



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DESKTOP BASED APPLICATION FOR NUMBERPLATE AND LICENSE RECOGNITION USING DEEP LEARNING

¹Shah Wali Shaheen, ²Mr. Jitender Kumar

^{1st} Research Scholar, ^{2nd} Assistant Professor

Department of Computer Science & Engineering
Vivekananda Global University, Jaipur, India

Abstract: In this research, we look at detecting and recognizing car license plate's problem in natural scene photographs. In a single forward transfer, here We introduce a deep neural network that could simultaneously locate license number plates and identify letters. The entire network could be trained from start to end. Unlike current techniques, which consider license plate detection and identification as two separate tasks that must be solved one at a time, our system incorporates the two tasks into one network that can solve them both simultaneously. It not only prevents intermediate errors from building up, but it also speeds up the processing. Four data sets containing photographs taken from different scenes under various conditions are evaluated for performance evaluation. Extensive tests demonstrate the efficacy and effectiveness of our suggested strategy.

Index Terms - Car Plate Detection, deep neural network, license identification.

I. INTRODUCTION

A In intelligent transportation systems, car license plate identification is critical. It has a wide range of possible uses, from defense to traffic control, and it has piqued researchers' interest in recent years. However, the majority of current algorithms are only effective in managed environments or with high-tech image capture systems. In an unregulated setting, reading license plates correctly remains a difficult job. The challenge is in extremely complex backgrounds, such as text on shop posters, walls, or bricks, as well as the accidental photographing situations, for instance lighting, distortion, smearing. Plate detection and recognition have traditionally been treated as two separate tasks that are solved using different approaches in previous research on license identification of plate and its detection. However, Plate identification and detection are closely related activities. The accuracy of identification could be enhanced by applying precise bounding boxes taken from detection methods, and the outcome of the identification could be used to exclude false positives and vice versa. As a result, we suggest a single structure in this paper to address these 2 tasks at simultaneously. A neural network is built that receives images as input and outputs the positions of license number plates and number plate labels in a highly efficient and accurate manner. We show how low-level attributes can be useful for detection and identification. Without using any heuristic rules, the entire network can be trained from start to finish. The following is a diagram of the network flow diagram. Given in Figure 1. This is the first work that we are aware of that combines license detection of plate and identification into a trainable network that simultaneously addresses both problems. We have made following contributions.

- single unified deep neural network is introduced that could identify plate number and recognize labels in a given photo the same time. The method as a whole escapes intermediary procedures such as character grouping or separation, as well as probabilistic methods for instance plate colors and characters space.
- Secondly, as compared to using various models, the Both detection and identification use convolutional features, resulting in fewer parameters. Moreover, the extracted features would have more details if both identification and recognition losses were optimized together. Experiments show that using the jointly trained model improves both detection and recognition efficiency.
- The resulting system is more effective because plate recognition is integrated directly into the detection pipeline rather than being addressed by separate models. Here one does not have to cropped the identified license plates from the image of input and tag them with our device via a separate network. On a Titan X GPU, the entire procedure takes 0.36 secs for a 600 600-pixel input

image. It's important to keep in mind that, while a variety of techniques for detection of text and identification in natural scenes are proposed, none of them have been shown to work. e.g., [1]– [8]. Our approach differs significantly from theirs. The most notable distinction is that our model could even be trainable from beginning to end, whereas some techniques combine outputs from several models to arrive at concluding detection and identification outcomes. Any pre-processing, such as character detection or character grouping, can be removed with this invention, and It is possible to prevent intermediate errors. Learned characteristics can be more discriminative, resulting in a better match. Better performance. In the next section, we'll look into some related research.

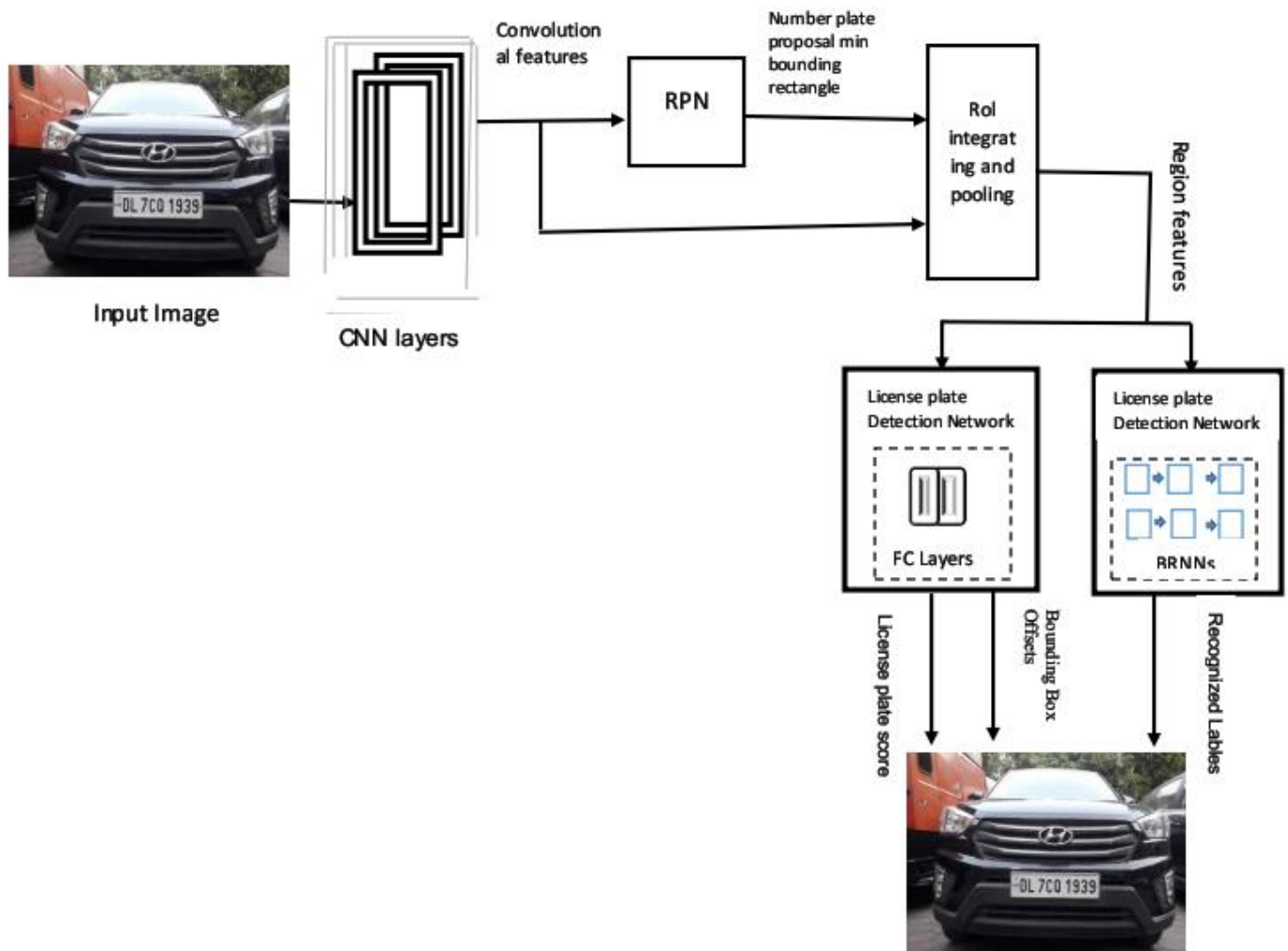


Figure 1 The final configuration of the proposed system. Some conv2D layers, a RPN for producing license plate proposals, a suggestion incorporating and pooling node, multi-layer perceptron's for detection of number plate and regression minimum bounding rectangle, and recurrent neural networks for recognition of number plate are all included in this system.

The remaining work of our research is laid out in the following manner. The second part contains a brief discussion of work that is connected. The integrated model is presented in Section 3 along with a detailed introduction to each component. In Section 4, experimental verifications are carried out, and lastly, conclusions are given

II. RELATED WORK

Since license number plate identification and recognition are commonly discussed in a separate manner, an overview of early research on this subject has been presented. Furthermore, we examine some other research on end-to-end scene text recognition and detection, which is relevant to this research.

License Plate Recognition: The aim of license plate character recognition is to use bounding boxes to locate license number plates in a picture. There are four types of existing methods. [9]– [11]: methods that are dependent on edges, colors, textures, and characters. Since license number plate are typically rectangular in frame with a particular ratio of aspect and have a superior density of edge than the rest of the picture, License plates are often detected using edge information. In [12] For plate detection, A method based on edges was built. Edge clustering was done using Expectation Maximization (EM), as nominee number plates, which removes regions with thick collections of edges and structures identical to plates. In [13], To bind regions with high density of edge and exclude sparse regions, in every column and row of a binary edge map, a new line density filter approach was suggested. Though edge-based methods are quick to calculate, they can't be used on complex images due to their sensitivity to unintended edges. The Color of the license number plate is normally dissimilar from the car's body color, so color-based methods are applied. In [14], By analyzing the target color pixels, a plate detection method was developed Strip scan was used to locate Iranian license plates using

a color-geometric design. Chang and his colleagues [15] centered on the different colors in RGB pictures, suggested a technique to make detection of Taiwan license number plates Colors for the foreground and history. They created a color edge sensor that detects the edges of black-white, red-white, and green-white. To detect inclined or deformed number plates, based on color procedures may be used.

However, In natural scene images, they are very vulnerable to varying lighting conditions., and they are unable to differentiate other artefacts in the picture that are the same color and size as the number plates. In texture-based methods to identify number plates, the unusual pixel intensity distribution in plate regions was used. Yu et al. [16] To get the horizontal and vertical information of a picture, I first used a wavelet transform. The projection data was then processed using Empirical Mode Decomposition (EMD) analysis to find the target wave crest, which specifies the place of a number plate. Giannoukos is a Greek author. et al. [17] SCW algorithm was designed to recognize numberplates in images centered on the plate texture's local irregularity property To speed up identification, Operator Context Scanning (OCS) was suggested. Texture-based methods have a higher computational complexity than edge or color-based methods because they use more distinguishing features. Given that license plates are made up of a string of letters, most recent research has focused on the letter-based function, which contains more detailed information. Zhou et al. [10] implemented the challenge of detecting numberplates as a visual clustering problem. For plate extraction, each character was given a Principal Visual Word (PVW), which includes geometric clues such as direction, height, and relative position. Li et al. [18] At the primary step, the MSER was used to extract candidate characters from images. The relationship between license plate characters was then represented using a Conditional Random Field (CRF). The belief propagation inference on CRF was eventually used to locate license plates. Llorca et al. [19] To detect isolate character regions, The MSER and the Stroke Width Transform (SWT) were used together. Finally, the probabilistic Hough transform was used to border the license plates. Character-based strategies are more dependable and have a higher recall rate. However, it is Other text in the image context has a significant impact on results. Preceding research on license number plate detection has usually segmented characters in the license number plate first, then used Optical Character Recognition (OCR) techniques to identify each segmented character. Extremal Regions (ER) were used, for example, in [20] to do segmentation of letters from coarsely sensed license number plates and improve number plate position. To identify the characters, restricted Boltzmann machines were used. In [12], For character segmentation, the Maximally Stable Extremal Regions method was used. For character recognition, by using LDA classifier extraction and categorization of local binary pattern is done. Hou et al. [21] Using Stroke Width Transform, which with approach of processing rotating licence plates, it was proposed to segment license plate characters. However, Character segmentation is a difficult process in and of itself, and it is vulnerable to being affected by image noise, shadows, and uneven lighting. It immediately affects plate recognition. Even if we have a powerful recognizer, the plate will not be recognized correctly if the segmentation is incorrect. With the advancement of deep neural networks, methods for recognizing the entire license plate without character separation have been suggested. Goodfellow et al. [22] suggested training a prediction model to read arbitrary multi-digit numbers using a broad scale distributed deep neural network. without individual character localization or segmentation. In [23], Hidden Markov Models (HMMs) were used to combine segmentation and optical character recognition, with the Viterbi algorithm determining the most likely label series. In [24], Plate recognition was once thought to be a sequence labelling problem. Convolutional Neural Networks (CNNs) were used to extract a sequence of feature vectors from the license plate bounding box using a sliding window method. To mark the sequential data without character separation, Recurrent Neural Networks (RNNs) with Connectionist Temporal Classification (CTC) [25] were used. End-to-end Scene Text Detection and Recognition System: For general text detection and recognition in natural scene images, a number of methods have been suggested. A heuristic approach to character grouping was used. The characteriness scores were combined with a dictionary to achieve word translation and recognition. Wang et al. [3] introduced a word detection technique in a given lexicon Using random ferns classifiers, a multi-scale sliding window-based approach was used to first localise character candidates. The Pictorial Structures (PS) formulation was used to find the best arrangement of a word from a given lexicon, which takes the positions and scores of sensed symbols as input and outputs all word bounding boxes and word labels at the same time. Matas and Neumann [26] suggested a quick algorithm based on character classification scores, character intervals, and language for selecting the best character sequence from a directed graph. Antecedents. based on character classification scores, character intervals, and language, proposed a quick algorithm for selecting the best character sequence from a directed graph.

preconditions The outputs are the sequence of regions and The best course caused their labels to appear. A separate character detection phase is often required in these methods. PhotoOCR [5] To achieve a advanced text detection recall, we combined the outputs of three separate text detection approaches. Over-segmentation of text lines was used to distinguish candidate character regions, and recognition was achieved using a combination of character classifiers and language model likelihoods. Jaderberg *et al.* [7] For word area proposal, we used a random forest classifier and a combination of complementary proposal generation techniques, as well as a random forest algorithm for word/no-word classification. A deep CNN was trained on a massive dictionary of 90k words to achieve word recognition. Liao *et al.* [8] Word detection and recognition were done separately using "TextBoxes" and "CRNN," where "TextBoxes" It's a 28-layer completely convolutional neural network that's been learned to generate word-level bounding boxes from input photos., and "CRNN" is a A 28-layer completely convolutional network that has been trained to generate word-level bounding boxes out of images of input. For the image-based sequence recognition task, "CRNN" [27] is a mixture of convolutional neural network, recurrent neural network, and CTC loss in one NN. Two separately trained models are used in these methods to detect and recognize text. The system that is the most similar to ours is [28], which proposes an all the time trainable scene text localization and identification framework. The detection of text had to be cropped out of the picture and recognized by another CNN model in this process. the identification and recognition losses were optimized jointly. In the recognition network, the

features must be calculated again. In our architecture, on the other hand, both identification and recognition modules share low-level convolutional features, allowing us to minimize parameters and save time.

III. MODEL

Unlike the approaches discussed above, our approach uses a single deep neural network to solve both detection and recognition. Our model includes a variety of convolutional layers for extracting discriminative features from license plates, an area proposal network tailored specifically for car license plates, a Region of Interest (RoI) pooling layer, multi-layer perceptron's for plate detection and bounding box regression, and RNNs with CTC for plate recognition, as shown in Figure 1. Plate detection and recognition can be accomplished simultaneously with this architecture, using only one network and a single forward evaluation of the input image. Furthermore, the entire network is trained end-to-end, with both localization and recognition losses jointly configured, and its efficiency has improved. We provide a comprehensive overview of each part in the subsections that follow.

A. Model Architecture

- 1) **Low-Level Feature Extraction:** To do extraction low level feature of ConvNet, we apply the OxfordNet network [29], . OxfordNet has thirteen layers of 3x3 convolutions, accompanied by non-linearity in the form of Rectified Linear Unit, five layers of 2x2 sample-based discretization process(max-pooling), and completely linked layers

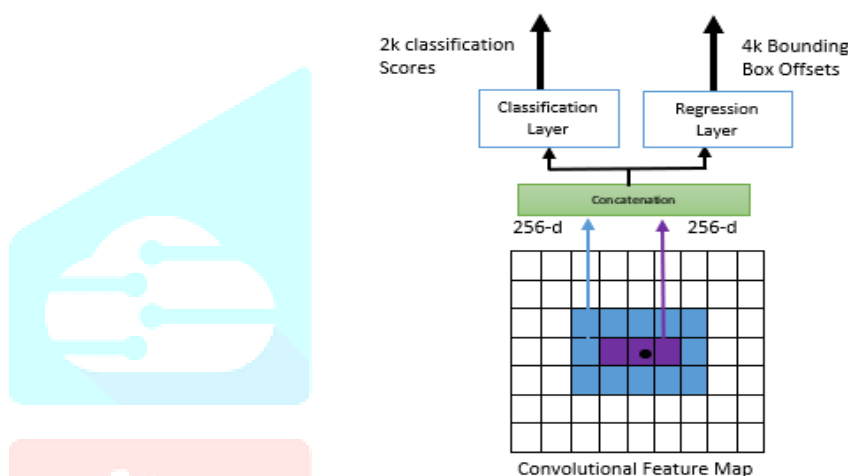


Figure 2 Making Plate Suggestions Two rectangular convolutional filters with rich contextual details are applied to each sliding window. The features are combined and fed into the plate/non-plate classification layer, as well as the regression layer, which calculates coordinate offsets in relation to k anchors at each point.

- 2) **Plate Proposal Generation:** Ren et al. [30] For object recognition, they created a RPN, the designed RPN produces nominee objects in photos. Region Proposal Network is a truly convolutional network that receives features which low-level convolutional as input and produces a series of possible output of bounding boxes. This could be prepared from beginning to end to produce proposals with higher-quality. We tweak Region Proposal Network in this paper to make it more appropriate for car licence plate proposals. We modelled 6 scales (heights 5, 8, 11, 15, 18, 22) having 5 ratios between height and width based on licence plates' scales and ratio of aspect in the datasets, resulting in k = 6 anchors at every position of maps of the input. also, as seen in Figure 2, 256-d rectangular kernels (W1 = 5, H1 = 4 and W2 = 4, H2 = 1) has been applied in place of the square one, as inspired by inception-RPN [31]. Across each position of sliding, the 2 kernels are added at the same time. local features which are patterns found in an image are extracted and formed into a 512-d feature vector by concatenating along the channel axis, that is fed into totally conv2D layer for number plate/no plate detection and regression box. These rectangle filters are more suited to artefacts along greater aspect ratios of number plate. The concatenated features, on the other hand, hold both local and contextual detail, which will help with plate classification. 2k scores at each position of sliding is for k anchors are generated by The plate classification layer on the feature diagram, indicating whether the anchors are likely to be license plates or not. The anchor box offsets to a neighboring ground-truth are output by the bounding box regression sheet, which outputs 4k values. The regression layer gives output four scalars (dr,dz,dv,di), these scales are the invariance translation and log-space height/distance transfer, given an anchor with the middle at (ra, za), width va, and height ha.

Equation for regression for bounding box

$$r = ra + drva, z = za + dyha,$$

$$v = va \exp(dw), i = ha \exp(di),$$

where r, z are the bounding box's core coordinates after regression, and v, I, are the bounding box's width and height.

There will be MN anchors in total for a convolutional function map of size $M \times N$. Those anchors are interchangeable and overlapping to a large extent. Furthermore, positive anchors are less than negative, resulting in bias during preparation if any of them are used. As a mini-batch, from single picture 256 anchors are randomly sampled with a positive/negative anchor ratio of up to 1:1. Any the hand labeled **bounding** boxes with an IoU score greater than 0.7 is chosen as an anchor.

Anchors with an IoU of less than 0.3 are considered negatives.

To ensure that any hand labeled box contains minimum single anchor which is positive, the anchors with the maximum Intersection over Union scores are considered positives. If there aren't sufficiently positive anchors, we use pessimistic anchors to fill up the gaps.

For box grouping, we use the binary logistic loss, and for box regression, we use the smooth L1 loss [30].

2) Plate Detection Network

The aim of the plate detection network is to determine if the proposed RoIs are car license plates and to refine the plate bounding box coordinates. To remove discriminative features for license plate identification, two completely dense layers having dropout rate of 1/2 and 2048 neurons are used. Each RoI's features are converted to a vector and transferred between the two completely linked layers. After that, the encoded features are fed into 2 different linear transformation layers, one for plates identification and the other for bounding box regression. The softmax likelihood of every Region of interest pooling as number plate/non-number plate is shown by the number plate classification layer's two outputs.

3) Plate Recognition Network

Based on the extracted area attributes, the number plate network recognition is used to identify every character in Region of interest pooling. We treat plate identification as a sequence marking issue to prevent the difficult job of character segmentation. The sequential features are labelled using bidirectional RNNs (BRNNs) with CTC loss [25], as seen in Figure 4.

B. Loss Functions and Training

During training, the entire network takes a picture, the number plate bounding rectangle, and the related labels as inputs, as we previously demonstrated., we integrate the number plate identification and recognition loss terms and training the recognition and detection networks together. after obtaining the samples and area features.

IV. Results

A. Datasets

The feasibility of our suggested approach is evaluated using four datasets.

The first dataset, CarFlag-Large, is made up of car licence plates from China. In all, we gathered 460000 photos. Fixed surveillance cameras take photographs from a frontal angle under various weather and lighting conditions, such as on sunny days, snowy days, or at dark, with 1600 2048 resolution. The numberplates are almost vertical. In the ground-truth file, only the closest licence plate in the picture is numbered. For instruction, we have used more than 3 lakh pictures, and for testing, we use 138000 pictures. The Application-Oriented License Plate (AOLP) database [12] is the second dataset. It contains a total of 2049 photographs of Taiwan licence plates. According to [12], this index is divided into three subsets of varying degrees of complexity and photographing conditions: Access Control (AC), Traffic Law Enforcement (LE), and Road Patrol (RP) Since we don't have any other photos of Taiwan licence plates, we choose photographs from various sub-datasets for training and research separate.

B. Evaluation Criterion

We use the "End-to-end" validation procedure for over-all spotting of text spotting in usual scenes [40]

IoU is described as

$$\text{IoU} = \frac{\text{area}(\mathbf{R}_{\text{dete}} \cap \mathbf{R}_{\text{gr}})}{\text{area}(\mathbf{R}_{\text{dete}} \cup \mathbf{R}_{\text{gr}})}$$

\mathbf{R}_{dete} and \mathbf{R}_{gr} are the detected bounding box and ground truth zones, respectively.

If Intersection over Union along with a ground-truth minimum bounding rectangle is greater than $\text{IoU} > 1/2$, the minimum bounding rectangle is called valid.

In our experiments, we quantify and present F-measures, which synthesise precision and recall using the equation below.

$$\text{F-measure} = \frac{2 \times (\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})}$$

C. Performance Evaluation on CarFlag-Large

1) Network Structure Analysis: After RoI Pooling, Feature Size: A series of tests with varying feature widths Y was carried out in order to determine the feature size X and Y after RoI pooling. Table II(a) shows that a longer Y will result in improved results, but it will also come with more parameters. In the following tests, we chose X = 4 and Y = 28 based on both efficiency and model scale.

D. Tables summary and results for our datasets

Table I

Dataset summary

Data set	Image Quantity	Image resolution (Height * width)	Number of Training / test Images	Number of plate	Plate size (height * width)	
CarFlag-Large	640000	16000 X 2048	322000/ 138000	138000	(20~56) X (85~265)	
AOLP	AC	2049	240 X 352	1368 / 681	681	(25~32) X (70~87)
	LE		480 X 640	1292 / 757	757	(28~48) X (80~133)
	RP		240 X 320	1438 / 611	611	(30~58) X (70~120)
Caltech-Cars	126	592 X 896	1626 / 126	126	(23~59) X (70~87)	
PKU Data	G1	810	728 X 1082	322000 / 810	810	(35~57) X (145~184)
	G2	700	728 X 1082	322000 / 700	700	(30~62) X (160~184)
	G3	743	728 X 1082	322000 / 743	743	(29~53) X (145~184)
	G4	572	1236 X 1600	322000 / 572	572	(30~58) X (158~170)
	G5	1152	1200 X 1600	322000 / 1152	1738	(20~60) X (136~168)

Table II

Experiments of ablation to verify the robustness of the recommended network. (1) the effect of feature map size on performance next region of interest pooling. The results show that a denser pooling will improve efficiency. The model size, on the other hand, increases in accordance ("m" means million). We selected 4 28 in order to balance model size and efficiency. in the experiments that follow. 2) the impact of the feature extraction technique in the (prn) on performance rectangular max pooling and an additional 2 convolutional layers lead to better results.

(a)

Size	End-to-End Performance (%)	Model size In detection net
4 x 16	95.45	73M
4 x 20	96.13	90M
4 x 24	96.76	107M
4 x 28	97.13	124M
4 x 32	97.15	141M

(b)

Method	End-to-End Performance (%)
Convs. +pooling	97.13
Average pooling	91.83
Max pooling	95.61

Table III

Carflag-large dataset experimental results. We use a two-stage baseline approach to evaluate the running speed and results of the collectively trained network. Our proposed model gives good accuracies in less time.

Method	End-to-End Performance (%)	Detection-only performance (%)	End-to-End Speed (per image single scale) (ms)
Ours(Jointly-trained)	97.13	98.33	310
Ours(Two-stage)	94.09	97.05	450

Table IV

Results of experimentation on the aolp dataset access control (ac), it is very easy to capture images as cars cross through a defined passage at any speed. Camera installed on roads are used to capture images. If there is violation of traffic law from the enforcement dataset done by car. The term rp (road patrol) applies to the applications of the camera. Images are taken from arbitrary viewpoints and distances from a patrolling vehicle. In terms of accuracy and functionality, our model gives good accuracy and performs well.

Table IV

Method	End-to-End Performance (%)			Detection-only performance (%)			End-to-End Speed (per image single scale) (ms)
	AC	LE	RP	AC	LE	RP	
Hsu et al. [12]	-	-	-	96	95	94	260
Li et al. [24]	94.85	94.19	88.38	98.38	97.62	95.85	1000-2000
Ours(Jointly-trained)	95.59	96.43	83.80	99.12	99.08	98.20	400

Table V

Caltech-cars dataset experimental results in comparison to previous methods, our method provides the best detection performance.

Table V

Method	Detection-only performance (%)	End-to-End Performance (%)
Zhou et al. [10]	89.83	-
Tian et al. [42]	90.70	-
Li et al. [24]	96.38	-
Kim et al. [43]	97.60	-
Ours (Jointly. Trained)	98.04	94.12

CONCLUSION

We introduce a jointly developed network for synchronized identification and recognition of car license plates in this article. Car license number plates could be detected and recognized simultaneously, with high precision and reliability, using this network. The model size is greatly reduced by exchanging convolutional features with both the identification and recognition networks. Without image being cropped, or character isolation, the entire network can be educated almost end-to-end. The benefit of our process is validated by a thorough assessment and analysis on three datasets using three different methods. We want to expand our model to include multi-oriented vehicle license number plates in the future. Furthermore, a time study reveals that Non-Maximum Suppression consumes about half of the total running time. As a result, Non-maximum Suppression can be optimized to improve speed of processing.

REFERENCES

- [1] L. Neumann and J. Matas, "A method for text localization and recognition in real-world images," in Proc. Asian Conf. Comput. Vis., 2011, pp. 770–783.
- [2] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2012, pp. 3538–3545.
- [3] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in Proc. IEEE Int. Conf. Comput. Vis., Nov. 2011, pp. 1457–1464.
- [4] T. Wang, D. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in Proc. IEEE Int. Conf. Pattern Recognit., Nov. 2012, pp. 3304–3308.
- [5] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, "PhotoOCR: Reading text in uncontrolled conditions," in Proc. IEEE Int. Conf. Comput. Vis., Dec. 2013, pp. 785–792.
- [6] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in Proc. Eur. Conf. Comput. Vis., 2014, pp. 512–528.
- [7] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," Int. J. Comput. Vis., vol. 116, no. 1, pp. 1–20, 2016.
- [8] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "TextBoxes: A fast text detector with a single deep neural network," in Proc. Nat. Conf. Artif. Intell., 2017, pp. 4161–4167.
- [9] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, "Automatic licenseplate recognition (ALPR): A state-of-the-art review," IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 2, pp. 311–325, Feb. 2013.
- [10] W. Zhou, H. Li, Y. Lu, and Q. Tian, "Principal visual word discovery for automatic license plate detection," IEEE Trans. Image Process., vol. 21, no. 9, pp. 4269–4279, Sep. 2012.
- [11] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, "A license plate-recognition algorithm for intelligent transportation system applications," IEEE Trans. Intell. Transp. Syst., vol. 7, no. 3, pp. 377–392, Sep. 2006.
- [12] G.-S. Hsu, J.-C. Chen, and Y.-Z. Chung, "Application-oriented license plate recognition," IEEE Trans. Veh. Technol., vol. 62, no. 2, pp. 552–561, Feb. 2013.

- [13] Y. Yuan, W. Zou, Y. Zhao, X. Wang, X. Hu, and N. Komodakis, "A robust and efficient approach to license plate detection," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1102–1114, Mar. 2017.
- [14] A. H. Ashtari, M. J. Nordin, and M. Fathy, "An Iranian license plate recognition system based on color features," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1690–1705, Aug. 2014.
- [15] S.-L. Chang, L.-S. Chen, Y.-C. Chung, and S.-W. Chen, "Automatic license plate recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 1, pp. 42–53, Mar. 2004.
- [16] S. Yu, B. Li, Q. Zhang, C. Liu, and M.-Q. H. Meng, "A novel license plate location method based on wavelet transform and EMD analysis," *Pattern Recognit.*, vol. 48, no. 1, pp. 114–125, 2015.
- [17] I. Giannoukos, C.-N. Anagnostopoulos, V. Loumos, and E. Kayafas, "Operator context scanning to support high segmentation rates for real time license plate recognition," *Pattern Recognit.*, vol. 43, no. 11, pp. 3866–3878, 2010.
- [18] B. Li, B. Tian, Y. Li, and D. Wen, "Component-based license plate detection using conditional random field model," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1690–1699, Dec. 2013.
- [19] D. F. Llorca et al., "Two-camera based accurate vehicle speed measurement using average speed at a fixed point," in *Proc. 19th Int. Conf. Intell. Transp. Syst.*, Nov. 2016, pp. 2533–2538.
- [20] C. Gou, K. Wang, Y. Yao, and Z. Li, "Vehicle license plate recognition based on extremal regions and restricted Boltzmann machines," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1096–1107, Apr. 2016.
- [21] Y. Hou, X. Qin, X. Zhou, X. Zhou, and T. Zhang, "License plate character segmentation based on stroke width transform," in *Proc. 8th Int. Congr. Image Signal Process.*, Oct. 2015, pp. 954–958.
- [22] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet, "Multidigit number recognition from street view imagery using deep convolutional neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–13.
- [23] O. Bulan, V. Kozitsky, P. Ramesh, and M. Shreve, "Segmentation and annotation-free license plate recognition with deep localization and failure identification," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2351–2363, Sep. 2017.
- [24] H. Li and C. Shen. (2016). "Reading car license plates using deep convolutional neural networks and LSTMs."
- [25] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.
- [26] L. Neumann and J. Matas, "On combining multiple segmentations in scene text recognition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 523–527.
- [27] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, Nov. 2017.
- [28] M. Busta, L. Neumann, and J. Matas, "Deep TextSpotter: An end-to-end trainable scene text localization and recognition framework," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2223–2231.
- [29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, Sep. 2014.
- [30] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards realtime object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
- [31] Zhong, Z., Jin, L., Zhang, S., & Feng, Z. (2016). Deeptext: A unified framework for text proposal generation and text detection in natural images. arXiv preprint arXiv:1605.07314.