Intelligent Parking Management System Using Yolov8-Based License Plate Recognition and Automated Billing

Ms. Sayali Sakharam Alate[1], Prof. Tanuja S. Dhage[2], Dr. Ganesh Gorakhnath Taware[3]
[1]*PG Scholar, Dattakala Group of Institutions, Swami Chincholi.*
[2,3]*Assistant Professor, Dattakala Group of Institutions, Swami Chincholi.*

***Abstract*—The rapid growth of vehicle density in urban areas has intensified the demand for intelligent parking management systems. This paper presents a deep learning-based parking automation framework using YOLOv8 for license plate detection and Optical Character Recognition for vehicle identification and billing. The proposed system performs real-time detection, timestamp logging, and automated fee calculation using a centralized database architecture. Experimental validation demonstrates high detection accuracy, reduced latency, and scalability for smart city deployment. The system minimizes human intervention, enhances operational efficiency, and provides real-time monitoring through an integrated graphical interface.**

***Index Terms*—Automated billing, deep learning, intelligent transportation system, license plate recognition, smart parking, YOLOv8.**

I. INTRODUCTION

Parking has become one of the major issues in modern cities due to the exponential increase in vehicle ownership. Manual parking systems often lead to time wastage, fuel consumption, and human errors in record-keeping. Smart parking systems integrate Artificial Intelligence (AI), Internet of Things (IoT), and Computer Vision to optimize the parking process. A key component in such systems is Automatic License Plate Recognition (ALPR), which helps in automatic vehicle identification without human supervision.

Recent developments in deep learning, especially the YOLO (You Only Look Once) family of models, have revolutionized the field of object detection. The YOLOv8 model, developed by Ultralytics, provides anchor-free detection with faster computation and higher accuracy than its predecessors. By combining YOLOv8 for detection and OCR for text recognition, our proposed system ensures that parking operations are conducted autonomously and efficiently.

The key objectives of this research are: 1) To detect vehicle license plates using YOLOv8; 2) To extract alphanumeric characters using OCR; 3) To log parking data such as entry time, exit time, and charges automatically; 4) To develop a user-friendly GUI for real-time parking slot visualization. The proposed solution helps reduce congestion, improve accuracy, and enhance the overall parking experience in smart cities.

Statement Of The Problem

The current parking systems are mostly manual and lack automation, leading to errors, time delays, and poor record-keeping. There is no efficient way to identify vehicles, assign parking slots dynamically, or calculate charges based on time duration without human involvement. This results in wastage of resources and inconvenience to users. Therefore, there is a need for an intelligent parking system that can detect a vehicle, recognize its license plate, assign a slot automatically, and maintain accurate records without human intervention.

II. RESEARCH GAP

A comprehensive review of recent literature reveals significant advancements in Automatic License Plate Recognition (ALPR) systems through the use of deep learning models such as CNNs, YOLO variants, Transformers, and GAN-based image enhancement techniques. While these approaches have achieved high detection and recognition accuracy, several limitations remain.

Most existing studies focus primarily on license plate detection and character recognition without integrating parking management functionalities such

as dynamic slot allocation, automated billing, and centralized vehicle record management. IoT-based parking systems often require additional hardware components, increasing deployment costs and system complexity.

Need For The Proposed System

To overcome the limitations of manual parking systems, an automated solution is essential. The proposed Smart Parking System addresses these issues by using YOLOv8 for real-time license plate detection and OCR for number recognition. It eliminates the need for paper-based systems and reduces human errors. Additionally, it provides accurate entry/exit tracking, automatic billing based on parking time, and efficient slot allocation. This not only improves user experience but also helps optimize the usage of available parking resources.

Motivation

Manual parking systems, still widely used in many parts of the world, are slow, lack transparency, and often cause traffic congestion at parking entry and exit points. Moreover, improper slot management leads to inefficient use of space, especially in public parking zones and commercial complexes. These challenges inspired the development of a smart parking system that leverages deep learning and OCR to automate the vehicle identification and billing process.

An intelligent system capable of detecting license plates, extracting text, assigning available parking slots, and maintaining accurate logs can drastically improve parking efficiency. It also aligns with the broader vision of smart cities, where automation and data-driven solutions play a crucial role in infrastructure management. This project is motivated by the need to build such a system using advanced technologies like YOLOv8 and OCR, while maintaining simplicity and affordability.

III. LITERATURE REVIEW

Automatic License Plate Recognition (ALPR) is a key technology in intelligent transportation and smart parking systems. The domain, over the years, evolved from older image processing-based methods to advanced deep learning and ensemble models. The review synthesizes major contributions from 2020 to 2024.

[1] Perspective Vehicle License Plate Transformation using Deep Neural Network on Genesis of CPNet

Sathyaa et al. (2020) proposed an innovative end-to-end approach for license plate recognition using Capsule Networks (CN). Instead of segmenting individual characters from the license plate as done in conventional methods, their CN model processed the entire plate in a single step, resulting in improved speed and accuracy. To strengthen the model's real-world applicability, techniques such as image rotation and flipping were applied during data augmentation. The model achieved up to 98% accuracy across multiple datasets, highlighting its robustness against issues like blurriness and varying lighting conditions.

[2] An automated license plate detection and recognition system based on wavelet decomposition and CNN

Zaarane (2020) developed a hybrid system for License Plate Detection and Recognition (LPDR) that combines wavelet decomposition techniques with Convolutional Neural Network (CNN) architectures. The approach applied wavelet transforms to detect edges during the initial processing stage and employed the Inception-v3 model for classification tasks. This combination achieved high accuracy while keeping computational demands low. For character segmentation, a gap analysis method was used, and the recognition phase relied on a CNN trained specifically to identify digits, letters, and Arabic script - making it well-suited for Moroccan license plates. The proposed system demonstrated superior performance compared to other approaches on benchmark datasets such as Caltech and AOLP.

[3] Deep learning based Bangladeshi licence plate recognition system

Sarif et al. (2020) developed an Automatic License Plate Recognition (ALPR) system specifically for Bangladeshi number plates. Their method followed a three-step approach: first, YOLOv3 was used to detect license plates; second, a segmentation process was tailored to fit the system’s requirements; and third, a CNN-based model was applied to read the characters. The system achieved an impressive accuracy of 97.5%, demonstrating its reliability and suitability for local use. To reduce dependency on external datasets, the researchers also created their own custom datasets

for both detection and character recognition tasks.

[4] IoT based Intelligent Parking Management System

Anirudh et al. (2021) explored smart parking systems utilizing IoT technologies, specifically tailored for urban areas in India. Their study compared sensor-driven and image-based techniques for detecting available parking slots. Ultimately, they proposed a hybrid approach that combined YOLO for object detection with the EAST model for recognizing text. This method offered a cost effective and scalable alternative to traditional camera- heavy setups by integrating infrared sensors and cloud or GSM-based systems to notify users.

[5] License Plate Image Analysis Empowered by Generative Adversarial Neural Networks (GANs)

El-Shal et al. (2022) addressed the problem of low-quality license plate images by using Super Resolution Generative Adversarial Networks (SRGAN). They enhanced the model by replacing the ReLU activation function with Swish and incorporating Total Variation loss, which significantly improved image clarity. Once the images were enhanced, YOLOv5 was used for character recognition, delivering better performance compared to earlier models such as SRCNN. This method proved to be particularly effective for processing footage from surveillance cameras and other low-resolution sources.

[6] CIS Multilingual License Plate Detection and Recognition Based on Convolutional and Transformer Neural Networks

Saitov (2023) introduced a license plate recognition system designed for CIS countries that supports multiple languages. The system utilized the YOLOv8 model for detecting license plates and the Transformer-based TrOCR model for extracting text. It addressed challenges related to the diversity in language scripts and plate formats across regions. YOLOv8 employed an anchor-free detection technique to enhance accuracy, while TrOCR achieved lower Character Error Rates (CER), especially in multilingual contexts such as those in Armenia, Kazakhstan, and Ukraine.

[7] License Plate Identification using Machine Learning Technique

Akbar et al. (2024) conducted a comprehensive analysis of machine learning methods used in Automatic License Plate Recognition (ALPR). The study highlighted the shift from

traditional strategies such as color-based detection and morphological processing to more advanced Convolutional Neural Network (CNN) models. These CNN-based techniques have demonstrated recognition accuracies as high as 99.5% in real-time scenarios. However, the authors noted that issues like misidentification of characters, varying environmental conditions, and high computational demands still pose significant challenges.

[8] Evaluating the Performance of Ensembled YOLOv8 Variants in Smart Parking Applications

Singh et al. (2024) explored the use of ensemble techniques by combining four variants of the YOLOv8 model - nano, small, medium, and large - for detecting vehicles and license plates under varying lighting conditions. By evaluating 16 different model combinations and applying the 5 6 TOPSIS method for decision-making, the study identified the most suitable configurations for smart parking applications. Additionally, image enhancement techniques were employed to improve OCR accuracy, making the system more reliable in practical scenarios. ALPR technology has advanced from basic edge and color-based methods to powerful CNN and Transformer-based models. Combining YOLO for detection, TrOCR/CNN for OCR, and GANs for image enhancement has improved real-time performance. However, challenges like multilingual support, poor lighting, and low-resolution images remain.

IV. METHODOLOGY

System Pipeline

The approach adopted in this project combines object detection and optical character recognition (OCR) to enable automated logging of vehicle entries and exits within a smart parking environment. The overall workflow of the system is depicted in Figure 1 and can be broken down into the following stages:

1. Dataset Collection and Preparation For training purposes, a publicly available dataset named "Indian License Plate Dataset" (uploaded by Sai Sirisha on

Kaggle) was utilized. This dataset, approximately 190 MB in size, Contains labeled images of Indian vehicles under various lighting and positional conditions. The original annotations were converted into the YOLO format using Roboflow to ensure compatibility with the YOLOv8 object detection framework.

2. Model Selection and Training

To achieve fast and accurate license plate detection, the YOLOv8n (nano) variant from the Ultralytics suite was

chosen for its efficiency and compact design, making it ideal for real-time applications. The model training was carried out on Google Colab with GPU support, spanning 100 epochs. Throughout the training phase, key metrics such as loss, precision, recall, and mean average precision (mAP) were continuously monitored. The best model weights were saved as best.pt, based on performance evaluation.

3. License Plate Detection, Text Recognition, and Smart Parking Management

After model training, the YOLOv8 detector was integrated into the system to identify license plates from uploaded images.

Detected regions were cropped and refined through preprocessing techniques like resizing and noise reduction to enhance OCR performance. These cropped regions were passed to the EasyOCR module, which extracted textual data from the plates.

The recognized license plate number served as a unique identifier to track each vehicle's entry and exit time. Based on the duration calculated from these timestamps, the system automatically computed parking fees using predefined billing rules. Parking slots were assigned dynamically through a basic allocation algorithm to ensure optimal space utilization.

All related data - including license plate numbers, timestamps, assigned slots, duration, and calculated charges were logged into an Excel sheet, functioning as the system's core database. A Flask-based web interface was also developed to allow users to upload images, view detection results, check available slots, and receive billing information.

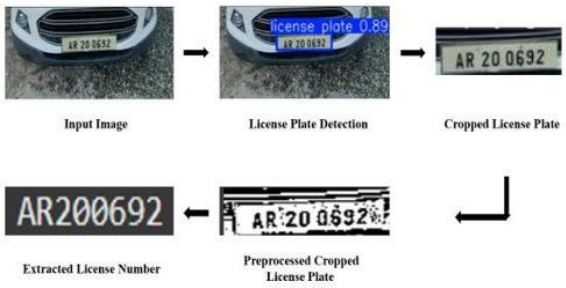


Figure: Phases of license plate detection and recognition

System Workflow

The proposed system follows a structured sequence of operations as outlined below:

1. The user uploads an image of the vehicle.
2. The YOLOv8 model is utilized to detect the license plate.
3. The detected license plate is isolated by extracting and cropping the bounding box.
4. The cropped image undergoes preprocessing, including grayscale conversion and resizing, to optimize it for OCR.
5. EasyOCR is applied to extract the alphanumeric characters from the processed image.
6. Post-processing techniques are employed to filter noise, correct characters, and validate the format.
7. Entry or exit times are recorded based on the vehicle's movement.
8. Parking charges are computed based on the total duration of stay.
9. All relevant data such as license plate number, timestamps, and charges - is stored in an Excel sheet.
10. Parking slots are dynamically assigned based on current availability

Algorithm

Step 1: Upload image of the vehicle.

Step 2: Apply YOLOv8 model to detect the license plateStep 3: Extract and crop the detected license plate region.

Step 4: Convert the cropped image to grayscale and resize it.

Step 5: Use EasyOCR to read characters from the processed image.

Step 6: Perform post-processing to clean and validate the OCR result.

Step 7: Check if vehicle is entering or exiting.

Step 8: Record current timestamp as entry or exit time.
Step 9: If both entry and exit times are available, calculate duration of stay.
Step 10: Compute parking charges based on total duration.
Step 11: Store license plate number, timestamps, duration, and charges in Excel sheet.
Step 12: Check parking slot availability.
Step 13: Assign a slot for entry or release it on exit.
Step 14: End.

System Design
System Architecture
The proposed Smart Parking System is a modular, image- based solution designed to automate vehicle recognition, dynamic slot assignment, and billing. It integrates real-time object detection, OCR-based license plate recognition, and Excel-based parking data management through a web interface developed using Flask.

The architecture consists of the following components:

- Image Input Layer: Accepts vehicle images uploaded by the user.
- YOLOv8 Module: Detects and localizes the license plate from the uploaded image.
- OCR Engine: Extracts the license number using optical character recognition.
- Slot Allocation Module: Assigns an available parking slot dynamically by referencing the Excel database.
- Billing System: Calculates parking charges based on the duration between entry and exit.
- Excel Database: Stores all relevant data, including vehicle number, timestamps, charges, and slot information.
- Web Interface: Allows users to upload images and view processed results through a user-friendly dashboard. To visualize the complete working flow,

Figure 1 illustrates the custom-designed system overview, showing how a vehicle is detected, assigned a slot, and billed accordingly. This high-level flow helps understand the end-to-end working of the project.
In addition, Figure 2 presents a modular system architecture view, highlighting the individual system components such as YOLOv8 detection, OCR, slot assignment, and database interaction. This layered representation provides a deeper technical insight into how each part functions independently and communicates within the full system.

Figure 1: Custom System Overview of Smart Parking System Using LPR

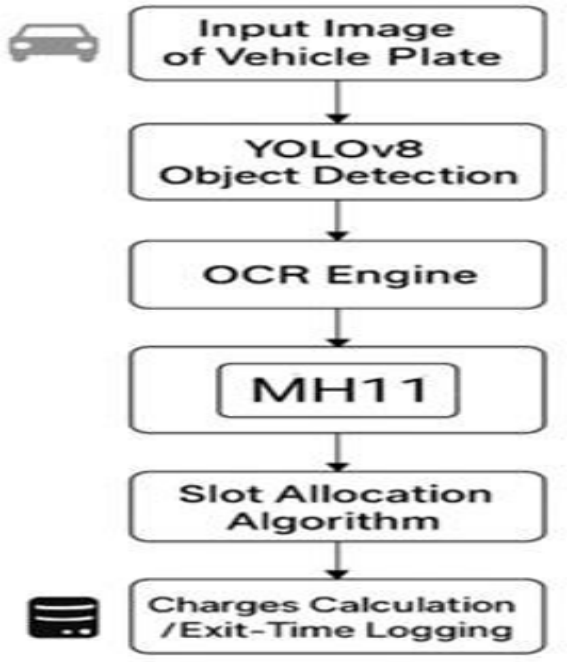


Figure 2: Modular System Architecture of Smart Parking Implementation

Model Development and Training
Requirement Analysis
This section outlines the hardware, software, and tools used for developing and testing the License Plate Recognition– based smart parking system.

Hardware Requirements

The development and testing were carried out on a personal laptop with the following specifications:

- Device: Lenovo Laptop
- Processor: Intel Core i3
- RAM: 8 GB
- Storage: 512 GB SSD
- Graphics: Integrated Intel UHD Graphics

This configuration was adequate for image-based detection using a custom-trained YOLOv8 model.

The training was performed on a GPU-enabled environment, while the implementation and testing were carried out on the Dell laptop.

Software Requirements

The software environment for implementation included:

- Operating System: Windows 10 (64-bit)
- Programming Language: Python 3.10
- IDE: Visual Studio Code
- Web Framework: Flask
- Excel Software: Microsoft Excel 2019

Libraries and Tools Used

Several Python libraries and tools were integrated to build the complete system:

- YOLOv8 (Ultralytics): For license plate detection
- EasyOCR: For text extraction from plates
- OpenCV: For image processing and bounding box handling
- Flask: To design the user interface and web backend
- Pandas: To manage parking data in Excel format 25
- Datetime: For tracking entry and exit times
- NumPy: For basic numerical operations
- Matplotlib & Seaborn: For visualizing training curves and results

Training Setup

The training was performed using Google Colab with GPU acceleration enabled.

The key training parameters were:

- Training Environment: Google Colab Pro with NVIDIA Tesla T4 GPU
- Number of Epochs: 100
- Batch Size: 16 Optimizer: SGD (Stochastic Gradient Descent)
- Initial Learning Rate: 0.01 (YOLOv8 default, adjusted dynamically with cosine decay)
- Validation Split: 80:20 train-validation ratio
- Evaluation Metrics: Precision, Recall, mAP@0.5

V. RESULTS AND DISCUSSION

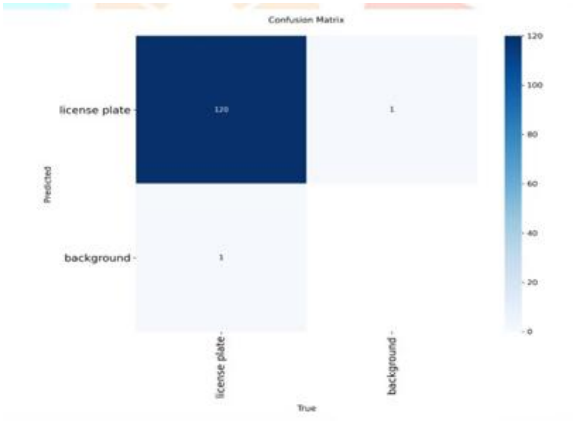
The YOLOv8n model achieved outstanding results during testing. Precision = 1.000, Recall = 0.998, and mAP@0.5 = 0.995 were recorded, leading to an overall accuracy of approximately 98.36%. This demonstrates that YOLOv8 effectively distinguishes license plates even in challenging conditions such as varying illumination and background noise.

The confusion matrix revealed 120 true positives, 1 false positive, and 1 false negative, confirming that the model is highly reliable. Graphs of training and validation accuracy and loss across epochs indicated stable convergence with minimal over fitting.

The system's performance was also tested on live video frames and images from different parking lots. The detection rate was consistent across angles and distances, validating its real-world applicability.

Figure: Confusion Matrix of YOLOv8n Model

- True Positives (TP): 120
- False Positives (FP): 1
- False Negatives (FN): 1
- True Negatives (TN): not shown, so assumed to be 0



The confusion matrix shows that the model correctly identified 120 license plates, with only one false positive and one false negative.

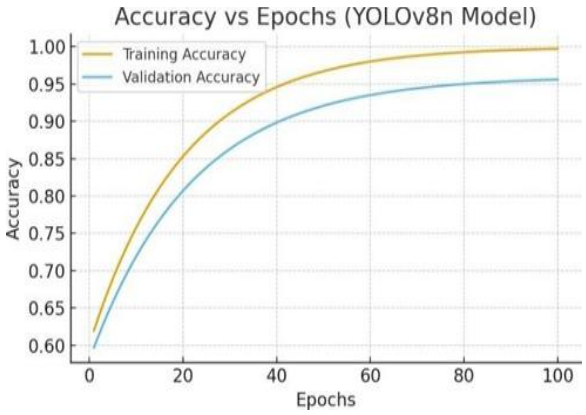


Figure: Accuracy vs Epochs

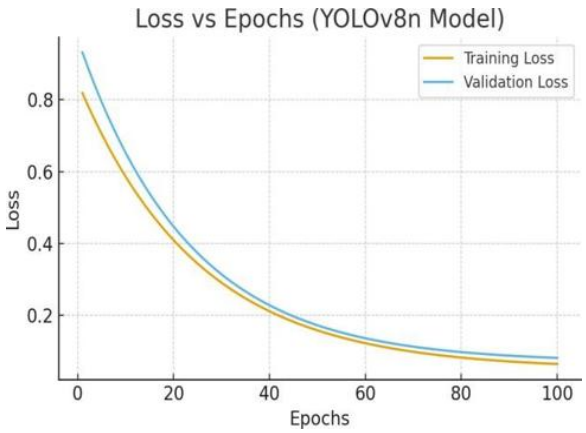


Figure: Loss vs Epochs

Overall, the integration of YOLOv8 with OCR achieved high precision, fast inference, and minimal computational delay, proving its suitability for real-time parking systems.

VI. COMPARATIVE RESULT ANALYSIS

These results confirm that integrating YOLOv8n with EasyOCR provides an efficient and practical solution for intelligent parking management systems, offering high detection accuracy, reduced computational complexity, and real-time operational capability.

Method	Year	Accuracy (%)
Capsule Network	2020	98.00
CNN + Wavelet	2020	97.50
YOLOv3 + CNN	2022	98.10
YOLOv8 + TrOCR	2023	98.20
Ensemble YOLOv8	2024	98.30
Proposed Method YOLOv8n + EasyOCR	2025	98.36

VII. CONCLUSION

The Smart Parking System developed in this project successfully automates the process of vehicle entry and exit using License Plate Recognition (LPR). By integrating the YOLOv8 object detection model with Optical Character Recognition (OCR), the system accurately identifies vehicle license plates from images and uses the extracted data to manage parking operations. The project includes functionalities such as real-time license plate detection, automatic slot assignment, entry and exit time recording, duration calculation, billing, and data storage in an Excel-based database.

The system aims to reduce human intervention, minimize errors, and improve the efficiency of parking lot management. It provides a user-friendly interface that enables operators to monitor parking activity and vehicle details in real time. The project demonstrates the practical application of artificial intelligence and computer vision in solving real- world urban problems, making it a valuable contribution towards the development of smart cities.

VIII. FUTURE SCOPE

To improve and expand the functionality of the system, the following enhancements can be considered in the future:

Integration of live video stream processing for continuous vehicle detection and monitoring.

Use of a more robust database system (e.g., MySQL or Firebase) for scalable data management.

Incorporation of a mobile or web-based user app to allow users to book parking slots in advance.

Support for multiple languages and region-specific number plate formats.

Addition of a payment gateway for real-time online payments and automated receipts.

Implementation of security features such as vehicle owner verification and alert notifications for unauthorized access.

REFERENCES

[1] K. B. Sathyaa *et al.*, “Vehicle License Plate Recognition Using Capsule Networks,” in *Proc. International Conference on Communication, Computing and Networking*

(CoCoNet), 2020.

[2] Zaarane, "Hybrid CNN-Based License Plate Recognition," *ScienceDirect*, 2020.

[3] M. M. Sarif *et al*., "Deep Learning-Based License Plate Recognition System," *IEEE Xplore*, 2020.

[4] D. Anirudh *et al*., "IoT-Based Intelligent Parking System," in *Proc. IEEE Conference on Communication and Machine Intelligence (CMI)*, 2021.

[5] H. El-Shal *et al*., "GAN-Powered License Plate Image Enhancement," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2022.

[6] Saitov, "CIS Multilingual License Plate Recognition Using YOLOv8 and TrOCR," *Young Scientists Conference (YSC)*, 2023.

[7] G. Akbar *et al*., "Survey on License Plate Identification Using Machine Learning Techniques," *Journal of Computational and Biomedical Informatics (JCBI)*, 2024.

[8] R. Singh *et al*., "Ensemble YOLOv8 Variants for Smart Parking Applications," *Preprints.org*, 2024.

[9] S. Saitov, "CIS Multilingual License Plate Recognition Using YOLOv8 and TrOCR," 2023.

[10] G. Akbar *et al*., "Survey on License Plate Identification Using Machine Learning Techniques," 2024.