Al-Nahrain Journal of Science
ANJS, Vol.25 (3), September, 2022, pp. 43-50

A Comparison Between the Use of CNN and Matching Templates in Recognizing the Iraqi License Plate Number

Shayma A. Hmdaoy* and Hanaa M. Ahmed

Computer Science Department, Technology University, Baghdad, Iraq

Article's Information

Received:
07.06.2022
Accepted:
23.08.2022
Published:
30.09.2022

Keywords:
SIFT
Contour
CNN
Templates matching
LPN

Abstract

One of the important applications that organize people's lives is the applications that relate to organizing the traffic regulation, detecting violators, detecting stolen cars, and managing car parks. Recognizing the license plate number (LPN) crosses the basic process in all previous applications. This process is affected by the surrounding light conditions while taking an image of the car. In addition to these problems, the plates of modern Iraqi cars (the so-called German number) suffer from words in silver color printed on the plate, which causes a senior problem when locating the LPN and recognizing it. To solve these problems, we presented in this research a comparison between the use of the SIFT and Contour algorithm in locating the LPN. While using CNN training models to recognize the PN achieved higher results than template matching.

DOI: 10.22401/ANJS.25.3.08
Corresponding author: cs.19.79@grad.uotechnology.edu.iq

1. Introduction

The primary means of transportation at present is the car. So that to regulate traffic and keep people safe, it is necessary to have systems characterized by accuracy in detecting drivers violating traffic laws, detecting stolen cars, or even in parking management [1,2]. Any system to recognize plate numbers passes through three primary stages: capturing the image of the car, locating the LPN, and finally recognizing the number [3].

The type of camera, camera position, and lighting conditions while taking the image all affect the image quality. The image quality of the captured image dramatically affects the final results of any system [4,5]. Therefore, preprocessing the image to improve its quality is considered one of the critical stages before determining the location of the license plate number [6]. Many filters can enhance the quality of an image, such as the Median filter, Sobel filter [7], and Gaussian filter [8].

Determine the location of the LPN in the image and isolate it from the rest of the image It is the basis of the work. Researchers have provided many algorithms for this purpose, Such as edge detection algorithms, which depend on defining the edges of all shapes in the image and then segmented the rectangular shape from them, like Sobel filters [9], Gabor filter, canny edge detection [10,11]. Some researchers relied on locating the LPN using color features [12]. Other researchers determine the location based on extracting the characteristics of the plate, Like the use of the (SIFT) algorithm [13]. Other researchers have used feature learning algorithms to determine the location, such as Viola_jones [14], CNN and YOLO [15,16,17].

After locating the LPN, the stage of identifying the numbers comes. One of the researchers dealt with LPN as a single image without extracting their numbers. Then match them with other LPN in the database using different algorithms, like matching templates [18], SIFT [19], CNN [5], etc. Or the letters are extracted from the PN and then matched using the OCR [20,21].

The colors and the writing language of the numbers differ from one country to another, so sometimes, different processing methods are needed. Iraqi LPN are divided into three types: The central and southern governorates used two types, the old model issued before 2003 and a modern model after 2003. The northern governorates use another model [22]. The old model before 2003 and the model used in the north of governorates are currently under change, so this research focuses on the model after 2003, the most common model among the three models.

2. Related Works

In (2017), the researchers proposed a PN detection system that first uses a Top Hot filter to remove small parts of the image with less than 70 pixels. Then using Otsu's thresholding to convert the image to a binary image. Then use the deletion method to delete parts whose area is less than 50 pixels. After locating and segmented the PN, they classified it into three types according to the area. If its area is less than 120, it is considered the old model. If it is less than 155, it is considered the modern model. Finally, if less than 200, it is considered the northern model. Then they cut and extract the numbers using. As for revealing the numbers, it was done after matching them with templates for numbers and letters that were previously stored. The template that gets the highest correlation is stored in a text file. The system has achieved an accuracy of up to 86.6%. The database used consists of 40 images taken in different lighting conditions [22].

In (2019), A template for the LPN, is designed, devoid of numbers and written words, then uses the object detection features to search for the location of the license plate number in the image. Then the system searches for the numbers inside the LPN after cutting it. The system uses a template matching algorithm to search for numbers by prepared number templates and stored in a database in advance. The system has been tested on 50 images of German-type LPN. The system verifies the numbers and neglects the letters on the LPN. The proposed system achieved an accuracy of 96% [23].

In (2020), the researchers proposed a three-stage number detection system. The first is to locate the LPN, the second is to segment letters and numbers, and the last stage is to identify the number of the plate. Canny edge detection was used at the stage of locating the LPN. Then binarization of the care license plate. Connect component analysis algorithm was used for segmenting the letters and numbers from the binary image. In the last stage, ANN was used to recognize the LPN. The network was trained using 460 images of letters and numbers. The network outputs are categorized into 10 Indian numbers and 14 Arabic letters. The proposed system achieved an accuracy of 91% [8]. The proposed system did not specify the number of images for each letter or number used for training and did not specify the method of evaluating the accuracy of the trained network.

In (2022), The PN location was determined using the SSD algorithm. CNN model were used to recognition LPN. This CNN models contains 22 classes, 10 classes for numbers from 0 to 9, and 12 classes for letters. Moreover, 1200 images trained the other model to recognize the names of the governorates. However, the system achieved an accuracy of 98% after testing it on 500 images. In addition, the system did not consider the letters containing dots when extracting the letters, as it specified the space of the symbols to be extracted between 200 and 2800, given that zero is the smallest object and the space is up to 200. The network was also trained on 12 letters only, without specifying what the letters are, knowing that there are currently 16 letters used on the number plate of Iraqi cars of the German type [24].

3. Proposed Approach

The research aims to find an efficient way to recognize the number of Iraqi license plates, especially for cars with the word Iraq printed on their number plates in silver color. These words cause a change in the characteristics of the outer edges of the PN and the numbers. The system consists of several stages, as follows:

1- Database preparing:

There is no database of Iraqi car numbers announced or used to compare previous research and the proposed system. Therefore, a database for the proposed system was created, and more than one method was tested in both the number plate extraction and the number discovery phases. A Samsung mobile phone camera captured 160 images for the database. Images were taken under different lighting conditions and at different angles.

2- Locate and segment the PN:

This stage is considered one of the critical stages in the system. Because to locate the PN, we need to define its outer edges. The system faced many problems, including writing the word on the LPN in silver color where the word is written in different positions on the edges and numbers, which leads to a change in the properties of the edges for both the PN and the numbers themselves, As shown in Figure 1. In addition to many other problems such as rotating the image and changing the brightness and noise.

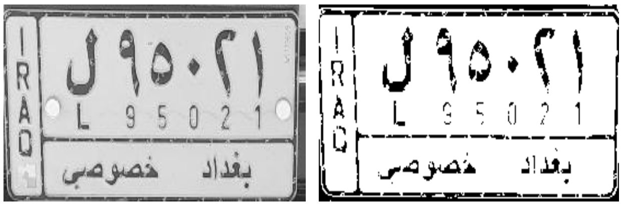

Figure 1. An example of modern car plate numbers.

The first problem that the proposed system will begin to solve is the problem of image rotation by using Projective transformations, which vary depending on the camera's position while taking the image. Projective transformations Increase the accuracy of the proposed system in detecting the features of objects in the image after adjusting the image coordinates [25,26]. Projective Transform Matrix has eight parameters (k11-k32) to transform (x,y) pixel coordinate to (x_1,y_1) pixel coordinate, as shown in equations (1,2,3).

$$\begin{bmatrix} x1 \\ y1 \\ 1 \end{bmatrix} = \begin{bmatrix} k11 & k12 & k13 \\ k21 & k22 & k23 \\ k31 & k32 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \text{...(1)}$$

$$x_1 = \frac{k11+k12+k13}{k31+k32+1} \quad \text{...(2)}$$

$$y_1 = \frac{k21+k22+k23}{K31+K32+1} \quad \text{...(3)}$$

The parameter's values are determined according to the image's angle, as shown in Figure 2.

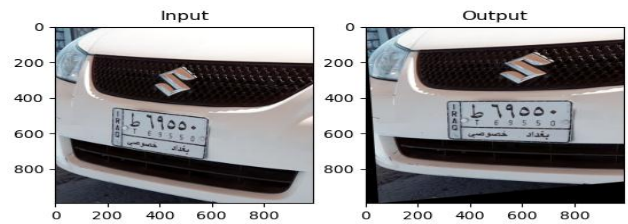

Figure 2. An example of rotating a car image.

The second problem that needs to be addressed is the problem of noise in the image. A bilateral filter was used to improve the captured image quality, which is characterized by smoothing the image, removing noise, and preserving edges simultaneously [27,28]. The bilateral filter has four parameters: input image, neighborhood diameter, **si**gma Color, and sigma Space.

After adjusting the rotation of the image if it needs to be modified and noise removed, it comes to the stage of locating the LPN and definitely. Two methods have been tried, SIFT and contour. The steps of their work and the final results will be explained separately.

2.1 Using SIFT to find the plate location:

SIFT is used to search for an object within an image or a two-image match. Four models of the license plate number were designed. The forms are devoid of numbers but contain the words that indicate the type of car, whether carrying, private, taxi, or government. In the proposed system, SIFT is used to detect and describe key points for all models of car PN images, as shown in Figure 3.

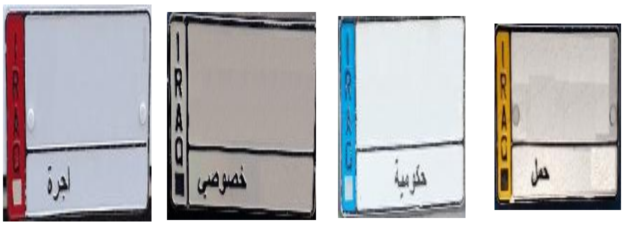

Figure 3. Models of the license plate number used with SIFT.

Then the key points are stored in a vector of 128 values. The SIFT repeats the same previous steps with the target car image. Keypoint descriptors are matched for each model individually with the keypoint descriptor of the target car image. Extract good keypoint according to the test ratio. The system determined the plate model that has the largest good keypoint. Then, determine the plate number's location in the target car image according to the locations of the good key points. Rotating the image to the appropriate position effectively contributed to improving the results, as shown in Figure 4.

Figure 4. Locate the PN by SIFT.

Despite the efficiency of the SIFT, it suffers from the problem of slowness in detecting the location of the plate, so a second method was used, which is the contour method, to locate the plate position.

2.2 Using contour to find the plate location:

A contour is a curve linking all the same-colored points along a border. Detecting image contours helps find image boundaries. Closed curves are different from edges. Thresholding an image before extracting contours improves its accuracy. Therefore, three different degrees of threshold (90,120,190) were used to solve the problem of diverse lighting and the color of the car compared to the color of the PN, as shown in Figure 5. Threshold values were manually determined by the experiment.

Threshold (90) Threshold (120) Threshold (190)

Figure 5. Using three different degrees of the threshold.

The LPN is rectangular, so the object that contains four sides are segmented. Here, another problem appeared: segmenting the incorrect parts due to other objects in the car characterized by their square shapes. The problem is solved using the following equation:

$$z = W/L$$

where W represents the width and L represents the rectangle's height, the object is segmented if the value of Z is between 2 and 2.5. The final results on the dataset proved that the contour was superior to the SIFT in terms of speed and accuracy. The SIFT achieved an accuracy of 86.2. The contour achieved an accuracy of 88.7.

2.3 Using OCR to recognize plate numbers:

Many researchers have used the method of matching templates matching or artificial intelligence networks to identify numbers. The two methods have been tried on the database, but in a different ways to solve the problem of writing that appears above the numbers, as follows:

1- OCR Using templet matching:

After locating the plate and segmenting it, converting it to a binary image, then using the erosion morphology method to fill in the small white spaces in the numbers and letters, the Arabic numbers and letters are extracted using the contour. The small English numbers are eliminated by extracting objects that exceed an area of only 200. Finally, the extracted numbers and letters are matched one by one with pre-prepared templates. The template with the highest matching score is placed in a list.

This method is faster and less costly in terms of calculations, compared to using CNN. But its accuracy is low, especially after using erosion on the numbers, which led to many prediction errors.

2- OCR based on CNN:
In this method, the initial stages of extracting numbers are similar to the previous method, where the image is converted to binary and the contour is used to find the outer limits of the number. But the difference is that after determining the outer limits of the numbers, a special function is used whose function is to find the rectangular perimeter closest to the outer limits of the number. The number is then cut from the color extracted PN according to the rectangle's boundaries. The images of numbers and letters extracted from the PN are then recognized using two pre-trained CNN models.

The CNN used to distinguish numbers was trained on a database containing 10 numbers from 0 to 9. Each number has 160 photos. CNN for letters was trained on a database of 16 letters for each letter 50 images. Letters and numbers were cut manually and in equal sizes from images of different license plates taken in different lighting conditions. Figure 6 contains sample images from the databases used in CNN training.

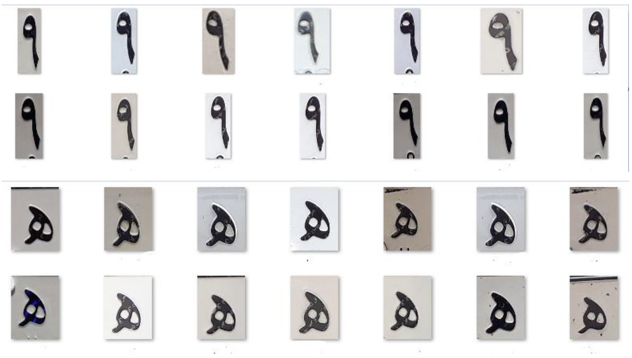

Figure 6. Samples of images used in network training

This CNN model has been used to train two models, one for numbers and the other for letters. In two models, the database was divided into 0.8 for training, 0.1 for validation, and 0.1 for testing.

To avoid the poor performance of the CNN model, which leads to overfitting problems, earlier stopping and checkpoint strategies are used. Early Stopping is used to stop training when the validation error reaches a minimum,[29]. While in a checkpoint, the model will only be saved if its validation set performance is the best[29]. The general structure of the proposed CNN model to classify the letters is illustrated by the following layers:

1. Convolution layer with 32 filters, a kernel size of (3*3), and a stride of one. The size of the input image is 32 * 32*3. The type of activation function is ReLu in this layer.
2. Convolution layer with 32 filters, a kernel size of (3*3), and a stride of one. The type of activation function is ReLu in this layer.
3. Max pooling layer with a pool size (3*3).
4. Convolution layer with 64 filters, a kernel size of (3*3), and an interaction function of type ReLu.
5. Convolution layer with 64 filters, a kernel size of (3*3), and an interaction function of type ReLu.
6. Convolution layer with 64 filters, a kernel size of (3*3), and an interaction function of type ReLu.
7. Max pooling layer with a pool size of (3*3).
8. Convolution layer with 128 filters, a kernel size of (3*3), and an interaction function of type ReLu.
9. Max pooling layer with a pool size of (3*3).
10. Flatten layer.
11. Dense hidden layer with 128 neurons, with ReLU activation function.
12. The last layer is the dense output layer with 16 neurons; one neuron for each class. The type of activation function is sotfmax in this layer.

Figure 7 contains a graph showing the topology of the proposed CNN model.

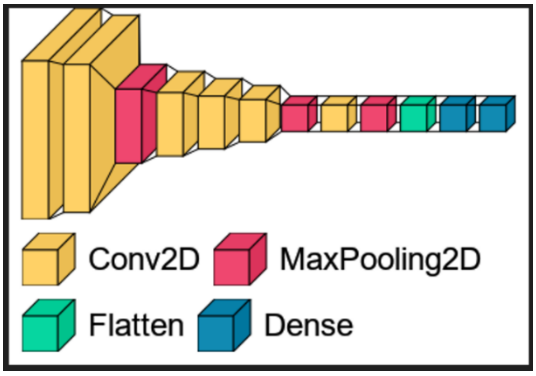

Figure 7. The proposed topology CNN model to recognition of symbol.

In order to evaluate the network performance, the Confusion Matrix and the classification report were used. The confusion matrix is one of the most important parts of classification metrics. The confusion matrix is a table that categorizes the number of correct and incorrect predictions by the response type, where it contains a set of values the are :true positives (TP), false positives (FP), false negatives (FN) and true negatives (TN), as show in Figure 6, [30]. as shown in Figure 8.

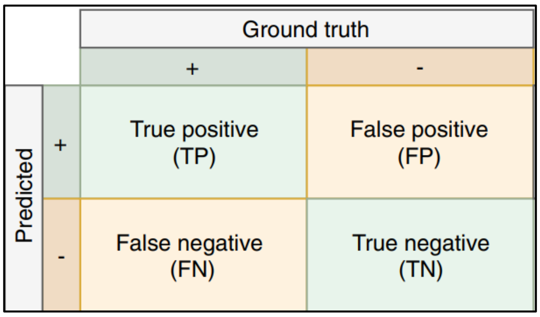

Figure 8. Confusion matrix [30].

The network performance can be better understood and analyzed using measurements other than the confusion matrix. Most of these measurements are based on values of the confusion matrix[31]. Among the most famous of these measures are the following:

1- Accuracy: It shows the number of cases that were correctly labeled, and it is calculated according to the following equation [31]:

$$Accuracy = \frac{(TP+TN)}{TP+TN+FP+FN}$$

2- Preision: It is worked out by finding the percentage of cases that were marked as positive. A model that gives more relevant results than irrelevant results has a higher precision value. It is calculated according to the following equation [31]:

$$Precision = \frac{TP}{TP+FP}$$

3- Recall: It is worked out by finding the percentage of "positive" cases that were labeled as such. A high recall value means that a model finds most of the right answers, even if it also finds some wrong answers. Recall is calculated according to the following equation[31]:

$$Recall = \frac{TP}{TP+FN}$$

4- F-Score: It is the middle ground between precision and recall.is calculated according to the following equation [31]:

$$F-measure = 2 * \frac{Precision*Recall}{Precision+Recall}$$

The number of observations in each class is a measure of support. Figures 9 represent confusion matrices describing the classification's performance of CNN models for recognizing letters and digits. While Table No1 contains the results extracted from the classifier's report to describe the performance of CNN training models to recognize letters and digests, respectively.

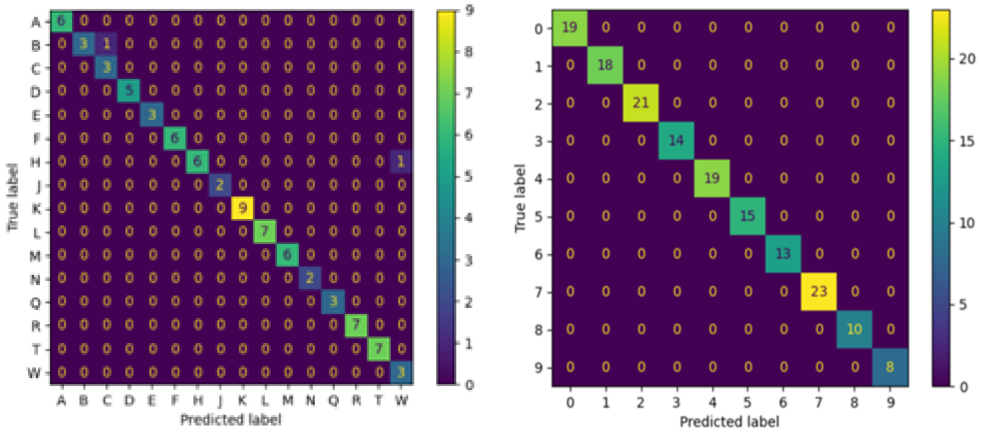

Figure 9. Describe the classification's performance of CNN models for recognizing letters and digits.

Table 1. results of classifier's report to recognition letters.

Class	Precision	Recall	f1-Score	Support
0	1.00	1.00	1.00	6
1	1.00	0.75	0.86	4
2	0.75	1.00	0.86	3
3	1.00	1.00	1.00	5
4	1.00	1.00	1.00	3
5	1.00	1.00	1.00	6
6	1.00	0.86	0.92	7
7	1.00	1.00	1.00	2
8	1.00	1.00	1.00	9
9	1.00	1.00	1.00	7
10	1.00	1.00	1.00	6
11	1.00	1.00	1.00	2
12	1.00	1.00	1.00	3
13	1.00	1.00	1.00	7
14	1.00	1.00	1.00	7
15	0.75	1.00	0.86	3

The following figure illustrates by graphing the data in Table No. 1

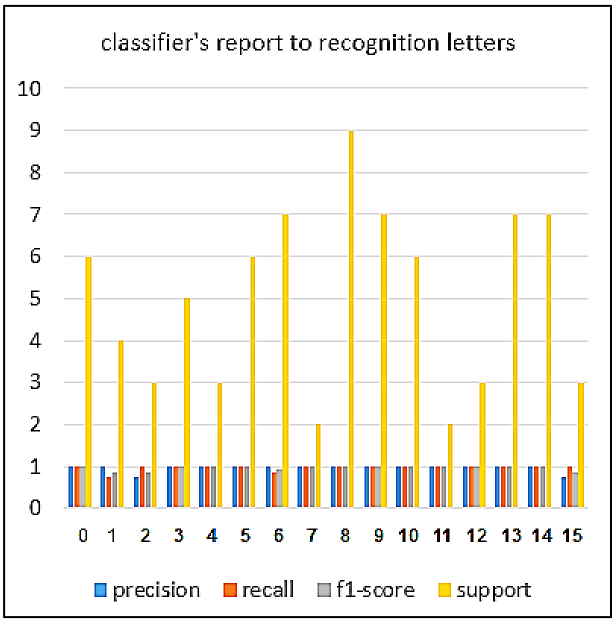

Figure 10. Results of classifier's report to recognition letters

Table 2. Results of classifier's report to recognition digits.

Class	Precision	Recall	f1-Score	Support
0	1.00	1.00	1.00	19
1	1.00	1.00	1.00	18
2	1.00	1.00	1.00	21
3	1.00	1.00	1.00	14
4	1.00	1.00	1.00	19
5	1.00	1.00	1.00	15
6	1.00	1.00	1.00	13
7	1.00	1.00	1.00	23
8	1.00	1.00	1.00	10
9	1.00	1.00	1.00	8

The following figure illustrates by graphing the data in Table No. 2

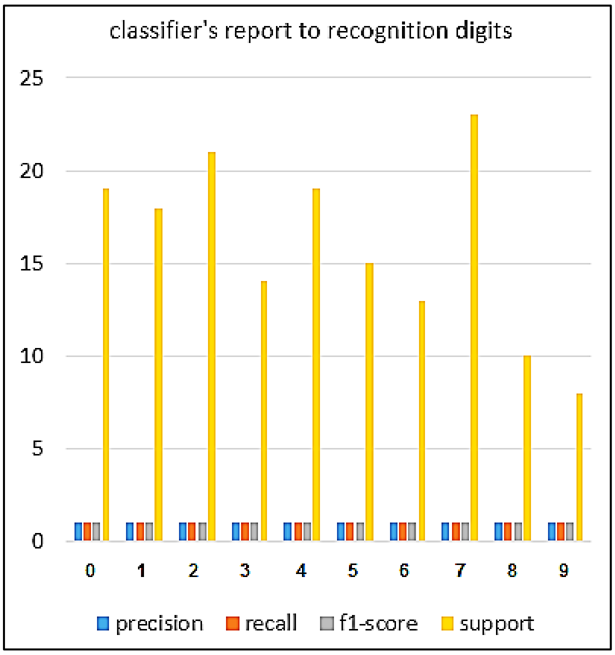

Figure 11. Results of classifier's report to recognition digits.

From the previous results, it is noted that the CNN model can recognize the letters with an accuracy of up to 97.4. while the CNN model for distinguishing numbers has achieved an accuracy of up to 100% on the test data.

The programming language used to implement the proposed system is Python.

The previous methods were combined to design a comprehensive system capable of locating and segmenting the LPN and then extracting numbers and letters to recognize them. The final results of combining the methods are shown in the following table:

Table 3. The final results of the combination of methods of segmenting and recognition of the LPN.

Methods	Accuracy	Speed
SIFT and templet matching	48.17	7.8418
Contour and templet matching	72.53	0.2343
SIFT and CNN	89.13	5.5232
Contour and CNN	96.47	2.177

Algorithm 1 illustrates the steps of Recognition the LPN Using CNN models.

Algorithm 1. Recognition the LPN Using the contour method and CNN models	
Input	the car image
Output	A list of LPN symbols
Step1	begin
Step2	Using the bilateral filter to remove the noise
Step 3	Convert the image to binary using an appropriate threshold
Step4	Using the contour method to locate and cut the PN of the car
Step5	Converting the deducted LPN to a binary image using the same threshold level as the previous one
Step 6	Extracting numbers and letters from the LPN using the contour as well
Step 7	Using CNN-trained models to identify the LPN
Step 8	End

The following figure contains pictures as examples illustrating the steps mentioned in the algorithm above.

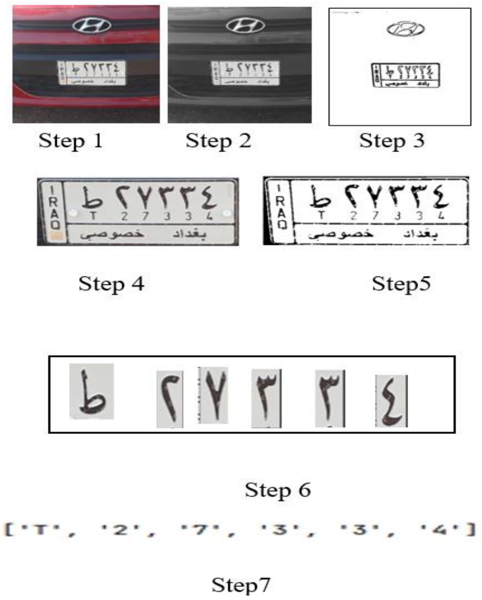

Figure 11. Examples illustrating the steps mentioned in the Algorithm 1.

Comparing the proposed system with previous works:

It is difficult to compare the proposed system with previous work in the same field of research due to the lack of a published database of Iraqi car numbers that was used in this research. Nevertheless, we made a general comparison with the research previously mentioned in the related works in terms of the type of problems that were not addressed as follows:

- **In (2017):** the researchers did not address the problem of having silver color writing above the numbers. In addition to, decreased accuracy due to the use of template matching method.
- **In (2019):** the researchers use a template prepared in advance to find the location of the LPN free of symbols using the method of templates matching, but this model did not take into account the presence of writing in silver color over the number plate.
- **In (2020):** The researchers used ANN to recognize the number plate, where the model was trained using 14 letters and 10 numbers. The system did not clarify how to distinguish between the number zero and the dots above the letters, then the number of letters used so far is 16, while the system used only 14 without mentioning what letters are used.
- **In (2022):** The researchers used CNN to recognize the number plate, where the model was trained using 12 letters and 10 numbers. The system did not clarify how to distinguish between the number zero and the dots above the letters, then the number of letters used so far is 16, while the system used only 14 without mentioning what letters are used.
- **The proposed system:** In comparison with previous research, the system has solved the problem of having silver color writing above the LPN. In addition to solving the problem of similarity between the number zero and the presence of dotes above some letters, where the letters were separated manually, as for the numbers, the contour was used to divide them, and two CNN models were used, one for the numbers and the other for the letters. The litter CNN model contains 16 characters, which are all the characters used so far. The system also dealt with the problem of image rotation to a degree that leads to a decrease in the accuracy of the results, which were not addressed in any of the previous systems, In addition, we focused in our research on improving the quality of images and increasing the number of samples in the database for each symbol used in training the CNN model, which contributed to raising the accuracy of the system.

4. Conclusions

The proposed system aims to build an integrated system capable of quickly and accurately identifying LPN. The proposed system can be used in several applications such as detecting stolen cares, reducing traffic accidents, organizing parking lots, etc. To determine the location of the LPN, two methods were tested, namely the SIFT and the contour. The results of the comparison between the two methods proved that the contour is faster and more accurate than SIFT. To recognize the LPN, two methods have been tried, namely, matching templates and CNN. The final results proved that the use of contour and CNN to recognize the LPN is more accurate and faster. The proposed system achieved an accuracy of up to 97.4%

References

[1] Bansal S.; Gupta M.; and Tyagi A. K.; “A Necessary Review on Optical Character Recognition (OCR) System for Vehicular Applications". Proc. 2nd Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2020: 918-922, 2020, doi: 10.1109/ICIRCA48905.2020.9183330.

[2] Abd Alhamza D. A. and Alaythawy A. D.; “Iraqi License Plate Recognition Based on Machine Learning". Iraqi J. Inf. Commun. Technol.; 3(4): 1–10, 2020, doi: 10.31987/ijict.3.4.94.

[3] Pustokhina I. V.; et. al, “Automatic Vehicle License Plate Recognition Using Optimal K-Means with Convolutional Neural Network for Intelligent Transportation Systems". IEEE Access, 8: 92907–92917, 2020, doi: 10.1109/ACCESS.2020.2993008.

[4] Lubna N. M. and Shah S. A. A.; “Automatic Number Plate Recognition : A Detailed Survey of". Sensors, 21(9): 3028, 2021, [Online]. Available: https://www.mdpi.com/1424-8220/21/9/3028/pdf

[5] Silva S. M. and Jung C. R.; “Real-time license plate detection and recognition using deep convolutional neural networks". J. Vis. Commun. Image Represent.; 71: 102773, 2020, doi: 10.1016/j.jvcir.2020.102773.

[6] Khinchi M. and Agarwal C.; “A review on automatic number plate recognition technology and methods". Proc. Int. Conf. Intell. Sustain. Syst. ICISS 2019 (Iciss: 363–366, 2019, doi: 10.1109/ISS1.2019.8908014.

[7] Mohammed F. G. and Jasem N. H.; “Review on License plate recognition algorithms in images and video". Int. J. Eng. Res. Adv. Technol.; 06(02): 18–25, 2020, doi: 10.31695/ijerat.2020.3599.

[8] Ali H. B. and Hathal M. S.; “Development of Iraqi License Plate Recognition System based on Canny Edge Detection Method". J. Eng.; 26(7): 115–126, 2020, doi: 10.31026/j.eng.2020.07.08.

[9] Menon A.; “Detection And Recognition of Multiple License Plate From Still Images". 2018 Int. Conf. Circuits Syst. Digit. Enterp. Technol.: 1–5, 2018.

[10] Ha P. S. and Shakeri M.; “License Plate Automatic Recognition based on edge detection License Plate Automatic Recognition based on Edge Detection". November, 2018, doi: 10.1109/RIOS.2016.7529509.

[11] Antar R.; “Automatic Number Plate Recognition of Saudi License Car Plates”. 12(2): 8266–8272, 2022.

[12] Zaheen M. Y.; Mohi-u-din Z.; Siddique A. A. and Qadri M. T.; “Exhaustive Security System Based on Face Recognition Incorporated with Number Plate Identification using Optical Character Recognition".

Mehran Univ. Res. J. Eng. Technol.; 39(1): 145–152, 2020, doi: 10.22581/muet1982.2001.14.
[13] Senbeto K. A.; Chung P. Y. K. and Zerihum D.; “Vehicle Plate Recognition for Ethiopian License Plate : based Sift Feature". Int. J. Eng. Res. Technol.; 8(06): 568–570, 2019, [Online]. Available: https://www.ijert.org/vehicle-plate-recognition-for-ethiopian-license-plate-based-sift-feature
[14] Jones V.; “A survey on identification of vehicle number plate using". 5(1): 2–4, 2019.
[15] Usmankhujaev S.; Lee S. and Kwon J.; “Korean license plate recognition system using combined neural networks". Adv. Intell. Syst. Comput.; 1003(July 2019): 10–17, 2020, doi: 10.1007/978-3-030-23887-2_2.
[16] Chen R.; “Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning ઙ". Image Vis. Comput.; 87: 47–56, 2019, doi: 10.1016/j.imavis.2019.04.007.
[17] Gnanaprakash V.; Kanthimathi N. and Saranya N.; “Automatic number plate recognition using deep learning". IOP Conf. Ser. Mater. Sci. Eng.; 1084(1): 012027, 2021, doi: 10.1088/1757-899x/1084/1/012027.
[18] Sonavane K.; “Survey on Automatic Number Plate Recognition (ANR)". 125(6): 6–9, 2015.
[19] Saadouli G.; Elburdani M. I.; Al-Qatouni R. M.; Kunhoth S. and Al-Maadeed S.; “Automatic and Secure Electronic Gate System Using Fusion of License Plate, Car Make Recognition and Face Detection". 2020 IEEE Int. Conf. Informatics, IoT, Enabling Technol. ICIoT 2020: 79–84, 2020, doi: 10.1109/ICIoT48696.2020.9089615.
[20] Farhat A.; Hommos O.; Al-zawqari A.; Al-qahtani A.; Bensaali F. and Amira A.; “Optical character recognition on heterogeneous SoC for HD automatic number plate recognition system". 2018.
[21] Rashtehroudi A. R.; “Iranian License Plate Recognition using Deep Learning”.
[22] Omran S. S.; “Iraqi Car License Plate Recognition Using OCR". March, 2017, doi: 10.24086/cocos17.19.
[23] Altameemi A. A. J.; Abbas H. H. and Alabaichi A.; “Iraqi vehicle license plate recognition using template matching technique". ICIC Express Lett. Part B Appl.; 10(6): 465–473, 2019, doi: 10.24507/icicelb.10.06.465.
[24] Abbass G. Y. and Marhoon A. F.; “Car license plate segmentation and recognition system based on deep learning". 11(4): 1983–1989, 2022, doi: 10.11591/eei.v11i4.3434.
[25] Jirasuwankul N.; “Effect of text orientation to OCR error and anti-skew of text using projective transform technique". IEEE/ASME Int. Conf. Adv. Intell. Mechatronics, AIM, July 2011: 856–861, 2011, doi: 10.1109/AIM.2011.6027057.
[26] Redzuwan R.; Radzi N. A. M.; Din N. M. and Mustafa I. S.; “Affine versus projective transformation for SIFT and RANSAC image matching methods". IEEE 2015 Int. Conf. Signal Image Process. Appl. ICSIPA 2015 - Proc.: 447–451, 2016, doi: 10.1109/ICSIPA.2015.7412233.
[27] F. F. Recognition, O. Detection, P. Recognition, U. Python, and H. Singh, Practical Machine Learning and Image Processing.
[28] Gavaskar R. G. and Chaudhury K. N.; “Fast Adaptive Bilateral Filtering". IEEE Trans. Image Process.; 28(2): 779–790, 2019, doi: 10.1109/TIP.2018.2871597.
[29] G. R. Liu, Machine Learning with Python. 2022. doi: 10.1142/12774.
[30] Jeppesen J. H.; Jacobsen R. H.; Inceoglu F. and Toftegaard T. S.; “A cloud detection algorithm for satellite imagery based on deep learning". Remote Sens. Environ.; 229(August): 247–259, 2019, doi: 10.1016/j.rse.2019.03.039.
[31] Chicco D. and Jurman G.; “The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation". BMC Genomics, 21(1): 1–13, 2020, doi: 10.1186/s12864-019-6413-7.