IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

AUTOMATIC LICENSE PLATE DETECTION AND RECOGNITION USING DEEP LEARNING

[1]JAYALAKSHMI M, [2]HEMALATHA S, [3] LATHIK MOGER,

[4]HEMANTH KUMAR NP, [5]ARPITHA J C.

[1]Student, [2]Student, [3]Student, [4]Senior Assistant Professor, [4]Assistant Professor
[1,2,3,4]Department of Computer Science and Engineering
[1,2,3,4] Alva's Institute of Engineering and Technology, Mijar, India
[5]Department of Bachelor of Computer Aplications.
[5]KLE Degree College,Nagarabhavi, Bangaluru, India

Abstract: The range of vehicles has elevated dramatically over the past decades. As the number of vehicles on the road grows, it becomes more difficult for law enforcement and traffic officials to keep track of each one. Reasons of management Automatic Recognition of License Plates. ALPR (Automatic License Plate Recognition) is a common surveillance device that records vehicle license plates. It became a picture that captures and recognizes their license plate numbers. This is an essential research subject for this time period. The coco dataset was used in our research is used to recognize license plates. Continuing with the strategy of We used multitask learning to recognize character strings. YOLO-v3 changed into used for recognition, and CRNN changed into used for category and our proposed method's category. We divided the photos into three sets for evaluation: 40% for the training set, 20% for the validation set, and 40% for the test set. We chose a lower threshold for the test set evaluation, 0.125, which resulted in a 99.82 recall. Our proposed method has an 86 % recognition rate, with 88 % of three letter plates and 99 % of four letter plates being recognized. Finally, utilizing temporal redundancy, the very last identity fee is significantly enhanced, attaining 96%. Our method boosts reputation charges from 93.58 to 96.1 %, surpassing Sighthound and Open ALPR through 9% and 4.9%, respectively.

***Index Terms* – CNN Model, ReadNet, YOLOv3, OpenALPR, Object Detection.**

I. Introduction

Automatic ALPR technology is gaining traction in a variety of applications, including safety and traffic control [1]. Different levels of scene analysis are required for different license plate identification applications, including identifying the categories of items in the scene, locating them, and defining the exact boundaries of each object. These scene analysis capabilities relate to three basic computer vision research tasks: image classification, object detection, and semantic (occurrence) segmentation, to name a few. Access control in structures and parking lots, law enforcement, stolen vehicle identification, traffic management, automated toll collection, and advertising learning are all common uses for License Plate Recognition Systems. There are many successful industrial systems [2] available, and there is still a lot of documentation or general public information on the ALPR system that uses deep learning algorithms for plate recognition and localization. Although, as illustrated in Figure 1, there are certain limits to cope with, such as specific detectors or viewing angles, proper illumination requirements, capturing in a predetermined region, and specific types of vehicles (they wouldn't find lps from bikes, lorries, or buses). Deep Learning (DL) methods emerged as an effective parameter in the current sector in this case. Vehicle license plate identification methods are commonly grouped into three groups based on Template Matching, characteristics, and motion information, which are frequently utilized and developed by various foreign and local researchers. Since it is clear that the demand for License Plate Detection and Recognition has existed for many years, a few research have already been conducted in this area [3][4].

The most prevalent method utilized by researchers is visual object detection, which is used as a core functional module for scene analysis in ALPR applications, piquing researchers' interest in this area. Because of the kind of open deployment environments, computerized scene evaluation at the ALPR platform will become extraordinarily demanding, posing severalth new demanding situations to item detection jobs and algorithms. The following are the principle demanding situations: (1) the way to manipulate the numerous adjustments usually encountered with inside the visible factors of items in acquiring images (for example, light, vision, small size, and ratio); (2) the way to use ALPR structures with inadequate reminiscence and computing strength at the same time as going for walks detection algorithms; (3) the way to manipulate real-time necessities and detection accuracies.

Figure 1. Lps of different layouts and notice the extensive variety in a Different format on different lp layouts.

Vehicle photos in the COCO dataset [5] can be found in a variety of locations and distances from the camera. Furthermore, the car is not often fully visible in the photograph. There aren't byany publicly to be had datasets for to the exceptional of our knowledge. ALPR with notes on cars, bikes, lps, and different objects characters as proven in FIGURE 1. As a result, we can identify two major issues. In our batch of data to begin, automotive and bicycle lps are the most common. Have various aspect ratios, preventing ALPR approaches from working. This constraint may be used to remove fake positives. Automobiles and bicycles also are available. Lps are available a number of shapes and sizes. We selected fine-tuning for ALPR due to the fact YOLO-stimulated fashions yielded sizable profits in item detection [6]. Fast-YOLO, on the other hand, is significantly faster but less exact than YOLOv3 [7], which we used in our study work. We use temporal redundancy since we're processing with video frames, so each frame were separated and then combine the findings to produce a more complete picture. For every automobile, a dependable calculation is required. According to the proposed technique is the YOLOv3 deep mastering algorithm. Outperforms COCO public automobile photos dataset consequences for a gadget that acknowledges license plates automatically.

II. OBJECT DETECTION ALGORITHM BY YOLOV3

Due to advancements in computational power (i.e., GPU and deep learning chips) and the availability of large-scale samples (e.g., COCO [8]), deep learning is becoming more popular. The learning process of a neural network is quick, scalable, and end-to-end. As a result, it has been widely utilized. The CNN model, in particular, has greatly improved picture categorization when compared to commonly used shallow approaches (e.g., Object detection, ReadNet [9]), and ReadNet [9] (e.g., Faster R-CNN [10]) and semantic segmentation (Mask R-CNN [11]), among other things. This detection framework has sparked a lot of attention, and many advanced object detectors based on it have been developed in recent years, CNN has been suggested.

Furthermore, the YOLO [12] series model is a Convolutional Neural Network-based real-time object identification system (CNN). Keeping Google's community structure in mind. The YOLO community, on the opposite hand, is doing the identical issue for cross-channel facts integration in 1x1 layers and additionally a 3x3 convolution layer. The network structure of YOLO is made up of two fully linked layers and 24 convolutional layers. Furthermore, Ali Farhadi corrected and presented YOLO v3 [7], which increases object identification performance by detecting small objects and dense or complex objects overlapping tiny items that are possibly the most common. In practical applications, a deep object detector is presented. The speed and accuracy of detection are very well balanced. Further key improvements include:

- Loss: YOLOv2's softmax loss is replaced, while the predicted object in YOLO v3 has a logistic loss. The selection of a class is more difficult when there are more variables, logistic regression is more successful. In the data collection, there are a lot of labels that overlap.
- Anchor point: YOLOv3 has nine anchor points instead of the five in YOLOv2, which improves Intersection over Union (IoU).
- Detection: YOLOv2 employs only one detecting, but YOLOv3 uses three, significantly improving the detection effects of small objects.
- Backbone: In YOLOv3, the Darknet-19 network from YOLOv2 is replaced by a darknet-53 network, enhancing object detection accuracy by extending the network. The most recent technology is used in our paper. The YOLOv3 model was used to do the detection dataset from the COCO.

A. Multitask Learning

Multi-task learning [14] is a character string recognition approach developed for licence plates. This method bypasses the character segmentation step and detects the image's character string (in this case, the cropped LP characters). Because there could be a lot of them. Each character has the ability to do a task on the network.

B. Convolutional Recurrent Neural Network

CRNN is a scene text recognition version [15] that consists of convolutional layers followed via way of means of recurrent layers, in addition to a selected transcription layer designed to create a tag collection from the per-body predictions. This layer is responsible for the input for a sequence labelling problem predicting a tag distribution x = x1, x2.... xn for each tag a single vector of features y= y1,y2…yn.

The CNN models used for license plate character detection and recognition are explained in the following sections. It's worth mentioning that all parameters (e.g., CNNS input, etc.) Are subject to change. The number of epochs, among other things, is defined here based on the validation group and as shown in the part in which the results of the experiments are reported.

III. METHODOLOGY

In this paper, we present a framework for detecting and recognizing LPs in images. Our strategy focuses on locating and reading LPs in challenging contexts. There are three essential aspects to the method:

- LPs Detection
- Character Segmentation
- Character Recognition

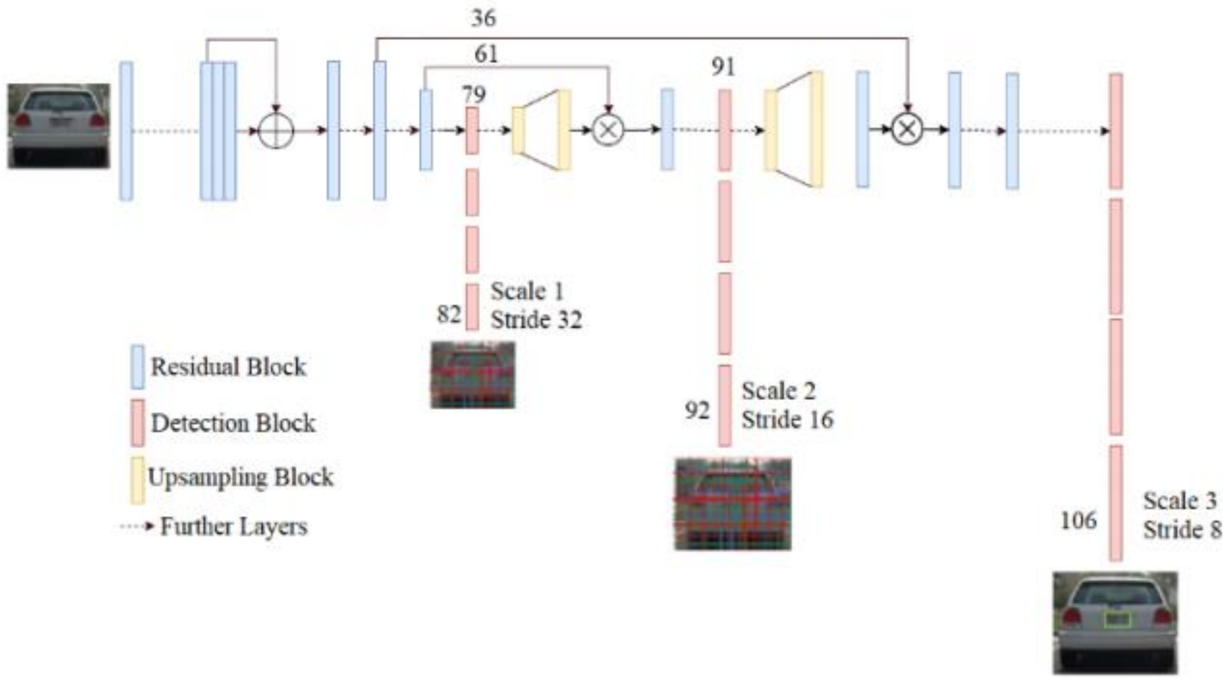

Figure 2: flow diagram of proposed alpr system

As demonstrated in FIGURE 2, we trained YOLOv3 to detect vehicles and LPs in our suggested technique. In a simple context, YOLOv3 should be able to detect LP, but in a more complex one, it may not. It might not be deep enough to perform this in complex scenarios. As a result, in order to use YOLOv3, we must make a change by considering the number of classes, the last number of layers. To forecast bounding boxes, confidence, and class probability, YOLO uses predicting boxes (A=9) and K class probability as given in Eq. (1).

$$\textbf{Filters} = (k+5)*A \qquad \text{-----} \quad (1)$$

Vehicle and LPs coordinates are used to train the CNN for vehicle LPs detection in order to detect LPs. In order to detect all of the automobiles in the validation set, we assessed that the confidence level with the lowest rate of erroneous positives.

Following LP detection, the character segmentation method (as given in TABLE I.) is trained using LP and character margins and coordinates. This margin is based on the validation set to, as explained in the previous stage. Verify that all of the characters correspond to the expected LP size. We also generate a training set to expand the training set. Every LP has a negative picture.

TABLE I. CHARACTER SEGMENTATION CNN ARCHITECTURE

Layer	Filters	Size
Conv	32	3x3
Max		
Conv	64	3x3
Max		
Conv	128	3x3/2
Conv	64	1x1
Conv	128	3x3
Max		
Conv	256	3x3/2
Conv	128	1x1
Conv	256	3x3
Conv	512	3x3/2
Conv	256	1x1
Conv	512	3x3
Conv	35	1x1
Detection		

We initially provide some padding (1-2 pixels) after LP detection and segmentation to enhance prediction because certain characters may now be properly segmented. Units are used to train the networks by passing segmented text with labels as input. Features collected using CNN are converted into feature vectors and then utilised as an input for the LTSM layer (as given in TABLE II.), assisting with the sequence layer problem and forecasting a label distribution.

TABLE II. CRNN LAYERS

Layer	Input	Size
Conv	64	3x3/1
Max		2x2/2
conv	128	3x3/1
Max		2x2/2
Conv	256	3x3/1
Conv	256	3x3/1
max		2x2/2
conv	512	3x3/1
batch		
conv	512	3x3/1
batch		
max		2x2/2x1
conv	512	
Layer	Input	Hidden
LTSM	512x1x40	256

IV. EXPERIMENTAL RESULTS

For training, we use the following parameters: 50k iterations, lr=[1-2,1-3,1-4,1-5] with steps of 10k, 20k, and 25k iterations.

A. Application Orientated LP (COCO) Evaluation

We split the pictures into three sets for evaluation: 40% for the training set, 20% for the validation set, and 40% for the test set.

(AC) (LE) (RP)

Figure 3. Detection results of lps detected by our method. Pointed detection of lps showing the robustness of algorithm even have different camera distance, illumination, angle, and several ambiguities.

- **Vehicle Detection**: To identify vehicles, we must first choose confidence levels. We are unable to recognise cars with 100% accuracy when confidence = 0.5 is used. In the validation dataset, all cars were correctly recognised with a confidence level of 0.25. As a result of this analysis, we set confidence for the test set at 0.125. We were able to get 100% recall with 99 % precision with this threshold, with only 5 false positives.
- **LP Detection:** Every vehicle with LPs was predicted within the bounding box in the validation set, as illustrated in FIGURE 3. As a result, vehicle patches were used to train the LP detection network. In both the validation and test sets, we achieved 100% recall and precision, which is an efficient consequence of our suggested technique.
- **Character Segmentation:** For the validation set assessment, we used the following confidence thresholds: 0.5, 0.25, and 0.125, all of which resulted in a 99.92% recall rate. So, to miss as few characters as possible, we set a lower threshold of 0.125, resulting in a 99.82 recall.
- **Character Recognition**: Padding values were introduced for character recognition 1 pixel for numbers and 2 pixels for characters in the validation set. We utilised data augmentation with a reversed character to obtain even better results. During the testing, we discovered that letter recognition may be enhanced by using augmentation and padding.

Our suggested technique achieved an 86% identification rate while ignoring temporal redundancy, identifying 88% of three letter plates and 99% of four letter plates. Taking use of temporal redundancy boosted results substantially. The final recognition rate, which was 96%, was considerably enhanced by using temporal redundancy. The recognition rate of our suggested technique increased from 93.58% to 96.1%(as given in TABLE III.). While exceeding Open ALPR and sighthound by 4.9% and 9%, respectively, the findings are demonstrated in the table below, demonstrating the suggested YOLOv3 model's high accuracy rate.

TABLE III. RECOGNITION RATES (%) WITH REDUNDANCY ACHIEVED BY PROPOSED SYSTEM INCLUDING PREVIOUS WORK ON ALOP DATASET.

ALPR	All Correct
Slighthound with redundancy	87.1
OpenALPR with redundancy	91.2
Proposed with redundancy	96.1

TABLE IV. TIME REQUIRED FOR NVIDIA 1080TI TO PROCESS THE ALGORITHM.

ALPR Stage	Time
Vehicle Detection	10.211ms
LP segmentation and Classification	2.31ms
Character Recognition	1.590ms
Total	14.111ms

Figure 4. The examples showing of accurately recognized LPs by our algorithm.

V. CONCLUSION

We suggested a real-time Automatic License Plate Recognition system employing YOLOv3 and CRNN in the proposal paper. In our modified network, the accuracy vs. speed trade-off was proved to be effective at every level. The suggested study deduces a unified technique for License Plate categorization and detection, which advances the findings utilising post processing criteria (redundancy). We overlooked mistakes in characters, those that were misclassified, and also in the amount of anticipated characters to be considered, according to the LP layout classes, which was necessary for achieving suitable results as shown in FIGURE 4. On the COCO datasets utilised in the studies, the average recognition rate is 96.1 %, surpassing Open ALPR and Sighthound by 4.9 % and 9%, respectively, demonstrating the efficacy of our proposed approach.

REFERENCES

[1]. 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, April 2016.

[2]. R. Laroca, E. Severo, L. A. Zanlorensi et al., "A robust realtime automatic license plate recognition based on the YOLO detector," in Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–10, Rio de Janeiro, Brazil, July 2018.

[3]. Salah Al-Shami, Ali El-Zaart, Rached Zantout, Ahmed Zekri and Khaled Almustafa. 2015 "A New Feature Extraction Method for Licence Plate Recognition", In 2015 Fifth International Conference on Digital Information and Communication Technology and Its Applications (DICTAP), Page(s) 64–69, IEEE 2015.

[4]. Nicolas Thome, Antoine Vacavant, Lionel Robinault, and Serge Miguet, "A cognitive and video-based approach for multinational License Plate Recognition", Machine Vision and Applications, Springer-Verlag, pp. 389-407, 2011.

[5]. Hsu, G.-S.; Chen, J.-C.; Chung, Y.-Z.; , "Application-Oriented License Plate Recognition," Vehicular Technology, IEEE Transactions on , vol.62, no.2, pp.552-561, Feb. 2013

[6]. Wu, F. Iandola, K. Keutzer, and P. H. Jin , "SqueezeDet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving," in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), July 2017, pp. 446–454

[7]. Redmon, J. and Farhadi, A. (2018) YOLO v3: An Incremental Improvement.

[8]. Lin, T.-Y. et al. Microsoft COCO: Common Objects in Context. in Computer Vision – ECCV 2014 (eds. Fleet, D., Pajdla, T., Schiele, B. & Tuytelaars, T.) 740–755 (Springer International Publishing, 2014).

[9]. He, K., Zhang, X., Ren, S. & Sun, J. Deep Residual Learning for Image Recognition. in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 770–778 (IEEE, 2016).

[10]. Ren, S., He, K., Girshick, R. & Sun, J. Faster R-CNN: Towards RealTime Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. 39, 1137–1149 (2017).

[11]. He, K., Gkioxari, G., Dollar, P. & Girshick, R. Mask R-CNN. in Proceedings of the IEEE international conference on computer vision 2961–2969 (2017).

[12]. Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016) You Only Look Once: Unified, Real-Time Object Detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 27-30 June 2016, 779-788.

[13]. Redmon, J. and Farhadi, A. (2017) YOLO9000: Better, Faster, Stronger. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 21-26 July 2017, 6517-6525G.

[14]. R. Gonc¸alves, M. A. Diniz, R. Laroca, et al., "Real-time automatic license plate recognition through deep multi-task networks," in 31th Conference on Graphics, Patterns and Images (SIBGRAPI), 110–117 (2018).

[15]. B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence 39, 2298–2304 (2017).