview of Automatic Number Plate Recognition (ANPR) Systems

Automatic Number Plate Recognition systems have become an essential part of vehicle identification and enforcement technology in recent decades. These systems involve multiple processes, including image acquisition, license plate detection, character segmentation, and optical character recognition. Throughout the past few years, an increasing number of traffic control, toll collection, and parking management systems have been using ANPR due to its capacity to quickly and correctly identify license plates of vehicles [11–13].

The process of identifying a vehicle's number plate involves multiple stages, beginning with detecting and isolating the plate from an image. To improve clarity, techniques such as converting to grayscale, reducing noise, and adjusting contrast are applied. Once the image is refined, it is processed through OCR technology, which extracts and converts the number plate details into readable text [14].

The development of deep learning and computer vision has undoubtedly revolutionized ANPR systems. For example, CNNs (Convolutional Neural Network) offer tremendous improvements in the identification of license plates. There are many variations of CNNs, but they all achieve higher recognition accuracy, for both character and vehicle plate detection under diverse and difficult conditions. Those techniques include YOLO and Single Shot MultiBox Detector, which have already been implemented in ANPR systems to enable real-time processing and high detection rates [15,16].

2.2 Techniques for Detecting Fake and Altered Number Plates

Advanced AI and image processing algorithms are now employed to detect fake and altered number plates by identifying inconsistencies and abnormalities indicating tampering. Some prominent techniques include:

1. **Convolutional Neural Networks (CNNs):** CNNs are extensively used for image recognition and classification tasks, including number plate detection. These networks are trained on datasets containing both genuine and fake number plates, identifying subtle alterations in texture, font, and character spacing [17,18].

2. **Generative Adversarial Networks (GANs):** GANs are effective in detecting fake number plates. They consist of a generator and a discriminator working in opposition to identify fake images. This adversarial training produces realistic fake number plates, enhancing the system's ability to detect forgeries [19,20].

3. **Optical Character Recognition (OCR):** OCR technologies convert the visual information of number plates into machine-readable text. When combined with machine learning, OCR can detect inconsistent characters and non-standard fonts and spacing on number plates, indicating tampering [21,22].

4. **Template Matching:** This technique involves comparing detected characters on a number plate with a known database, easily identifying discrepancies in font and character placement [19].

5. **Self-Attentive Mechanisms:** These mechanisms within neural networks focus on critical image regions, improving the model's ability to detect subtle changes in number plates [23,24].

6. **YOLO:** YOLO is a fast and accurate real-time object detection algorithm, suitable for real-time number plate recognition at fuel stations. Its speed allows for immediate detection and action against fake number plates using confidence (Eq. (1)) [25,26]. YOLOv8, the latest version of the YOLO series, was released in 2023 and was utilized during this research.

2.3 Comparison of YOLO-Based Approaches with Alternatives

1. **YOLO Variants Performance:** Numerous studies have explored the effectiveness of various YOLO versions for license plate detection. Table 1 represents a study that compares YOLOv5, YOLOv7, YOLOv8,

and YOLOv9, ultimately identifying YOLOv8 as the top performer for real-world applications. Its standout feature was its impressive performance in cloud-based environments, where it achieved an accuracy rate of 78%, making it the most suitable choice for practical deployment [27].

Table 1: Yolo results comparison for numberplate detection [27]

Sr. no.	YOLO version	Accuracy: validation (%)	Accuracy: testing on cloud (%)
1	YOLOv5	83	71
2	YOLOv7	84	52
3	YOLOv8	83	78
4	YOLOv9	73	70

2. **Fuzzy Logic:** Fuzzy logic systems face several drawbacks when compared to YOLO for tasks like number plate detection, especially in areas such as real-time performance, accuracy, adaptability, and automation. Fuzzy logic relies on manually crafted rules and membership functions, which makes it slower and less efficient for detecting number plates in fast-paced scenarios, such as high-speed traffic. It also tends to struggle in challenging environments—like low-light conditions, motion blur, or skewed images—where YOLO, as a deep learning-based model, excels due to its ability to generalize and maintain accuracy [28,29].

3. **Morphological Methods:** YOLO outperforms traditional morphological methods when it comes to detecting vehicle number plates, thanks to its speed, accuracy, adaptability, and robustness. Morphological methods, which depend on image processing techniques like dilation, erosion, edge detection, and thresholding, are rule-based and often struggle in real-world situations. They tend to falter under challenges such as changing lighting conditions, diverse plate designs, motion blur, and obstructions. In contrast, YOLO, as a deep learning-based model, can automatically learn features from data, making it far more effective in handling complex and varied scenarios [30].

4. **Faster R-CNN and SSD:** Faster R-CNN (Region-Based Convolutional Neural Network) is a two-stage object detection framework that first identifies potential regions of interest and then classifies them, leading to high accuracy but increased computational complexity. Because it processes images in multiple steps, it can be slow and less suitable for real-time applications [31,32]. On the other hand, SSD (Single Shot MultiBox Detector) takes a single-stage approach, predicting object categories and bounding boxes in one go. This makes it much faster than Faster R-CNN, making it a better choice for real-time detection. However, this speed comes at a cost—SSD may sacrifice some accuracy, especially when detecting smaller objects, as it relies on lower-resolution feature maps for predictions. Tables 2 and 3 demonstrate that YOLO operates in real-time while utilizing fewer computational resources compared to Faster R-CNN and SSD.

Table 2: Yolo comparison results with faster R-CNN and SSD-A [33]

Sr. no.	Algorithm	Inference time (ms)	FPS
1	YOLOv4	25	40
2	SSD (MobileNet)	45	22
3	Faster R-CNN (ResNet-50)	120	8

Table 3: Yolo comparison results with faster R-CNN and SSD-B [33]

Sr. no.	Algorithm	GPU memory usage (MB)
1	YOLOv4	2800
2	SSD (MobileNet)	3200
3	Faster R-CNN (ResNet-50)	5400

2.4 Comprehensive Comparative Analyses between OCR Approaches and Alternative Techniques

1. **Tesseract:** The proposed system uses EasyOCR for character recognition, which brings several key benefits over Tesseract, a commonly used OCR engine. EasyOCR is built to handle a wider variety of fonts, languages, and text orientations, making it more reliable in real-world situations where license plates might have unusual fonts, distortions, or uneven lighting. On the other hand, Tesseract, while effective for clear, high-quality text, often has trouble with noisy or distorted inputs and typically requires significant preprocessing to deliver similar results. EasyOCR also stands out for its real-time processing capabilities and its ability to handle multiple languages without additional setup, making it a better fit for applications like fuel station drive-off detection, where speed and flexibility are crucial. These strengths make EasyOCR the preferred choice for the proposed system, especially in environments with diverse and challenging conditions [34].

2. **Keras-OCR:** EazyOCR is often preferred over Keras-OCR for vehicle number plate detection because it offers a complete end-to-end solution, handling both text detection and recognition within a single model. This makes it simpler, more efficient, and easier to implement in real-world applications. It is also more robust in challenging conditions, such as blurry, noisy, or distorted images, which are common when capturing vehicle plates. This is largely due to its deep learning-based CRNN (Convolutional Recurrent Neural Network) models, which enhance accuracy even in difficult scenarios. In contrast, Keras-OCR requires a separate text detection model, like EAST, making the process more complex. Additionally, EazyOCR supports a wider range of languages and non-standard fonts, making it better suited for recognizing number plates from different regions. Its simple API and minimal setup requirements further improve ease of use, whereas Keras-OCR requires more configuration and customization. Overall, EazyOCR stands out for its accuracy, speed, and versatility, making it a better choice for vehicle number plate detection in varied and complex environments [34–36]. Fig. 1's comparison shows that EasyOCR is more reliable than Keras and Tesseract.

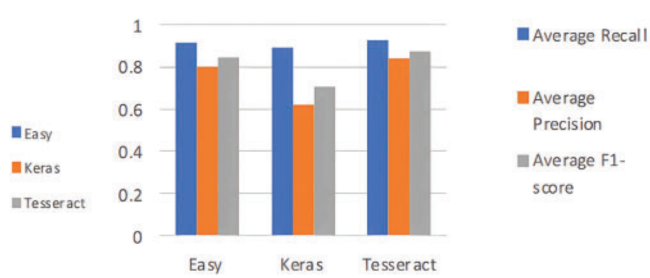


Figure 1: OCR comparison results with Tesseract and Keras-OCR [33]

2.5 YOLO and OCR: Superior Technologies for Fake Number Plate Detection

Among the various technologies used for detecting fake number plates, YOLO and OCR stand out due to their unique features and excellent performance.

1. **YOLO: Real-Time Object Detection:** YOLO is renowned for its speed and accuracy in real-time object detection. It processes images as a single regression task, predicting bounding boxes and class probabilities directly from full images, making it ideal for real-time applications like fuel station number plate detection [25,37].

2. **OCR: Precise Character Recognition:** OCR technology reads and converts the alphanumeric characters on number plates into machine-readable text. Advanced OCR systems can detect inconsistencies in characters, such as non-standard fonts and spacing, indicating fake number plates [17,38]. Fig. 2 illustrates a comparison of the confidence scores between OCR.

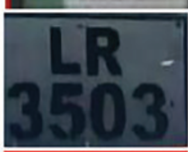
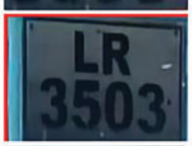
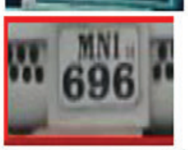
Extracted Frames	OCR Recognition Result	OCR Confidence Score (%)	AlexNet Based CNN Recognition Result	AlexNet Based CNN Confidence Score (%)
	FDS4014	100	FD54014	86
	FDS4014	100	F0S404	77
	FD5401	86	FD404	83
	LR35D3	89	LR3503	100
	LR3503	100	LR3SD3	67
	LR3503	100	LR303	91
	MN6	67	M696	80
	MNI696	100	MNI696	100

Figure 2: OCR reliability compared with CNN [15]

3. **Combined Strengths of YOLO and OCR:** Integrating YOLO and OCR technologies results in a robust system for detecting fake number plates (Fig. 3). YOLO's real-time object detection capabilities complement OCR's precise character recognition, enhancing overall system performance and reliability [17,25].

Image	Number of detections for given period	Detected Number	Actual Number
	5	CK15FKE	
	20	CK15FKB	CK15FKB
	8	CK15FKD	

Figure 3: Combination of technology help to identify most accurate numberplate characters

The review highlights significant advancements in ANPR systems and various AI techniques developed to detect counterfeit and modified number plates. It accentuates the strong and weak points of each technique and offers some examples proving the efficiency of ANPR system use in different environments. Also, the review focuses on the importance of developing a powerful AI-based detection system to decrease the threat of fuel station drive-offs with the help of the combination of YOLO and OCR technologies designed for better real-time detection and higher accuracy.

3 Proposed Methodology

3.1 Methodology Diagram

Fig. 4 illustrates the research methodology, which follows a structured process. The system begins with Data Acquisition and Preparation, where high-resolution CCTV cameras capture vehicle images, followed by image pre-processing to enhance quality. The next phase is Software Development, involving the creation of algorithms for image processing and data management. The AI Model Development stage follows, where the YOLOv8 model is trained for license plate detection, and EasyOCR is integrated for character recognition. OCR Integration ensures the accurate extraction of alphanumeric characters from the detected plates. In the System Integration and Testing stage, all components are combined and tested for functionality. Interface Development creates a user-friendly web interface for real-time monitoring. The final stage, Evaluation and

Refinement, is an ongoing process of system evaluation and improvement. The methodology concludes with Preparation for Implementation, which defines the necessary steps for deploying the system in real-world fuel pump scenarios.

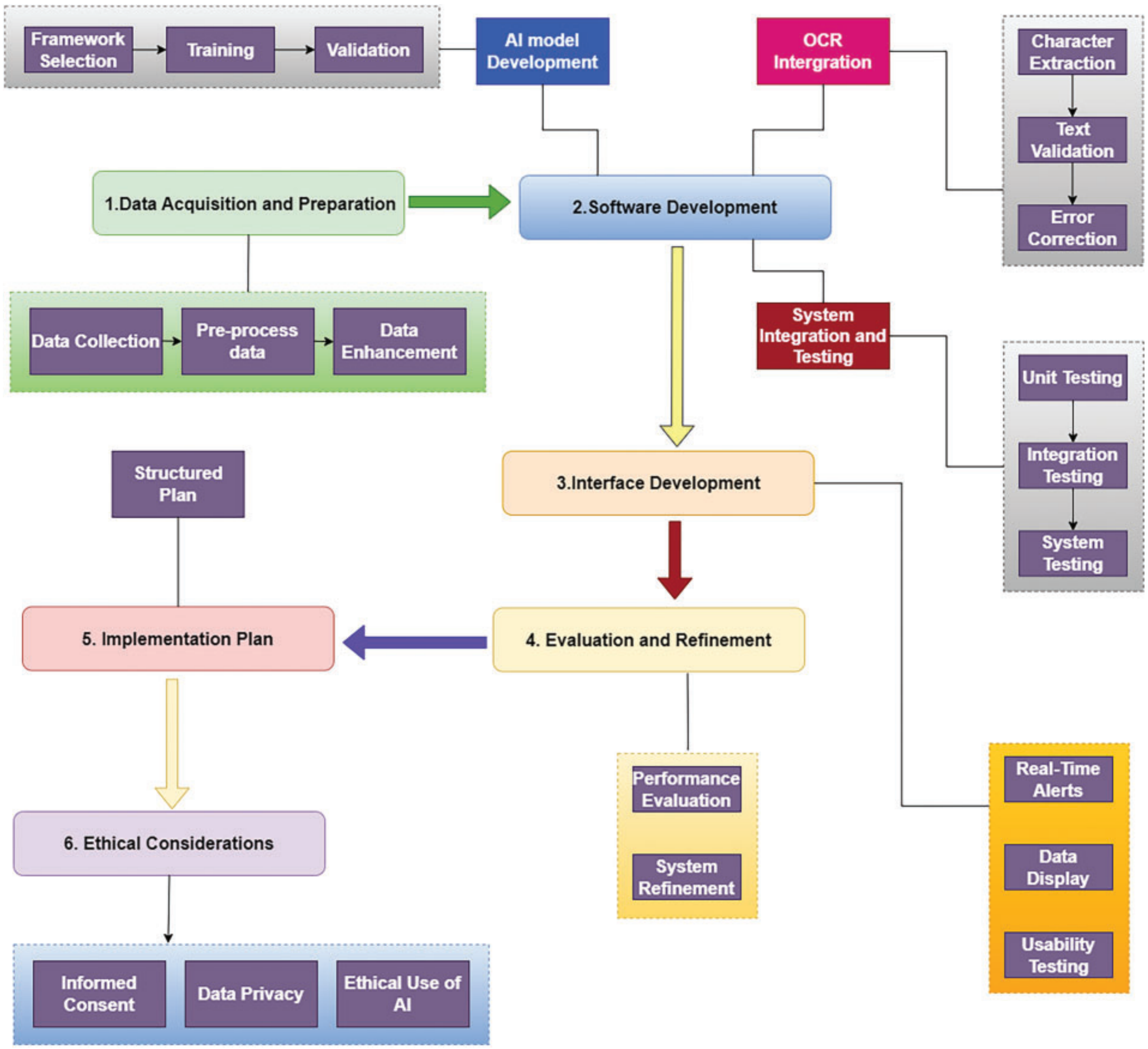


Figure 4: Methodology diagram for AI-driven number plate detection and verification

3.2 Image Samples

To illustrate the data used in this study, Fig. 5 shows sample images from the dataset. These images capture different lighting conditions, angles, and environments to enhance AI model performance. The diversity in the dataset is crucial for training models that can perform well in real-world scenarios.

Figure 5: Sample images for number plate detection and recognition

3.3 Pseudo Code of the Study

The pseudo-code of this study is outlined in Algorithm 1 below.

Algorithm 1: Pseudocode of the study

Step 1: Initialize the system
Load YOLOv8 for plate detection, EasyOCR for character recognition, and connect to the MOT History API. STATE **Step 2:** Capture footage
Continuously collect images from the fuel station's CCTV cameras.
Step 3: Preprocess images
Resize and enhance the images for consistent and clear analysis.
Step 4: Detect number plates
Use YOLOv8 to identify and locate the number plates in the images.
Step 5: Recognize characters
Extract and convert the plate's alphanumeric characters using EasyOCR.
Step 6: Verify plates
Cross-check the recognized plate with vehicle data using the MOT History API.
Step 7: Detect fakes
Flag any mismatches as potential fake or altered plates. Issue alerts: Provide real-time alerts on suspicious vehicles to the fuel station operator.
Step 8: Monitor and refine
Track the system's performance and retrain the models as needed.
Step 9: Optional vehicle tracking
Use the SORT (Simple Online and Real-time Tracking) algorithm to ensure the correct association of number plates with vehicles across frames.

3.4 Data Analysis Method

Regarding the data analysis method, one of the focuses is to evaluate the performance of AI models to detect and recognize number plates. The important metrics are accuracy, precision, recall, and F1 score. The confidence score indicates how likely it is that the bounding box contains an object whether the bounding box contains an object and the accuracy of the bounding box.

Confidence Score:

$$\text{Confidence} = P(\text{Object}) \times \text{IOU}_{\text{pred}}^{\text{truth}} \tag{1}$$

In Eq. (1), where $P(\text{Object})$ is the probability that an object is present, and $\text{IOU}_{\text{pred}}^{\text{truth}}$ is the Intersection Over Union between the predicted box and the ground truth. The IOU metric measures the overlap between the predicted bounding box and the actual bounding box, indicating the accuracy of the detection.

Image Preprocessing The input image $I \in \mathbb{R}^{H\times W\times 3}$ is resized to a fixed size of 640 × 640 pixels:

$$I_{\text{resized}} = \text{resize}(I, 640 \times 640) \tag{2}$$

Bounding Box Prediction For each grid cell (i, j), the model predicts multiple bounding boxes. Each box is parameterized as:

$$\text{Bounding box} = (t_x, t_y, t_w, t_h, \text{confidence}, c_1, c_2, \dots, c_C) \tag{3}$$

where:

- t_x, t_y: Offset for the center of the bounding box relative to the grid cell.
- t_w, t_h: Width and height scaling factors.
- confidence: Probability that an object exists in the bounding box.
- $c_1, c_2, \dots, c_C$: Class probabilities.

The final bounding box coordinates are computed as:

$$b_x = \sigma(t_x) + c_x, \quad b_y = \sigma(t_y) + c_y \tag{4}$$
$$b_w = p_w e^{t_w}, \quad b_h = p_h e^{t_h} \tag{5}$$

where c_x, c_y are the top-left coordinates of the grid cell and p_w, p_h are anchor box dimensions. The sigmoid function σ constrains t_x, t_y to the range $[0, 1]$.

Loss Function The total loss function $\mathcal{L}$ combines classification loss, localization loss, and objectness loss:

$$\mathcal{L} = \lambda_{\text{class}}\mathcal{L}_{\text{class}} + \lambda_{\text{box}}\mathcal{L}_{\text{box}} + \lambda_{\text{obj}}\mathcal{L}_{\text{obj}} \tag{6}$$

Classification Loss The classification loss is computed using cross-entropy for the predicted class probabilities p_{ij} and true labels y_{ij}:

$$\mathcal{L}_{\text{class}} = -\sum_{i,j} y_{ij} \log(p_{ij}) \tag{7}$$

Localization Loss The localization loss measures the error in the predicted bounding box coordinates (b_x, b_y, b_w, b_h) compared to the ground truth (g_x, g_y, g_w, g_h) using smooth L1 or generalized IoU (GIoU):

$$\mathcal{L}_{\text{box}} = \text{smooth}_{L1}(b_x - g_x) + \text{smooth}_{L1}(b_y - g_y) + \text{smooth}_{L1}(b_w - g_w) + \text{smooth}_{L1}(b_h - g_h) \tag{8}$$

Objectness Loss The objectness loss is computed as binary cross-entropy for the predicted confidence score $\hat{p}_{\text{obj}}$ and the ground truth p_{obj}:

$$\mathcal{L}_{\text{obj}} = -\left[p_{\text{obj}} \log(\hat{p}_{\text{obj}}) + (1 - p_{\text{obj}}) \log(1 - \hat{p}_{\text{obj}})\right] \tag{9}$$

Non-Maximum Suppression (NMS) To remove overlapping bounding boxes, non-maximum suppression (NMS) is applied. For each class, the boxes are sorted by confidence score, and IoU (Intersection over Union) is calculated as:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{10}$$

Boxes with high IoU are suppressed to retain the best candidates.

3.5 YOLO Model Component and Properties

YOLO is a real-time object detection framework that uses a single-stage CNN for fast and efficient image processing. It includes an input processing module, a feature extraction backbone (like CSPDarkNet in YOLOv8), and a detection head for predicting bounding boxes and object classes. YOLO excels in speed and accuracy, making it suitable for real-time applications like autonomous vehicles and surveillance. It supports multi-scale detection, anchor boxes for precision, and non-maximum suppression to eliminate duplicate detections. Highly scalable, YOLO operates efficiently on both high-performance GPUs (Graphics Processing Unit) and edge devices, making it ideal for dynamic AI applications that require real-time processing within a couple of seconds [26,39]. YOLO is a fast, real-time object detection framework that uses a single-stage CNN to process an entire image and detect multiple objects in one go. It features an input processing module for resizing and normalizing images, a feature extraction backbone (e.g., CSPDarkNet in YOLOv8), and a detection head that predicts bounding boxes, class probabilities, and confidence scores. Known for its high-speed inference, multi-scale detection, and scalability across various hardware platforms, YOLO is ideal for applications such as autonomous vehicles, surveillance, and number plate recognition [8,37].

3.6 EasyOCR Component and Properties

EasyOCR is an open-source Optical Character Recognition (OCR) system that uses deep learning models, including CNNs and LSTMs (Long Short-Term Memory networks), for accurate text detection and recognition. It features preprocessing for image normalization, CRAFT-based (Character Region Awareness) text detection, and CRNN-powered (Convolutional Recurrent Neural Network) text recognition, along with post-processing for accuracy enhancement. Supporting over 80 languages, EasyOCR is optimized for real-time performance, working efficiently on both CPUs (Central Processing Unit) and GPUs while handling printed, handwritten, and low-quality text. The YOLOv8 + EasyOCR system extracts number plate characters, which are then verified through the MOT History API to detect fraudulent plates, significantly improving real-world fraud detection at fuel stations [22].

3.7 Model Training and Evaluation

We use YOLOv8 to train the model. The data was collected from a public image repository on Kaggle and processed through several steps, as outlined below.

1. **Data Augmentation:** Image resizing, accurate annotation, and rotation were done on the selected dataset to clean it for training. This helps improve the accuracy of the trained model [39].

Steps that follow for data Augmentation:

- All images to 640 × 640 pixels to ensure uniform input size for YOLOv8
- Applied Gaussian and median filtering to remove background noise and artefacts, improving OCR readability
- Removed colour variations to focus on character recognition
- removed low-quality or corrupted images from the dataset to avoid model mislearning

2. **Model Optimization:** Hyperparameter tuning is used to optimize model performance, adjusting parameters like learning rate, batch size, and number of epochs.

3. **Validation and Testing:** Trained models are evaluated using various metrics such as accuracy, precision, recall, and F1 score. The predicted results are used to verify the accurate identification of number plates. [15]. Fig. 6 shows how to validate the AI model.

Figure 6: Model training and evaluation process

Validation methods used:

- The dataset was split into training (80%) and (20%) testing subsets to ensure that the model was tested on unseen data.
- Performance was validated by comparing AI predictions against manually verified number plate data.
- The OCR-extracted number plates were cross-referenced with the MOT History API to validate authenticity (faulty Numberplate also validated here).

3.8 Real-Time Implementation

Trained models are used to identify car number plates and cars from CCTV footage. Additionally, the SORT algorithm is used to track cars with number plates. OCR-extracted characters are filtered and sent to the MOT History API to validate the vehicle.

1. **Integration with CCTV Systems:** This system used CCTV footage, which helped reduce infrastructure costs. A server is used to run the application for real-time processing and the web application.

2. **Alert Mechanism:** To generate alerts, we use a web interface that provides the required vehicle details to identify the vehicle. If no details are found, the interface will prompt with a red colour indicator, which will help the fuel station staff quickly identify fake number plates and prevent unauthorized fueling.

3. **Performance Monitoring:** Continuous monitoring ensures system reliability and accuracy. Regular updates and model retraining are performed based on new data and feedback from fuel station operators.

4. **SORT Algorithm Integration into the AI System:** SORT (Simple Online and Real-time tracking) is an algorithm used for object tracking. This system is employed to track vehicles that have been identified by YOLO, allowing consistent association with the detected number plates across frames in real time [40]. This makes it easier to compile the number plates corresponding to each vehicle when extracting the results. SORT tracks vehicles using inputs such as bounding box coordinates from YOLOv8 detections, a detection confidence score, the frame number for continuity, and an initially unassigned object ID that is later generated for persistent tracking. The algorithm outputs updated bounding box coordinates, a unique object ID for each vehicle, its trajectory across frames, and, if visible, a matched number plate. Integrated into the AI system, YOLOv8 detects vehicles, SORT assigns tracking IDs, and EasyOCR extracts number plate characters, ensuring consistency even if the plate becomes partially occluded. Suspicious vehicles are flagged if the MOT History API detects forged plates

The methodology described in this chapter presents a way to follow in order to create a system of detecting fake and changed number plates of fuel stations with the help of AI. The system, with its AI-based models, integration with pre-existing CCTV infrastructure, and real-time processing ensures safety and reduces the economic loss caused by drive-offs.

4 Analysis and Results

When compared to previous approaches, our system demonstrates superior accuracy and real-time performance. While YOLOv5-based ANPR systems typically achieve around 78%–80% detection accuracy [41]. Our implementation with YOLOv8 and EasyOCR improved this to 84% in real-world CCTV footage. Moreover, unlike traditional ANPR systems that rely on blacklists, integrating the MOT History API allows us to detect fraudsters who alter number plates, significantly increasing detection reliability.

4.1 Experimental Design

In the experimental design, a model is trained to accurately detect number plates in real-time at fuel stations using low-resolution CCTV videos with YOLOv8.

4.2 Data Sources

The system utilizes datasets from platforms like Kaggle, which include annotated vehicle number plate images. This approach ensures a diverse and comprehensive dataset, enhancing the model's robustness and generalization. Fig. 7 displays a sample of the images used during the annotation process.

4.3 Model Training and Evaluation

The YOLO algorithm was used for training the number plate detection model. The data was divided into training, testing, and validation sets to analyze performance. The training process involved hyperparameter tuning and data augmentation techniques to enhance model robustness. Figs. 8 and 9 represent the evaluation results of the model.

4.4 Data Collection Process

The process begins with the acquisition of selected datasets, complemented by the generation of synthetic data when necessary. This approach ensures a diverse and comprehensive dataset that reflects a wide range of real-world scenarios.

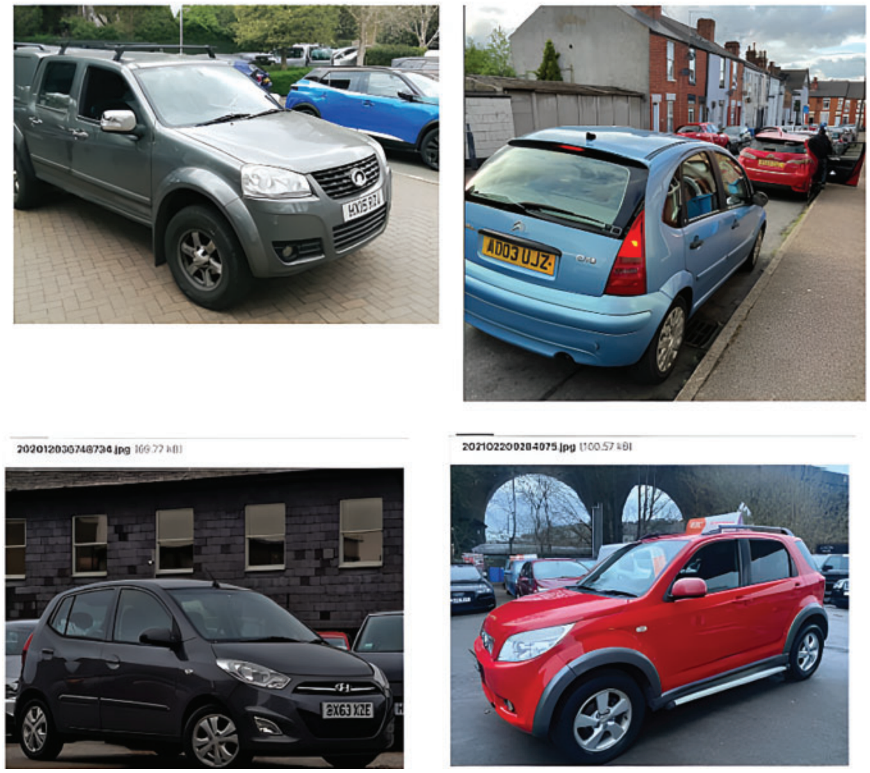


Figure 7: Sample data of dataset

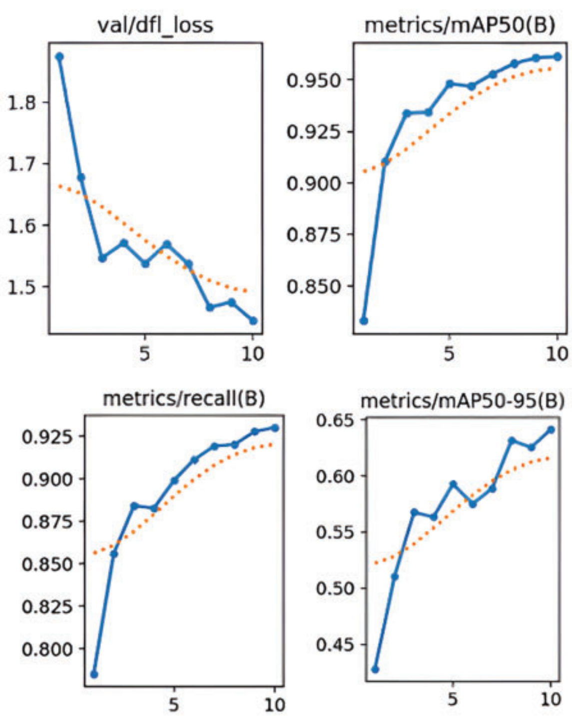


Figure 8: Model training and evaluation results

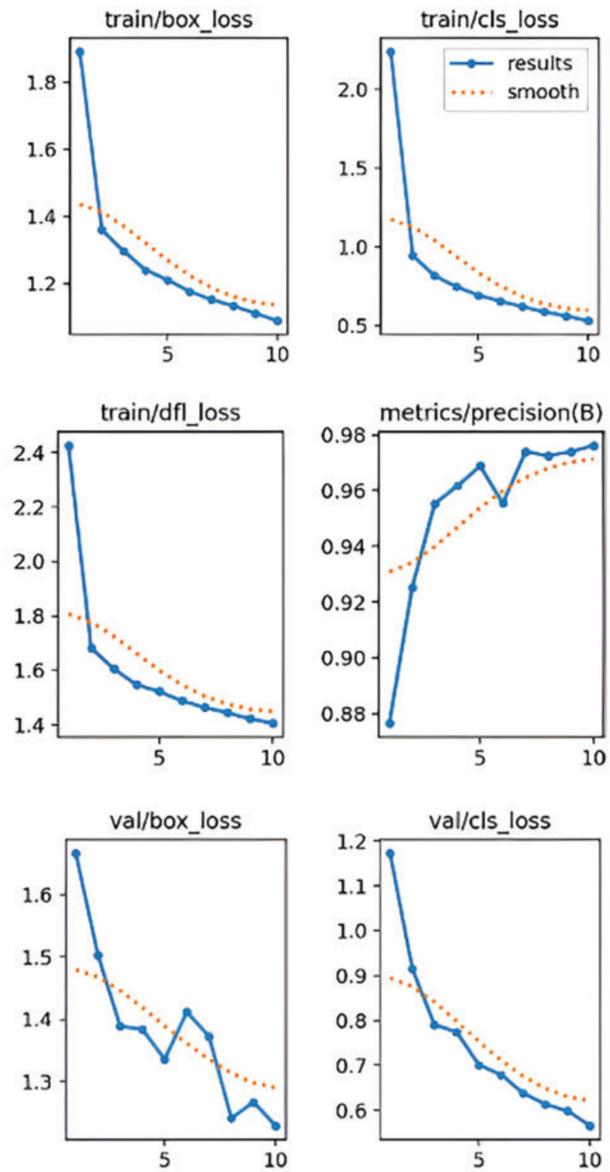


Figure 9: Model training and evaluation results

Preprocessing is then applied, including resizing images to a uniform size, normalizing pixel values, and converting them to grayscale if required. These steps help standardize the input, making it suitable for training the AI model.

Subsequently, each image is manually annotated using specialized annotation tools. Although this step can be time-intensive, it is vital for constructing a high-quality and reliable training set. Following annotation, a quality assurance check is performed by cross-validating a random subset of labelled images. This process verifies the accuracy of the labels and supports any necessary refinements.

4.5 Data Augmentation Techniques

To increase dataset diversity and reduce overfitting, data augmentation techniques were applied to simulate real-world conditions

- Simulated different viewing angles of vehicles
- Improved model robustness to varying lighting conditions
- Modeled real-world scenarios where plates appear blurred due to movement
- Created altered/fake plates with different fonts, colours, and distortions to enhance the model's ability to detect forgery.

4.6 Evaluation of YOLO Model Performance in Number Plate Detection

The performance of the YOLO model was assessed in terms of precision, recall, and mean average precision metrics, which provide comprehensive insight into how accurate the model is in detecting the number plates. The precision-recall curve supports the model's dependability in recognizing number plates, as it represents the point at various threshold levels at which the performance of the deep learning model can be maximized. Additionally, the confusion matrix provided a clear understanding of the model's performance, highlighting the true positive, true negative, false positive, and false negative with the number of correctly and incorrectly classified instances as shown in Fig. 10.

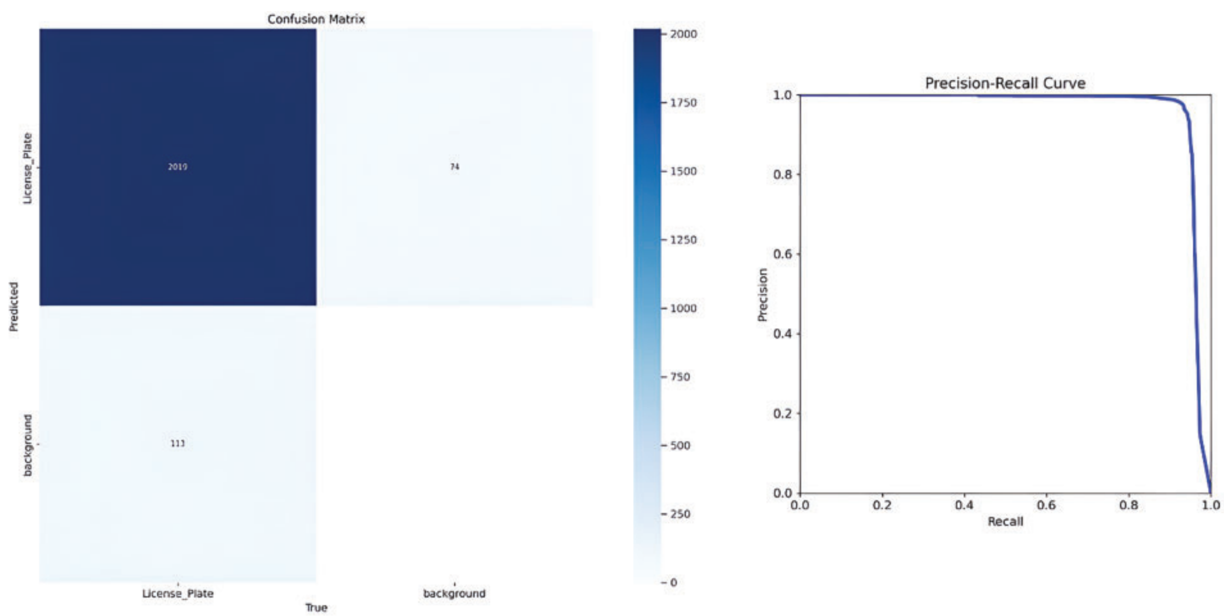


Figure 10: Performance metrics, precision-recall curve, and confusion matrix 8

The combination of preprocessing and augmentation played a crucial role in achieving high accuracy. The data cleaning and enhancement steps ensured that the model learned from high-quality images, reducing false positives. The augmented dataset allowed YOLOv8 and EasyOCR to generalize well to different environments, including varying lighting, camera angles, and obstructions. This led to an 84% (Fig. 11) success rate in real-world CCTV footage.

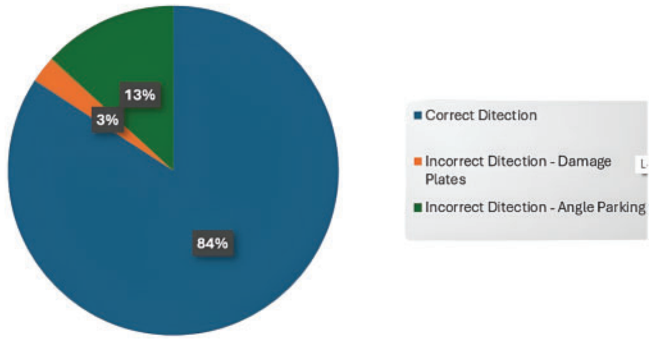


Figure 11: Results analysis

4.7 Predictive Analysis

The predictive analysis involved testing the model on a separate dataset to predict the presence of number plates. The results showed a high degree of accuracy, with the model correctly predicting the presence of number plates in various environmental conditions. Fig. 12 shows the results of the predictive analysis, demonstrating the model's effectiveness.

Figure 12: (Continued)

Figure 12: Predictive analysis results

4.8 OCR Integration Test

The OCR number plate recognition was tested on 50 vehicles. Initial tests showed that only 12 number plates were recognized correctly. After integrating character mapping and optimizing for UK number plate recognition, the system accurately identified 46 out of 50 number plates, resulting in a 92% accuracy rate. OCR image recognition time average is 216.2 ms for the 50 number plates.

4.9 Real Environment Test Results

The real environment tests were conducted at various fuel stations to assess the system's performance under real-world conditions. The system accurately detected and verified number plates under varying vehicle speeds and angles, as depicted in Fig. 13, as well as across diverse lighting and weather conditions, as shown in Fig. 14. These evaluations validated Fig. 14 the system's robustness and practical applicability.

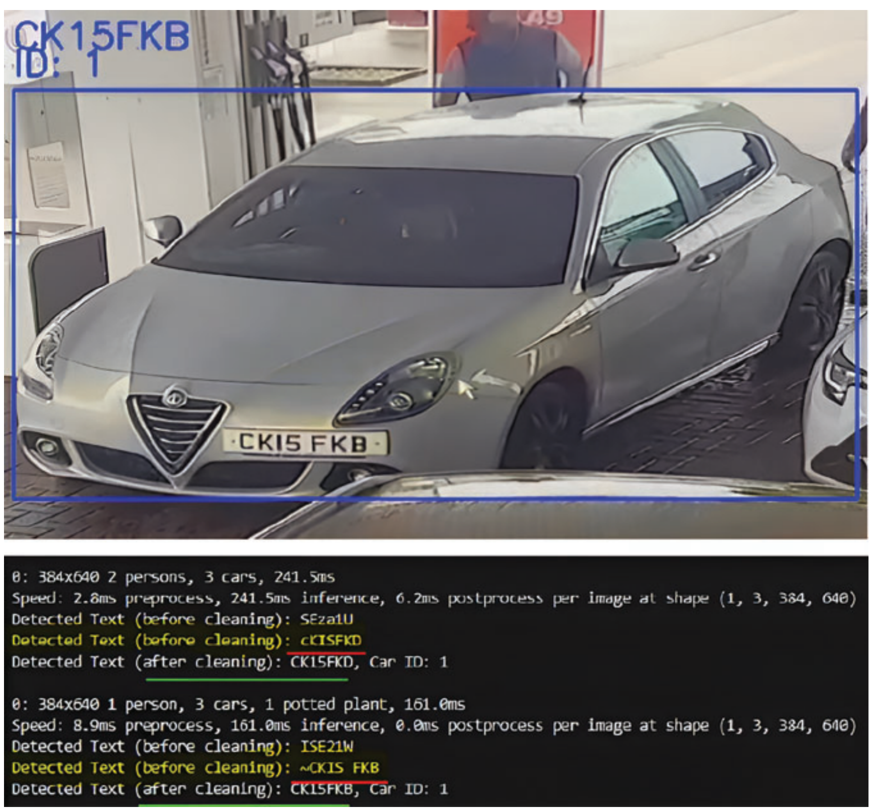


Figure 13: Results with obstacle

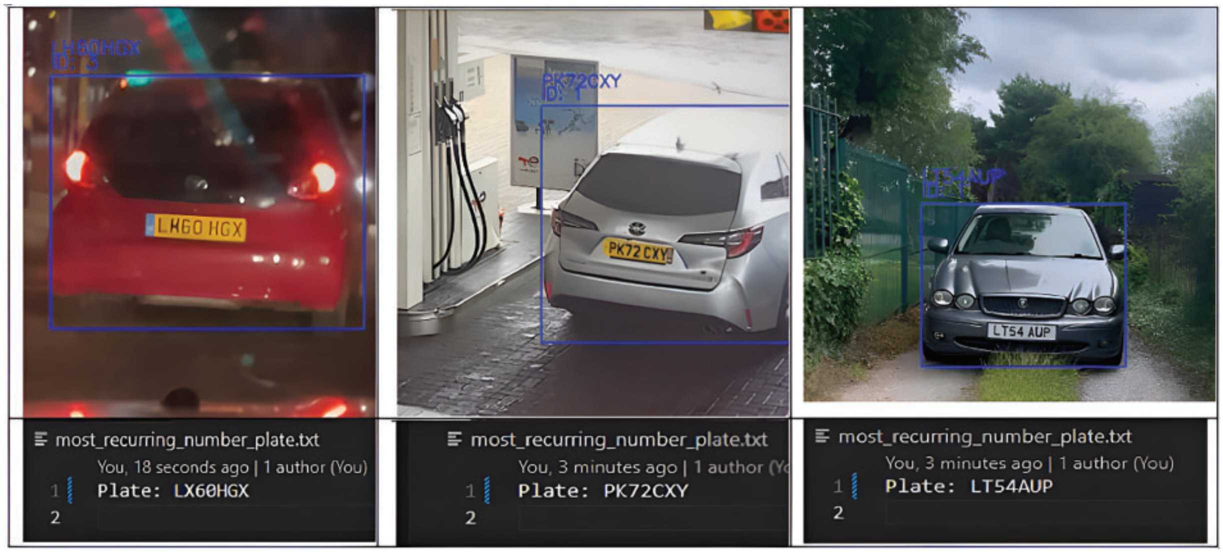


Figure 14: Results based on various environment conditions

Table 4 shows the results obtained from the training of the AI model.

1. The training losses (box_loss, cls_loss, dfl_loss) consistently decrease over epochs, demonstrating model improvement.
2. The validation losses also decrease but may fluctuate slightly, which is normal as the model generalizes to unseen data.
3. By epoch 10, the losses are lower compared to earlier epochs, suggesting the model is converging.

Table 4: Training results table

epoch	train or box_loss	train or cls_loss	train or dfl_loss	val or box_loss	val or cls_loss	val or dfl_loss
1	1.8924	2.241	2.4267	1.6665	1.1726	1.8749
2	1.3607	0.94519	1.6815	1.5033	0.91583	1.6784
3	1.2969	0.81681	1.6055	1.389	0.79095	1.5465
4	1.2402	0.74689	1.547	1.3841	0.77442	1.5708
5	1.2115	0.69295	1.5218	1.3362	0.70128	1.5374
6	1.1765	0.65609	1.4879	1.4119	0.6795	1.5693
7	1.1528	0.62043	1.4638	1.3729	0.63827	1.5372
8	1.1333	0.58808	1.444	1.2416	0.61349	1.4663
9	1.1115	0.56124	1.4217	1.2672	0.59887	1.4745
10	1.0892	0.53121	1.4051	1.2294	0.56595	1.4449

4.10 Web Interface API Test

The web interface API test showed 100% accuracy in integrating with the API, providing real-time alerts and verification results to the operators. The user-friendly interface allowed for easy monitoring and management of detected number plates. Fig. 15 shows the web interface used for real-time monitoring.

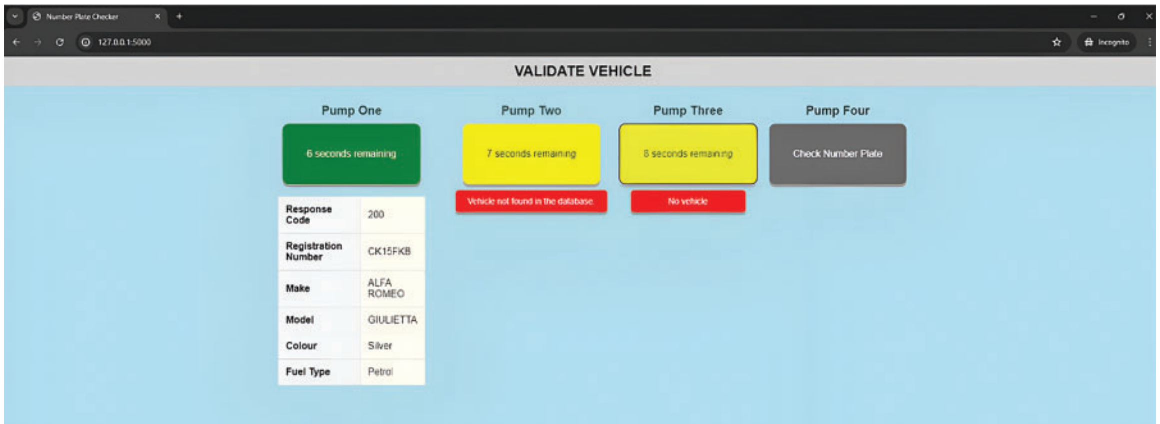


Figure 15: Web interface for real-time monitoring

The analysis and results demonstrate the effectiveness of the proposed AI-driven system in detecting and verifying number plates at fuel stations. The integration of YOLO for real-time detection and EasyOCR for character recognition, combined with robust data preprocessing and augmentation techniques, resulted in a highly accurate and reliable system. The system's performance metrics and integration test results validate its potential to reduce fuel station drive-offs and enhance security measures.

4.11 Hardware Requirement

This system consists of a number of software components and libraries, which necessitates considerable hardware resources to operate efficiently as shown in Table 5.

Table 5: Hardware requirements

Components	Minimum hardware (For testing)	Recommended hardware (Processing at fuel stations)
CPU (Central Processing Unit)	Intel Core i5 (10th Gen) AMD Ryzen 5	Intel Core i7 (12th Gen) AMD Ryzen 7
RAM (Random Access Memory)	8 GB DDR4	16 GB DDR4 or higher
GPU (Graphics Processing Unit)	NVIDIA GTX 1650 (4 GB VRAM) or CPU only	NVIDIA RTX 3060 (6 BG VRAM) or higher
Storage	256 GB SSD	512 GB SSD (NVMe preferred)
Operating system	Windows 1011, Ubuntu 20.04+	Ubuntu 20.04+ (preferred for AI/ML)
Camera input	Basic IP Camera or CCTV footage (720p)	High-resolution IP cameras (1080p or higher)
Network	Stable internet connection (for API calls)	Fiber optic or 4G/5G for real-time API verification
Real-time processing	Limited, may have delays	Smooth real-time detection and tracking

5 Discussion

In the United Kingdom, fuel station owners are currently grappling with significant losses due to the increased rate of fuel station drive-offs. These cases have been on the rise in previous years, and given the high numbers, the police are currently unable to address the issue. As it stands, fuel station owners are the most affected parties by the increased prevalence of such crimes. Although many fuel stations have tried to curb fuel station drive-offs using costly ANPR cameras and systems, they are unaffordable for smaller fuel stations. The reliance on blacklisted data hardly identifies the number of place changes or detects the first-time drive-offs. This paper outlined the prototype system that offers a low-cost approach to the issue of franchise fuel stations by using CCTV cameras to monitor vehicles. The prototype system utilizes the YOLOv8, OCR, character mapping, and the MOT History API to increase the detection and prevention of drive-offs at fuel stations.

Our preprocessing and augmentation strategies were instrumental in improving the AI system's performance. By ensuring high-quality input data, the model made more precise predictions. Augmentation techniques helped the system adapt to different environmental conditions, reducing false negatives in low-light and occlusion scenarios. As a result, the system demonstrated a high detection rate in real-world conditions, reinforcing the importance of comprehensive data preparation in AI-driven number plate recognition.

Based on the analysis results, the AI model developed for number plate recognition demonstrates accurate performance in identifying number plates, with effectiveness evaluated through metrics such as precision, recall, and mean Average Precision (mAP). Precision, which measures the proportion of correctly identified number plates out of all predicted plates, helps determine the optimal threshold that balances precision and recall, maximizing the model's effectiveness. The OCR technology in this model uses techniques like grayscale processing, template matching, and denoising to enhance accuracy, with outputs mapped according to the UK number plate format to further increase accuracy. Although designed for low-resolution videos, the model's accuracy improves significantly with high-resolution or ANPR cameras. The identified number plates are validated using the MOT History API, ensuring 100% accuracy in verifying vehicle parts, with this step affecting only the maintenance of the API. By combining OCR with YOLO, the model enhances real-time vehicle detection, number plate detection, and identification, with an additional algorithm enabling object tracking to ensure accurate association between the number plate and the relevant vehicle. Testing at a fuel station under CCTV cameras showed an 84% success rate and 100% accuracy in web interface results. This AI model and prototype have the potential for further enhancement in various real-world scenarios, as they can achieve high success rates with high-resolution cameras.

6 Conclusion

The evaluation of the YOLO model for number plate detection is thoroughly detailed. Its performance is assessed through precision, recall, mAP metrics, precision-recall curves, and confusion matrix analysis. These results demonstrate that the model is well-suited for daily applications requiring reliable number plate detection. The confusion matrix evaluation reveals where the model excels and where it struggles, providing specific insights into its detection capabilities and limitations.

Generally, the YOLO model is a highly accurate and efficient number plate recognition system. First, adding UK-style number plate characters to the training data improved the model's accuracy. Furthermore, integrating the YOLO model with ANPR cameras would make plate detection extremely precise. However, continuous monitoring is required to track the proportion of false positives and false negatives relative to total detections. Subsequent studies should focus on optimizing detection threshold levels and revising training datasets to further improve performance. Additionally, implementing a character mapping algorithm improves result accuracy, strengthening usability in real-world applications. Ongoing testing will ensure continued error reduction and long-term consistency.

7 Future Work

Various areas of research and potential system enhancements could significantly expand the functionality of the proposed AI-driven detection system. While the current integration of YOLO and EasyOCR is sufficient, future improvements might include other AI methods, such as Generative Adversarial Networks (GANs), to generate a larger dataset of synthetic number plates for more robust training and reinforcement learning, which would allow the system to adapt and improve based on real-world feedback over time. This system could also be applied beyond fuel stations to other vehicle identification scenarios like parking lots, toll booths, and border control points, where modifications for different environmental conditions might be necessary. Enhancing real-time processing capabilities is critical for instantaneous responses, suggesting a need for decentralized processing where data analysis and decision-making occur at the same location. Additionally, adding multi-factor authentication, such as combining number plate recognition with vehicle make and model or driver facial recognition, would further secure the system. This system has limited capability in handling non-standard numbers, which is an area that requires improvement.

Acknowledgement: The authors are grateful to all editors and anonymous reviewers for their comments and suggestions and thank all the members who have contributed to this work with us.

Funding Statement: This study received no financial support or funding from any organization.

Author Contributions: The authors confirm their contributions to the paper as follows: Milinda Priyankara Bandara Gamawelagedara, Mian Usman Sattar, and Raza Hasan contributed to the study's conception and design. Data collection was conducted by Milinda Priyankara Bandara Gamawelagedara and Raza Hasan, while analysis and interpretation of results were carried out by Milinda Priyankara Bandara Gamawelagedara and Mian Usman Sattar. The draft manuscript was prepared by Milinda Priyankara Bandara Gamawelagedara, Raza Hasan, and Mian Usman Sattar, with manuscript guidance and revision provided by the journal. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Tham AYX, Chin CS. YOLOv5 and residual network for intelligent text recognition on degraded serial number plates. In: International Conference on Engineering Applications of Neural Networks; 2024 Jun 27–30; Corfu, Greece. p. 301–14.
2. Liu Q, Chen SL, Chen YX, Yin XC. Improving license plate recognition via diverse stylistic plate generation. Pattern Recognit Lett. 2024;183:117–24. doi:10.1016/j.patrec.2024.05.005.
3. Puranic A, Deepak K, Umadevi V. Vehicle number plate recognition system: a literature review and implementation using template matching. Int J Comput Appl. 2016;134(1):12–6. doi:10.5120/ijca2016907652.
4. Ćorović A, Ilić V, Durić S, Marijan M, Pavković B. The real-time detection of traffic participants using YOLO algorithm. In: 2018 26th telecommunications forum (TELFOR); 2018 Nov 20–21; Belgrade, Serbia. p. 1–4.
5. Mukherjee S, Tyagi H, Tyagi P, Singh N, Bhardwaj S. OCR using python and its application. J Adv Zool. 2023;44(S3):1083–92. doi:10.17762/jaz.v44iS-3.1062.

6. Romsom E. Global oil theft: impact and policy responses. WIDER working paper; 2022. [Internet]. [cited 2025 Mar 23]. Available from: https://www.wider.unu.edu/sites/default/files/Publications/Working-paper/PDF/wp2022-16-global-oil-theft-impact-policy-responses.pdf.
7. Thaiparnit S, Khuadthong N, Chumuang N, Ketcham M. Tracking vehicles system based on license plate recognition. In: 2018 18th International Symposium on Communications and Information Technologies (ISCIT); 2018 Sep 26–29; Bangkok, Thailand. p. 220–5.
8. Gurney R, Rhead M, Lyons V, Ramalingam S. The effect of ANPR camera settings on system performance. In: 5th International Conference on Imaging for Crime Detection and Prevention (ICDP 2013); 2013 Dec 16–17; London, UK.
9. Tang J, Wan L, Schooling J, Zhao P, Chen J, Wei S. Automatic number plate recognition (ANPR) in smart cities: a systematic review on technological advancements and application cases. Cities. 2022;129:103833. doi:10.1016/j.cities.2022.103833.
10. Gunawan TS, Mutholib A, Kartiwi M. Performance evaluation of automatic number plate recognition on android smartphone platform. Int J Electr Comput Eng. 2017;7(4):1973–82. doi:10.11591/ijece.v7i4.pp1973-1982.
11. Arora M, Jain A, Rustagi S, Yadav T. Automatic number plate recognition system using optical character recognition. Int J Sci Res Comput Sci Eng Inf Technol. 2019;5(2):986–92. doi:10.32628/CSEIT1952280.
12. de Vyvere B Van, Colpaert P. Using ANPR data to create an anonymized linked open dataset on urban bustle. Eur Transp Res Rev. 2022;14(1):17. doi:10.1186/s12544-022-00538-1.
13. Robinson A, Venter C. Validating traffic models using large-scale automatic number plate recognition (ANPR) data. J S Afr Inst Civ Eng. 2019;61(3):45–57. doi:10.17159/2309-8775/2019/v61n3a5.
14. Salma Saeed M, ur Rahim R, Gufran Khan M, Zulfiqar A, Bhatti MT. Development of ANPR framework for Pakistani vehicle number plates using object detection and OCR. Complexity. 2021;2021(1):5597337. doi:10.1155/2021/5597337.
15. Khan MG, Salma, Saeed M, Zulfiqar A, Ghadi YY, Adnan M. A novel deep learning based ANPR pipeline for vehicle access control. IEEE Access. 2022;10:64172–84. doi:10.1109/ACCESS.2022.3183101.
16. Kumar A, Zhang ZJ, Lyu H. Object detection in real time based on improved single shot multi-box detector algorithm. EURASIP J Wirel Commun Netw. 2020;2020(1):204. doi:10.1186/s13638-020-01826-x.
17. Kashyap Y. An advanced ANPR system using OCR in Matlab. Int J Res Appl Sci Eng Technol. 2017;5(9):1284–8.
18. Li Z, Liu F, Yang W, Peng S, Zhou J. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Trans Neural Netw Learn Syst. 2021;33(12):6999–7019. doi:10.1109/TNNLS.2021.3084827.
19. Han BG, Lee JT, Lim KT, Choi DH. License plate image generation using generative adversarial networks for end-to-end license plate character recognition froma small set of real images. Appl Sci. 2020;10(8):2780. doi:10.3390/app10082780.
20. Gui J, Sun Z, Wen Y, Tao D, Ye J. A review on generative adversarial networks: algorithms, theory, and applications. IEEE Trans Knowl Data Eng. 2021;35(4):3313–32. doi:10.1109/TKDE.2021.3130191.
21. Chavan C, Hembade S, Jadhav G, Komalwad P, Rawat P. Computer vision application analysis based on object detection. Int J Sci Res Eng Manag. 2023;7(4):1–6. doi:10.55041/IJSREM19015.
22. Singh A, Bacchuwar K, Bhasin A. A survey of OCR applications. Int J Mach Learn Comput. 2012;2(3):314–8. doi:10.7763/IJMLC.2012.V2.137.
23. Song W, Shi C, Xiao Z, Duan Z, Xu Y, Zhang M, et al. Autoint: automatic feature interaction learning via self-attentive neural networks. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management; 2019 Nov 3–7; Beijing, China. p. 1161–70.
24. Xie G, Li Q, Jiang Y. Self-attentive deep learning method for online traffic classification and its interpretability. Comput Netw. 2021;196(5):108267. doi:10.1016/j.comnet.2021.108267.
25. Chandu VE, Harshitha C, Sree VN, Ganesh E. Object detection and estimating the distance between detected objects using deep learning algorithms. Int J Sci Res Eng Manag. 2023;7(4):1–5.
26. Kang CH, Kim SY. Real-time object detection and segmentation technology: an analysis of the YOLO algorithm. JMST Adv. 2023;5(2):69–76. doi:10.1007/s42791-023-00049-7.

27. Asaju CB, Owolawi PA, Tu C, Wyk EV. Cloud-based license plate recognition: a comparative approach using you only look once versions 5, 7, 8, and 9 object detection. Information. 2025;16(1):57. doi:10.3390/info16010057.
28. Ashbaugh B, Boitano J. Advantages and disadvantages of controller designs using fuzzy logic. [Internet]. [cited 2025 Mar 23]. Available from: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=d51e8f09834c8ba5791b1e15b9dbc36c6d21b7ec.
29. Lubna Mufti N, Shah SAA. Automatic number plate recognition: a detailed survey of relevant algorithms. Sensors. 2021;21(9):3028. doi:10.3390/s21093028.
30. Moussaoui H, Akkad NE, Benslimane M, El-Shafai W, Baihan A, Hewage C, et al. Enhancing automated vehicle identification by integrating YOLO v8 and OCR techniques for high-precision license plate detection and recognition. Sci Rep. 2024;14(1):14389. doi:10.1038/s41598-024-65272-1.
31. Podorozhniak A, Liubchenko N, Sobol M, Onishchenko D. Usage of mask R-CNN for automatic license plate recognition. Adv Inf Syst. 2023;7(1):54–8. doi:10.20998/2522-9052.2023.1.09.
32. Reswara E, Suakanto S, Putra SA. Comparison of object detection algorithm using YOLO vs. faster R-CNN: a systematic literature review. In: Proceedings of the 2023 6th International Conference on Big Data Technologies; 2023 Sep 22; Qingdao, China. p. 419–24.
33. Aboyomi DD, Daniel C. A comparative analysis of modern object detection algorithms: YOLO vs. SSD vs. faster R-CNN. ITEJ (Inf Technol Eng J). 2023;8(2):96–106. doi:10.24235/itej.v8i2.123.
34. Poudel U, Regmi AM, Stamenkovic Z, Raja S. Applicability of ocr engines for text recognition in vehicle number plates, receipts and handwriting. J Circuits Syst Comput. 2023;32(18):2350321. doi:10.1142/S0218126623503218.
35. Kathirvel A, Blesso Danny J, Gandu Shalem Preetham, Joe Hinn TO, Roak Kennedy C, Aldrin Immanuel J. Systematic number plate detection using improved YOLOv5 detector. In: 2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN); 2023; Vellore, India. p. 1–6.
36. Anand K, Nath MN, Naidu N. Virtual toll booth based on number plate recognition system using Yolo V8 and easy OCR. In: 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI); 2023 Dec 21–23; Chennai, India. p. 1–8.
37. Huang R, Pedoeem J, Chen C. YOLO-LITE: a real-time object detection algorithm optimized for non-GPU computers. In: 2018 IEEE International Conference on Big Data (Big Data); 2018; Seattle, WA, USA. p. 2503–10.
38. Kumar chinnaiyan V, Balaji S, Beny R, Kavin jayasuriya E. Automatic number plate recognition system. Grenze Int J Eng Technol. 2021;7(1):905–8.
39. Sun T, Chen H, Liu H, Lou H, Duan X. HPS-YOLOv7: A high precision small object detection algorithm. Res Sq. 2023. doi:10.21203/rs.3.rs-2813484/v1.
40. Duan C, Li X. Multi-target tracking based on deep sort in traffic scene. J Phys: Conf Ser. 2021;1952(2):022074. doi:10.1088/1742-6596/1952/2/022074.
41. Raj S, Gupta Y, Malhotra R. License plate recognition system using YOLOv5 and CNN. In: 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS); 2022; Coimbatore, India. p. 372–7.