YOLO Based Recognition Method for Automatic License Plate Recognition

Waqar Riaz, Abdullah Azeem, Gao Chenqiang, Zhou Yuxi, Saifullah, Waqas Khalid
School of Communication and Information Engineering
Chongqing University of Posts and Telecommunications
waqar.riaz@hotmail.com

***Abstract*— Over the past two decades, the number of vehicles have increased intensely. With this increase, it is gradually more difficult to track each vehicle for law enforcement and traffic management purposes. Automatic License Plate Recognition (ALPR) is a popular surveillance system that captures vehicle images and identifies their license plate numbers and became an important research topic of this era. In our study, AOLP dataset is used for the license plate recognition. Keeping the strategy of multi task learning for character string recognition we employed YOLOv3 for the recognition and CRNN for classification for our proposed method. For evaluation, we allocated 40% images to the training set, 20% to the validation set and 40% to the test set. For the test set evaluation we choose lower threshold i.e. 0.125 achieving 99.82 recall. Our proposed method achieved 86% recognition rate with recognizing 88% three letter and 99% of four letter plates. In the end while using temporal redundancy, final recognition rate is significantly improved i.e. 96%. Our proposed method improves recognition rates from 93.58% to 96.1% having outperforming Sighthound and OpenALPR by 9% and 4.9%, respectively.**

Keywords—YOLOv3, CRNN, OpenALPR, Object Detection

I. Introduction

Automatic ALPR technology is constantly gaining popularity, particularly in safety and traffic management systems [1]. Different license plate detection applications require different levels of scene analysis, including identifying the types of objects in the scene, locating these objects, and determining the exact boundaries of each object. These scene analysis functions correspond to three basic research tasks in the field of computer vision, commonly known as image classification, object detection and semantic (occurrence) segmentation. License Plate Recognition Systems are used regularly for access control in structures and parking areas, law enforcement, stolen automobile detection, traffic management, automated toll collection and advertising learning. There are lots of successful industrial systems [2] available and are still much documentation or general public information regarding the ALPR system using deep learning algorithms used in plate localization and recognition. Although, there are certain constraints to deal with, such as particular detectors or viewing angles, appropriate lighting requirements, capturing at a predetermined region, and particular kinds of vehicles (that they wouldn't find LPs from vehicles like bikes, trucks or buses) shown in Figure 1. Within this situation, Deep Learning (DL) methods appeared as an efficient parameter in the current field.

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

The vehicle license plate detection algorithms, which are often used and proposed by several international and domestic researchers, are usually classified into three categories based on Template Matching, features, and motion information. Since we can see that the need of License Plate Detection and Recognition has not developed now but rather numerous years back, a few studies have already been done in this field[3][4].

Visual object detection is probably the most common one adopted by the researchers, and it is used as a basic functional module for scene analysis in ALPR applications hence, increasing the interest of researchers in this region. Due to the diversity of open deployment atmosphere, automatic scene analysis running on the ALPR platform becomes very demanding, which introduces many new difficulties to object detection tasks and algorithms. These challenges mainly include: (1) how to manage the various changes commonly encountered in the visual aspects of objects in attaining images (for example, light, vision, small size, and ratio) (2) how to use ALPR platforms with inadequate memory and computing power while running detection algorithm; (3) How to manage the real-time requirements and detection accuracies.

The AOLP dataset [5] have vehicle images in various places and distances from the camera. Moreover, in some instances, the car isn't completely visible in the picture. To the best of our knowledge, there aren't any public datasets for ALPR using annotations of automobiles, bikes, LPs, and characters. Thus, we could point out two dominant challenges in our dataset. To begin with, generally, vehicle and bike LPs have different aspect ratios, not permitting ALPR methods to use this constraint to filter false positives. Also, car and bike LPs have various designs and positions. As significant improvements in object detection were attained via YOLO-inspired models [6], we chose fine-tuning to get it for ALPR. On the other side, Consequently, Fast-YOLO is considerably quicker but less precise than YOLOv3 [7] adopted in our work. Since we're processing movie frames, we also employ temporal redundancy like we process every frame individually and then combine the results to make a more robust calculation for every automobile. Based on the YOLOv3 deep learning algorithm, the proposed method outperforms results on AOLP public vehicle images dataset for an Automatic License Plate Recognition system.

II. Object Detection Algorithm By YOLOv3

Driven by the progress in computing power (i.e., GPU and deep learning chips) and the accessibility of large-scale samples (e.g., COCO [8]), in accordance to the fact that deep neural network has a fast, scalable and end-to-end learning framework, so it has been extensively practiced. In particular, compared with commonly used shallow methods, the CNN model has significantly improved image classification (e.g., ResNet [9]), and object detection (e.g., Faster R-CNN [10]) and semantic segmentation (Mask R-CNN [11]), etc. This detection framework has provoked great interest as well as, in recent years, many advanced object detectors based on CNN have been proposed.

Moreover, the YOLO[12] series model is a real-time object detection system based on Convolutional Neural Network (CNN). Keeping the network architecture of Google Net, the YOLO network is doing the same only the difference in 1x1 layers for cross channel information integration and also convolution layer of 3x3. YOLO is based on 2 fully connected layers and 24 convolutional layers in network structure. Ali Farhadi [13] published YOLO v2, highlighting intensive enhancement in the speed and accuracy of the algorithm. Furthermore, Ali Farhadi rectified and proposed YOLO v3, which further improves the object detection performance [7], it detects small objects and dense or overlapping compact objects that may be the most popular deep object detector in practical applications, presenting the accuracy and speed of detection are very balanced. Further key improvements include:

- **Loss:** Softmax loss of YOLOv2 being substituted by logistic loss in YOLO v3. While the predicted object class is more complicated, especially selecting logistic regression is more effective when there are many overlapping labels in the data set.
- **Anchor point:** YOLOv3 uses 9 anchor points instead of 5 anchor points of YOLO v2, thus improving Intersection over Union (IoU).
- **Detection:** YOLOv2 uses only 1 while YOLOv3 uses 3 detections, by which detection effects of small objects get impressively improved.
- **Backbone:** In YOLOv3, the Darknet-19 network of YOLOv2 is switched with a darknet-53 network, thereby improving the accuracy of object detection by network deepening. Our paper uses the latest YOLOv3 model to perform the detection of the AOLP dataset.

A. Multitask Learning

Multi-Task learning [14] is just another strategy for character string recognition created for license plates. This process skips the character segmentation point and immediately recognizes the character string of an image (here, the cropped LP characters). As there could be numerous characters, each character is entitled as a task on the network.

B. Convolutional Recurrent Neural Network

CRNN is a version constructed for scene text recognition [15], which is made up of convolutional layers followed by recurrent layers, along with a custom transcription layer was made to convert the per-frame predictions to a tag sequence. This layer manages the input for a sequence labeling issue, forecasting a tag distribution $x = x1, x2, \ldots, xn$ for every single feature vector $y = y1, y2, \ldots, yn$ in the feature map.

In the subsequent sections, we explain the CNN models utilized for license plate character detection and recognition. It's worth noting that all parameters (e.g., CNNs input, number of epochs, amongst others) given here are defined depending on the validation group and presented in the further section, where the experiments have been reported.

III. Methodology

Here we cast a framework which detects and recognizes LPs in a given image. Our approach is designed to focus on finding and reading LPs in tough environments. This type of method can be divided into three main parts:

- LPs Detection
- Character Segmentation
- Character Recognition

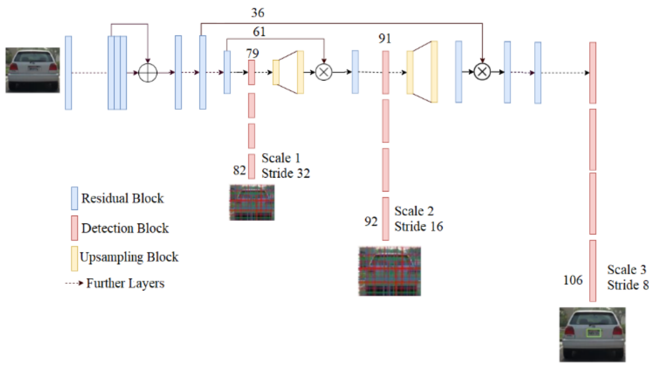

Figure 2 Flow diagram of the proposed ALPR system

In our proposed method, we trained YOLOv3 for vehicle and LPs detection, as shown in FIGURE 2. In a simple scenario, YOLOv3 should be able to detect LP, but in more complex scenes, it might not be deep enough to achieve this task. Therefore, in order to use YOLOv3, we need to change the last number of layers considering the number of classes. YOLO utilizes predicting boxes $(A=9)$ and K class probability as shown in Eq. (1) to predict bounding boxes, confidence, and class probability.

$$filters = (K + 5) * A \tag{1}$$

In order to detect LPs, vehicle and LPs coordinates are used to train the CNN for vehicle LPs detection. In the validation set, in order to detect all the vehicles, we evaluated the confidence threshold having the lowest false positive rate.

After LP detection, the character segmentation algorithm (architecture shown in TABLE I.) is trained using LP with margins and coordinates of characters. As described in the previous stage, this margin is based on the validation set to make sure that all the characters lie with predicted LP size. In order to increase the training set, we also generate a negative image of every LP.

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		

After LP detection and segmentation, we first introduce some padding (1-2 pixels) in order to improve prediction because some characters might now be well segmented. In order to train the networks, segmented characters with labels are passes Units as input. For recognition, we use the CRNN algorithm [15], which is designed for text recognition. Given the LP, containing the characters, features extracted using CNN is transformed into feature vectors and then used as an input for the LTSM layer, helping sequence layer problem and predicting 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

In this section, we conduct the experiments to verify the effectiveness of the proposed algorithm. We use NVidia 1080Ti to conduct these experiments. We use following parameters for training; 50k iterations, lr = $[1^{-2}, 1^{-3}, 1^{-4}, 1^{-5}]$ with steps 10k, 20k, 25k iterations.

A. Application Orientated LP (AOLP) Evaluation

The AOLP dataset [5] consists of 2049 images, which are divided into three categories, including 611 Road patrol images, 757 law enforcement, and 681 Access control (AC). These Images are captures in different weather conditions, different locations, and illumination conditions. For evaluation, we divided 40% images to the training set, 20% to the validation set, and 40% to test set.

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:** In order to perform vehicle detection, first, we evaluate confidence thresholds. Using confidence = 0.5, we are unable to detect vehicles with complete accuracy. With confidence 0.25, all vehicles were successfully detected in the validation dataset. So, based on this evaluation, we set confidence to 0.125 for the test set. Using this threshold, we accomplished 100% recall with 99% precision giving 5 false positives.
- **LP Detection:** In the validation set, every vehicle with LPs was predicted within the bounding box, as shown in FIGURE 3. Therefore, the LP detection network was trained based on vehicle patches. As expected, we achieved 100% recall and precision both in validation and test set, which itself an efficient result of our proposed approach.
- **Character Segmentation:** For validation set evaluation, we set the confidence threshold: 0.5, 0.25 and 0.125, achieving 99.92% recall regardless. So, in order to miss a few characters as possible, we chose a lower threshold i.e., 0.125 achieving 99.82 recall.
- **Character Recognition:** We introduced padding values: 1 pixel for digits and 2pixels for characters in the validation set. In order to achieve even better results, we used data augmentation with a flipped character. During the evaluation, we concluded that with augmentation and padding, letter recognition can be improved.

While ignoring temporal redundancy, our proposed method achieved 86% recognition rate with recognizing 88% three letter and 99% of four letter plates. Results were improved significantly while taking advantage of temporal redundancy. Using temporal redundancy, the final recognition rate was significantly improved, i.e., 96%. Our proposed method improved recognition rates from 93.58% to 96.1%. While, outperforming OpenALPR and slighthound by 4.9% and 9% and the results are proved in the given FIGURE 4 showing the high accuracy rate of proposed YOLOv3 model.

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

In the proposed document, we compiled a real-time Automatic License Plate Recognition system which is proposed using YOLOv3 and CRNN. At every stage, the accuracy and speed trade-off was proven efficient in our modified network. A unified methodology for License Plate classification and detection has been deduced in the proposed work, which advances the results using post processing rules (redundancy). Current approach was essential for obtaining appropriate results subsequently, accordingly to the LP layout classes, we ignored errors occurred in characters, those which were misclassified, and also in the number of predicted characters to be considered. The average recognition rate is 96.1% on AOLP datasets used in the experiments while outperforming OpenALPR and Sighthound by 4.9% and 9% respectively showing the effectiveness of our proposed system.

ACKNOWLEDGMENT

This work was supported by the Chongqing Research Program of Basic ResearcH AND FRONTIER TECHNOLOGY UNDER Grant cstc2018jcyjAX0227.

REFERENCES

[1] 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, 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] B. Wu, F. Iandola, P. H. Jin, and K. Keutzer, “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 Real-Time 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).