Received: 20 January 2024 | Revised: 16 July 2024 | Accepted: 11 August 2024
DOI: 10.37917/ijeee.21.1.21 Early View | June 2025

Open Access

Iraqi Journal for Electrical and Electronic Engineering
Original Article

# License Plate Detection and Recognition in Unconstrained Environment Using Deep Learning

**Heba Hakim*[1], Zaineb Alhakeem[2], Hanadi Al-Musawi[1], Mohammed A. Al-Ibadi[1], Alaa Al-Ibadi[1]**
[1]Computers Engineering Department, University of Basrah, Basrah, Iraq
[2]Chemical Engineering and Oil Refining Department, Basrah University for Oil and Gas, Basrah, Iraq

Correspondance
*Heba Hakim
Computers Engineering Department, University of Basrah,
Basrah, Iraq
Email: hiba.abdulzahrah@uobasrah.edu.iq

**Abstract**
***Real-time detection and recognition systems for vehicle license plates present a significant design and implementation challenge, arising from factors such as low image resolution, data noise, and various weather and lighting conditions.This study presents an efficient automated system for the identification and classification of vehicle license plates, utilizing deep learning techniques. The system is specifically designed for Iraqi vehicle license plates, adapting to various backgrounds, different font sizes, and non-standard formats. The proposed system has been designed to be integrated into an automated entrance gate security system. The system's framework encompasses two primary phases: license plate detection (LPD) and character recognition (CR). The utilization of the advanced deep learning technique YOLOv4 has been implemented for both phases owing to its adeptness in real-time data processing and its remarkable precision in identifying diminutive entities like characters on license plates. In the LPD phase, the focal point is on the identification and isolation of license plates from images, whereas the CR phase is dedicated to the identification and extraction of characters from the identified license plates. A substantial dataset comprising Iraqi vehicle images captured under various lighting and weather circumstances has been amassed for the intention of both training and testing. The system attained a noteworthy accuracy level of 95.07%, coupled with an average processing time of 118.63 milliseconds for complete end-to-end operations on a specified dataset, thus highlighting its suitability for real-time applications. The results suggest that the proposed system has the capability to significantly enhance the efficiency and reliability of vehicle license plate recognition in various environmental conditions, thus making it suitable for implementation in security and traffic management contexts.***

**Keywords**
**Big Data, Deep Learning, Intelligent Systems, License Plate Detection, License Plate Recognition.**

## I. Introduction

With the proliferation of monitoring cameras integrated into Intelligent Transportation Systems (ITS) across various roadway locations and the growing number of vehicles, license plate identification has emerged as a key area of interest. Since these cameras output images or video frames, Digital Signal Processing and machine vision techniques are employed to extract valuable insights for developing various ITS applications. License plate detection systems play a crucial role in maintaining law and order on roads, enhancing traffic safety, and efficiently managing parking resources. These systems rely on complex computer vision methods for the detection and identification of license plates.

However, the design of license plate detection and recognition system is a challenge problem. Despite the advancements in technology that improve its efficiency and feasibility, there are still many issues that need to be addressed carefully. The first issue is that the images utilized to recognize the license plate number are obtained from unconstrained environments. The license plate detection and recognition system should accurately identify the characters on the license plate regardless of the image's resolution, quality, illumination conditions and camera's position relative to the plate's plane. Nevertheless, as these constraints are increased, the ability of the system to handle variant cases and scenarios is reduced. The second issue encountered by the license plate detection and recognition system is the variability in the design of license plate. The designs vary from one country to another. Different designs may be observed within the same country following many factors such as the creation date of the license plate, the issuance location within the country or the vehicle type. It is recommended that the design of license plate detection and recognition system used in a specific country and resolve all associated technical challenges. The third issue is that the license plate and its characters should be accurately recognized in real time. These are the main issues that each license plate detection and recognition system must address to be used in real-world scenarios.

Consequently, there is still potential for improving their performance and ability to interact with various types of license plates. The first systems used analog cameras which are not very appropriate with such systems, while the video camera shots and the digital cameras' images are the best choice that gives a good performance of recognition as Sarfraz et al. show in their works [1, 2].

Many Automated License plate detection and recognition systems have used convolutional neural networks (CNN) due to their outstanding performance. In [3, 4] end to end License Plate Recognition (LPR) was introduced based on the You Look Once Technique (YOLO) algorithm for the detection of the vehicle and its license plate. Even with its outstanding results in constrained scenarios, the presented system is inadequate for License Plate Recognition in the real world.

Yonetsu et al. [5] utilized a two-stage YOLOv2 algorithm to achieve accurate detection for License plates (LP) in night scenarios to overcome the limitation of YOLOv2 in the detection of a small object such as a license plate. Even though it achieved a good performance at night scenes, it had a problem in detection a distant LP and also some signboards were wrongly detected as LP.

A YOLOv3 algorithm was used in [6, 7] to divide an image into rectangular areas to calculate the value of confidence by predicting bounding boxes for the objects and the probability of class for each area. While the system is accurate and fast, it depends on the object size within an image. That means the objects trained on large sizes may not be accurately detected if they appear smaller during the testing process.

In [8, 9] edge information by using the density of vertical edge was used to extract the LP in a vehicle's image. To ensure a high recall, the dilation morphological process was carried out on binarized images to filter out the false positives. The algorithm's limitations include its lack of robustness for scenes in real world and its dependence on the manual setting of the horizontal and vertical spacing between characters in LP.

LP detection in [10] utilized a combination of non-linear filters, transformations of tophat morphological and the Sobel filter. However, a limitation of this system involves the use of the Sobel filter in vertical edge detection which can leads to the generation of false gradients due to the presence of noise. In [11],a single-shot multi-box method was used for LP detection, while the vertical projection technique was applied for character segmentation. Since the system has a low frame rate (fps), it is unsuitable for real-world scenes.

Silva et al. [12] and Abbas et al. [13] demonstrated the use of CNN in detection and rectification of distorted license plates before applying them to the recognition system. It produced good performance but the test dataset didn't have many challenging scenes.

Elhadi et al. [14] and Siddiq et al. [15] used a model of a CNN that is configured to have a faster response in the detection and recognition of Sudanian license plates. The model in [14] detected only the numbers without the recognition of the state of registration or other features. While [15] achieved 92.8% accuracy on a dataset that didn't include very challenging scenarios.

Rashid et al. used You Look Once Technique (YOLO) to detect and recognize one type of Iraqi license plates through an Android Application with accuracy of 90.9% [16] .

Shehata et al. proposed two different systems for LP detection and two systems for character recognition, based on Faster R-CNN deep learning technique and Connected Components Label (CCL) statistical method. Each system has its own accuracy for LP recognition, the best one has 95% of Egyptian plates [17].

Habeeb et al. used YOLOv2 to detect and recognize the Iraqi license plates in different types. They conducted a comparative analysis with Support Vector Machine (SVM), Neural Network (NN). While the SVM and NN has faster recognition time, YOLOv2 achieved best recognition accuracy about 86% [18].

Antar et al. used Candy edge detection algorithm with horizontal projection to segment the license plate, followed by Optical Character Recognition (OCR) to recognize characters/numbers in Saudi LP. The Saudi plates have both Arabic and English Letters and Characters [19].

Youssef et al. used YOLO v3 to recognize and detect the Egyptian LP. This model can detect and recognize the Egyptian LP which has only Arabic letters and characters with a good performance of overall recognition [20]. Latif et al. used K-Nearest Neighbor (KNN) with OCR to detect and recognize the Iraqi LP with accuracy about 92%. This system has a very good results with one type of Iraqi LP not all the types (the old and new ones) [21].

With the help of a CNN model and spatial pyramid pooling, Henry et al. [22] were able to eliminate the need of processing the fixed-size images in a recognition. In spite of the good results, it still takes more time to collect images in the same size for the batch processing. Additionally, it is limited to work in a constrained environment.

It is important to note that while certain license plate detection and recognition approaches may have demonstrated impressive performance, they have not been tested under real-world conditions, where complexity is an unavoidable factor. The majority of training and testing datasets consist of tens of thousands of vehicle images acquired under controlled and standard conditions, characterized by low acquisition noises and suitable brightness and contrast. It is important to highlight that, as far as our investigation goes, there is currently no deep learning approach that offers exceptional performance and accuracy in the context of Iraqi License plate recognition. The variation of Iraqi license plates and the absence of training data with annotated and determined classes are the primary causes for this claim. All of these reasons inspired us to propose an approach for Iraqi vehicle license plate detection and recognition, offering significant accuracy and real-time performance. As opposed to the most of cases in which some interdependencies exist among their stages, the proposed system provides an end-to-end robust method for License Plate Detection and Recognition (LPDR) in non-standard environments. Utilizing the YOLOv4 algorithm [23] it detects the license plate position and recognizes the characters/digits on the license plate. The system demonstrates sufficient efficiency to handle challenging scenarios and effectively manages the trade-off between high accuracy and real-time performance when compared to previously published state-of-the-art methods.

The main hampering challenge in the success of these systems is the use of several types of license plates. As shown in Fig. 1 Iraqi car plates have different fonts, colors, and sizes. This paper aims to employ an efficient method for automatic License Plate Detection. The multi-stage processes have been adopted for real-time performance because they offer moderate accuracy while also having less computational complexity. Both License Plate Detection (LPD) and Character Recognition (CR) stages are considered classification processes. For each of these stages, deep CNN methods have been utilized based on Yolov4 object detection architecture. YOLOv4 has been used to detect the vehicle's license plate with recognizing its characters, since it provides high accuracy in real time. The main contributions of this proposed work are:

Fig. 1. Different types of Iraqi car license.

- A real time multi-stage deep learning based system has been implemented for vehicle license plate detection and recognition with higher accuracy in real world scenarios than the works compared therein.
- Two separate YOLOv4 techniques have been trained and fine-tuned for LPD and CR phases on a large amount of realistic data. Moreover, our method is significantly more robust than methods previously published, as it provides promising results in challenging circumstances.
- The used data is collected under different weather, illumination and noise conditions. Illumination rectification has not been employed in recently published systems, as far as we are aware.
- The various techniques of augmentation on scarce data are used to expand the dataset.

The rest of the paper has organized as follow: Section 2 describes the proposed system with its mechanism for work. The discussion and results are provided in third section. Finally, the conclusion of this paper is given in section 4.

## II. The Proposed System

The aim of the proposed system is to utilize an automated application for entrance gate security system; consequently, this work has focused on training the model under such circumstances.

When the vehicle reaches the gate security system, a distance sensor detects the vehicle presence and its distance from a gate. At this time, a camera starts to capture image. The captured image is processed to extract a vehicle number plate. The region of vehicle number plate is used as input to the character segmentation. After that, the character recognition process is implemented to predict a vehicle number. The recognized plate number will be matched with the recorded numbers in a database. If it matches, then a gate will be opened, otherwise, it indicates that vehicle has no authentication; resulting in the gate remaining closed and the system denies vehicle entry. A warning email is also sent to the authority of the gate management. This data is stored in the database along with specific information, such as the vehicle's entry and exit time.

The block diagram in Fig. 2 shows the proposed License Plate Recognition (LPR) system in two main stages: (1) license plate detection (LPD), (2) Character recognition (CR). In LPD phase, if one or more Iraqi car license plates are detected at different distances in a captured image, the system will crop and separate them from a given image. As well as the characters and digits that recognized inside the area of cropped car license plate can be done through the classification process. Since a real-time and high accuracy are needed in LPD and CR, the YOLOv4 was chosen for these two sages. YOLOv4, as described in the previous section, is a better selection to the proposed method, since the process of license plate detection should be in real-time and most license plate characters are in small size.

According to Fig. 2, an image is the input of the system while the system's output is a series of character/digits representing the context contained within the license plate of vehicles. The proposed system starts by capturing image and followed by applying enhancement methods to improve the quality of the captured image. After that, a deep learning algorithm is used to extract the license plate from the improved image and recognize the numbers and characters that are found inside the extracted license plates. Each number or character is treated as an object in a classification process. The processes of license plate extraction and recognition are ignored when no license plate is detected from the first stage. The architecture of the mentioned algorithm YOLOv4 is depicted in Fig. 3, which contains CSPDarknet53 (its backbone) [24] , SPP [25] with multiple kernal sizes(1, 5, 9, 13), PANet [26] (its neck), and YOLOv3 [27] as its head. The cross-stage-partial (CSP) is integrated into every partial dense block to improve the learning capability of the CNN. Feature maps serve as the feature extractor output, which are connected to the module of feature enhancement. To improve feature fusion and receptive field, the SPP and PANet are used for feature enhancement. An input image size of the proposed method LPD is 416 x 416, where as a size of 224 x 224 is used for CR system. These resolution values were determined empirically to achieve the best system performance. In Yolo deep architecture [28], a grid segmentation has been performed on the image with grid sizes of M x M. Two parameters are needed in the predict method: bounding boxes (N) and score of confidence (S). To find the parameter (S), the evaluation metric Intersection over Union (IoU) should be calculated to compare the dimension and location of predicted bounding box and ground truth as shown in (1)

$$IoU = (Overlap\,Area\,(X,Y))/(Union\,Area\,(X,Y)) \quad (1)$$

where X and Y represent two bounding boxes. When two boxes are alike in location and dimension that means IoU has higher values. The following subsections describe each stage in details.

### *A. License Plate Detection System*

The input of the first stage (LPD) is a color/gray-scale set of the vehicle images. YOLOv4 was used with changing the last convolution layer to the one class that represents Iraqi vehicle license plate. Other parameters of this technique such as anchors and confidence threshold remained the same as the default setting to get better license plate detection and a better IoU. The quality of images is improved by using Gaussian optimization to reduce image noise/blur, while histogram equalization is used to obtain better quality image. Iraqi vehicle license plate is only one class that will be detected by the classification method. In our system, transfer learning is utilized to apply pre-trained model of YOLOv4 on our custom dataset. The license plates are cropped from the input image after detection, and they are saved as a separate object to be processed in the next stages.

### *B. Character Recognition System*

A set of detected license plate images is the input to the character recognition system (CR). The license plates are cropped from the input image after detection by taking the coordinates of bounding box of the plate. The detected plate should be cut with more pixels in each direction to ensure that the largest part of the license plate appears. The system considers the CR stage as a separate classification issue.

The character recognition system (CR) contains the character segmentation and recognition stages. The CR system also used YOLO v4 due to its ability to detect the small objects in a license plate such as characters and digits. In this phase, the system should classify 23 classes (10 digits and 13 characters).The images of Iraqi license plate resized to 214 x 214 before applied on YOLO v4.

Iraqi license plates come in different colors. Each color means a specific type of vehicle i.e. white for private car, blue for government car, yellow for carry car, and Red for Taxi. So, binary threshold is applied to make the license plate background in black and text in white. However, a color check test is necessary because a red license plate has white background and black text after applying binary thresholding process. In such cases, an inverse binary threshold is applied. The color check test is performed by examining the color of the maximum

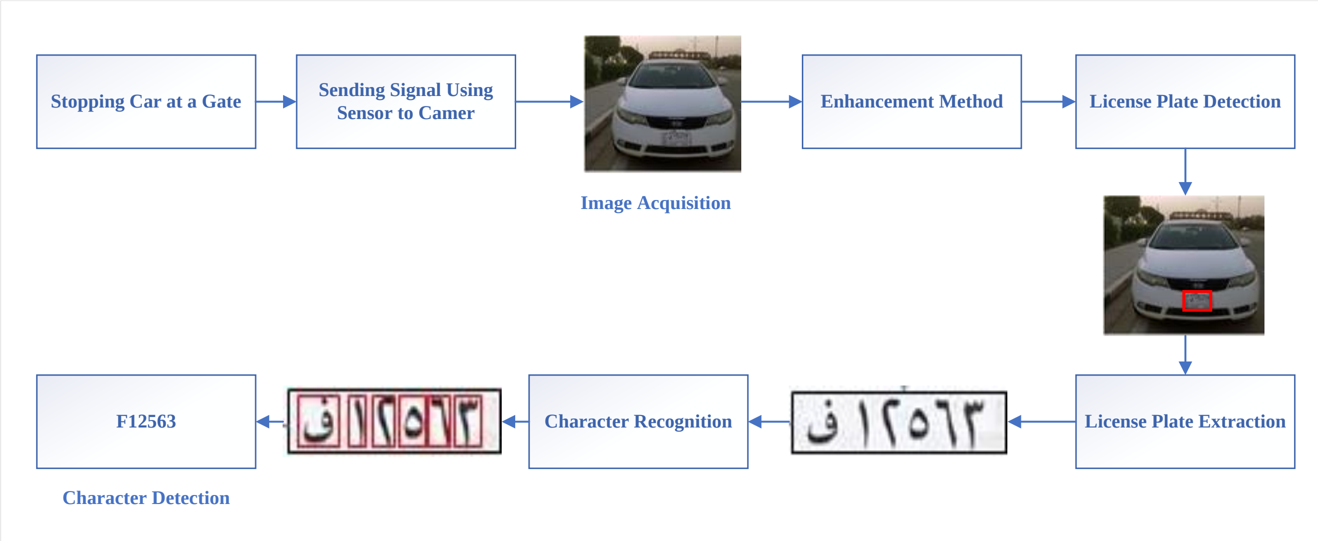


Fig. 2. Block diagram of the proposed license plate recognition system.

number of pixels in an image to determine the background color.
After that, Noise removal process is applied to remove the noise and unwanted pixels from the background. This process is done while scanning the connected components in the image and getting the coordinates and dimensions of each component. The dimension of each component should be the same as the Arabic numbers or letters used in Iraqi license plate.
In Arabic language the number zero is represented by the symbol (.) which looks like to a dot-shaped character. Due to a possible noise in an image, it may be challenging to distinguish whether it is the number zero or is just a noise. To determine whether a dot-shaped character is actually number zero, certain properties such as the size and position in an image and proportion to other characters can be examined and tested. The zero number has aspect ratio and its position lies in the vertical center or below it. By using these properties, it becomes possible to distinguish between a zero and the noise. The process is applied on all components in order to remove a noise from the background. After recognizing all characters/digits in an extracted LP, they were ordered according their distance from left part of a license plate.

### *C. Dataset*

License plates are different and specific to each country, so the algorithms used for LPR system are also different and specific. In many countries, the license plates are written in English language. The license plates also have a similar design for all cities in the country. Therefore, the recognition process of the characters will be facilitated. In Iraq, license plates have different forms, fonts and shape depending on the region that registered the vehicles. That makes LPR process more difficult than others. As shown in Figure 1 Iraq's license plates have many shapes depends on region, type and date of the license plate. The dataset is about 4000 Iraqi car license plate images. Our dataset covers different types of Iraqi car license plate. These images are collected from many different resources and from the real functional system which makes them more challenging than others in the standard conditions. For license plate detection system, 85% of dataset samples were used for training while the others for evaluation. The same training and evaluating data were considered for Characters recognition system.
When the data was gathered, the labeling process using LabelImg software was adopted to make annotation\labeling of license plate for a small set of data. After training a LPD system on this small data, the remaining data were applied on LPD system to generate annotation on them. Subsequently, the automatic labeling process on the remaining images was done, followed by applying a validation test. The dataset has two folders: one for images and other for annotations which contains coordinates of the license plate location (xmin, xmax, ymin, ymax) for the car image.
To enhance the detection and recognition accuracy for a model that based on deep learning, a large amount of data is required. Therefore, data augmentation methods such as brightness variation, gray-scaling and rotation were applied on images on both LP and their characters/digits to increase the size of training dataset. This leads to improve the performance of both detection and recognition systems and also prevent overfitting during the process of training. The dataset, which is used for

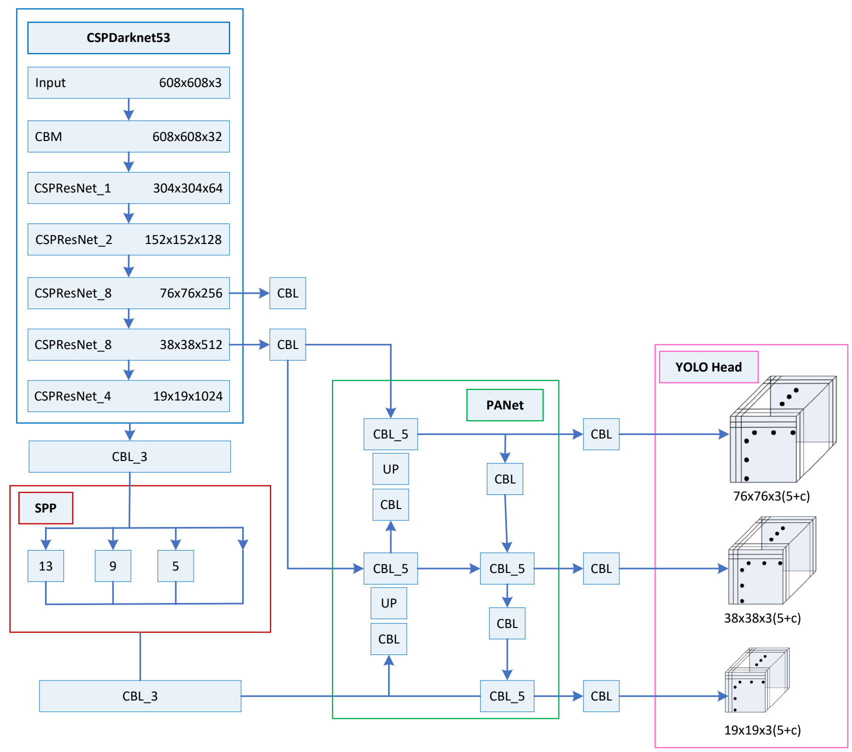


Fig. 3. The architecture of Yolov4 algorithm. The blocks included are CBL:Conv.+ Batch Norm + Leaky ReLU; CBM: Conv.+ Batch Norm + Mish; and UP: Upsampling.

training and evaluating the proposed system, represents color images with different illumination, resolutions, noise, shooting angles and shadow under various weather conditions. The deep learning algorithm (YOLOv4) is trained using training dataset, which includes ground truth annotations.
The performance of our proposed module has been evaluated on a wide range of real-world condition images some of which were captured by mobile phone cameras while others were collected from real surveillance systems. These images make the system more challenging compared to a system with standard conditions. In addition to the color images in a test data, grayscale images were gathered to show that the system can also be applied to the output of infrared cameras. For performance evaluation of the proposed system, the typical metrics are used including precision and recall. Fig. 4 demonstrates the complete approach of the proposed system. After applying the LPD module to an image, the character recognition module is the next step. A dataset including all digits and characters on Iraqi car license plates has been utilized for the character recognition system. The cropped LP from the LPD stage was used as inputs of character recognition stage. The characteristics of CR dataset can be seen in Table I, and as shown, the dataset includes 10 digits and 13 characters. An equivalent label is considered for each class. Accordingly, generate the training data for 23 classes, the segmentation process for character and digital is manually done.

## III. Discussion and Results

The effectiveness of both Iraqi car license plates detection and character recognition has been validated using the dataset described in the previous section. This work includes two kinds of datasets: car license plate dataset and numbers/character dataset. The training samples were input into the LPD system to provide the final system. The trained model has been evaluated on a personal computer in Python 3.7 using different

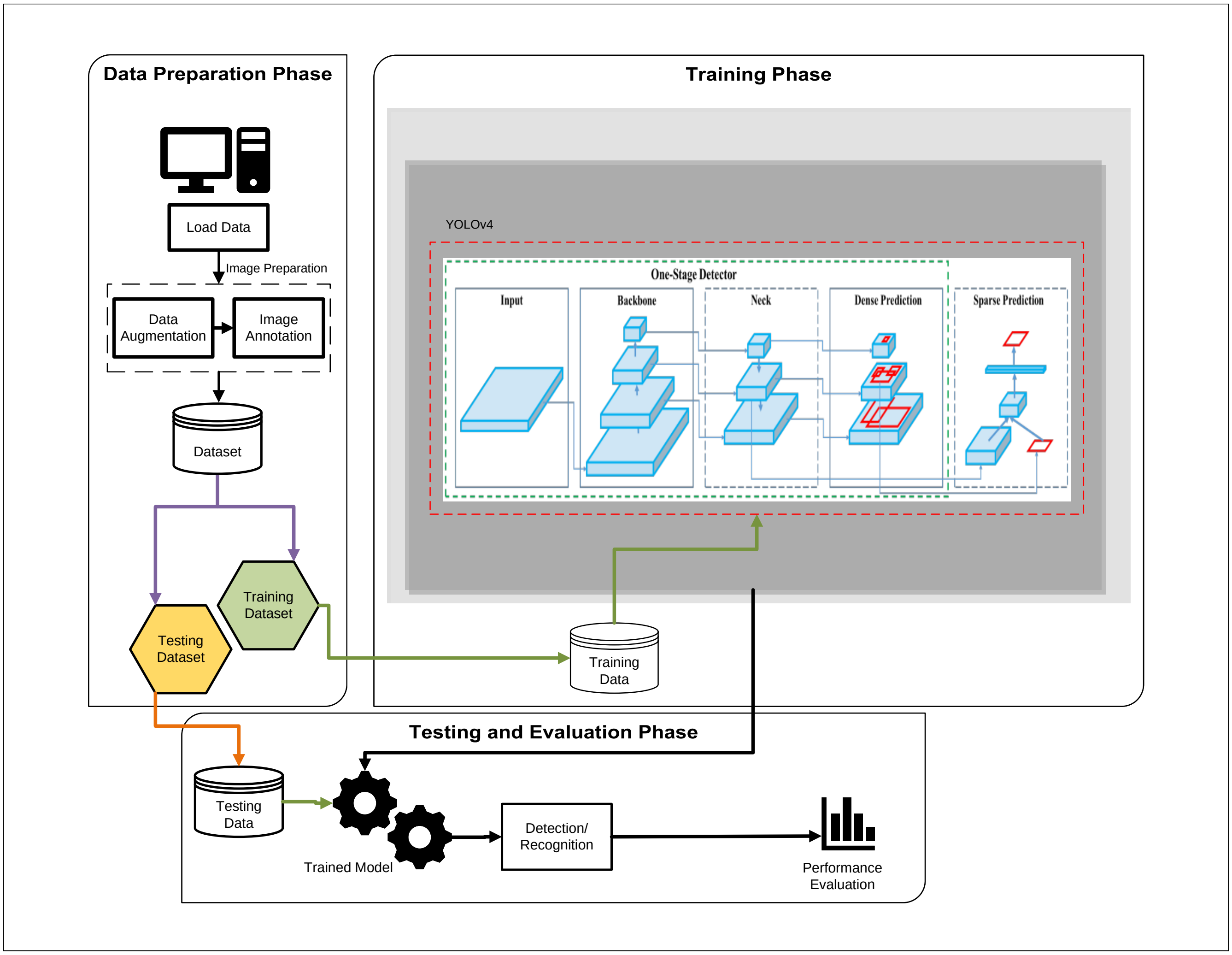


Fig. 4. The overall approach of the proposed system.

images of car with Iraqi license plate. Some of these samples has been captured by a camera of mobile phone, while others are collected from realistic surveillance system. Since the classes' number in our LPD and CR is different from the default configuration of YOLOv4 algorithm, this value was configured based on the number of classes. For tuning the network, there is a unique class for LPD system and the learning rate was $1 \times e^{-4}$ and epochs number was 100 epochs. Since the loss function did not improve from 1.64801 after 35 epochs, the training process stopped. For CR system, the classes' number is 23 class as mentioned in the previous section. The cropped image from LPD stage is fed for training in this stage with 45 epochs and loss function was 2.45. The training process for CR system was similar to training of LPD system and was carried out on the same H/W device.
Precision and Recall measurements were used to evaluate the system performance. To calculate Precision, number of correctly classified instances is dividing by a total number of classified samples as described mathematically in the following equation (2):

$$Precision = true\ positive/(true\ positive + false\ positive) \tag{2}$$

While Recall can be determined by dividing correctly classified instances number on a total number of ground truth data as follows (3):

$$Recall = true\ positive/(true\ positive + false\ negative) \tag{3}$$

To evaluate the performance of LPD system, the detected bounding boxes of license plate versus the coordinate on the ground-truth labels was checked. The values of intersection

TABLE I.
THE EXISTING ARABIC CHARACTERS IN IRAQI CAR LICENSE PLATES AND THEIR EQUIVALENT CHARACTER/NUMBERS THAT IS USED IN TRAINING STAGE.

| Arabic Character | Equivalent Label | Arabic Number | Equivalent Label |
|---|---|---|---|
| أ | A | ٠ | 0 |
| ب | B | ١ | 1 |
| ج | J | ٢ | 2 |
| ط | T | ٣ | 3 |
| د | D | ٤ | 4 |
| و | W | ٥ | 5 |
| ر | R | ٦ | 6 |
| ه | H | ٧ | 7 |
| ى | E | ٨ | 8 |
| ف | F | ٩ | 9 |
| ن | N | | |
| م | M | | |
| ك | K | | |

over union (IOU) are used to calculate the overlap area between the detecting bounding box and the ground-truth labels. The value of IOU is used after non-max suppression (NMS) as crucial value for evaluating ”confidence scores associated with specific class”. The IOU value is 0.5 and the the object threshold is 0.3 which means if the IOU is greater than 30%, it is considered successful, whereas less than 50% will be considered as detection fault. As indicated in Table II, the precision is much accurate. The performance of CR system was

TABLE II.
THE PERFORMANCE OF DETECTION OF IRAQI CAR LICENSE PLATES

| | |
|---|---|
| Object threshold | 0.3 |
| IoU | 0.5 |
| Non-Maximum suppression | 0.5 |
| Precision | 0.9812 |
| Recall | 0.9910 |

presented in Table III. The precision values are approximately equal for digits and characters (0.981), while the recall value for digits (0.995) is better than characters (0.989). So, the CR system can effectively recognize the character and digits in different Iraqi car license plate. Overall, many experiments have been done to show the proposed system performance in commercial use. The model can detect car license plate and obtain the test from the detected license plate correctly with accuracy of 95.07%. The test data includes car images captured in various resolution, brightness, illumination conditions and various shooting angles. The sizes of images in our dataset are different, therefore the proposed system does not depend on the resolution of image and can deal with small images which have low quality. The time distribution on the given dataset has been calculated to show the real time performance of our proposed system. Fig. 5 details the results of evaluation for a random amount of images. In most cases, the execution time for detection of license plate is 50ms, while characters and numbers are recognized in CR system in 80ms. The average time is required for LPD system is 53.78ms to detect license plate, while, CR system needs an average time of 72.86 ms to classify characters inside a detected car license plate. In this regards, the average end-to-end time to classify a sequence of characters for a given car image is 118.63 ms. These calculations prove the performance of the proposed system in a real time. The LPD stage fails to localize the LP in case the LPD system acquires an image with a partially visible license plate. Additionally, low illumination during night time and the presence of headlight beams of the vehicle reduce the LPD performance. The precision of the CR stage is directly affected by the accuracy of the LPD stage. Thus, if the LPD stage successfully detects the license plates for the vehicle which is far away from a camera, the characters/digits inside LP are unreadable, leading to misclassifications in CR stage. Table IV demonstrates the comparison among various versions of deep learning (YOLO) method which are used in LPD and CR phases. Our approach overcomes the limitation of existing license plate detection techniques, which

TABLE III.
ACCURACY OF CHARACTER RECOGNITION SYSTEM

| Data | Accuracy | Data | Accuracy |
|---|---|---|---|
| A | 100% | 0 | 99.81% |
| B | 99.54% | 1 | 97.38% |
| J | 100% | 2 | 100% |
| T | 89.97% | 3 | 98.26% |
| D | 100% | 4 | 94.34% |
| W | 91.07% | 5 | 99.91% |
| R | 99.99% | 6 | 95.89% |
| H | 100% | 7 | 100% |
| E | 93.25% | 8 | 99.76% |
| F | 100% | 9 | 92.69% |
| N | 98.11% | | |
| M | 100% | Total | 97.28% |
| K | 96.56% | | |

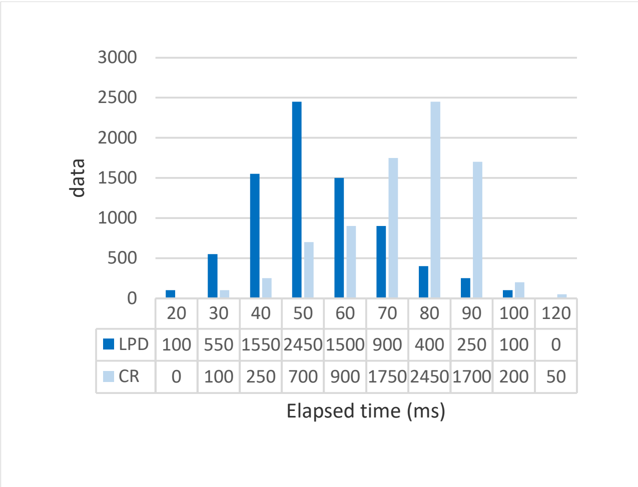


Fig. 5. The execution time for LPD and CR phases on number of data.

TABLE IV.
THE PERFORMANCE OF DETECTION OF IRAQI CAR LICENSE PLATES

| Reference | Elapsed Time | Overall Accuracy |
|---|---|---|
| [18] using YOLO v2 | 192.3 sec | 85.56% |
| [16] using YOLO v1 | 3.035 sec | 90.9% |
| [20] using YOLO v3 | - | 92.46% |
| Proposed method using YOLOv4 | 0.11863 sec | 95.07% |

often perform well in controlled environments but poorly in real-world scenarios. This disparity arises from their dependence on datasets with ideal conditions such as low noise, consistent brightness, and contrast, making them unsuitable for the diverse and challenging conditions of Iraqi license plate recognition. Our proposed system provides a robust end to end method for Iraqi vehicle license plate detection and recognition. Unlike many existing systems with interdependencies among stages, our system effectively detects license plate and accurately recognizes characters /digits under challenging conditions.

To the best of our knowledge, no deep learning method for Iraqi license plate recognition has achieved both super performance and accuracy. This is primarily due to the variation of Iraqi vehicle LP and the lack of annotated training-data and defined classes.

Also a comparison between our system and other existing systems has been shown in Table V, VI and VII respectively using three dataset: (1) UFPR dataset [3] which contains 4500 Brazilian plates images with resolution of 1920x1080, (2) Caltech dataset that available online (http://www.medialab.ntua.gr/research/LPRdatabase/) has 126 images of US plates with 896 x 592 resolution, and (3) Media Lab dataset that available online (http://www.medialab.ntua.gr/research/LPRdatabase) consisting of 726 images of Greek plates. As shown, the proposed work achieved the highest accuracy among the first two dataset on all competing algorithms. In last dataset, there is a small difference in accuracy. Our system achieved accuracy of 97.6%, while [29] had 97.9% due to the difference in subsets of training and testing. In addition, this dataset considers as a less challenging since all other systems have very high accuracy.

TABLE V.
THE PERFORMANCE OF THE PROPOSED SYSTEM AGAINST OTHER SYSTEMS USING UFPR DATASET

| Reference | Accuracy |
|---|---|
| [3] 2018 | 78.3% |
| [30] 2018 | 55.6% |
| [4] 2019 | 82.5% |
| Our proposed system | 92.8% |

TABLE VI.
THE PERFORMANCE OF THE PROPOSED SYSTEM AGAINST OTHER SYSTEMS USING CALTECH DATASET

| Reference | Accuracy |
|---|---|
| [31] 2016 | 77.5% |
| [32] 2018 | 94.1% |
| [4] 2019 | 96.1% |
| [22] 2020 | 97.8% |
| Our proposed system | 98.3% |

TABLE VII.
THE PERFORMANCE OF THE PROPOSED SYSTEM AGAINST OTHER SYSTEMS USING MEDIA LAB DATASET

| Reference | Accuracy |
|---|---|
| [29] 2018 | 97.9% |
| [22] 2020 | 96.9% |
| Our proposed system | 97.6% |

The proposed system based on YOLOv4 for both detection and recognition stages of LP. Since this algorithm is well-known for its accuracy and efficiency in detection, the processing time of overall system is quite low in addition to system accuracy. The efficiency of the proposed system that applied on various benchmark dataset is shown in Table VIII.

TABLE VIII.
THE EFFICIENCY OF THE PROPOSED SYSTEM ON DIFFERENT DATASETS

| Dataset | Execution time |
|---|---|
| UFPR dataset | 117.36 ms |
| Caltech dataset | 116.25 ms |
| Media Lab dataset | 117.78 ms |
| Our custom dataset | 118.63 ms |

## IV. CONCLUSIONS

A real-time system for Iraqi car license plate detection and recognition has been proposed in this paper. Due to the various backgrounds, font styles, font color, and sizes of the car license plates with non-standard characters in Iraq, a deep-learning method is applied to improve the efficiency of the license plate recognition system. Our system contains two deep-object detection networks, and a deep-learning algorithm has been implemented for both. The YOLO v4 method has been used for this purpose to have the advantage of high accuracy and real-time performance. Initially, the license plate detection (LPD) system finds the location of the license plate from the input image and then the detected license plate is processed by the character recognition (CR) system to recognize the license plate contents. The proposed system has been trained on a dataset collected in different weather, illumination and noise conditions. By testing the system on realistic data, the accuracy of the system is 95.07%. In addition, the average time of processing is 118.63 ms show the proposed system operates in real-time for each image. Extensive experiments have been carried out on different vehicle datasets to investigate the performance of our proposed system in comparison with the prior systems in license plate detection and recognition. Future work may include securing the privacy and identity of vehicle license plates across various parts of the system. Additionally, employing high dynamic range (HDR) can more effectively capture details in regions of extreme brightness and darkness, thus improving the system's accuracy.

## CONFLICT OF INTEREST

The authors have no conflict of relevant interest to this article.

## REFERENCES


[1] M. Sarfraz, M. J. Ahmed, and S. A. Ghazi, "Saudi arabian license plate recognition system," in *2003 International Conference on Geometric Modeling and Graphics, 2003. Proceedings*, pp. 36–41, IEEE, 2003.

[2] M. Ahmed, M. Sarfraz, A. Zidouri, and W. Al-Khatib, "License plate recognition system," in *10th IEEE International Conference on Electronics, Circuits and Systems, 2003. ICECS 2003. Proceedings of the 2003*, vol. 2, pp. 898–901 Vol.2, 2003.

[3] R. Laroca, E. Severo, L. A. Zanlorensi, L. S. Oliveira, G. R. Gonçalves, W. R. Schwartz, and D. Menotti, "A robust real-time automatic license plate recognition based on the yolo detector," in *2018 international joint conference on neural networks (ijcnn)*, pp. 1–10, IEEE, 2018.

[4] R. Laroca, L. Zanlorensi, G. Gonçalves, E. Todt, W. Schwartz, and D. Menotti, "An efficient and layout-independent automatic license plate recognition system based on the yolo detector. arxiv 2019," *arXiv preprint arXiv:1909.01754*.

[5] S. Yonetsu, Y. Iwamoto, and Y. W. Chen, "Two-stage yolov2 for accurate license-plate detection in complex scenes," in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–4, IEEE, 2019.

[6] Z. Liu, Z. Wang, and Y. Xing, "Wagon number recognition based on the yolov3 detector," in *2019 IEEE 2nd international conference on computer and communication engineering technology (CCET)*, pp. 159–163, IEEE, 2019.

[7] A. R. Youssef, A. A. Ali, and F. R. Sayed, "Real-time egyptian license plate detection and recognition using yolo," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 7, 2022.

[8] V. Jain, Z. Sasindran, A. Rajagopal, S. Biswas, H. S. Bharadwaj, and K. Ramakrishnan, "Deep automatic license plate recognition system," in *Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing*, pp. 1–8, 2016.

[9] M. Al-Yaman, H. Alhaj Mustafa, S. Hassanain, A. Abd AlRaheem, A. Alsharkawi, and M. Al-Taee, "Improved automatic license plate recognition in jordan based on ceiling analysis," *Applied Sciences*, vol. 11, no. 22, p. 10614, 2021.

[10] C. Gou, K. Wang, Y. Yao, and Z. Li, "Vehicle license plate recognition based on extremal regions and restricted boltzmann machines," *IEEE transactions on intelligent transportation systems*, vol. 17, no. 4, pp. 1096–1107, 2015.

[11] Q. Wang, "License plate recognition via convolutional neural networks," in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pp. 926–929, IEEE, 2017.

[12] S. M. Silva and C. R. Jung, "License plate detection and recognition in unconstrained scenarios," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 580–596, 2018.

[13] G. Y. Abbass and A. F. Marhoon, "Iraqi license plate detection and segmentation based on deep learning," *Iraqi Journal for Electrical and Electronic Engineering*, vol. 17, no. 2, pp. 102–107, 2021.

[14] Y. Elhadi, O. Abdalshakour, and S. Babiker, "Arabic-numbers recognition system for car plates," in *2019 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE)*, pp. 1–6, IEEE, 2019.

[15] I. R. Khan, S. T. A. Ali, A. Siddiq, M. M. Khan, M. U. Ilyas, S. Alshomrani, and S. Rahardja, "Automatic license plate recognition in real-world traffic videos captured in unconstrained environment by a mobile camera," *Electronics*, vol. 11, no. 9, p. 1408, 2022.

[16] S. R. Ahmed, M. R. Ahmed, D. A. Majeed, A. H. Hammed, A. salam Daham, and E. H. Hammed, "Android application to retrieve car details from car plate numbers," in *IOP Conference Series: Materials Science and Engineering*, vol. 928, p. 032026, IOP Publishing, 2020.

[17] M. Shehata, M. T. Abou-Kreisha, and H. Elnashar, "Deep machine learning based egyptian vehicle license plate recognition systems," *arXiv preprint arXiv:2107.11640*, 2021.

[18] D. Habeeb, F. Noman, A. A. Alkahtani, Y. A. Alsariera, G. Alkawsi, Y. Fazea, A. M. Al-Jubari, *et al.*, "Deep-learning-based approach for iraqi and malaysian vehicle license plate recognition," *Computational intelligence and neuroscience*, vol. 2021, 2021.

[19] R. Antar, S. Alghamdi, J. Alotaibi, and M. Alghamdi, "Automatic number plate recognition of saudi license car plates," *Engineering, Technology & Applied Science Research*, vol. 12, no. 2, pp. 8266–8272, 2022.

[20] A. R. Youssef, A. A. Ali, and F. R. Sayed, "Real-time egyptian license plate detection and recognition using yolo," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 7, 2022.

[21] H. H. Yaba and H. O. Latif, "Plate number recognition based on hybrid techniques," *UHD Journal of Science and Technology*, vol. 6, no. 2, pp. 39–48, 2022.

[22] C. Henry, S. Y. Ahn, and S.-W. Lee, "Multinational license plate recognition using generalized character sequence detection," *IEEE Access*, vol. 8, pp. 35185–35199, 2020.

[23] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[24] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "Cspnet: A new backbone that can enhance learning capability of cnn," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 390–391, 2020.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[26] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8759–8768, 2018.

[27] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[28] H. Hakim and A. Fadhil, "Survey: Convolution neural networks in object detection," in *Journal of Physics: Conference Series*, vol. 1804, p. 012095, IOP Publishing, 2021.

[29] J. Zhuang, S. Hou, Z. Wang, and Z.-J. Zha, "Towards human-level license plate recognition," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 306–321, 2018.

[30] G. R. Gonçalves, M. A. Diniz, R. Laroca, D. Menotti, and W. R. Schwartz, "Real-time automatic license plate recognition through deep multi-task networks," in *2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*, pp. 110–117, IEEE, 2018.

[31] H. Li and C. Shen, "Reading car license plates using deep convolutional neural networks and lstms. arxiv 2016," *arXiv preprint arXiv:1601.05610*.

[32] H. Li, P. Wang, and C. Shen, "Toward end-to-end car license plate detection and recognition with deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 1126–1136, 2018.