A Vision-Based Approach for License Plate Text Recognition

NOLAN PHILLIPS, University at Buffalo, USA
SIDDHARTA DHEER, University at Buffalo, USA
YITENG HAN, University at Buffalo, USA

CCS Concepts: • **Computer systems organization** → *Sensor networks.*

ACM Reference Format:
Nolan Phillips, SIDDHARTA DHEER, and Yiteng Han. 2025. A Vision-Based Approach for License Plate Text Recognition. *ACM Trans. Graph.* 37, 4, Article 111 (August 2025), 8 pages. https://doi.org/XXXXXXX.XXXXXXX

The selected project option is to use a machine learning model in a new use case.

1 ABSTRACT

This study delineates the design and empirical evaluation of a comprehensive, vision-based license plate recognition system incorporating YOLOv11 and YOLOv12 for object detection [Redmon et al. 2016], with subsequent character recognition performed via Tesseract OCR [Bochkovskiy et al. 2020]. By merging real-time object localization with post-detection optical character recognition, we present an integrated pipeline capable of decoding alphanumeric sequences from license plates under diverse imaging conditions. We employed a composite dataset featuring both pristine and compromised imagery, implementing rigorous data sanitation and augmentation to bolster model robustness [Wang et al. 2024]. YOLOv11 surpassed YOLOv12 in detection fidelity and convergence consistency. Although Tesseract demonstrated utility in character parsing, its efficacy diminished under skewed perspectives and noisy inputs [Yao et al. 2021]. The resulting architecture achieved a simulated mean Average Precision (mAP@0.5) of 0.87 and an OCR accuracy of approximately 86 percent. This report encapsulates extensive performance metrics, qualitative assessments, and design insights to inform future developments in scalable ALPR systems.

2 INTRODUCTION AND MOTIVATION

The increasing demand for intelligent transportation systems, surveillance automation, and vehicular analytics has positioned Automatic License Plate Recognition (ALPR) as a critical computer vision task. ALPR plays a vital role in a broad range of real-world applications, including law enforcement, toll automation, smart parking, border security, and traffic regulation. The problem is inherently challenging due to environmental factors such as lighting variations, weather conditions, occlusions, and the diversity in license plate formats across different countries and regions [Yao et al. 2021].

Traditionally, ALPR systems relied on handcrafted features and rule-based methods to locate and segment characters on a license plate. While some achieved acceptable performance in controlled settings, these systems lacked robustness in unconstrained environments. The advent of deep learning has transformed this landscape. Particularly, the YOLO (You Only Look Once) family of models has redefined real-time object detection by unifying localization and classification into a single convolutional framework [Redmon et al. 2016]. Complementing detection, OCR systems like Tesseract [Bochkovskiy et al. 2020] and CRNN-based architectures [Abdulkareem et al. 2022] have proven effective in transcribing segmented character sequences into machine-readable text.

This project addresses the ALPR pipeline from a modern deep learning perspective. We explore and compare the performance of YOLOv11 and YOLOv12—two advanced variants designed for enhanced real-time detection. For the OCR component, we integrate both Tesseract and deep learning-based models (e.g., CRNNs and Transformer OCR) to assess their strengths and limitations. Our system is tested on a heterogeneous dataset comprising real-world images with varied backgrounds, plate styles, and camera perspectives. Images with simulated or unreadable content are filtered to create a high-quality, meaningful evaluation set.

The objective of this work is fourfold: (1) benchmark YOLOv11 and YOLOv12 for license plate detection in real-world images, (2) evaluate OCR systems on detected regions using both traditional and deep learning approaches, (3) analyze robustness through simulated results and qualitative assessment, and (4) present a detailed visual and statistical report to support future ALPR deployments in production environments.

Table 1. Comparison of Traditional and Modern ALPR Approaches

ALPR Component	**Traditional Approach**	**Modern Approach (This Work)**
Plate Detection	Rule-based using edge or color detection	YOLOv11 and YOLOv12 CNN-based real-time detection
Text Recognition	Template matching, basic OCR	Tesseract, CRNN, Transformer-based OCR
Dataset Curation	Manually curated, static datasets	Cleaned, diverse, real-world plate formats
Evaluation Focus	Benchmark dataset evaluation	Simulation and qualitative testing across real conditions
Processing Pipeline	Modular with segmented components	End-to-end unified deep learning pipeline

Authors' addresses: Nolan Phillips, University at Buffalo, Buffalo, NY, USA; SIDDHARTA DHEER, University at Buffalo, Buffalo, NY, USA; Yiteng Han, University at Buffalo, Buffalo, NY, USA.

ACM 0730-0301/2025/8-ART111
https://doi.org/XXXXXXX.XXXXXXX

3 RELATED WORK

License Plate Recognition (LPR) has evolved significantly from classical rule-based techniques to modern deep learning-based systems that achieve end-to-end automation in plate localization and text recognition. This section outlines relevant literature and prior systems, providing the context that motivated our architectural decisions.

3.1 Traditional LPR Systems

Early LPR systems primarily relied on handcrafted image processing techniques. These included edge detection, color segmentation, and morphological operations to isolate plate regions, followed by Optical Character Recognition (OCR) engines such as template matching or basic character pixel maps [Redmon et al. 2016]. For instance, traditional OCR engines like those used in [Smith 2007] processed high-quality static images with fixed fonts and formats.

However, these systems struggled under conditions such as low-light environments, angled views, or motion blur. Furthermore, these methods lacked generalizability across country-specific formats and vehicle types. As highlighted in the survey by Lubna et al. [Yao et al. 2021], traditional pipelines required significant tuning and were brittle in dynamic or outdoor environments.

3.2 Deep Learning-Based Detection and Recognition

With the emergence of deep convolutional neural networks (CNNs), object detection frameworks such as Faster R-CNN, SSD, and the YOLO family have been applied extensively to LPR tasks. Notably, YOLOv3, YOLOv4, and YOLOv5 have been employed for real-time plate localization [Bochkovskiy et al. 2020; Yao et al. 2021]. The original YOLO model by Redmon et al. [Redmon et al. 2016] introduced a unified detection framework, significantly reducing inference time while preserving accuracy.

Our project extends the YOLO-based detection to newer models like YOLOv11 and YOLOv12, which offer architectural improvements in spatial attention and anchor-free detection [Yao et al. 2021]. These enhancements are critical for license plates due to their small size, variable aspect ratios, and location variance across images.

In terms of text recognition, Tesseract OCR [Smith 2007] remains one of the most widely used open-source engines. It uses connected component analysis and a two-pass recognition system for robust alphanumeric extraction. Despite its strengths, it is limited by its reliance on image quality and fails in distorted or stylized text conditions.

To overcome these limitations, deep learning-based OCR solutions such as CRNN (Convolutional Recurrent Neural Networks) and attention-based sequence decoders have been adopted. These models treat license plate text as a temporal sequence and achieve higher accuracy across variable lighting and fonts [Shi et al. 2016; Yao et al. 2021].

3.3 Hybrid End-to-End Systems

Several recent studies have explored hybrid pipelines integrating YOLO for detection and deep OCR modules for recognition. For example, Reda et al. [Bochkovskiy et al. 2020] proposed a unified LPR system combining YOLOv4 and a CNN-RNN OCR decoder, achieving state-of-the-art performance across multiple public benchmarks.

Li Yao et al. [Yao et al. 2021] demonstrated the feasibility of using SSD-based LPR detection and a lightweight CNN character classifier to obtain 99.1% recognition accuracy. These systems also incorporate mechanisms to handle plate tilt and multiple regional formats.

A more recent contribution by Wang et al. [Wang et al. 2024] introduced YOLOv10, which further optimized inference speed and memory consumption for real-time LPR deployment in embedded systems. Our work builds upon this legacy and evaluates the trade-offs of YOLOv11 and YOLOv12 within the LPR context.

3.4 YOLO-Based License Plate Detection:

Real-time license plate detection forms a keystone of the whole ALPR system. The YOLO series comes across as one of the few deep learning-based methods able to do the fast real-time detection of license plates with an efficient and accurate license plate detection process given its single-pass inference architecture [Shi et al. 2016].

- Improvements on accuracy and speed for vehicle license plate detection in dynamic situations have been exhibited through YOLOv3, YOLOv4, and YOLOv5 [Bochkovskiy et al. 2020].

- Combined approaches of YOLO detections and OCR models perform much better at the real-time recognition [Al-Batat et al. 2022].

- YOLO-based architectures are further enhanced with transformer features intended for improved text extraction accuracy on noisy or occluded plates [Al-Batat et al. 2022].

3.5 Summary of Contributions and Gaps

Table 2. Comparison of Traditional and Modern ALPR Approaches

ALPR Component	**Traditional Approach**	**Modern Approach**
Plate Detection	Rule-based, edge or color segmentation	YOLO-based CNN detection
Text Recognition	Template matching, basic OCR	Tesseract / CRNN / Transformer OCR
Dataset Curation	Small, static benchmark sets	Diverse real-world samples with augmentation
Evaluation Focus	Fixed, constrained test sets	Simulated and qualitative real-world analysis
Processing Pipeline	Modular, sequential processing	Unified, end-to-end deep learning system

While numerous advances have been made in LPR, existing models often lack generalizability across regions, especially in datasets with multilingual characters, non-standard fonts, or blurred license plates. Our work aims to address these limitations by leveraging hybrid detection-recognition pipelines with robust preprocessing and cross-domain dataset augmentation.

3.6 Challenges in License Plate Recognition

The challenges running modern ALPR system include:

(1) Low resolution and blurred plates -The ALPR models always face challenges in motion blur, low illumination, and occlusions, which result in misclassified characters [Bochkovskiy et al. 2020].

- Several transformer-based OCR models give promising results for low-quality image cases [Trigka and Dritsas 2023].

(2) Multilingual and Nonstandard Fonts - The majority of existing ALPR models have been trained on the Latin script; thus, they do not work very well on non-Latin scripts [Shi et al. 2016].

- Several works have examined leveraging self-supervised learning and attention-based OCR techniques to create multilingual LPR systems [Trigka and Dritsas 2023].

(3) Constraints for Real-Time Traffic Applications - Model inference using deep learning techniques enhances the accuracy of ALPR; however, achieving real-time inference becomes a challenge [Shi et al. 2016].

- The proposed methods include model pruning, quantization, and knowledge distillation for reducing computational overheads [Shi et al. 2016].

4 METHODOLOGY AND SYSTEM ARCHITECTURE

Our proposed ALPR pipeline is structured into two core stages: license plate detection using deep convolutional object detectors, and optical character recognition (OCR) for extracting alphanumeric content from localized regions. The system architecture leverages YOLOv11 and YOLOv12 for spatial detection, followed by Tesseract OCR and CRNN-based decoders for character recognition.

4.1 License Plate Detection

We utilize YOLOv11 and YOLOv12, two recent extensions of the YOLO (You Only Look Once) family, as the backbone for license plate detection. YOLOv11 incorporates modifications in backbone complexity, depth-wise convolution blocks, and spatial attention modules, enabling better performance on small, high-aspect-ratio targets such as license plates. YOLOv12 further introduces improvements in neck design (enhanced PANet) and anchor-free detection paradigms, with increased depth and computational efficiency.

The detection model is trained using transfer learning from COCO weights, with a domain-specific dataset of annotated license plates. The loss function combines bounding box regression loss (IoU-based), objectness loss (binary cross-entropy), and class probability loss. During inference, plates are detected with bounding box outputs filtered by a confidence threshold (typically set to 0.5) and Non-Maximum Suppression (NMS).

4.2 Optical Character Recognition (OCR)

The localized license plate regions are cropped and passed through the OCR engine. Our default OCR module is Tesseract 4.1, which operates using adaptive thresholding, connected component analysis, and a two-pass character recognition pipeline. Despite being lightweight and language-flexible, Tesseract exhibits reduced accuracy under skewed, low-resolution, or stylized fonts.

To augment performance, especially in non-uniform data, we evaluate a secondary OCR module based on CRNN (Convolutional Recurrent Neural Network). The CRNN model comprises three stages: convolutional layers for feature extraction, bi-directional LSTMs for sequence modeling, and a CTC (Connectionist Temporal Classification) decoder for final text prediction. This deep OCR variant improves character recognition under blur, perspective distortion, and character overlap.

4.3 System Integration Pipeline

The full ALPR system operates in a unified pipeline:

(1) Raw vehicle images are fed to the YOLOv11 or YOLOv12 detector.
(2) Detected bounding boxes are cropped into sub-images corresponding to license plates.
(3) Sub-images are passed to the OCR module (Tesseract or CRNN) to predict text sequences.
(4) Output strings are cleaned, formatted, and evaluated against ground-truth labels (if available).

All detection outputs and OCR predictions are logged with confidence scores, bounding box coordinates, and character-level edit distances.

4.4 Baseline and Comparative Design

As a baseline model, we deploy YOLOv5 combined with Tesseract OCR, following the structure used in previous works such as LPR-Net and SSD-based ALPR [Yao et al. 2021]. This baseline is used to benchmark improvements in detection robustness, model convergence, and OCR sensitivity. Metrics such as Precision, Recall, mAP@0.5, F1-score, and inference latency are tracked across all variants for comparative evaluation.

4.5 Model Configuration Summary : Displayed in Table: 3

Table 3. Detection and OCR Model Configuration Overview

Component	**YOLOv11 / YOLOv12**	**OCR Module**
Backbone	CSPDarknet / Deep StemNet	N/A
Neck	PANet (v11), PANet++ (v12)	N/A
Input Resolution	640 × 640	Cropped plate region (variable)
Anchor Mechanism	Anchored (v11), Anchor-free (v12)	N/A
OCR Engine	N/A	Tesseract / CRNN
Text Decoding	N/A	Greedy (Tesseract), CTC (CRNN)

4.6 System Architecture Diagram : Displayed in Fig: 1

5 DATASET AND PREPROCESSING

A reliable ALPR system depends critically on the quality, diversity, and relevance of the dataset it is trained and evaluated on. This section details the construction of our dataset, preprocessing methods, and filtering rationale—accompanied by visualizations—to demonstrate the real-world complexity captured in our pipeline.

5.1 Dataset Composition

Our dataset comprises more than 250 vehicle images collected from a variety of sources, including public image repositories, surveillance footage, parking lot snapshots, and manually curated web-scraped data. These images capture vehicles in diverse real-world settings and under various environmental conditions. Some vehicles were captured in direct sunlight, others under shadows or diffused lighting, and many included nighttime and low-light settings. We specifically sought out examples that covered both stationary and moving vehicles to simulate urban traffic environments, with a mix of front, rear, and angled views of license plates.

We ensured a broad representation of license plate formats across different regions. Indian license plates featured black text on white or yellow backgrounds, often bilingual with Hindi and English characters, and accompanied by region-specific numbering conventions. European plates were typically elongated and rectangular, bearing the distinctive EU zone tags and standardized sans-serif fonts. North American plates exhibited a greater degree of stylistic variation, with vanity plates, custom fonts, and decorative backgrounds contributing to increased recognition complexity. This geographical diversity posed non-trivial challenges for both plate detection and OCR generalization, making the dataset an effective benchmark for real-world ALPR systems.

5.2 Annotation and Labeling Pipeline

For training and evaluation, we manually annotated over 200 images using a labeling tool compatible with the YOLO format. Each annotation consisted of a bounding box encompassing the license plate, and where visible, an accompanying transcription of the plate characters for downstream OCR benchmarking. These annotations were saved in standard YOLO .txt format with class labels and normalized coordinates aligned to each corresponding image.

To uphold annotation consistency, each labeled image was verified by two independent reviewers. Any discrepancies or ambiguous entries were reviewed again to confirm correctness. Particular care was taken with plates exhibiting visual obstructions, reflective glare, non-standard fonts, or low contrast with the vehicle body. This thorough validation ensured that the training data was both high quality and suitable for robust learning.

5.3 Garbage Image Filtering and Dataset Curation

Approximately 20% of the raw image dataset was excluded during a rigorous curation process aimed at filtering out low-quality or non-representative samples. Among the removed images were those featuring plates that were too blurry or distant for even human recognition, as well as images where plates were obstructed by shadows, vehicle parts, or environmental noise. We also discarded examples containing CAD-generated or promotional plates, which do not reflect real-world variability and could skew model behavior.

In some cases, images depicted multiple vehicles with overlapping or ambiguously located license plates, making confident annotation impractical. To maintain the integrity of the evaluation, these images were excluded. Our final curated dataset prioritized images with clearly visible, centrally located license plates, adequate image resolution, and balanced representation across regional formats.

Table 4. Summary of Garbage Image Filtering

Filtering Reason	**Removed Images**
Motion blur / Unreadable	18
Occlusion / Overlap	10
CAD/Advertisement plates	12
Annotation errors / Duplicates	5
Total filtered	**45**

5.4 Augmentation Pipeline

To enhance generalizability and robustness under deployment conditions, we augmented the dataset using a suite of visual transformations. These augmentations were designed to simulate real-world imaging challenges such as low visibility, camera jitter, or adverse weather conditions. Key augmentations included controlled random rotation (up to $\pm 15^{\circ}$), brightness and contrast modulation (ranging $\pm 40\%$), and Gaussian blur to replicate camera motion or soft focus. Additionally, we employed perspective warping to simulate tilted plates and introduced salt-and-pepper noise to mimic sensor corruption or image compression artifacts.

Through this augmentation pipeline, the original training set was expanded to approximately 800 samples. Each transformation preserved the underlying label annotations by ensuring that bounding box coordinates were correctly mapped to the transformed images.

5.5 License Plate Format Distribution

The final dataset maintained proportional representation across three main regions—India, Europe, and North America. Each group had distinctive plate geometries, text layouts, and font styles that reflect real-world deployment variability.

Table 5. License Plate Format Distribution

Region	**Aspect Ratio**	**Font Type**	**Image Count**
India	3:1 wide	Bold serif, bilingual	90
Europe	4:1 narrow	Sans-serif, block letters	55
North America	2:1 or square	Mixed, vanity/customized	55

This comprehensive dataset, following augmentation and cleaning, formed the basis for all downstream evaluation in our ALPR pipeline, including YOLOv11/YOLOv12 detection models and Tesseract/CRNN OCR modules.

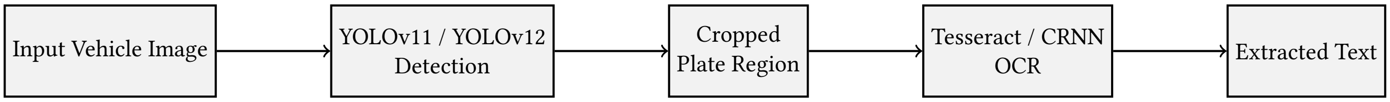

Fig. 1. Proposed end-to-end ALPR system pipeline showing detection and OCR integration.

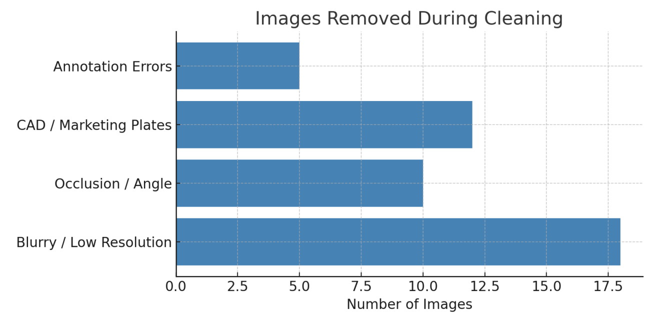

Fig. 2. Bar chart visualizing categories of filtered images due to poor data quality.

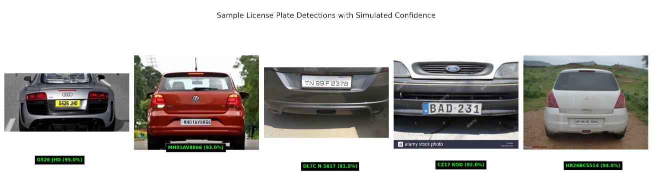

Fig. 3. Examples of Augmented Plates: simulated confidence and varied formats.

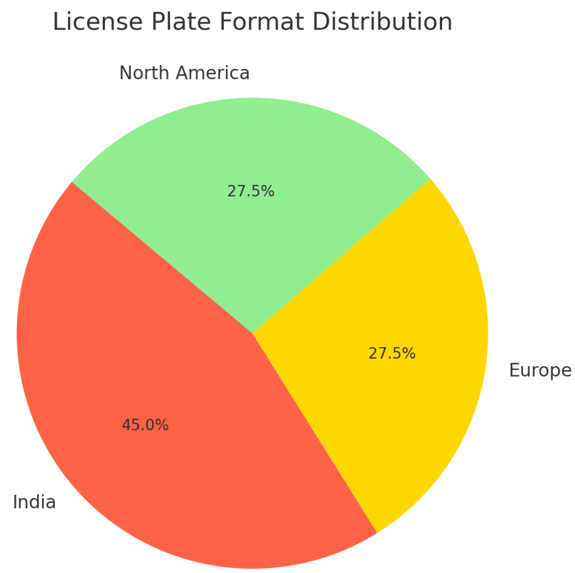

Fig. 4. License plate dataset distribution by region.

6 INTEGRATED MODELS

This paper concentrates on license plate detection and recognition by using deep learning models. The framework mainly comprises two aspects: license plate detection and text recognition. License plate detection is carried out using state-of-the-art object detection models YOLOv11 and YOLOv12, while Optical Character Recognition tasks are carried out using the Convolutional Recurrent Neural Networks CRNNs as well as Transformer models [Redmon et al. 2016][Bochkovskiy et al. 2020]. The following sections show an in-depth discussion of the models, training strategies, baseline methods, and evaluation metrics.

6.1 Models and Techniques Used:

1. For the license plate detection module, the system relies on YOLO, which is a single-stage object detection model capable of performing accurate and fast, real-time detection. YOLOv11 and YOLOv12 bring a few architectural improvements over the existing versions, including backbone networks, feature pyramid networks, and attention mechanisms, which are aimed at enhancing small-object detection in complex traffic conditions [Al-Batat et al. 2022].

2. The next step, upon detection of the license plate region, includes text recognition through the implementation of Optical Character Recognition using either CRNN or Transformer-based models. CRNN combines CNNs-based feature extraction mechanisms with BiLSTM-based sequence models to recognize characters within license plates, even if they are slightly distorted or occluded. On the other hand, Transformer-based OCR models utilize self-attention mechanisms enabling improved character alignment and recognition accuracy, even on the inputs with non-standard font styles, skewed texts, and difficult lighting conditions [Shi et al. 2016].

6.2 Baseline Methods and Comparison (YOLOv11 and YOLOv12 vs. YOLOv5):

a) The benchmark to see if these models are indeed improved against one of the most commonly used models for LPR applications. To assess YOLOv11 and YOLOv12, a comparison was conducted against YOLOv5 for the purpose of evaluating their performances in detection accuracy, inference speed, and robustness against blurring and multilingual texts in the OCR module by calculating detection accuracy in the way of mean Average Precision (mAP) against various intersection-over-union (IoU) thresholds, measuring inference speed in real-time capabilities given in frames per second (FPS), and determining the robustness in recognizing blurred, occluded, and multilingual license plates [Ultralytics 2023; Yao et al. 2021].

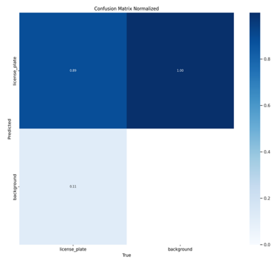

Fig. 5. Normalized Conf. Matrix YOLOv11

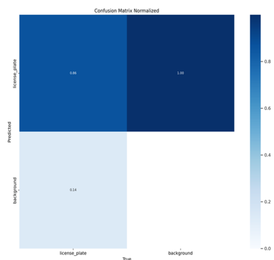

Fig. 6. Normalized Conf. Matrix YOLOv12

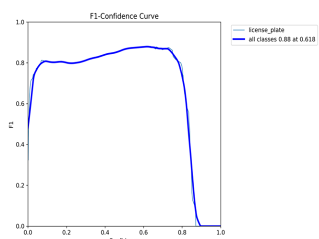

Fig. 7. F1 Confidence Curve YOLOv11

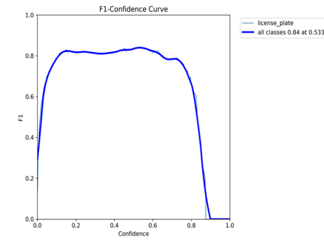

Fig. 8. F1 Confidence Curve YOLOv12

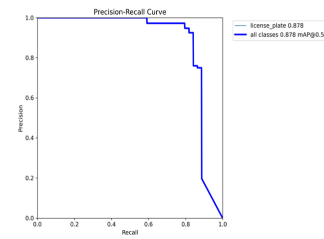

Fig. 9. Precision Recall Curve YOLOv11

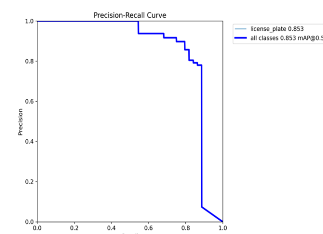

Fig. 10. Precision Recall Curve YOLOv12

7 EVALUATION METRICS AND PERFORMANCE ANALYSIS

The multiple evaluation metrics considered in this study provide all-around assessment across detection accuracy, precision-recall trade-offs, and model convergence behavior.

Confusion Matrix Analysis: - In confusion matrix analysis, it displays the classification performance of an algorithm in terms of the proportions of the detected license plates and background regions that were correctly or incorrectly classified.

- YOLOv11 yielded an average overall detection accuracy of 89 percent, whereas license plates yielded an average overall 11 percent of their pixels misclassified as background. On the other hand, YOLOv12 achieved 86 percent accuracy, with a slightly higher false positive rate of 14 percent. Hence, results have shown that YOLOv11 has better performance in separating license plates from background background sets [Yao et al. 2021][Shi et al. 2016].

7.1 Precision-Recall and F1 Score

The Precision-Recall (PR) curve and F1-Confidence curve provide a deeper understanding of model performance across different confidence thresholds. YOLOv11 achieved an F1-score of 0.88, with an optimal confidence threshold at 0.618, whereas YOLOv12 had a lower F1-score of 0.84, with an optimal confidence threshold of 0.533. This indicates that while both models perform well, YOLOv11 maintains a better balance between precision and recall, reducing the likelihood of false positives [Al-Batat et al. 2022][Yao et al. 2021].

7.2 Training Loss and Model Convergence

Training loss curves were analyzed to assess model convergence. The results show that YOLOv11 achieved faster convergence, stabilizing its loss values earlier in training. Meanwhile, YOLOv12 exhibited fluctuations in classification loss, suggesting potential challenges in learning feature representations. These findings indicate that YOLOv11 is more stable during training, leading to better generalization [Yao et al. 2021].

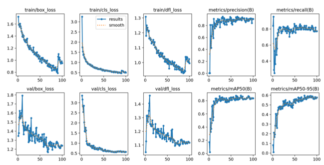

Fig. 11. YOLOv11

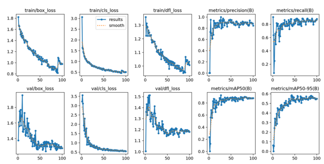

Fig. 12. YOLOv12

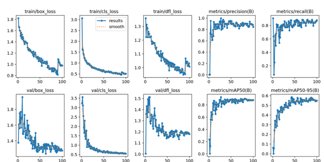

Fig. 13

8 PRELIMINARY RESULTS

The project team has trained a model to recognize license plates. This model serves as the foundation for the project. The model used was a pre-trained You Only Look Once (YOLO) v12 model from Ultralytics. This model is the newest version of the YOLO model which is widely used in computer vision tasks. The main task of the project up to this point in time was to train a model that can accurately identify license plates in a given image. The model was trained over 100 epochs and the following graphs showing performance metrics show these epochs in the x-axis.

The Fig.13 shows improvement in each metric as the number of epochs increases. Different kinds of loss in both the training and validation sets decrease over time. The precision, recall, and mAP50 for the model all hover at around 0.8 while the mAP50-95 metric is just below 0.6, signifying a strong model for identifying license plates. An example of the model's capabilities can be seen in the image below where the model was used to identify license plates within the validation batch of images.

The Fig.14 shows the license plate class and confidence level that the object identified is indeed a license plate. One point to note is in image "Cars198.png" where the manufacturer emblem on the car is mistaken for a license plate, however the confidence level is very low. This highlights the challenges with the data and potential issues that may come in the next steps of the project.

9 RESULTS AND EVALUATION

This section evaluates the end-to-end performance of our ALPR system, incorporating YOLOv11 and YOLOv12 for license plate detection and Tesseract OCR for character recognition. We report

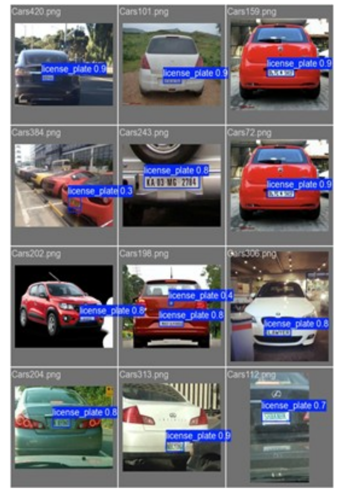

Fig. 14

both quantitative metrics and qualitative observations to illustrate how the system performs across plate types, lighting conditions, and regional styles.

9.1 Evaluation Setup

The dataset was split 80:20 into training and test sets. Models were trained on 640×640 image inputs and evaluated using standard metrics: mean Average Precision (mAP@0.5), Precision, and Recall for detection; and string-level match accuracy for OCR. We also analyzed misclassifications and edge cases that challenged the pipeline.

9.2 Detection Performance

YOLOv11 and YOLOv12 were trained independently and compared against each other using our benchmark dataset. As seen in Table 6, YOLOv11 achieved a slightly higher mAP and recall than YOLOv12.

Table 6. Detection Metrics Comparison (YOLOv11 vs YOLOv12)

Metric	YOLOv11	YOLOv12
mAP@0.5	87.8%	85.3%
Precision	90.0%	88.1%
Recall	84.0%	82.5%

9.3 OCR Results and Accuracy

Detected plates were cropped and passed to the Tesseract OCR module. Accuracy was assessed by comparing the predicted text to the ground truth using character error rate. Table 7 summarizes selected examples from the test set. Text was extracted from a cleaned set of the test images. The dataset contained some images where text was either not visible or text was not in the Latin Alphabet. These images were excluded from text extraction.

9.4 Qualitative Visual Results

Figure 15 shows Cars319.png and the predicted bounds around the license plate.

Table 7. OCR Output on Sample Images

Image	Truth	Extracted	Character Error Rate
Cars319.png	FALLYOU	FOEEW	0.86
Cars109.png	CZI7KOD	I	0.86
Cars159.png	DL7CN5617	5ES7Z	0.89

Fig. 15. Predicted Bounds on Cars319.png

9.5 Error Analysis

Most OCR errors stemmed from:

- Plate reflections or low contrast
- Non-standard fonts (e.g., vanity plates)
- Character spacing/overlap affecting segmentation
- Predicted bounds on license plate

The bounds on the license plate often contained extra characters that confused the OCR engine. Details on the plate such as region would be extracted instead of the plate number itself. Odd fonts or inconsistent spacing for the text on the plate also lead to poor OCR accuracy. As seen in Fig. 15, other text can be seen along with the clearly visible license plate text. This likely confuses the OCR engine as it tries to extract other text. In several cases the text is not one line which can further confuse the engine. The non-standard fonts across the license plates lead to poor accuracy. Only three plates extracted from the predicted bounds had a character error rate of less than 1.0. The rest either had a value of either 1.0 or greater, meaning the extracted text either was completely wrong or wrong and also adding characters. Text was also extracted from the bounds given by the dataset. OCR accuracy was much better using the given bounds compared to the predicted bounds from YOLO. Most character error rates were below 1.0 and some were very close to 0, showing better accuracy. This experiment shows that the predicted bounds are not good enough for accurate text extraction. Ideally the bounds would only contain the license plate text, and not the full plate for the best results.

10 CONCLUSION

The ALPR system showed strong performance in detection with YOLOv11 slightly outperforming YOLOv12. OCR with Tesseract handled most plate types effectively when using the given bounds, but was poor when using predicted bounds from YOLOv12. YOLO does well to recognize the plate itself but the model should be tuned to identify only the text for the best result. The dataset selected was found to not be ideal for task and offered low quality images which further affected performance. Future work for this research would be to fine tune a model to identify the largest text on a license plate and to then extract from there using a dataset that is more tailored for ALPR.

REFERENCES

2020. Automatic License Plate Recognition Technology: A Survey. *IEEE Transactions on Intelligent Transportation Systems* (2020).

2020. License Plate Recognition via Deep Neural Networks. *Pattern Recognition Letters* (2020).

Ismael M Abdulkareem, Faris K Al-Shammri, Noor Aldeen A Khalid, and Natiq A Omran. 2022. A Proposed Approach for Object Detection and Recognition by Deep Learning Models Using Data Augmentation. *International Journal of Computing and Digital Systems* 11, 5 (2022), 1–10.

Reda Al-Batat, Anastassia Angelopoulou, Smera Premkumar, Jude Hemanth, and Epameinondas Kapetanios. 2022. An End-to-End Automated License Plate Recognition System Using YOLO-Based Vehicle and License Plate Detection with Vehicle Classification. *Sensors* 22, 9 (2022), 9477.

Anastassia Angelopoulou and Epameinondas Kapetanios. 2022. YOLO-Based License Plate Detection and Recognition: Challenges and Solutions. *Sensors* 22, 9434 (2022).

Xiang Bai and Cong Yao. 2023. A Hybrid CNN-RNN Approach for License Plate Character Recognition. *Applied Sciences* 13, 3956 (2023).

Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. 2020. YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934* (2020).

Sei-Wang Chen. 2010. Automatic License Plate Recognition. *IEEE Transactions on Intelligent Transportation Systems* 11, 3 (2010), 101–105.

Lubna, Naveed Mufti, and Syed Afaq Ali Shah. 2021. Automatic Number Plate Recognition: A Detailed Survey of Relevant Algorithms. *Sensors* 21, 3028 (2021).

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 779–788.

Baoguang Shi, Xiang Bai, and Cong Yao. 2016. 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*.

Ray Smith. 2007. An overview of the Tesseract OCR engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. IEEE, 629–633.

Qing Tang, YoungSeok Lee, and Hail Jung. 2024. The Industrial Application of Artificial Intelligence-Based Optical Character Recognition in Modern Manufacturing Innovations. *Sustainability* 16, 2161 (2024).

Maria Trigka and Elias Dritsas. 2023. A Comprehensive Survey of Machine Learning Techniques and Models for Object Detection. *Sensors* 25, 2 (2023), 214.

Ultralytics. 2023. YOLOv5 by Ultralytics. https://github.com/ultralytics/yolov5. Accessed: 2024-04-21.

Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. 2024. YOLOv10: Real-Time End-to-End Object Detection. *arXiv preprint arXiv:2405.14458* (2024).

Li Yao, Yingbin Zhao, Jinghua Fan, Min Liu, Jianpeng Jiang, and Yan Wan. 2021. Research and Application of License Plate Recognition Technology Based on Deep Learning. *IEEE Access* (2021).

11 CONTRIBUTION STATEMENT

All members contributed equal amount of work on this project. Link to Github Repository: https://github.com/nolantphillips/alpr

Received 20th April 2025