International Conference on Recent Trends in Data Science and its Applications
DOI: rp-9788770040723.026

Automatic Number Plate Detection Using Deep Learning Techniques

R.Athilakshmi
Department ofComputational Intelligence, SRM Institute of Science and Technology,
Chennai, India

RaghavKhatoria
Department of Computational Intelligence, SRM Institute of Science andTechnology,
Chennai, India

SubhasriSrinivasan
Department ofComputational Intelligence, SRM Institute of Science and Technology,
Chennai,India

***Abstract*—Automatic Number Plate Recognition (ANPR) is a technique that reads a vehicle's license plate using photographs given to the Deep Learning model as its input. Existing ways to find and identify license plates on cars are severely limited by the low resolution and significant loss of edge data in plate photos. The process of automatic identification of license plates necessitates a high level of accuracy when cars are moving rapidly, and number plate abstraction is a difficult task due to number arrangement and environmental factors. The primary objective of this work is to create an Automatic Number Plate Detector that aims to recognize the text present in the number plates of cars, vans, bikes, trucks, and other vehicles that are either stationary or in motion with good accuracy. The recognition model was created by training it on a large dataset. The deep learning models Faster-RCNN and Yolo were employed to fetch the Region of Interest, followed by OCR for character extraction.**

***Keywords*— YOLOv5, RoI, Automatic Number Plate Recognition, Faster RCNN, OCR**

I. Introduction

Automatic Number Plate Recognition (ANPR) uses Deep Learning models to read license plates automatically and precisely. For various number plate recognition applications, multiple levels of image analysis are needed, such as recognising the types of objects in the image, locating these objects, and determining the precise boundaries of each object. Number plate recognition systems are frequently used to manage traffic, identify stolen automobiles, regulate access to buildings and parking lots, collect tolls automatically, and determine which advertisements are most effective. This ANPR system is supported by an abundance of documentation and general public knowledge. Nevertheless, there are constraints to consider, such as specific detectors or viewing angles, adequate illumination requirements, recording in a specific zone, and specific vehicle types. In this scenario, Deep Learning (DL) techniques emerged as a useful parameter.

Faster RCNN is superior to both RCNN and fast RCNN. In this approach, a convolutional neural network receives an image as input and produces a convolutional feature map. Then, a different network is used to predictthe region proposals. Following that, the suggested regions are reshaped using a RoI pooling layer. This layer classifies the picture within the region and forecasts theoffset values for the bounding boxes. OCR stands for "Optical Character Recognition." It was first identified as a computer vision problem. A simple OCR engine functions by storingnumerous font and text image templates. The OCR then uses pattern-matching algorithms to compare text images to its internal database, character by character. The proposed work involved working with a manually picked dataset that contains both Indian number plates. The vehicles used here include cars, trucks, bikes, army vehicles, autos, and taxis. The environmental conditions of these photos are either blurred, dusty, clear, or in motion. In this study, the Automatic Number Plate Detection System is divided into 2 processes, namely

i) Number Plate Detection

This is the first phase in which the Region of interest is selected. A bounding box is created to determine the exact coordinates that surround the vehicles' license plates. The construction of this detector will be carried out using the YOLO [1] transfer learning model and Faster RCNN [2],[3]which will be trained on a dataset consisting of a diverse variety of vehicle images, like normal vehicles, taxis, army vehicles, bikes, etc.

ii) Digit Number Recognition

This is the second phase of the model. After the bounding boxes and the coordinates are saved, the digits and characters are extracted from them with the highest accuracy possible. The images will be cropped to the size of the coordinates received in the first phase. This new image is fed to an OCR for digit recognition.

II.Literature Survey

The main task of Automatic License plate Recognition (ALPR) is to localize and recognize license plates in vehicle images. The conventional ALPR has two components, i.e., License Plate Detection (LPD) and License Plate Recognition (LPR). To help readers comprehend the work done in ALPR systems, thefollowing section highlights the reviews of various techniques used for LP detection and LP Recognition in literature. LP detection can typically be divided into two groups: one-stage and two-stage LP detection. Fast R-CNN [4] and Faster RCNN [5], which produce substantial region recommendations in the first stage and enhance them in the second, are the basic works of the two-stage detection networks. A unified framework called TE2E (Towards End to End) in which the model used Region Proposal Network (RPN) for License Plate(LP) detection and Recurrent Neural Networks(RNN) for character recognition. Here, the TE2E framework adopts

DOI: rp-9788770040723.026

two-stage detection involves the RoI pooling layer and high-quality bounding box proposals method for LP detection and RNN for character recognition. However, their findings revealed that RNNs were ineffective for character recognition [6]. To increase effectiveness, Xu et al developed a CNN-based architecture called Roadside Parking Network (RPnet). However, as the character's finer characteristics are lost, small-size images have a negative impact on recognition performance [7]. Another approach involved image Processing Techniques for plate detection and the OCR algorithm for text extraction. They got a success rate of around 85% in the extraction of the number plates and 80% in OCR. In the next work, the authors utilized a Faster-RCNN model for number plate detection. In that, the contrast operation was performed using Top-hat and Black- hat morphological transformations. They also performed character segmentation and recognition and utilized different optimization algorithms for recognizing the number plate [8].

Next, A region proposal network (RPN) was developed by Ren et al. to provide almost cost-free region proposals by sharing full-image convolutional features with the detection network. Fully convolutional networks that predict object limits and object scores at every place are known as RPNs. The RPN is trained from start to finish to provide high quality region suggestions, which Fast R-CNN uses for detection. They also combined Fast R-CNN and RPN into a single network by sharing their convolutional features—using the pretty popular concept of neural networks with 'attention' methods, the RPN component instructs the unified network on where to look [5]. Another research involved YOLOv3 for plate detection and OCR for image enhancement and recognition. However, their method achieved an accuracy of nearly 90% for number plate detection and 91.5% for number plate recognition [9].

In the next research, the authors combined YOLOv4 and contour methods for detecting and recognizing the number plates. However, the results achieved by the contour model are comparatively low when compared to existing methods. In another research, the authors compared two methodologies - YOLO and Traditional Image Processing for finding a suitable method for number plate detection. The accuracy, recall, and precision for the traditional image processing technique were found to be 72%, 0.92, 0.92 whereas for YOLO it was 90%, 0.91, and 0.92 respectively [10]

Some of the limitations observed in these papers are:

- Most of the images are of cars and not any other vehicle.
- Only vehicles with non-transporting number plates are used.
- Models which were used are outdated.
- Characters are detected wrongly such as 's' as 8 and 'o' as 0. The rest of this paper is organized as follows: Section III describes the proposed architecture while Section IV elaborates on the models used and Section V includes the discussion of obtained results. Section VI concludes the paper with possible scope for further investigations.

III. System Architecture

The initial stage of the project involved image collection and dataset preparation using a customized dataset. The data collection includes 1160 images that were captured while a car was moving through metropolitan traffic. These images were extracted from 50 distinct videos, each with a frame rate of 40 frames per second (FPS) and a duration of one second (1S). Additionally, the UFPR-ALPR dataset [11] was utilized, which consists of 4,500 completely annotated pictures taken by 150 vehicles in real-world situations. The model was trained with a custom dataset that included 5660 images drawn from both the collected images and the UFPR-ALPR dataset. The annotations in the datasets were converted into different formats depending on the deep learning models used. This dataset is then used to train the model and detect the number plate, given an image of the vehicle at different angles. This research utilized Faster RCNN and You Only Look Once (YOLO) models for detecting the number plate. Although both algorithms are reasonably good, YOLO surpasses Faster R-CNN in terms of accuracy, speed, and efficiency. YOLO is an end-to-end object detection model which predicts the bounding box and probability of each class and classifies the entire image at once. Finally, the coordinates of the number plate will be predicted. Now, this number plate will be extracted from the given image with the help of the coordinates predicted and a new dataset will be created that contains only the digits of the number plate, which will then be fed to the OCR for reading its digits.

1. *Image acquisition and Dataset preparation*

This step involves preparing a dataset composed of 1160 images of different types of vehicles, like cars, bikes, taxis, and army vehicles, with Indian license plate numbers and foreign number plates. To train the dataset on the YOLOv5 and Faster RCNN models, annotations were required for each image. These annotations provided the coordinates of the bounding box that the model needed to detect from an image. The LabelImg tool was used for this purpose, which is a graphical image annotation tool developed using Python and utilizing Qt for its graphical interface (version 0.92). The proposed framework is shown in Figure 1.

In this research, a new dataset was developed that includespictures of cars, heavy vehicles, and buses. A novel framework was proposed that combined YOLOv5 with a modified OCR model which can be trained end-to-end for number plate detection and recognition. Additionally, a comparison was made between the proposed model and a state-of-the-art model in the literature for the detection and recognition of number plates.

2. *Number Plate Detection*

YOLO (You Only Look Once) is a single-shot detection method that detects objects in an image in one go, rather than having to scan the image multiple times or in a sliding window fashion [12]. This makes it much faster than other object detection algorithms, although it may not be as accurate.

DOI: rp-9788770040723.026

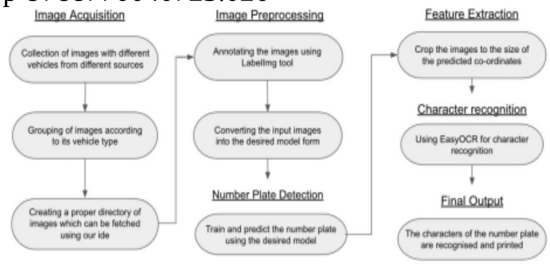

Fig1. System architecture

YOLO divides the input image into a grid of cells and predicts the presence of object-bounding boxes and class probabilities for each cell. If the center of an object falls into a cell, that cell is responsible for detecting that object. YOLO also uses anchor boxes, which are predefined bounding boxes of different aspect ratios, to improve detection accuracy. There have been several versions of YOLO released, with the latest being YOLOv5. YOLO has gained popularity due to its fast detection speed and good performance on a variety of object detection tasks.

3. Optical Character Recognition

OCR (Optical Character Recognition) is a technique that translates printed text into a digital version. The wide variety of typefaces and styles used to write a character in the printed image makes it difficult to recognize the letters. The image must be pre-processed before an OCR algorithm can be selected. The text is straightened, despeckled, and transformed from colour to binary image - an image with just two colours, black and white - in this stage. Following pre processing, the feature detection algorithm detects a character by studying the image's lines and strokes.

The recognition method then examines the character in its entirety and recognizes a textual line by searching for rows of white pixels separated by rows of black pixels. It can also determine where an individual character begins and finishes. It then transforms the character's picture into a binary matrix, where white pixels represent 0s and black pixels represent 1. It, using the distance formula, calculates the distance from the matrix's center to the farthest 1, and then draws a circle with that radius and divides it into moregranular pieces. Figure 2 shows the image character tobinary conversion.

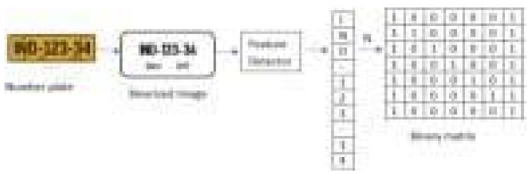

Fig 2. Image character to binary matrix conversion

At this stage, the OCR system will compare each subpart to a database that consists of matrices where each matrix represents a character in various typefaces to determine which character it has the most in common with statistically. Doing this for each line and character makes it simple to convert printed media to digital. Character recognition using feature extraction is shown in Figure 3.

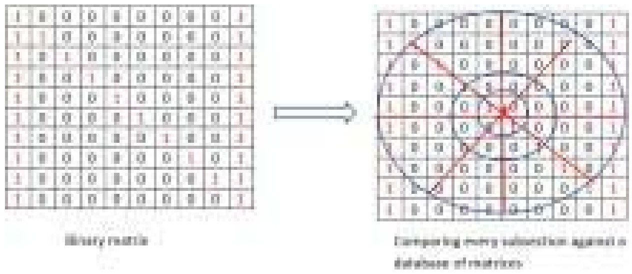

Fig 3. Character recognition using Feature extraction

IV. Models Used

A. Faster RCNN

Early in the 2010s, Ross Girshick and colleagues atMicrosoft Research created the object detection method known as R-CNN (Regions with Convolutional Neural Networks). R-CNN creates a set of features by taking an entire image as input and processing it through a CNN. The image is then categorised into one of several predetermined classes using these features. R-CNN uses a sliding window method, where it moves a window over the image and categorises each window, to find objects in the image. Given that running the CNN numerous times is required, this can be computationally expensive. [13].

R-CNN has been improved, and Fast R-CNN accelerates the object detection procedure. In order to create a set of features, a CNN is used to process a whole image as input. The image is then classified into one of a number of predetermined classes, and the bounding boxes for image objects are predicted using these attributes. Fast R-CNN avoids executing the CNN more than once by using a method known as region of interest (RoI) pooling, in contrast to R-CNN, which employs a sliding window approach. [4].

Faster R-CNN is a type of object detection algorithm and is an even faster version of Fast R-CNN. Feature Network, Region Proposal Network (RPN), and Detection Network are the three neural networks that make up Faster-RCNN. The Feature Network is an image categorization network that has already been trained, similar to VGG but lacking a few top/last layers. The form and structure of the original image are preserved in this network's output. RPNs are typically basic networks with three convolutional layers. Two layers—one is used for classification and the other one is used for bounding box regression—are fed from a common layer. As a result, a few bounding boxes known as ROIs are created. The RPN and Feature Network provide input to the Detection Network, which then uses that information to create a bounding box and the final class. It usually has four dense or completely linked layers. Two layers are stacked on top of one another that are common to both a classification layer and a bounding box regression layer. To make it easier to categorise only the contents of certain boundary boxes, the features are trimmed in line with those boxes. [5].

Faster R-CNN has a number of advantages over other object detection algorithms, but its speed in both training

DOI: rp-9788770040723.026

and inference is by far its biggest benefit. It is also accurate, making it a popular choice for object detection tasks. The prediction of the Number Plate by the Faster RCNN model is shown below in Figure 4.

Fig 4. Plate detection using Faster RCNN

B. YOLO (You Only Look Once)

YOLO primarily uses Computer Vision for object recognition and picture categorization and it divides animage into an NxN grid and extracts k bounding boxes from each grid. The network generates an offset value and class probability for each of the bounding boxes [12]. The bounding boxes are chosen and used to locate the object inside the image if their class probability is higher than a predetermined threshold value as shown in Figure 5. YOLO outperforms other object identification algorithms by orders of magnitude (45 frames per second). The YOLO algorithm's drawback is that it struggles to pick up minute details in the image. For instance, due to spatial restrictions, it might struggle to recognize a flock of birds.

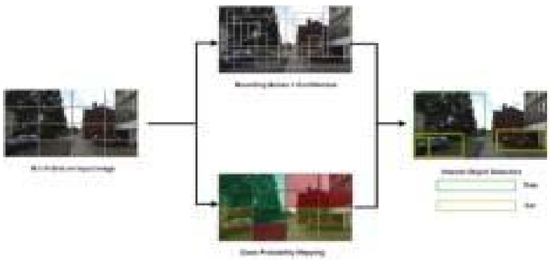

Fig 5. YOLO architecture

There are many versions of YOLO available, out of which YOLOv4 was considered the best real-time object detection algorithm on the standard MS COCO dataset as per the MAP benchmark. On the other hand, YOLOv5 which was introduced in 2020 has high speed, low volume, and high precision. In this research, YOLOv5 will be used to ensure high detection speed and accuracy. The backbone, detection head, and feature pyramid network are the three fundamental structures of the YOLOv5 network model. The feature pyramid network then gathers features from multiple scales and transfers them to the detection network after the backbone network extracts features from numerous images at various sizes. The object bounding box is the result of the detection network using the image features to forecast the item category in the image. [14]. Both YOLO and Faster RCNN require the annotations to be in a specific format so that they can be used as input. For YOLO, the bounding box coordinates of each image have to be in the format: class, x center, y center, width, and height. Each image has its corresponding annotations stored in a text file with the same name as the image file. For faster RCNN, the annotations for all images have to be stored in a text file, with each row containing: file path,x1,y1,x2,y2, and class name, which can be created by using the annotations obtained from LabelImg for YOLO. The prediction of the Number Plate by the YOLOv5 model is shown below in Figure 6.

Fig 6. Plate detection using Faster YOLOv5

C. OCR

OCR stands for Optical character recognition. It is a method that encodes the text that is present in visuals so that a machine can read it. For this purpose, Easy OCR was used. Easy OCR uses CRNN for character recognition. Feature extraction using Resnet, sequence labelling using LSTM, and decoding using CTC make up its three primary parts. The OCR analysis transforms the input, which is a digital image of printed or handwritten text, into a machine-readable digital text format. The digitised image is then broken up into smaller pieces by OCR, which performs analysis to look for text, words, or character blocks. After being further divided into components, these blocks of characters are then contrasted with a Character Dictionary. Figure 7 and 8 show the input image given and the output image detected by the YOLO model. Figure 9 shows the output extracted by the OCR model.

Fig 7. Input image given to Number Plate detector

Fig 8. Output from the Number Plate detector

Fig 9. The result given by OCR

V. Experimental Results

A comparison was made between the transfer learning methods used in this work and other deep learning methods such as Morphological methods [15], Projection Methods

DOI: rp-9788770040723.026

[16], and Feature salience methods [17]. As shown in Table 1, it can be concluded that not all the transfer learning methods are suitable for this type of problem, as only Yolo is performing well in terms of computing speed with a computing speed of 0.25s and an accuracy of 95%. On the other hand, Faster RCNN did not perform well in terms of accuracy but was still faster than the methods used in [15][16] and [17]. Table 1 shows the comparison of differentnumber plate detection algorithms with existing methods in the literature.

TABLE 1. COMPARISON OF DIFFERENT NUMBER PLATE DETECTION ALGORITHMS AND EXISTING METHODS IN THE LITERATURE

Algorithm	Accuracy	Computation time(in sec)
Morphological methods [15]	90%	0.60s
Projection methods[16]	89%	0.90s
Feature salience method [17]	93%	0.50s
Faster RCNN	86%	0.70s
YOLOv5	**95%**	**0.25s**

TABLE 2.COMPARISON OF DIFFERENT NUMBER PLATE RECOGNITION ALGORITHMS WITH EXISTING METHODS IN THE LITERATURE

Algorithm	Computation time(in sec)	Recognition rate(%)
i-novel [18]	471s	73.1%
Template matching [19]	1000s	93-94%
ANN using featureextraction [20]	75s	92.2%
YOLOv5+OCR	**75s**	**97%**

As shown in Table 2, the model, which is a combination of YOLO and Easy OCR, outperforms the other methods in terms of both computation time and recognition rate. Specifically, it has a computation time of 75s and a recognition rate of 97%, as compared to other methods such as those in references [18], [19], and [20]. Hence, from the above comparison of results, it can be concluded that the best accuracy for number plate detection was achieved using YOLOv5. By applying this model, the license plate of the vehicle can be localized to obtain its coordinates. Then, feature extraction is used to crop the image to the size of the coordinates of the number plate obtained from the earlier prediction so that it can be used as an input to the OCR. For number plate recognition EasyOCR is used which extracts the characters into a string format.

VI. CONCLUSION AND FUTURE WORK

The proposed research utilized YOLOv5 and OCR, to jointly train a network for number plate detection and recognition respectively. The proposed methodology outperformed a few other strategies that have already been published in the literature in terms of computation time and recognition rate. This framework can also be applied to smart traffic, which identifies license plates of vehicles that violate traffic laws, or to optimized parking, which keeps track of the vehicles parked. The advantage of the proposed approach is that it handles all types of license plates with different sizes and shapes. Although the proposed algorithm is efficient in detecting different types of Indian license plates, we also believe that the same model should be applied to other country vehicle databases to detect and extract the number plate effectively.

REFERENCES

[1] W. Riaz, A. Azeem, G. Chenqiang, Z. Yuxi, Saifullah and W. Khalid, "YOLO Based Recognition Method for Automatic License Plate Recognition," 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications(AEECA), Dalian, China, pp. 87-90, 2020,doi: 10.1109/AEECA49918.2020.9213506.

[2] P. Ravirathinam and A. Patawari, "Automatic License Plate Recognition for Indian Roads Using Faster RCNN," 2019 11th International Conference on Advanced Computing (ICoAC), Chennai, India, pp. 275-281, 2019, doi: 10.1109/ICoAC48765.2019.246853.

[3] N.Ap, T.Vigneshwaran, M.Arappradhan,and R. Madhanraj, “Automatic Number Plate Detection in Vehicles using Faster R-CNN,” 2020 International Conference on System, Computation, Automation and Networking (ICSCAN), pp. 1-6, 2020.

[4] R. Girshick, "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, pp. 1440-1448, 2015,doi: 10.1109/ICCV.2015.169.

[5] S.Ren, K. He, R. Girshick,and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, pp. 1137-1149, 2017.

[6] H. Li, P. Wang and C. Shen, "Toward End-to-End Car License Plate Detection and Recognition With Deep Neural Networks," in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 3, pp. 1126-1136, March 2019, doi: 10.1109/TITS.2018.2847291.

[7] ZhenboXu, Wei Yang, AjinMeng, Nanxue Lu, Huan Huang, Changchun Ying, and Liusheng Huang, “Towards End-to-End License Plate Detection and Recognition: A Large Dataset and Baseline”. In Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII. Springer-Verlag, Berlin, Heidelberg, pp. 261–277, 2018, https://doi.org/10.1007/978-3-030-01261-8_16

[8] H.Li,and C. Shen, “Reading Car License Plates Using Deep Convolutional Neural Networks and LSTMs,”2016,ArXiv, abs/1601.05610.

[9] B. Srilekha, K. V. D. Kiran and V. V. P. Padyala, "Detection of License Plate Numbers and Identification of Non-Helmet Riders using Yolo v2 and OCR Method," 2022 International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, pp. 1539-1549, 2022,doi: 10.1109/ICEARS53579.2022.9751989.

[10] Gomathy, V., Janarthanan, K., Al-Turjman, F., Sitharthan, R., Rajesh, M., Vengatesan, K., &Reshma, T. P. (2021). Investigating the spread of coronavirus disease via edge-AI and air pollution correlation. ACM Transactions on Internet Technology, 21(4), 1-10.

[11] Laroca, Rayson,Severo, Evair,Zanlorensi, Luiz,Soares de Oliveira, Luiz,Gonçalves, Gabriel, Schwartz, William, Menotti, and David. (2018). A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector. 10.1109/IJCNN.2018.8489629.

[12] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 779-788, 2016,doi: 10.1109/CVPR.2016.91.

[13] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, pp. 580-587, 2014,doi: 10.1109/CVPR.2014.81.

[14] Rajesh, M., &Sitharthan, R. (2022). Introduction to the special section on cyber-physical system for autonomous process control in industry 5.0. Computers and Electrical Engineering, 104, 108481.

[15] Khan, K., Imran, A., and Rehman, H.Z.U. et al. “Performance enhancement method for multiple license plate recognition in challenging environments,” J Image Video Proc., p. 30, 2021, https://doi.org/10.1186/s13640-021-00572-4.

[16] P. Prabhakar, P. Anupama and S. R. Resmi, "Automatic vehicle number plate detection and recognition," 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kanyakumari, India, pp. 185-190, 2014, doi: 10.1109/ICCICCT.2014.6992954.

[17] Z. X. Chen, C.-Y. Liu, F.L. Chang and G. Y. Wang, "Automatic License-Plate Location andRecognition Based on Feature Salience,"

DOI: rp-9788770040723.026
in IEEE Transactions on Vehicular Technology, vol. 58, no. 7, pp. 3781-3785, Sept. 2009, doi: 10.1109/TVT.2009.2013139.

[18] S. Shastry, G. Gunasheela, T. Dutt, D. S. Vinay and S. R. Rupanagudi, "“i” — A novel algorithm for optical character recognition (OCR)," 2013 International Mutli Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), Kottayam, India, pp. 389-39 Re3, 2013,doi: 10.1109/iMac4s.2013.6526442.

[19] S. Goel and S. Dabas, "Vehicle registration plate recognition system using template matching," 2013 International Conference on Signal Processing and Communication (ICSC), Noida, India, pp. 315-318, 2013, doi: 10.1109/ICSPCom.2013.6719804.

[20] M. UsmanAkram, Z. Bashir, A. Tariq and S. A. Khan, "Geometric feature points based optical character recognition," 2013 IEEE Symposium on Industrial Electronics & Applications, Kuching, Malaysia, 2013, pp. 86-89, doi: 10.1109/ISIEA.2013.6738973.