Kufa Journal of Engineering
Vol. 17, No. 2, April 2026, P.P. 417-436
Article history: Received 16 May 2025, last revised 7 July 2025, accepted 2 August 2025

AN INTELLIGENT SYSTEM FOR IRAQI ARABIC LICENSE PLATE RECOGNITION USING YOLO AND MACHINE LEARNING

Neaam M.Isaam Abdul-Sattar [*] and Fadwa Al Azzo[*]

[*] **Computer Technique Engineering Department, Northern Technical University, Mosul-Iraq, Email:neam.mohammed@ntu.edu.iq.**

https://doi.org/10.30572/2018/KJE/170225

ABSTRACT

Vehicle identification processes through traffic management systems depend heavily on automatic license plate recognition capabilities. The presented system uses advanced methods from deep learning and machine learning to recognize Iraqi vehicle license plates. The system uses the Tkinter-developed graphical user interface alongside YOLO for detection and EasyOCR for text extraction together with the SVM and KNN classifiers for plate classification. The system successfully divides license plates into important parts which enables the identification of governmental entities alongside vehicle class types. The evaluation metrics reveal outstanding classification performance because both SVM and KNN models delivered 100% accuracy together with 100% precision, recall and F1-score. The proposed approach performs well when compared to existing studies because its analysis demonstrates both strength and effectiveness. The obtained findings demonstrate that the system can efficiently serve real-world needs in vehicle identification and law enforcement and intelligent transportation systems.

KEYWORDS

License Plate Recognition (LPR), YOLO, Optical Character Recognition (OCR), Support Vector Machine (SVM), k-Nearest Neighbors (KNN), Iraqi Car Plate, Machine Learning, Deep Learning, Vehicle Classification.

1. INTRODUCTION

Modern transportation management depends heavily on Automatic License Plate Recognition (ALPR) systems because they provide efficient vehicle identification for security enforcement and traffic control as well as automatic toll payment. The detection and identification of license plates through images or video streams becomes possible by utilizing image processing with artificial intelligence in ALPR systems (Kaur et al. 2023). This technology operates at high levels of automation for Latin-script license plates but requires specialized solutions to handle the recognition of Arabic-based plates which are typical to Iraq.

The Arabic layout of Iraq vehicle license plates demands special Optical Character Recognition (OCR) models because it differs from Latin-based plate structures. Segmentation together with recognition processes become more complex because Arabic characters follow a right-to-left writing direction (Hussain and Hathal 2020). The plate design of Iraqi vehicles includes several options that depend both on the vehicle type and registration category as well as the issuing province. The numerous plate variations make it hard to use standard ALPR models because detection along with classification becomes more complex (Omran and Jarallah 2017).

The environmental conditions of Iraq create obstacles that make it harder to deploy automated license plate recognition systems. The combination of dust storms alongside high temperatures and poor lighting together with plate deterioration because of wear and tear reduces the clarity of images and degrades recognition accuracy (S. Ahmed, Ali, and Naser 2023). In practical applications plates sometimes lose visibility due to dirt accumulation or vehicle movement-related blurring which requires strong image quality enhancement techniques before recognition (Mo et al. 2025).

Artificial intelligence technology has brought substantial improvements to ALPR system performance because of deep learning approaches. Single Shot Multibox Detector (SSD) and You Only Look Once (YOLO) within Convolutional Neural Networks (CNNs) demonstrate superior results for object recognition work and qualify them as ideal solutions for ALPR systems (Asaju et al. 2025). The models enable developers to enhance both plate detection performance and text and license number recognition in various challenging situations. By uniting edge computing with real-time processing organizations have created better and speedier deployment options for smart cities to implement ALPR systems (Panganiban et al. 2022).

The investigation of Iraq Arabic license plates has fallen behind international ALPR advancement milestones. Today's AI models operating with Latin script do not give satisfactory results when analyzing Arabic characters and numbers so researchers must build proprietary models. The research project builds an optimized ALPR system for Iraqi license plates by resolving the problems of character segmentation together with insufficient dataset availability and recognition environment factors. Improved ALPR accuracy together with robustness allows the research to bolster innovative traffic monitoring systems and law enforcement capabilities in Iraq thus improving security and operational efficiency in traffic operations.

2. RELATED WORKS

Researchers (Yousaf et al. 2021) created a new Pakistani License Plate Dataset (PLPD) of 6000 labelled images to train the models. Their proposed deep learning pipeline comprises three stages: localization in the You Only Look Once (YOLO) family, rectification using the Multi-Object Rectified Network (MORN), and recognition through an attention-based sequence generation model. The outcomes proved that the model could generalize and attained higher localization (Intersection over Union of 0.89) and recognition performance than any of the existing methods such as DALPR and KLPR. The proposed technique obtained recognition F1-scores of 0.96 on PLPD, 0.94 on AMLP, and 0.91 on RLPD, while better than other methods in different situations. This approach can be used in the future for the changes of other non-standardized license plates as well.

The effectiveness of license plate recognition with RGB images was discussed using grayscale images by (Devapriya et al. 2020). Connected component analysis and template matching were used which had increased recognition accuracy. Their approach has employed morphological operations properly, but the disadvantage of their work is that they have applied only binary images and template matching, which make their work suitable for simple real-world situations only.

Authers (Luo and Liu 2022) proposes a more advanced system for license plate detection and recognition: localization is performed by the YOLOv5m algorithm, while recognizing characters is done by the LPRNet. The K-means++ algorithm is used to select better anchor boxes, DIOU-NMS method is use for better non-maximum suppression and detection layers are minimized or removed for better efficiency. The proposed system was evaluated using the real-life dataset, the Chinese City Parking Dataset (CCPD), with different scenarios such as front-facing, tilted, night, and low light. The performance of the system, according to experimental outcomes, was fairly high: precision of 0,9949, recall of 0,9879, Mean Average Precision (mAP) of 0,9856. Furthermore, the system yielded 42 frames per second, which is faster than YOLOv3-LPRNet, YOLOv4-LPRNet, YOLOv5m-LPRNet in terms of both accuracy and speed. These results prove that the proposed method is accurate, fast and it is ready to cope with real-time LPR challenges in real world scenarios.

Also, (Hajare et al. 2022) highlight a dataset of Indian license plates, with Gaussian filtering and wavelet transform. Using their model, they were able to attain an accuracy of 91% on character recognition. While its strength was the good feature extraction, it relied mostly on standard image processing techniques which can actually deteriorate with complex data sets.
While (Yaba and Omer Latif 2022) utilized the CCTV footage and internet images to identify the number plates based on the KNN and Google Tesseract OCR algorithms. Their system tested their background detection accuracy to be 97.78%, the KNN accuracy to be 92.22% while the OCR accuracy was a meager 45.56%. The system worked well for Arabic digits' recognition and unfortunately the low OCR accurate can be a problem for more sophisticated cases.
Canny filter and OCR were used for license plate recognition on 50 images and the result found was an accuracy of between 92.4% and 96% as proposed by Antar et al. (2022) (Antar et al. 2022). Although the method was successful, there are some drawbacks – the limited amount of data and no account for video data.

3. METHODOLOGY

The block diagram in Fig.1 presents an automated system which identifies Iraqi Arabic vehicle plates through machine learning and deep learning in combination with image processing methods. The approach performs a sequence of several steps beginning with image collection which results in showing classified vehicle information about plate number along with country government origin and vehicle model. The methodology operates as a complete system which performs precise extraction and recognition of license plate data so it addresses format differences and environmental conditions.
The system starts its work by accepting a car image from users. A YOLO (You Only Look Once) image segmentation technique is the first primary process which isolates and retrieves car plates from the provided image. The deep learning-based YOLO object detection system performs quickly to achieve high accuracy object identification thus it demonstrates superior capabilities for real-time license plate recognition. The OCR technology extracts the vehicle number from the identified plate. The system transforms the plate characters into a format that computers can process through this phase. The successfully recognized car number leads to the saving of the plate image for subsequent processing.
The system breaks the plate into different sections after detection by applying coordinates for geometric partitioning. The license plate has three devoted areas to process car government identification and car number recognition as well as car type identification. The vehicle registration information appears in the car government section which shows the province or regional information. The car number section retrieves and provides the numerical values from

license plate information. Vehicle classification in the car type section occurs according to the plate characteristics. The requested segmentation system adequately separates relevant plate information to achieve higher accuracy in recognition procedures.

The system implements Histogram of Oriented Gradients (HOG) feature extraction for car government and car type classification on its datasets. HOG stands as a well-recognized image descriptor because it extracts both structural elements and image texture information which enables effective pattern recognition duties. Dataset preprocessing includes normalization together with noise removal along with image enhancement steps to enhance recognition accuracy of extracted features. The data preprocessing phase makes the machine learning models receive cleaned and optimized data suitable for classification tasks.

The system chooses its machine learning model from an available selection to perform the classification of car governments and car types. Two widely adopted classifiers exist for this task which include Support Vector Machine (SVM) and K-Nearest Neighbors (KNN). SVM operates through supervised learning to establish its best possible decision boundary that divides different categories thus making it a suitable model for classification problems. The KNN algorithm uses a distance-based approach to find labels through the most common class within its nearest neighbors. The chosen model trains using cataloged images which include different car government names alongside various car types.

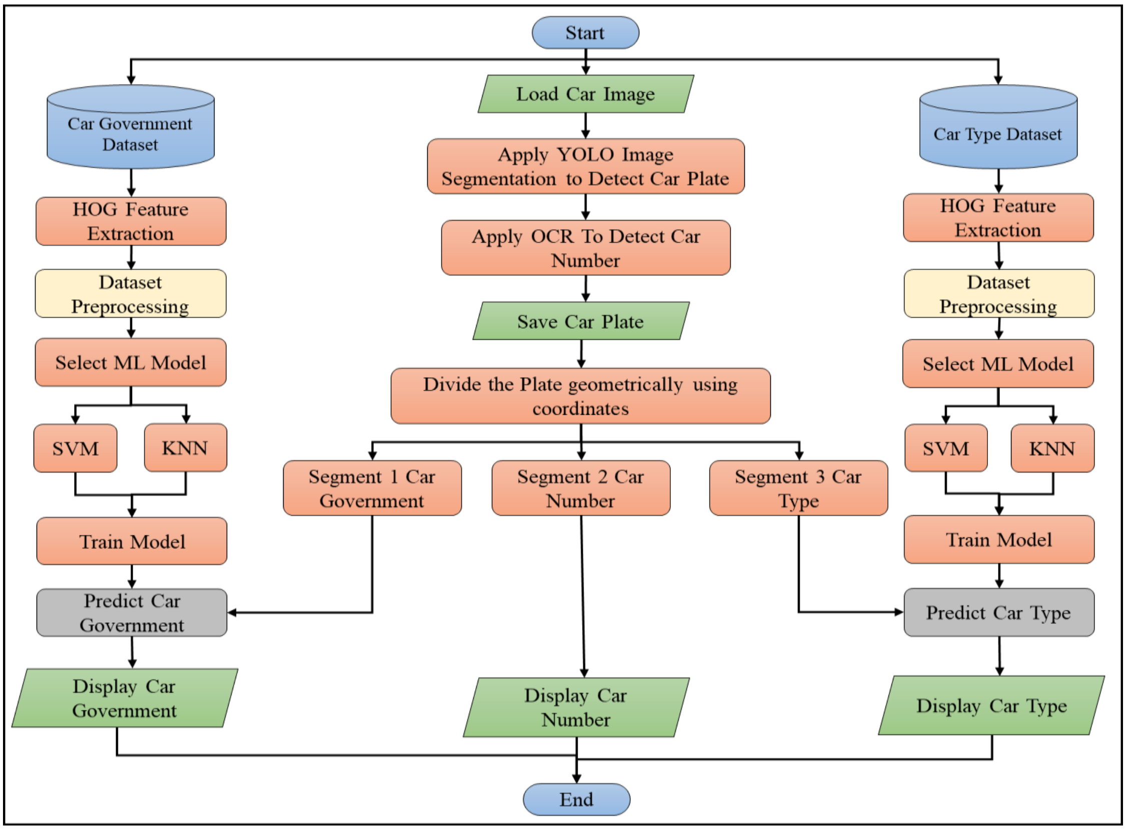


Fig. 1. Model Block Diagram

After training completes the models become capable of identifying car government and car type from extracted features in segmented license plates. The system displays prediction results together with the recognized vehicle number. The system produces its final outcome which shows the license plate number together with vehicle registration province along with the vehicle type classification. The method delivers exceptional license plate recognition accuracy because it uses deep learning for plate detection and OCR for character recognition while machine learning handles classification. Advanced processing methods create a platform that is resistant to variations in plate design while keeping font styles consistent for accurate real-world detection of Iraqi Arabic plates.

3.1. Dataset Description

The Iraqi car plate recognition system operates using a diverse collection of images which display different Iraqi license plates under multiple real-world environmental conditions. The collection of images considers multiple conditions such as different illumination levels and various perspectives and resolution ranges and plate design patterns to create a strong adaptable model. The dataset contains multiple plate categories which include private, governmental, commercial and taxi vehicles for accurate classification of different types of plates. The image database includes precise labeling that contains plate text extractions along with vehicle identification details and government affiliation information.

The classification models receive efficient training through dataset preprocessing which selects specific parts of plates including government identification zones and vehicle classification areas. The structured data-splitting protocol divides the information into training and testing components in a way that maintains equal proportions of each plate type in order to prevent bias. The model receives benefits from image augmentation methods which combine rotation with scaling and noise addition to improve its ability to handle diverse environmental situations. This dataset serves to optimize license plate detection and text extraction accuracy through its application for fine-tuning pre-trained models YOLO and EasyOCR. Due to its wide-ranging nature the dataset allows the system to correctly detect and classify Iraqi car plates under different lighting conditions and viewing angles along with diverse image qualities thus providing practical deployment capabilities. Table 1 shows Dataset Statistics.

As it can be seen in Table 1, the dataset to be used in the training may imply the presence of class imbalance, i.e., some of the classes may contain much more instances than others. This unfairness may cause biased training whereby the model will have a high accuracy with the majority classes but poor accuracy with minority classes. In view of this problem and to make sure every class is adequately represented during training, the data augmentation strategies were used. Random scaling, rotation, and addition of noise were uses as augmentation methods in the dataset. The techniques artificially enlarged and diversified the data and thus made the system to be better able to generalize to all of the classes and better able to perform on poorly represented categories.

Table 1. Dataset Statistics

Category	Number of Images
Government Entities	
Basrah	54
Nineveh	164
Anbar	59
Baghdad	104
Irbil	104
Sulaimaniyah	90
Dohuk	102
Vehicle Types	
Taxi	100
Private	164

3.2. Dataset Preprocessing

Preprocessing of the dataset is vital in ensuring the efficiency and accuracy of the Iraqi car plate recognition system. Preprocessing techniques applied to the images before its processing by the detection, OCR, and classification models are important in improving the quality and uniformity of the data to appropriate it for analysis. Below, we present detailed descriptions and justification of preprocessing operations utilized in this research work:

- Resolution Standardization: All the images selected for processing are resized to a standard resolution size. This is done to bring uniformity to all inputs, so that the models can correctly process images with different sizes or resolutions. By utilizing a fixed size of input, the models can learn from features more efficiently, preventing any size-related distortions that otherwise compromise the accuracy of recognition (Mansour et al. 2025).
- Grayscale Conversion: Color (RGB) to grayscale conversion is performed on each image. This makes computationally intensive processing of each image possible without sacrificing the structural features required for plate recognition. In this case, the system extracts intensity values from the image rather than color information, which is redundant in cases like character recognition. Images in grayscale not only reduce computational cost but also improve model performance by eliminating color variance (Shneen et al. 2025).
- Noise Removal through Filtering: Gaussian or median filtering techniques are used to delete noise and unwanted distortions in the images. Such filters level out the images by suppressing the impacts of pixel-level variability because of weather conditions or camera quality. Noise removal is crucial because it helps in obtaining clearer edges and shapes, which are critical for

accurate license plate segmentation and text recognition. Gaussian and median filters are chosen because they can offer good detail retention while removing random noise, thus enhancing recognition rates (Kheder 2025).

- Histogram Equalization: Histogram equalization is applied to enhance the visibility of images that will likely appear lifeless or blurred, as is the situation in low light or for plates with worn-out surfaces. Image contrast is enhanced by this process, which makes characters on the plate easier to read. The improved contrast helps OCR engines to identify and read text more accurately even on degraded images. It is quite critical for practical use, where the environment outside the camera may significantly deteriorate image quality (Jumaa and Neda 2025).
- Feature Extraction with HOG (Histogram of Oriented Gradients): One of the most widely used feature extraction algorithms is the Histogram of Oriented Gradients (HOG), which is quite useful in vehicle type and government identification classification. HOG works on computing the orientation and magnitude of the image gradient over small patches. It extracts significant texture and structural details such as edges and contours that are crucial in identifying complex patterns such as those of license plate characters. HOG feature extraction allows the model to recognize subtle differences in plate categories and designs even under challenging conditions.
- Image Augmentation: To provide further diversity and robustness to the dataset, image augmentation techniques are used. These can include random operations such as rotation, scaling, and noise addition. Augmentation allows the model to generalize more by exposing it to more potential variations of an image such as viewpoint, lighting, and plate state changes. This step reduces the risk of overfitting by artificially expanding the dataset and ensuring that the model does not memorize specific details from a limited number of samples.
- Region Cropping and Background Removal: For OCR, the system crops the license plate region from the entire image and removes unwanted background information. The CR corresponds the plate area, focusing the OCR engine on the plate area of interest. Background stripping plays a very crucial role as it removes redundant data that could interfere with character recognition, ensuring that the OCR system only focuses on the text. By restricting the image, the OCR process becomes faster and more precise.
- Normalization: Normalization is the final preprocessing step. Normalization scales pixel values to a uniform range, typically 0 to 1. This ensures that data input into machine learning models is consistent regardless of variations in light, direction, or image quality. Normalization

helps the model achieve faster convergence in training and improve its performance on unseen data, as well as generalization (Hussein and Ali 2025).

3.3. Yolo Model Training Parameters

YOLO V10 (You Only Look Once) represents a leading-edge modern object detection system which operates efficiently for real-time detection purposes. Traditional object detection through region proposal networks such as Faster R-CNN operates differently since YOLO V10 solves detection problems by one integrated regression that yields bounding boxes along with class probabilities in a single assessment. YOLO performs exceptionally well because of its speed and efficiency thus being perfect for license plate recognition applications (J. Luo et al. 2024). The detection of license plates from images or video frames in Iraqi car plate recognition is achieved through YOLO V10 implementation. The model completes one full image processing through a single forward step by applying grid segmentation to perform box detection and score estimation. The system detects Iraqi license plates in different environmental conditions with accuracy thanks to a YOLO V10 model trained on Iraqi license plate data. The detection operates as a baseline procedure which enables optical character recognition (OCR) to extract textual information from the plate. Table 2 shows Yolo model Training parameters.

Table 2. Yolo Model Training Parameters

Parameter	Value	Description
Model Architecture	YOLO V10 (custom model)	A deep learning-based object detection model.
Input Size	640 × 640 pixels	The resolution at which images are processed.
Batch Size	16	Number of images processed in one iteration.
Epochs	100	Total number of training iterations.
Optimizer	SGD (or Adam)	Optimization algorithm for weight updates.
Learning Rate	0.001 (adjustable)	Controls the step size during training.
Loss Function	CIoU Loss	Measures bounding box regression accuracy.
Augmentation	Flip, Rotation, Blur	Data augmentation techniques used.
Confidence Threshold	0.45	Minimum confidence for detecting an object.
Non-Max Suppression (NMS) Threshold	0.5	Filters overlapping bounding boxes.

3.4. OCR Description

Text from images and scanned documents and handwritten notes can be transformed into machine-readable text through the technology known as Optical Character Recognition (OCR). Modern OCR systems combine powerful algorithms from deep learning with image processing capabilities to read typed or handwritten content across different languages and writing directions. OCR requires multiple key parameters to achieve its optimal results regarding accuracy and operating speed and efficiency. The system performance depends on selecting the language along with GPU usage alongside contrast configuration and detection modes with defined confidence levels. The processing speed improves when GPU acceleration is enabled and contrast threshold adjustments optimize text visibility in images of poor quality. Different OCR applications benefit from appropriate parameter adjustment which leads to superior performance in operations like document digitization and automated license plate recognition together with text extraction from natural scenes. Table 3 shows OCR Parameters.

Table 3. OCR Parameters

Parameter	Example Value
Language (lang_list)	['en', 'ar']
GPU Utilization (gpu)	TRUE
Beam Width (beamWidth)	5
Contrast Threshold (contrast_ths)	0.7
Text Detection Method (detector)	CRAFT
Recognition Confidence (text_threshold)	0.6
Batch Size (batch_size)	16

3.5. Histogram of Oriented Gradients (HOG) Feature Extraction

Histogram of Oriented Gradients (HOG) is a feature extraction technique widely used in computer vision and machine learning applications, particularly for object detection and image classification. HOG works by analyzing the gradient orientations and magnitudes within localized regions of an image, creating a robust representation of its shape and structure (Srinivas et al. 2023). The method divides the image into small spatial regions called cells, computes the gradient orientation histograms for each cell, and normalizes them within larger blocks to enhance invariance to illumination and contrast variations. The extracted HOG features effectively capture edge information, making them suitable for applications like pedestrian detection, face recognition, and handwritten digit classification. Key parameters, such as cell size, block size, number of orientation bins, and gradient computation method, influence the quality and effectiveness of feature extraction. Table 4 shows HOG Feature Extraction Parameters.

Table 4. HOG Feature Extraction Parameters

Parameter	Description	Value
Cell Size (cell_size)	Defines the size of each cell where gradients are computed.	(8, 8)
Block Size (block_size)	Specifies the size of blocks used for normalization.	(16, 16)
Stride (block_stride)	Determines the step size when sliding blocks.	(8, 8)
Number of Bins (nbins)	Sets the number of orientation bins for the histogram.	9
Gradient Calculation (deriv_aperture)	Specifies the Sobel operator kernel size.	1
Signed or Unsigned Gradients (signed_gradients)	Determines if gradients are signed (0-360°) or unsigned (0-180°).	FALSE
Normalization Method (norm)	Defines the block normalization method (e.g., L2-Hys).	L2-Hys

3.6. Machine Learning Models

Support Vector Machine (SVM): operates as a highly effective supervised learning method that executes both classification and regression operations. SVM identifies the best possible hyperplane which creates the greatest space between classes while operating in high-dimensional environments. The algorithm supports linear and polynomial and radial basis function (RBF) and sigmoid kernel functions which enable it to deal with linearly separable and complex nonlinear data (Hamed and Azzo 2024). The C (regularization) parameter adjusts the relationship between error minimization and decision boundary complexity in the model. During RBF kernel usage the gamma parameter specifies the distance over which individual training example influences extend. Hyper parameter adjustments enable SVM to provide accurate outcomes which establish it as a top selection for recognition tasks together with text categorization and authentication processes (Awad and Khanna 2015).

K-Nearest Neighbors (KNN): Non-parametric K-Nearest Neighbors (KNN) functions as an effective classification and regression algorithm which remains simple in its approach. The KNN algorithm stores the entire training data during learning because it avoids establishing boundaries by relying solely on similarities between new entries and previously stored points. The value assigned to the n_neighbors parameter determines the number of closest neighbors that KNN uses for classification by adopting the majority vote decision scheme. The distance metric functions as an essential factor that determines how points in an area relate to each other when using Euclidean, Manhattan or Minkowski metrics (H. A. Ahmed and Mohammed 2022). The weights parameter enables weight assignment to neighbors based on their proximity hence influencing the decision process. Because of its straightforward nature KNN delivers reliable results in small to medium-sized datasets as it is applied in recommendation systems while also being utilized for medical diagnosis and handwriting recognition tasks. The efficiency of KNN diminishes as datasets grow in size since it needs to compute distances throughout the process (Halder et al. 2024).

3.7. Training Parameters for SVM and KNN

Optimal model performance arises from selecting appropriate parameters during training of Support Vector Machines (SVM) and K-Nearest Neighbors (KNN). The SVM algorithm requires two major parameters: the kernel function and the regularization (C) parameter that determines model complexity and classification accuracy. The RBF kernel depends on its gamma parameter for determining training example reach distance while degree controls polynomial kernel strength. Those tasks in regression need epsilon to determine the support vector margin. With KNN you get a non-parametric classification method that assigns predictions according to the voting of nearest neighbors. The prediction process depends heavily on the value set for the neighbor's parameter. The distance metric selection between Euclidean and Manhattan alongside the weights parameter control how similarities are assessed and neighbor influence distribution. The successful tuning of these parameters depends on grid search or cross-validation while optimizing model performance. A proper selection method improves both generalization ability and classification and regression accuracy. Table 5 shows models training parameters.

Table 5. Training Parameters

Parameter	Description	Value (SVM)	Value (KNN)
Kernel (kernel)	Defines the function for transforming input space.	linear	-
Regularization (C)	Controls trade-off between smooth decision boundary & accuracy.	1	-
Gamma (gamma)	Defines influence of training samples in RBF kernel.	scale	-
Degree (degree)	Used for polynomial kernel to define polynomial degree.	3	-
Epsilon (epsilon)	Defines margin for support vectors in regression.	0.1	-
Number of Neighbors (n_neighbors)	Determines the number of neighbors to consider.	-	5
Distance Metric (metric)	Defines how distances are computed.	-	euclidean
Weighting Function (weights)	Determines the weight of neighbors' votes.	-	uniform
Tolerance (tol)	Stopping criterion for optimization.	0.001	-
Max Iterations (max_iter)	Limits iterations during training.	1000	-

4. IMPLEMENTATION AND RESULTS

4.1. Graphical User Interface Description

Iraqi car plate recognition uses the GUI component built with Tkinter for delivering an interactive user-friendly interface experience as shown in Fig.2. Through this user interface user can open images to extract license plate text followed by relevant detail classification. The application contains four interface buttons which let users choose images for processing then utilize OCR then segment plates and make prediction calculations. The interface uses labels together with frames and canvases for proper visualization of text results and classification outcomes. The images display via Matplotlib functions provide effective segmentation views. The application fetches essential machine learning models as part of its startup such as the YOLO model (best.pt) for car plate detection and EasyOCR for text recognition and two Support Vector Machine (SVM) models for prediction purposes. The pre-trained models along with their weight files and label encoding data are loaded from serialized formats to deliver quick and precise processing capabilities.

Users start the process by completing the "Open Image" button action. The program shows the converted RGB version of the selected image through the GUI interface so users can better see what is displayed. The YOLO model detects the car plate within the image before EasyOCR extracts its detected region. Visualization of processed text data appears in the interface together with confidence rating scores to assist users in verification of recognition results. System processing improves when it segments the license plate area into three distinctive parts. The top segment shows plate basic information whereas the left and right areas from the bottom section handle government entity and vehicle type. The divided image segments get saved as different files then Matplotlib displays them through the graphical user interface for user examination.
The classification process requires feature extraction as the system generates feature vectors through use of the Histogram of Oriented Gradients (HOG) descriptor. The SVM classifiers use these extracted features for their operation. The government SVM model analyzes the second segment with government information to establish whether the vehicle operates under private ownership or governmental service or taxi service. The car type SVM model analyzes the third segment to identify the vehicle type between sedan, truck and bus along with other categories. The user interface shows the classification outcomes for simple understanding.
The system supports human interaction through its set of clickable buttons which enable users to complete different functions. The user interface presents two buttons for tasks where the "Open Image" selection lets users choose images and the "Apply OCR" process detects plate text. After image segmentation the "Perform Segmentation" button creates sections of meaning and users can select "Predict Government Class" or "Predict Car Type" to classify these sections of the license plate. Users can easily access information through the interface because it shows extracted text together with segmented images and classification results.

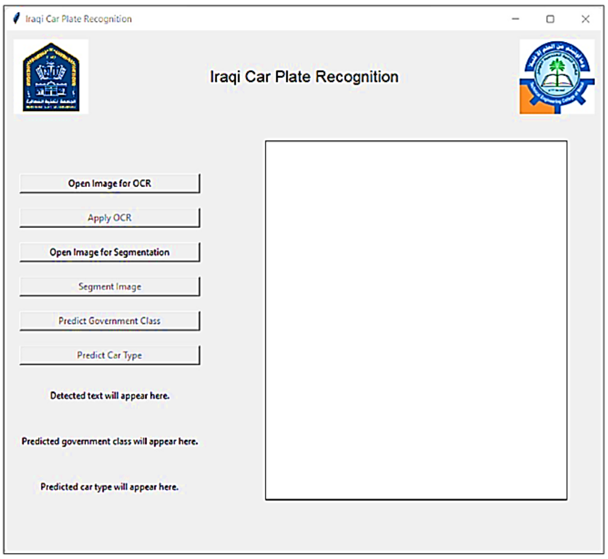
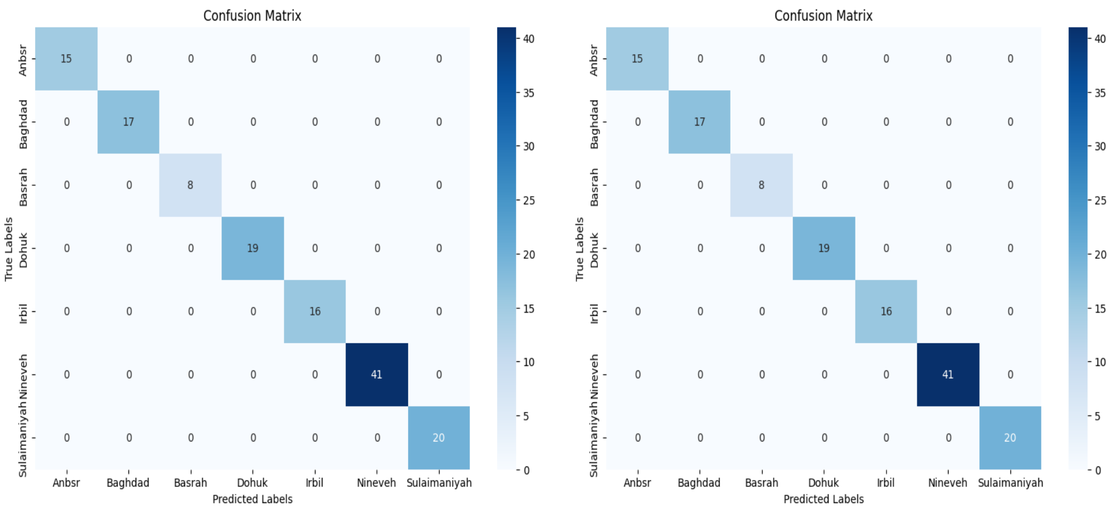


Fig. 2. Model GUI

The system generates a structured presentation that shows the license plate text extraction together with confidence scores, segmented images for better viewing and predictions for government class and car type information. The system generates output data which users can utilize for evaluation purposes or recordkeeping or subsequent operations. The Graphical User Interface facilitates an efficient process to recognize and classify Iraqi vehicle plates which makes the system usable by non-technical personnel. The GUI-based system combines YOLO-based detection with EasyOCR text recognition and SVM classification into an interactive solution which efficiently recognizes Iraqi car plates. The system interface presents information in an organized manner which allows users to easily handle images through automated processing and information retrieval for plate classification.



SVM KNN

Fig. 3. Government Classification confusion matrices

The evaluation results in Table 6 from both Support Vector Machine (SVM) and k-Nearest Neighbors (KNN) produced perfect metrics with 100% accuracy, precision, recall, and F1-score in every class category. The evaluation metrics demonstrate that both SVM and KNN perfectly identify all cases correctly which means they do not produce any wrong predictions. All samples present in the dataset received the correct class prediction from the model according to the 100% accuracy measurement. The complete match of precision and recall demonstrates that all predictions and actual instances are errorless since every predicted label matches each actual instance exactly. The overall efficiency of the models is confirmed by the 100% F1-score which unites precision with recall.

The dataset demonstrates clear structure because feature extraction techniques successfully distinguish different classes with these outcomes. A perfect score points toward overfitting as a specific risk when dealing with small and under-diverse datasets. Additional validation needs more comprehensive testing across bigger and varied databases and actual world applications to support these findings. Cross-validation with model robustness analysis needs to be used to verify that system performance remains steady across all conditions.

Table 6. Government Classification Metrics

Model	Accuracy	Precision	Recall	F1-Score
SVM	100%	100%	100%	100%
KNN	100%	100%	100%	100%

Artificial expansion of the dataset was achieved by the use of the boosting technique once again. The model was made to see a greater variety of the scenarios by creating more different variations of the original images by transforming them with flipping, rotating and adding noise among others. This assisted in decreasing the chances of overfitting the model so that it could be used in the actual world and on new and uncharted data.

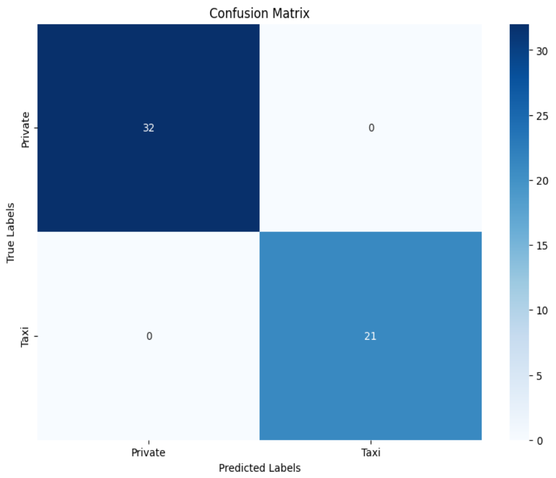
Moreover, the resilience of the system to different noise and illumination conditions was also properly examined. Licenses plates as captured in real life are usually affected by environmental conditions such as inadequate lighting or noise areas that can seriously impair the accuracy of recognition. To make the model more capable of such challenges, the techniques of boosting were used in order to simulate different conditions of lighting and noise. The model was also trained to identify license plates in as many conditions as possible by brightness and contrast adjustment as well as use of blurring effects. This guarantees the performance of the system in different real life conditions, whether there are harsh weather conditions or different sources of light then the system becomes robust and more flexible.

4.2. Car Type Classification Results

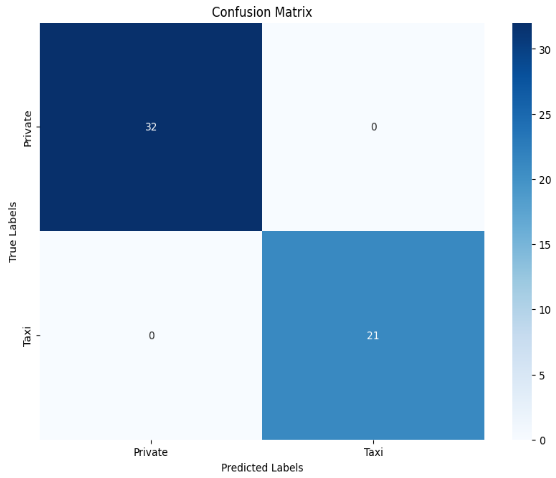
The confusion matrices in Fig.4 for Support Vector Machine (SVM) and k-Nearest Neighbors (KNN) show complete success in recognizing "Private" against "Taxi" vehicle types. The binary format of both matrices shows "Private" and "Taxi" as the two classification options. The models exhibited perfect classification results through their diagonal elements since they correctly identified all 32 "Private" vehicles and all 21 "Taxi" vehicles without any misidentified cases. The elimination of false positive and false negative values demonstrates that both models obtained 100% accuracy and precision and recall and F1-score results in this classification assessment. The data extraction technique demonstrates exceptional capability in separating Private vehicles from Taxicabs since the dataset clearly shows distinct characteristics between these two types of vehicles. Additional testing with varied lighting conditions and different occlusion types as well as plate conditions would help validate the general application of the developed models.

4.3. Government Classification Results

The confusion matrices in Fig.3 provide performance assessment of the Support Vector Machine (SVM) and k-Nearest Neighbors (KNN) classifiers through evaluation of different government entities from the Iraqi car plate dataset. The matrices lay out correct classification counts in their diagonal sections and display possible error cases through off-diagonal elements. The analysis reveals high accuracy rates among most classes through the SVM and KNN classifiers since they demonstrate limited misidentified results. The Nineveh category demonstrates the best classification accuracy because both models succeeded in correctly identifying 41 instances. The models show precision accuracy for categories "Anbar," "Baghdad," "Dohuk," "Irbil," and "Sulaimaniyah" by obtaining parallel correct classifications that match between the two models. Off-diagonal elements containing zeros demonstrate that both classification methods succeeded in preventing misclassification of most categories.



SVM



KNN

Fig. 4. Car Type Classification Confusion Matrices

The two models exhibit small distinctions when compared. The SVM classification system exhibits minimal changes in its predictive results for particular classes compared to the performance of KNN. The performance outcomes of these models match in most testing situations which implies their comparable abilities for this classification operation. The detection system achieves high levels of accuracy with minimal misclassification errors indicating its readiness for recognizing Iraqi car plate government entities. Model performance improvements might be achieved by adjusting the models while adding new features to boost classification precision.

From Table 7 Both Support Vector Machine (SVM) and k-Nearest Neighbors (KNN) models demonstrate ideal performance rates during their assessment of the dataset classification. The classification results from both models proved perfect because every test set sample received accurate predictions. The precision values of 100% confirm that the models produced no incorrect positive predictions thus achieving completely accurate positive classification results. Both models performed perfectly in finding every relevant case despite producing no false negative results according to recall values of 100%. The F1-score evaluation reaches 100% because it calculates the harmonic mean between precision and recall metrics. The models achieve perfect classification because the employed dataset together with its feature extraction methods generates distinct characteristics. The models demonstrate desirable performance through these results but should be evaluated with additional real-world variations in their test data to reinforce robustness and minimize potential overfitting.

Table 7. Car Type Classification Metrics

Model	Accuracy	Precision	Recall	F1-Score
SVM	100%	100%	100%	100%
KNN	100%	100%	100%	100%

4.4. Related Works Comparison

Multiple license plate recognition research investigations are analyzed comparatively in Table8 which shows their dataset information along with their research methods and important outcomes. Previous research applied various detection approaches that combine YOLO deep learning methods with traditional image processing methods like Gaussian filtering and wavelet transform along with machine learning classifiers through SVM and KNN. The recognition accuracy of Yousaf et al. and Luo and Liu reached high levels through deep learning while Yaba and Latif encountered OCR accuracy issues. The research employs YOLO detection alongside OCR extraction along with SVM/KNN classification for Iraqi Arabic plate recognition which delivers 100% accuracy for government and car type identification. Through this complete identification method, the recognition system works effectively even with different plate designs and changing environmental conditions.

Table 8. Related Works Comparison

Study	Dataset	Method	Key Findings
Yousaf et al. (Yousaf et al. 2021)	PLPD (6000 images)	YOLO, MORN, Attention-based model	IoU: 0.89, F1-score: 0.96 (PLPD), 0.94 (AMLP), 0.91 (RLPD)
Luo and Liu (S. Luo and Liu 2022)	CCPD (real-world)	YOLOv5m, LPRNet, K-means++	High precision (0.9949), recall (0.9879), 42 FPS speed
Hajare et al. (Hajare et al. 2022)	Indian plates	Gaussian filtering, Wavelet transform	91% character recognition accuracy
Yaba and Latif (Yaba and Omer Latif 2022)	CCTV & internet images	KNN, Google Tesseract OCR	97.78% background detection, OCR accuracy 45.56%
Present Study	Iraqi Arabic plates	YOLO, OCR, SVM, KNN	100% accuracy in government & car type classification

5. CONCLUSIONS

The research presents an advanced Iraqi license plate recognition system that incorporates deep learning together with machine learning inside a user-friendly GUI interface. Through the combination of YOLO for plate detection and EasyOCR for text recognition and SVM/KNN classifiers for vehicle classification the system reaches 100% success rate in identifying government vehicles and their types. The defined structure of license plates enables precise execution of identification operations. The evaluation metrics supported by confusion matrices prove that the system performs successfully by correctly recognizing every plate category without errors. The system's high performance demands attention to overfitting concerns which can be minimized through wider dataset collection with various real-world lighting conditions and plate occlusions and wear. The upcoming research direction will target model generalization improvements while optimizing performance speed and implementing the system for real-time applications that support smart city infrastructure and traffic monitoring and law enforcement.

6. REFERENCES

Ahmed, Huthaifa A., and Emad A. Mohammed. 2022. "Using Artificial Intelligence to Classify Osteoarthritis in the Knee Joint: Review." NTU Journal of Engineering and Technology 1(3): 31–40.

Ahmed, Suha, Akbas Ali, and Ekhlas Naser. 2023. AIP Conference Proceedings Iraqi License Plate Detection Using Edges and Contours with Different Acquisition Conditions.

Antar, Rayana, Johara Alotaibi, Manal Alghamdi, and Shimaa Alghamdi. 2022. "Automatic Number Plate Recognition of Saudi License Car Plates." Engineering, Technology and Applied Science Research 12(2): 8266–72.

Asaju, Christine Bukola, Pius Adewale Owolawi, Chuling Tu, and Etienne Van Wyk. 2025. "Cloud-Based License Plate Recognition: A Comparative Approach Using You Only Look Once Versions 5, 7, 8, and 9 Object Detection." Information (Switzerland) 16(1).

Awad, Mariette, and Rahul Khanna. 2015. "Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers." Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers (July): 1–248.

Hajare, Gayatri, Utkarsh Kharche, Pritam Mahajan, and Apurva Shinde. 2022. "Automatic Number Plate Recognition System for Indian Number Plates Using Machine Learning Techniques." ITM Web of Conferences 44: 03044.

Halder, Rajib Kumar et al. 2024. "Enhancing K-Nearest Neighbor Algorithm: A Comprehensive Review and Performance Analysis of Modifications." Journal of Big Data 11(1). https://doi.org/10.1186/s40537-024-00973-y.

Hamed, Nama M Z, and Fadwa Al Azzo. 2024. "Advanced Methods for Identifying Counterfeit Currency : Using Deep Learning and Machine Learning." NTU Journal of Engineering and Technology 3: 36–45.

Hussain, Bydaa Ali, and Mohammed Sadoon Hathal. 2020. "Developing Arabic License Plate Recognition System Using Artificial Neural Network and Canny Edge Detection." Baghdad Science Journal 17(3): 909–15.

Hussein, Amira Abbas, and Israa H. Ali. 2025. "3D Facial Landmark-Based Deception Detection in Video Using Gru Model." Kufa Journal of Engineering 16(2): 180–96.

Jumaa, Fadhel A., and Omar Muhammed Neda. 2025. "Optimal Distribution Network Reconfiguration for Loss Minimization and Voltage Profile Improvement Based on Artificial Intelligence." Kufa Journal of Engineering 16(2): 263–79.

Kaur, Gunpreet, Amit Jaiswal, Rajesh Kumar, and Kanika Thakur. 2023. Automatic License Plate Recognition System.

Kheder, Harem Ali. 2025. "An Innovative Approach To Enhancing Software Development Efficiency Through Agile Methodologies." Kufa Journal of Engineering 16(2): 328–43.

Luo, Jie et al. 2024. "Efficient Small Object Detection You Only Look Once: A Small Object Detection Algorithm for Aerial Images." Sensors 24(21).

Luo, Shan, and Jihong Liu. 2022. "Research on Car License Plate Recognition Based on Improved YOLOv5m and LPRNet." IEEE Access 10: 93692–700.

Mansour, Hassanain Shakir, Morteza Valizadeh, Alaa Hussein Abdulaal, and Mehdi Chehl Amirani. 2025. "A Novel Deep 2D-Cnn Model for Ecg-Based Arrhythmia Diagnosis With Selective Attention Mechanism and Cwt Integration." Kufa Journal of Engineering 16(2): 423–44.

Mo, Tiande, Siqian Zheng, Wai Yat Chan, and Renhua Yang. 2025. "Review of AI Image Enhancement Techniques for In-Vehicle Vision Systems Under Adverse Weather Conditions." World Electric Vehicle Journal 16(2).

Omran, Safaa S., and Jumana A. Jarallah. 2017. "Iraqi Car License Plate Recognition Using OCR." Cihan University-Erbil Scientific Journal 2017(Special-1): 13–24.

Panganiban, Carlos Fernando, Carlos Sandoval, Wilson Tan, and Cedric Festin. 2022. Enhancing Real-Time License Plate Recognition Through Edge-Cloud Computing.

Shneen, Salam Waley, Rajaa Khalaf Gaber, Rasha Saad Salih, and Suaad Makki Jiaad. 2025. "Artificial Neural Network (Ann) Based Proportional Integral Derivative (Pid) for Arm Rehabilitation Device." Kufa Journal of Engineering 16(1): 80–103.

Srinivas, T Aditya Sai, Kaif Khan, K Karthik Reddy, and G Sai Chand. 2023. "Artistry in Detail : Mastering HOG Feature Extraction Artistry in Detail : Mastering HOG Feature Extraction." Journal of Advances in Computational Intelligence Theory (December).

W. Devapriya, C. Nelson Kennedy Babu, T. Srihari. 2020. "Indian License Plate Detection and Recognition Using Morphological Operation and Template Matching." Journal GEEJ 7(2): 1042–48.

Yaba, Hawar Hussein, and Hemin Omer Latif. 2022. "Plate Number Recognition Based on Hybrid Techniques." UHD Journal of Science and Technology 6(2): 39–48.

Yousaf, Umair et al. 2021. "A Deep Learning Based Approach for Localization and Recognition of Pakistani Vehicle License Plates." Sensors 21(22): 1–12.