Efficient License Plate Recognition via Holistic Position Attention

Yesheng Zhang, Zilei Wang*, Jiafan Zhuang
University of Science and Technology of China
{ysyzhang, jfzhuang}@mail.ustc.edu.cn, zlwang@ustc.edu.cn

Abstract

License plate recognition (LPR) is a fundamental component of various intelligent transportation systems, and is always expected to be accurate and efficient enough in real-world applications. Nowadays, recognition of single character has been sophisticated benefiting from the power of deep learning, and extracting position information for forming a character sequence becomes the main bottleneck of LPR. To tackle this issue, we propose a novel holistic position attention (HPA) in this paper that consists of position network and shared classifier. Specifically, the position network explicitly encodes the character position into the maps of HPA, and then the shared classifier performs the character recognition in a unified and parallel way. Here the extracted features are modulated by the attention maps before feeding into the classifier to yield the final recognition results. Note that our proposed method is end-to-end trainable, character recognition can be concurrently performed, and no post-processing is needed. Thus our LPR system can achieve good effectiveness and efficiency simultaneously. The experimental results on four public datasets, including AOLP, Media Lab, CCPD, and CLPD, well demonstrate the superiority of our method to previous state-of-the-art methods in both accuracy and speed.

Introduction

At present, intelligent transportation systems (ITS) have been widely applied in various fields, *e.g.*, improving transportation security (Zhang et al. 2011) and enhancing productivity (Anagnostopoulos et al. 2006). License plate recognition (LPR) is one of the key techniques in ITS since the license plates are generally used as the identification of vehicles. For example, a scout car uses the LPR deployed at the edge device to identify the vehicles violating the traffic rules. In the real-world applications, it is always expected that the LPR system is highly efficient along with enough accuracy as the edge devices usually have limited resources. However, it is still challenging for LPR to achieve good trade-off between high accuracy and efficiency, especially for the complicated images as shown in Fig. 1.

For the LPR task, we indeed need extract the two key attributes of a license plate, *i.e.*, semantic category and position information for each character, in order to produce the corresponding character sequence. Traditional methods take an intuitive way that first detects all characters in a license plate and then recognizes the cropped character subimages one-by-one. Obviously, the two involved procedures are performed separately. Nowadays, the recognition of single character has been sophisticated by exploiting the powerful capacity of deep learning. But character detection is still a challenging problem in practice because it is usually sensitive to variations of lighting, angles, fonts, colors, and background (Du et al. 2012; Saha 2019).

Figure 1: Hard license plate examples and corresponding recognition results by our proposed method. They are from AOLP, Media Lab, and CCPD from top to bottom, respectively. Here some complicated situations are shown, including uneven lighting, blurring, staining, and serious tilting.

Recently, some methods attempt to extract the semantic and position information jointly, and consequently better recognition performance can be achieved. Here we roughly divide them into three broad categories. The first one is to utilize the RNN/LSTM techniques (Li and Shen 2016; Li, Wang, and Shen 2018; Cheang, Chong, and Tay 2017; Kessentini et al. 2019) over densely extracted features in a sliding-window manner, which encodes both semantic and position information. However, the mismatch between receptive fields and character areas would bring noises to the features, and meanwhile the dense feature extraction and sequential processing would hinder the run-time performance. The second one is to employ multiple classifiers (Xu et al. 2018; Špaňhel et al. 2017; Jain et al. 2016; Gonçalves et al. 2018) to separately recognize the characters at different positions. Here the same features are fed into the classifiers without position information and the classifiers are expected

*Corresponding Author.

to automatically focus on different character positions. That is, character position information is implicitly encoded by different classifiers. Nevertheless, the classifiers for semantic recognition are difficult to accurately extract the position information and also cannot utilize the diversity of characters at different positions. The third one is to exploit semantic segmentation to explicitly segment the license plate images. For example, (Zhuang et al. 2018) performs semantic segmentation on the input image and then uses the object counting technique to determine the number of characters belonging to the same semantic. Here the subtle post-processing (*e.g.*, connected component analysis) is needed, which involves the tuning of many hyper-parameters for different datasets and thus hinders the generalization of the method.

In this paper, we propose a novel LPR framework aiming at simultaneously achieving high recognition accuracy and computational efficiency. Our key idea is to explicitly disentangle position and semantic information for producing more discriminative features, and then use a shared classifier to recognize characters at different positions that can better exploit diversity of characters. To this end, we propose a holistic position attention (HPA) consisting of position network and shared classifier. Specifically, position network is to directly encode the position information of characters in a license plate by explicitly disentangling from the semantic information. Evidently, the produced HPA maps have good explainability, each of which represents a certain position of character sequences. Shared classifier is to recognize the characters at different positions in a unified and parallel way. Here the character features at different positions are obtained by modulating the global semantic features with the corresponding HPA map. Through the cooperation of position network and shared classifier, our proposed method is end-to-end trainable, the characters at different positions can be processed concurrently, and no post-processing is needed. These characteristics benefit our method to achieve high accuracy and efficiency simultaneously. The experimental results on the benchmark datasets well demonstrate the superiority of our proposed method to previous state-of-the-art methods in both accuracy and speed. Fig. 1 gives the recognition results of some challenging license plates by our proposed method.

The contributions of this work are summarized as follows.

- We propose a novel HPA approach consisting of position network and shared classifier, which benefits the LPR system to achieve high efficiency along with better recognition accuracy.
- We comprehensively study the effect of the key designs in LPR. Particularly, it is shown that the end-to-end training and shared classifiers for different positions are important to the LPR performance.
- We experimentally evaluate the proposed method on the AOLP, Media Lab, CCPD, and CLPD datasets, and the results show that our method outperforms previous state-of-the-art methods.

Related Work

Here we roughly divide different LPR methods into four categories according to the used key techniques.

Two-stage Method. Before explosion of deep learning, traditional LPR methods mostly follow a similar pipeline, *i.e.*, explicit character detection followed by independent character recognition. So we call them two-stage methods. Character detection is essentially to decompose the image of a license plate into the character subimages. In practice, it has became the main obstacle of LPR since its results are easily affected by environmental conditions, *e.g.*, varying light, low resolution, motion blurring, and object deformation. To alleviate this issue, some heuristic algorithms are introduced, such as maximally stable extreme region (MSER) (Hsu, Chen, and Chung 2012), connected component analysis (CCA) (Anagnostopoulos et al. 2006), and vertical projection (Zhu, Dianat, and Mestha 2015; Yu and Kim 2000; Duan et al. 2005). Afterwards, CNN is introduced to both character detection (Laroca et al. 2018; Dong et al. 2017) and character recognition (Hsu, Chen, and Chung 2012; Dong et al. 2017; Laroca et al. 2018). But the performance of character detection still falls behind the requirement of real-world applications, *i.e.*, extracting position information of characters in a license plate is still challenging.

Sequential Method. This type of methods adopt some sequential model to recognize the involved characters. Typically, the sliding window is used to densely extract character probabilities or features from the plate images, and a sequential method is employed to perform recognition. Specifically, CNN (Li and Shen 2016) are used for feature extraction. Sweep OCR (Bulan et al. 2017) and fully convolutional network (FCN) (Wu and Li 2016; Zherzdev and Gruzdev 2018) can be used for character probability sequence acquisition. Recurrent neural network (RNN) with long short-term memory (LSTM) (Li and Shen 2016), hidden markov model (HMM) (Bulan et al. 2017), greedy or beam search (Zherzdev and Gruzdev 2018), and customized non-maximum suppression (NMS) (Wu and Li 2016) can be used for character sequence inference. Benefiting from elimination of character detection, these methods can usually achieve better performance. However, the mismatch between receptive fields and real character regions would bring noises to extracted features. Recently, some sequential methods introduce the attention mechanism to tackle the issue. For example, (Zhang et al. 2020) utilizes LSTM to yield attention weight for each character and then the feature for recognition is obtained by weighted sum. DAN (Wang et al. 2020) proposes to decouple attention from text before gate recurrent unit (GRU) recognizer. But the sequential decoding involved in these methods would hinder the run-time performance.

Multi-classifier based Method. To avoid extracting the position information, (Xu et al. 2018; Špaňhel et al. 2017; Jain et al. 2016; Gonçalves et al. 2018) propose to utilize multiple classifiers to perform LPR. Specifically, the global features are first extracted from the entire plate image, and then they are fed into different classifiers. Here the classifiers are expected to automatically focus on different character regions.

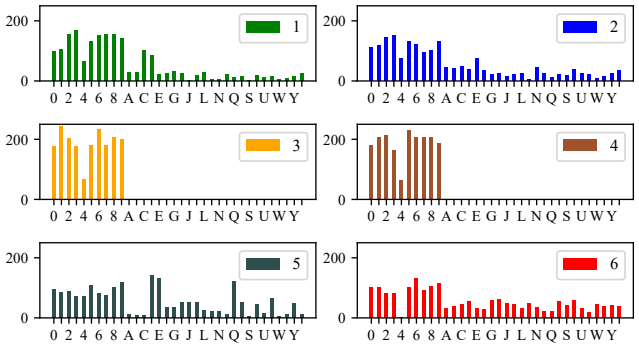

Figure 2: Distribution of license plate characters at different positions. Here the AOLP dataset is particularly adopted, in which each license plate contains 6 characters. It can be seen that their distributions are quite inconsistent.

That is, the position information is implicitly represented by different classifiers. However, the classifiers primarily designed for recognition are difficult to accurately localize the characters. In addition, the character distributions at different positions are highly inconsistent, as shown in Fig. 2. Only using the characters at single position during training would bias the classifier and cannot fully utilize the diversity of character data.

Segmentation based Method. This type of methods formulate LPR into a semantic segmentation task, which produces the recognition results over pixels rather than one semantic label for each character. Through some subtle post-processing, they can achieve better recognition performance. For example, (Zhuang et al. 2018) introduces semantic segmentation to LPR for the first time, and it boosts the performance of LPR to a new level. In (Zhuang et al. 2018), however, the post-processing techniques (*e.g.*, CCA, character counting) need be carefully designed, which would suffer from the tuning of many hyper-parameters along with unsatisfactory run-time performance.

In fact, the license plates have some priors that can be utilized by LPR. First, the number of characters in a license plate is usually fixed or has little variation. Second, the layout and style of characters is common regular given a plate type. In this paper, we propose a novel LPR method to deeply exploit these priors with aims of simultaneously achieving high accuracy and efficiency. Specifically, we propose HPA to explicitly extract the position information of each character, a shared classifier to concurrently recognize characters, and an end-to-end network to avoid post-processing.

Our Method

Overview

In this work, we aim to get an accurate and efficient LPR method by deeply exploring the inherent characteristics of license plates. To this end, we propose a novel LPR framework with holistic position attention (HPA), as shown in Fig. 3. Specifically, we first adopt a backbone network to transform the input image into the global features, which have lower spatial resolution than the images. Then we use the semantic network and position network to separately extract the semantic and position information of each character in a license plate, which result in the semantic features and position attention maps, respectively. Finally, we propose to use a shared classifier to perform character recognition by taking the semantic features and position attention maps. That is, only one classifier is trained and then different characters in a license plate can be concurrently processed during inference. The output of shared classifier is exactly the final recognition result of the license plate.

In our proposed framework, the main challenge is how to effectively fuse the semantic and position information with intrinsically encoding the order of different characters. To tackle the issue, we propose a novel HPA consisting of position network and shared classifier. Specifically, we use position network to produce holistic position attention maps, each of which represents the position information of one character in the license plate. The spatial size of HPA maps is same as that of semantic features, and the sequence of maps naturally express the order of characters. We use these maps to separately modulate the semantic features from semantic network with element-wise multiplication, and consequently the features of each character are correspondingly produced. Finally, a shared classifier takes the character features to complete the recognition. Note that we can optionally add supervision signals to semantic and position networks if the ground truth is provided. In practice, the semantic or position features would be first upsampled to the resolution of images and then transformed into the results of semantic segmentation for introducing the supervision.

From Fig. 3, we can observe the following characteristics of the proposed LPR framework. First, we explicitly disentangle the semantic and position information of characters by two networks in parallel. The sensitive character detection or time-consuming sliding window operation in previous works is avoided. Second, the order of characters in a license plate is naturally encoded, and the recognition results are directly yielded by the network. Obviously, the subtle post-processing in (Zhuang et al. 2018) can be avoided. Third, we use a shared classifier to conduct character recognition for different positions. So the diversity of characters can be fully utilized, unlike the multi-classifier based methods (Xu et al. 2018; Špaňhel et al. 2017; Jain et al. 2016; Gonçalves et al. 2018). This is important in practice since the license plates in the real scenarios usually present large variation of character distributions for different positions, as shown in Figure 2. Fourth, our proposed framework is highly parallel, which benefits our method to be efficient and have a larger space of engineering optimization. To be specific, semantic network and position network are parallel, and different characters of a license plate can be concurrently recognized.

Semantic and Position Network

In our proposed method, the semantic network is used to produce the semantic features and the position network is used to produce the position attention maps. To more clearly

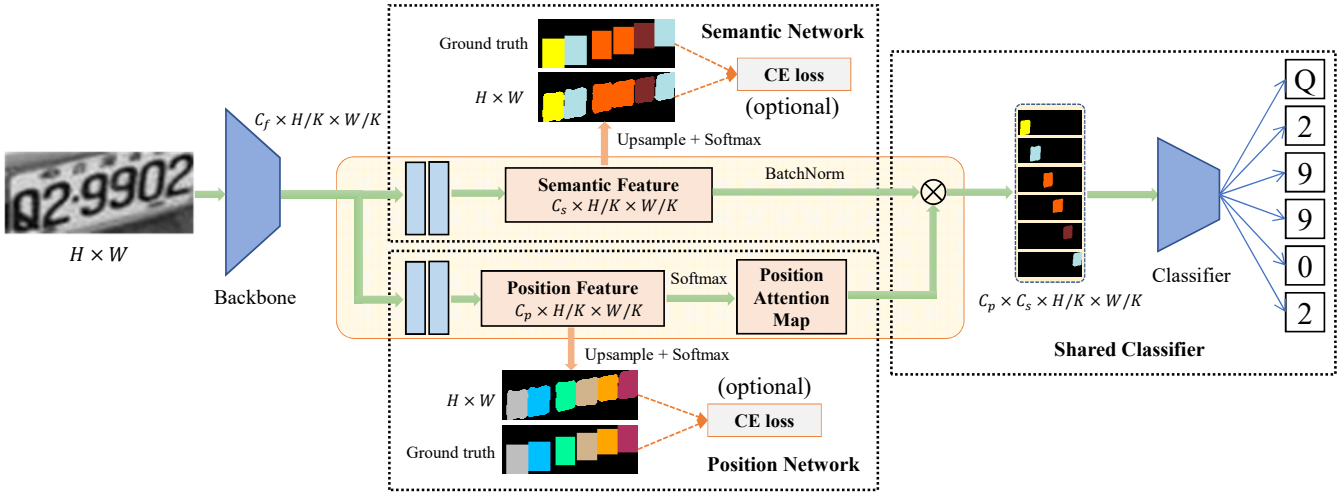

Figure 3: Illustration of our proposed LPR method, which consists of four main components: backbone network, semantic network, position network, and shared classifier. Backbone network is to extract the global features of input image. Semantic network and position network are to produce semantic features and position attention maps, respectively. Shared classifier is to perform character recognition that takes the semantic features modulated by the attention maps. Here C_f is the channel number of the global features, and C_p, C_s are the number of characters in a license plate and number of character classes, respectively. Best viewed in color.

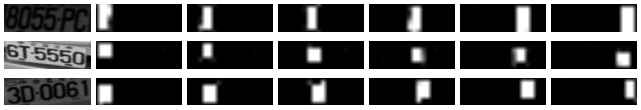

Figure 4: Visualization of HPA maps. Here the attention maps for different characters in a license plate are orderly exhibited from left to right.

explain them, here it is assumed that the ground truth is provided. For semantic network, the ground-truth labels give a bounding box for each character and the pixels in the bounding box are annotated according to the semantic class of characters. Similarly, the ground-truth labels for position network are with bounding boxes, but the pixel label in a bounding box is determined by the position of characters in a license plate. For example, the position label is from $\{1, 2, ..., 6\}$ if the involved license plates have 6 characters at most. In our proposed method, the position information need be transformed into the position attention maps, as shown in Fig. 4. In our experiments, we produce them by applying *softmax* to the position features.

The semantic and position networks connect the same backbone network to share global features and produce the semantic and position features by appending different heads. So we choose an off-the-shelf network for common feature extraction. In our implementation, BiSeNet (Yu et al. 2018) is particularly adopted due to its high efficiency. Specifically, BiSeNet uses Spatial Path and Context Path to extract spatial and context information separately, and then utilizes feature fusion module (FFM) to fuse them. Here we split BiSeNet into $\text{Net}_{backbone}$ and Net_{task}. $\text{Net}_{backbone}$ consists of spatial and context paths and is used for common feature extraction. Net_{task} contains FFM and is used to produce semantic or position features. Note that $\text{Net}_{backbone}$ can adopt different base network, *e.g.*, ResNet-18, -34, -50, and -101 in the original paper. We will evaluate different base networks in our experiments.

Shared Classifier

The shared classifier takes the semantic features and HPA maps to produce the recognition result of character sequence. Here we propose to construct the discriminative features for each character in order to achieve high recognition accuracy. Particularly, the spatial attention way is adopted with the element-wise multiplication operator, in which the HPA map of each character is used to modulate the semantic features separately. As a result, we can get a sequence of character features for recognition. Evidently, different characters can be concurrently processed. In this stage, the background channel is discarded before multiplication due to their irrelevance to character recognition.

In our implementation, we use a rather lightweight network for the shared classifier. To be specific, a 5×5 convolutional layer followed by batch normalization and ReLU is used to process features, and a global pooling followed by a fully connected layer is used for character classification.

In the real-world applications, the LPR system may need to process the license plates containing variable-number characters (*e.g.*, Media Lab). Indeed, our proposed method can handle such a situation. Specifically, we use the maximum number of characters among all license plates to set C_P, and introduce a *void* class to represent the semantic of null character. During training, if the actual number of characters $C < C_P$, the $C_P - C$ null characters are appended to the character sequence. During inference, if a void class is

predicted, it would be directly discarded.

Loss Function

The proposed LPR method can be trained in an end-to-end manner. Here we consider three supervision signals, *i.e.*, character semantic, character position, and character sequence. Here the first two are used only when their ground truth is provided. The cross entropy (CE) loss is adopted for all of them, and we denote the corresponding losses by L_S, L_P, and L_C, respectively. The overall loss function is the linear combination of the three losses, namely,

$$L = \alpha_1 L_S + \alpha_2 L_P + \sum_{i=1}^{m} L_{C_i}. \quad (1)$$

Here m is the number of characters in a license plate and L_{C_i} represents the loss for the i-th character. Both α_1 and α_2 are the control parameters to balance the three loss terms, and they are empirically set to 1.0 in our experiments. If we need remove the supervision from character semantic or character position, the corresponding control parameter is set to 0.

Experiments

In this section, we experimentally evaluate the proposed LPR method. Here four challenging public license plate datasets are used, *i.e.*, AOLP (Hsu, Chen, and Chung 2012), Media Lab (Anagnostopoulos et al. 2008), CCPD (Xu et al. 2018), and CLPD (Zhang et al. 2020). This work focuses on LPR, and we use the images of license plates detected by the YOLOv4 model (Bochkovskiy, Wang, and Liao 2020) that is learned on the training set of CCPD. We also provide the results on the cropped images of license plates with ground-truth bounding boxes for applicable datasets. As for the performance metrics, we follow previous methods (Xu et al. 2018) and adopt the recognition accuracy over the entire license plates. That is, the recognition result of a license plate would be considered false as long as its IoU with the ground-truth bounding box is less than 70% or one of characters is wrongly predicted. Besides, we also calculate frame-per-second (FPS) to measure the run-time performance.

Implementation Details

We use ResNet-18, -34, -50, -101 (He et al. 2016) as the base network in backbone, and the pretrained models on ImageNet (Deng et al. 2009) are used for initialization. For our used BiSeNet, the spatial size of features is 8 times smaller than that of input images. Before training AOLP dataset, our LPR network is pretrained on the synthetic license plate dataset provided by (Zhuang et al. 2018). During training, we use a mini-batch of the size 256 on 2 GTX1080Ti GPUs and the RMSProp optimizer (Ruder 2016). The initial learning rate is set $2e-5$ for each dataset. Throughout the experiments, the license plate images are resized to 50×160 according to (Zhuang et al. 2018), and the aspect ratio is close to that of real license plates. Here we adopt some common data augmentation strategies, such as noising, blurring, color jittering, rotation, projection, and cropping.

Datasets

AOLP contains 1874 license plate images from Taiwan. The dataset is divided into three subsets: access control (AC) with 681 images, law enforcement (LE) with 582 images, and road patrol (RP) with 611 images. Appearance of license plates are different for three subsets (*e.g.*, perspective and lightness variance). We follow the same training/test split as in (Li and Shen 2016; Zhuang et al. 2018), *i.e.*, two subsets is used for training and the remaining one is for test. So we conduct three experiments, and then calculate their recognition accuracy separately. Besides, we evaluate the run-time performance of all open-source methods, in which FPS is calculated with the same input size (*i.e.*, 50×160) and the average performance on 5000 forward computation is reported. All evaluation experiments are performed on a server with one NVIDIA GTX 1080Ti GPU, Intel(R) Xeon(R) CPU E5-2640 v4, and about 500 GB memory.

Media Lab collects 706 Greek license plate images, which is divided into a normal subset with 427 images and a difficult subset with 279 images. The dataset does not provide the official training/test split. For fair comparison, we follow the baseline methods and use the images in the normal subset for test. Since images in Media Lab is too scarce to train an effective model, we follow (Zhuang et al. 2018) to perform 4-fold cross validation. Specifically, we evenly divide the normal subset into four parts randomly. We then conduct four experiments, for each of which three parts together with the difficult subset are used for training and the remaining one is for test.

CCPD (Xu et al. 2018) is a Chinese license plate dataset with $290K$ license plate images, which is the largest public LPR dataset at present. CCPD contains a base subset with $200K$ images and some small subsets. Here we conduct the experiments on the base subset for fair comparison since the official training/test split is available (*i.e.*, $100K$ for training and $100K$ for test). For this dataset, only the results using the supervision of character sequence are reported.

CLPD (Zhang et al. 2020) contains 1200 images with different vehicle types, which are collected from all 31 provinces in China. For fair comparison, we use the same training strategy as in (Zhang et al. 2020), *e.g.*, treating the whole data as the test set and the model is trained on the CCPD-base dataset.

Experimental Results

AOLP Table 1 gives the results, and we have the following observations. First, our proposed method achieves the best recognition accuracy compared with previous state-of-the-art methods no matter if the detected bounding boxes or ground truth are used. Particularly, an accuracy of near 100% on all subsets is achieved when ResNet-101 is equipped in the base network. Second, our method can simultaneously get good accuracy and speed. For example, our method equipped with ResNet-50 outperforms previous methods and commercial systems in both recognition accuracy and run-time performance. Third, compared with (Hsu, Chen, and Chung 2012) that uses the character detection, our method brings significant accuracy improvement. Com-

Method	AC	LE	RP	FPS
(Hsu, Chen, and Chung 2012)	88.50	86.60	85.70	-
(Li and Shen 2016)	94.85	94.19	88.38	-
(Wu and Li 2016)*	97.90	97.60	98.20	-
(Silva and Jung 2018)	-	-	98.36	90
(Zhuang et al. 2018)*	99.41	99.31	99.02	25
SNIDER(Lee et al. 2019)*	-	-	99.18	-
(Zhang et al. 2019)*	-	-	99.67	-
(Zhang et al. 2020)*	97.30	98.30	91.90	-
OpenALPR*	92.36	86.08	93.13	-
Sighthound*	94.57	96.56	89.03	-
Ours (ResNet-18)*	99.27	98.97	99.84	**191**
Ours (ResNet-18)	98.83	97.94	98.20	-
Ours (ResNet-34)*	99.12	99.31	99.84	158
Ours (ResNet-34)	98.79	98.97	98.36	-
Ours (ResNet-50)*	99.41	99.48	99.84	106
Ours (ResNet-50)	99.12	98.97	98.53	-
Ours (ResNet-101)*	**99.56**	**100.00**	**99.84**	57
Ours (ResNet-101)	99.27	99.14	98.85	-

Table 1: Performance comparison on the AOLP dataset. AC/LE/RP denotes the corresponding subset is used for test. "*" denotes using the ground-truth bounding boxes of license plates.

Method	Average Accuracy
(Anagnostopoulos et al. 2006)	86.00
(Zhu, Dianat, and Mestha 2015)	87.33
(Zhuang et al. 2018)*	97.89
OpenALPR*	68.38
Sighthound*	92.04
Ours (ResNet-18)*	98.13
Ours (ResNet-18)	95.78
Ours (ResNet-34)*	98.59
Ours (ResNet-34)	95.78
Ours (ResNet-50)*	98.59
Ours (ResNet-50)	96.72
Ours (ResNet-101)*	**98.83**
Ours (ResNet-101)	96.96

Table 2: Performance comparison on the Media Lab Dataset. "*" denotes using the ground-truth bounding boxes of license plates.

pared with (Zhuang et al. 2018) that employs some post-processing techniques, our method achieves much better run-time performance due to avoiding post-processing.

Media Lab The experimental results on the Media Lab dataset are reported in Table 2. It can be seen that our method outperforms all baseline methods. Particularly, our method equipped with ResNet-101 achieves a new state-of-the-art. In particular, the average accuracy of 98.83% means that only 5 license plates failed in total.

CCPD & CLPD Table 3 shows the recognition performance on the CCPD-Base and CLPD datasets. We can see that our method equipped with ResNet-18 is very close to the state-of-the-art method in (Xu et al. 2018; Zhang et al. 2019) on CCPD-Base, and can get a higher accuracy using ResNet-101. As for CLPD, our proposed method remarkably outperforms previous state-of-the-art methods.

Method	CCPD-base	CLPD
(Xu et al. 2018)	98.50	66.5
(Zhang et al. 2019)	99.00	-
(Zhang et al. 2020)	99.60	70.80
Ours (ResNet-18)	99.57	78.50
Ours (ResNet-34)	99.61	79.50
Ours (ResNet-50)	99.62	82.17
Ours (ResNet-101)	**99.65**	**82.50**

Table 3: The recognition accuracy on the CCPD-base and CLPD datasets.

Ablation Study

In this section, we conduct ablation study to show the effect of key designs in our proposed method. Here the AOLP dataset is particularly used, the setting equipped with ResNet-101 is adopted, and the images of license plates are cropped using the ground-truth bounding boxes.

Effect of HPA Maps Here we explore the effect of HPA maps in constructing character features. If the maps are not involved, we cannot construct the specified features for the characters at different positions, and thus have to use multiple classifiers to separately recognize them. Here we construct a setting that replaces the shared classifier with 6 independent classifiers, and then uses the same semantic features from semantic network to feed them. For fair comparison, we also use the separated classifiers for the situation that applies the HPA maps. Table 4 gives the experimental results, and they well demonstrate the necessity and effectiveness of explicitly encoding the position information of characters.

Method	AC	LE	RP	MP
w/ HPA (MC)	97.65	94.33	99.51	-
w/o HPA (MC)	83.55	61.17	96.40	-
SC + ResNet-101	99.56	100.00	99.84	66.77
MC + ResNet-101	97.65	94.33	99.51	71.21
SC + ResNet-18	99.27	98.97	99.84	18.23
MC + ResNet-18	92.51	86.94	99.02	22.66

Table 4: Ablation study on holistic position attention (HPA) and shared classifier (SC). Here MC represents the multiple classifiers and MP denotes the model size with the unit of million parameters.

Training methods	AC(%)	LE(%)	RP(%)
$S \rightarrow P \rightarrow C$	99.27	99.31	99.51
$P \rightarrow S \rightarrow C$	90.31	88.49	98.20
$S + P \rightarrow C$	99.27	99.66	99.51
$S + P + C$	99.56	100.00	99.84

Table 5: Performance comparison of different training strategies. Here "S", "P", and "C" represent semantic network, position network, and shared classifier, respectively. "$\rightarrow$" and "$+$" correspond to the one-by-one training and joint training.

Effect of Shared Classifier Here we explore the advantage of shared classifier. We conduct experiments with the setting of ResNet-18 and -101. For comparison, the features for character recognition are kept same, and we just use 6 separated classifiers to replace the shared classifier. As shown in Table 4, the shared classifier can significantly boost the recognition accuracy because the character data at different positions can be fully utilized during training. Obviously, the shared classifier can also reduce the model size.

End-to-End Training Here we investigate the effect of end-to-end training. In our LPR system, three key networks need be trained, *i.e.*, semantic network, position network, and shared classifier. In particular, we can train different networks in a one-by-one way, in which the trained network is frozen when training the consequent networks, or in a joint way. Table 5 gives the results for different training strategies. It can be seen that jointly training the three networks (*i.e.*, the end -to-end training) achieves the best performance.

Supervision Signal Here we investigate the extra supervision signals for semantic and position networks. Table 6 gives the results of different combinations of supervisions. From the results, we can see that more supervisions are helpful, but our method can still achieve good performance without semantic and position supervisions.

Failure Case Analysis

In this section, we analyze the failure cases to deeply explore the performance of our proposed method. Specifically, we exhibit all failure cases on AOLP and Media Lab datasets from the best results, as shown in Fig. 5, which contains 4 failed samples for AOLP and 5 failed samples for Media Lab. Here we list the license plate image, generated semantic map and position maps, ground truth, predicted character sequence, and cause of failure. We can observe that the failures are mainly caused by the poor quality of the license plate images, which would lead to bad semantic features. In particular, it is observed that the predicted position maps are all correct for the failed samples, which well demonstrates the robustness of our proposed holistic position attention.

Method	AC(%)	LE(%)	RP(%)
Both	99.56	100.00	99.84
w/o P	99.56	99.83	99.84
w/o S	99.41	99.31	99.84
Neither	99.41	99.31	99.67

Table 6: Effect of supervision signals. Here "S" and "P" corresponds the supervision signals for semantic and position networks.

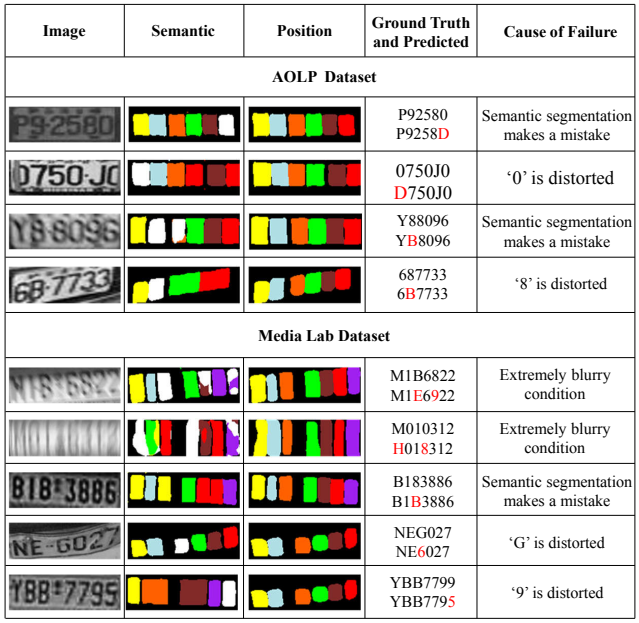

Figure 5: Failure cases on the AOLP and Media Lab dataset. Best viewed in color.

Conclusion

In this paper, we proposed a novel LPR method that can simultaneously achieve high recognition accuracy and computational efficiency. Specifically, we introduce a holistic position attention to explicitly encode the position information of characters in a license plate and build a shared classifier to perform character recognition. The experimental results on four public datasets verified the effectiveness of our method, which outperforms previous state-of-the-art methods in both recognition accuracy and run-time performance. Due to the highly parallel design, our proposed method has a large space of engineering optimization.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant 61673362 and 61836008, Youth Innovation Promotion Association CAS (2017496).

References

Anagnostopoulos, C. N. E.; Anagnostopoulos, I. E.; Loumos, V.; and Kayafas, E. 2006. A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Trans. Intell. Transport. Syst.* 377–392.

Anagnostopoulos, C.-N. E.; Anagnostopoulos, I. E.; Psoroulas, I. D.; Loumos, V.; and Kayafas, E. 2008. License plate recognition from still images and video sequences: A survey. *IEEE Trans. Intell. Transport. Syst.* 377–391.

Bochkovskiy, A.; Wang, C.-Y.; and Liao, H.-Y. M. 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv preprint arXiv:2004.10934* .

Bulan, O.; Kozitsky, V.; Ramesh, P.; and Shreve, M. 2017. Segmentation-and annotation-free license plate recognition with deep localization and failure identification. *IEEE Trans. Intell. Transport. Syst.* 2351–2363.

Cheang, T. K.; Chong, Y. S.; and Tay, Y. H. 2017. Segmentation-free vehicle license plate recognition using ConvNet-RNN. *arXiv preprint arXiv:1701.06439* .

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*, 248–255.

Dong, M.; He, D.; Luo, C.; Liu, D.; and Zeng, W. 2017. A CNN-Based Approach for Automatic License Plate Recognition in the Wild. In *BMVC*.

Du, S.; Ibrahim, M.; Shehata, M.; and Badawy, W. 2012. Automatic license plate recognition (ALPR): A state-of-the-art review. *IEEE Trans. Circuits Syst. Video Technol.* 311–325.

Duan, T. D.; Du, T. H.; Phuoc, T. V.; and Hoang, N. V. 2005. Building an automatic vehicle license plate recognition system. In *RIVF*, 59–63.

Gonçalves, G. R.; Diniz, M. A.; Laroca, R.; Menotti, D.; and Schwartz, W. R. 2018. Real-time automatic license plate recognition through deep multi-task networks. In *SIBGRAPI*, 110–117.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Hsu, G.-S.; Chen, J.-C.; and Chung, Y.-Z. 2012. Application-oriented license plate recognition. *IEEE Trans. Veh. Technol.* 552–561.

Jain, V.; Sasindran, Z.; Rajagopal, A.; Biswas, S.; Bharadwaj, H. S.; and Ramakrishnan, K. 2016. Deep automatic license plate recognition system. In *ICVGIP*, 1–8.

Kessentini, Y.; Besbes, M. D.; Ammar, S.; and Chabbouh, A. 2019. A Two-Stage Deep Neural Network for Multi-norm License Plate Detection and Recognition. *Expert Syst. Appl.* 159–170.

Laroca, R.; Severo, E.; Zanlorensi, L. A.; Oliveira, L. S.; Gonçalves, G. R.; Schwartz, W. R.; and Menotti, D. 2018. A robust real-time automatic license plate recognition based on the YOLO detector. In *IJCNN*, 1–10.

Lee, Y.; Lee, J.; Ahn, H.; and Jeon, M. 2019. SNIDER: Single Noisy Image Denoising and Rectification for Improving License Plate Recognition. In *ICCV*, 0–0.

Li, H.; and Shen, C. 2016. Reading car license plates using deep convolutional neural networks and lstms. *arXiv preprint arXiv:1601.05610* .

Li, H.; Wang, P.; and Shen, C. 2018. Toward end-to-end car license plate detection and recognition with deep neural networks. *IEEE Trans. Intell. Transport. Syst.* 1126–1136.

Ruder, S. 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747* .

Saha, S. 2019. A Review on Automatic License Plate Recognition System. *arXiv preprint arXiv:1902.09385* .

Silva, S. M.; and Jung, C. R. 2018. License plate detection and recognition in unconstrained scenarios. In *ECCV*, 580–596.

Špaňhel, J.; Sochor, J.; Juránek, R.; Herout, A.; Maršík, L.; and Zemčík, P. 2017. Holistic recognition of low quality license plates by cnn using track annotated data. In *AVSS*, 1–6.

Wang, T.; Zhu, Y.; Jin, L.; Luo, C.; Chen, X.; Wu, Y.; Wang, Q.; and Cai, M. 2020. Decoupled Attention Network for Text Recognition. In *AAAI*, 12216–12224.

Wu, Y.; and Li, J. 2016. License plate recognition using deep FCN. In *ICCSIP*, 225–234.

Xu, Z.; Yang, W.; Meng, A.; Lu, N.; Huang, H.; Ying, C.; and Huang, L. 2018. Towards End-to-End License Plate Detection and Recognition: A Large Dataset and Baseline. In *ECCV*, 255–271.

Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; and Sang, N. 2018. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *ECCV*, 325–341.

Yu, M.; and Kim, Y. D. 2000. An approach to Korean license plate recognition based on vertical edge matching. In *SMC*, 2975–2980.

Zhang, J.; Wang, F.-Y.; Wang, K.; Lin, W.-H.; Xu, X.; and Chen, C. 2011. Data-driven intelligent transportation systems: A survey. *IEEE Trans. Intell. Transport. Syst.* 1624–1639.

Zhang, L.; Wang, P.; Li, H.; Li, Z.; Shen, C.; and Zhang, Y. 2020. A Robust Attentional Framework for License Plate Recognition in the Wild. *IEEE Trans. Intell. Transport. Syst.* .

Zhang, S.; Tang, G.; Liu, Y.; and Mao, H. 2019. Robust license plate recognition with shared adversarial training network. *IEEE Access* 697–705.

Zherzdev, S.; and Gruzdev, A. 2018. LPRNet: License Plate Recognition via Deep Neural Networks. *arXiv preprint arXiv:1806.10447* .

Zhu, S.; Dianat, S. A.; and Mestha, L. K. 2015. End-to-end system of license plate localization and recognition. *J Electron Imaging* 023020.

Zhuang, J.; Hou, S.; Wang, Z.; and Zha, Z.-J. 2018. Towards Human-Level License Plate Recognition. In *ECCV*, 306–321.