FAFNet: A False-Alarm Filtering Algorithm Base on Deep Learning for License Plate Recognition

Qingbiao Zhou\textsuperscript{1}, Jie Fang\textsuperscript{1, *}, Yanming Du\textsuperscript{1}, Kun Gao\textsuperscript{2}

\textsuperscript{1}Zhejiang Industry Polytechnic College
\textsuperscript{2}Hangzhou Computer Vision Research Institute of ZKTeco Co., Ltd.

*Corresponding author e-mail: zhouqingbiao@zjipc.com

Abstract. In this paper, we propose an efficient False-Alarm Filtering Algorithm: FAFNet (False-Alarm Filtering Network) to solve the problems of false-alarm in Chinese License Plate Recognition (LPR) scenes. Firstly, FAFNet is a light-weight convolutional neural network which can solve the false-alarm problems in real time on embedded devices, and has achieved a reasonable performance. Secondly, we propose an approach of model internationalization without any retraining, so that the FAFNet model can be available in foreign LPR false-alarm problems, which reduces the training cost and labor cost. Finally, we create a large dataset with more than 200 thousand images from LPR scenes of the real world, which contains various conditions of false alarm. Our experimental results indicate that the proposed approach performs a high accuracy for filtering the false-alarm problems of different countries and can be used on embedded devices in real time.

Keywords: False-Alarm Filtering Algorithm, Chinese License Plate Recognition

1. Introduction
In recent years, license plate recognition plays more and more important role in our life with the development of our society. It is widely used in traffic management, digital security surveillance, vehicle recognition, urban parking management. Nowadays license plate recognition technologies \cite{1,2,3,4} have already made a great progress and can realize high-precision recognition in complex conditions.
However, license plate recognition technology still remains many problems to be solved, one of which is the false-alarm. False-alarm means that the non-license plate regions in the scene of license plate recognition are recognized as license plates, which brings great inconvenience to urban parking management. The main reasons for false-alarm are the complex background texture of the non-license plate regions and the similarity between characters in various scenes and the characteristics of license plate region. Some examples of false-alarm is showed in Figure 1, which is marked as yellow box.

In order to solve the false-alarm problems mentioned above, this paper proposes FAFNet algorithm. We know that although the regions of real license plate and false-alarm plate have similar features, the regions around them are different. The region around the real license plate is mainly car head, while the region around the false-alarm plate is some other things. Therefore, we can use the difference of the features of the region around the license plate to classify the real license plate and the false-alarm plate. In this paper, a reasonable rectangular region is cropped according to the license plate coordinates obtained by license plate recognition and relevant interception parameters, which is used as the input of FAFNet to judge the false-alarm. The process of filtering the false-alarm plate is showed in Figure 2.

FAFNet is based on the deep convolutional neural network, which can be trained end-to-end. Recent studies have shown great improvement of convolutional neural network in many computer vision tasks such as image classification, object detection and so on. However, it is still a challenge for running these tasks on embedded devices. FAFNet is a very efficient neural network, which takes only 0.024GFLOps to make a single forward pass.

Our main contributions can be summarized as follows:

A high precision and real-time method is proposed to solve the false-alarm problems of license plate recognition.

A method of model internationalization is proposed to solve the problem that the false-alarm filter model cannot be universal due to the different size of license plates in different countries.
A large dataset was produced, all from industry, and included a variety of scenarios: fuzzy images, low light conditions, physical effects, and various weather scenarios.

The remainder of the work is organized as follows. In Section 2, we review the details of our work, including the Cropping Rule and the FAFNet design. In Section 3, we introduced our dataset and experiments. The experimental results verify the excellent performance of the FAFNet. In Section 4, we discuss the solution to the problem of model internationalization. Section 5 contains a conclusion of the paper.

2. The proposed method
This section consists of two parts: (1) we describe the Cropping Rule that generating the input of FAFNet and determine the cropping parameters; (2) we introduce the FAFNet design in detail.

2.1. The cropping rule
As we mentioned above, the input image of FAFNet is generated from the original images based on the cropping parameters and the license plate coordinates obtained by LPR. In addition, we distinguish the real license plate from the false-alarm plate mainly by using the feature of vehicles’ head. Therefore, the Cropping Rule can be described as follows: the cropped image according to the license plate coordinates must contain a complete car head.

In this paper, we set the cropping parameter \( m \) as the ratio of the cropped image to the license plate in width, while \( n \) as the ratio in height. The center of the cropped image is set as the center point of the upper boundary of the license plate, which can obtain more useful information in the vertical direction:

\[
\begin{align*}
    w_{\text{crop}} &= m \times w_{lp} \\
    h_{\text{crop}} &= n \times h_{lp} \\
    \text{center}_x_{\text{crop}} &= \text{center}_x_{lp} \\
    \text{center}_y_{\text{crop}} &= \text{center}_y_{lp} - h_{lp}/2
\end{align*}
\]

where \( \text{crop} \) means the cropped image, \( \text{lp} \) means the license plate; \( w, h, \text{center}_x, \text{center}_y \) represent the width, height and the coordinates of center respectively.

Based on the Cropping Rule and lots of empirical testing, we found that \( m = 3, n = 8 \) is the most suitable cropping parameter for Chinese license plates. Figure 3 shows some cropped images. The blue boxes of Figure 3(a) represent the coordinate position of the license plate returned by the license plate recognition algorithm. The green boxes of Figure 3(b) represent the cropped region. The FAFNet input images are showed in Figure 3(c).
2.2. FAFAFNet

With the development of deep neural networks, many classical classification networks have been proposed, such as the early LeNet [16] and Alexnet [17], and later the more powerful VGGNet [5], GoogLeNet [6, 7] or ResNet [8]. Though these powerful networks work well for computer vision tasks, they cannot meet the needs of embedded devices in terms of model size and recognition speed. Therefore, a new lightweight network is designed based on the existing network construction tricks.

Our network architecture is inspired by the DenseNet [9] model for image classification. Compared with the traditional networks mentioned above, DenseNet has fewer parameters, smaller model size and faster computing speed, which meets the requirements of building a lightweight network. Meanwhile, its unique dense architecture deepens the depth of the network, endowing the model with implicit supervision and regularization, which enables the model to maintain high quality accuracy.

**Dense block.** Dense block is a major part of the network, which is concatenated by numbers of dense cells. In this paper, \([1 \times 1 \text{ conv}, 3 \times 3 \text{ conv}]\) is defined as the dense cell of the network. The \(3 \times 3 \text{ conv}\) is considered as a composite function, which contains three consecutive operations: \(3 \times 3 \text{ convolution, followed by a batch normalization (BN)} \) [10] and a leaky rectified linear unit (Leaky ReLU). The \(1 \times 1 \text{ conv}\) acts as the bottleneck layer before the \(3 \times 3 \text{ conv}\), which has been noted in [7,8] to reduce the number of input feature maps, and thus to improve computational efficiency. The specific structures of “conv”, dense cell and dense block are shown in Fig.4. In the dense cell, \(1 \times 1 \) convolution outputs \(\lambda k (\lambda > 1)\) feature maps, \(3 \times 3 \) convolution outputs \(k\) feature maps.
Figure 4. The structures of “conv”, dense cell and dense block

In the dense cell, 3 × 3 convolution outputs k feature maps while 1 × 1 convolution outputs \( k \lambda (\lambda > 1) \) feature maps. A dense block outputs mk feature maps if it has m dense cells.

**Transition layer.** As an important part of FAFNet, the transition layer is mainly used to connect the dense blocks in the network. It consists of an 1 × 1 conv and a 2 × 2 max pooling layer. The main function of 1 × 1 conv is to fuse the output features of dense block. The 2 × 2 max pooling layer down sampling the outputs after the 1 × 1 conv.

**Implementation details.** In the paper, FAFNet has 4 dense blocks and 3 transition layers. The input size of FAFNet is 112 × 112 × 3. Before the first dense block, a 3 × 3 conv layer with 8 output channels is used to process the input image. For 3 × 3 convolutional layers, stride is set as 1 and each side of the inputs is zero-padded by one pixel to keep feature map size unchanged. We set the transition layer between two adjacent dense blocks, doubling the number of output channels after each transition layer. The number of the dense cell in each dense block is \{1, 2, 4, 4\} and for each dense cell, we set \( k = 8, \lambda = 2 \). After the last dense block, we set a 1 × 1 convolutional layer with a linear activation, whose output channels are equal to the number of the labels. Then a global average pooling is applied and followed by a softmax classifier. The structure of FAFNet is shown in Table 1.

**Table 1.** The structure of FAFNet

| Layers             | Output Size | FAFNet                  |
|--------------------|-------------|-------------------------|
| Convolution        | 112 × 112 × 8 | 3 × 3 conv, stride = 1  |
| Pooling            | 56 × 56 × 8  | 2 × 2 max, stride = 2   |
| Dense Block 1      | 56 × 56 × 16 | \[1 × 1 conv_{3 × 3}\] × 1 |
| Transition Layer 1 | 56 × 56 × 16 | 1 × 1 conv, stride = 1  |
|                    | 28 × 28 × 16 | 2 × 2 max, stride = 2   |
| Dense Block 2      | 28 × 28 × 32 | \[1 × 1 conv_{3 × 3}\] × 2 |
| Transition Layer 2 | 28 × 28 × 32 | 1 × 1 conv, stride = 1  |
|                    | 14 × 14 × 32 | 2 × 2 max, stride = 2   |
| Dense Block 3      | 14 × 14 × 64 | \[1 × 1 conv_{3 × 3}\] × 4 |
| Transition Layer 3 | 14 × 14 × 64 | 1 × 1 conv, stride = 1  |
|                    | 7 × 7 × 64   | 2 × 2 max, stride = 2   |
| Dense Block 4      | 7 × 7 × 96   | \[1 × 1 conv_{3 × 3}\] × 4 |
| Convolution        | 7 × 7 × 2    | 1 × 1 conv, stride = 1  |
| Classification     | 1 × 1 × 2    | 7 × 7 global average    |

SoftMax
3. Experiments and results
In this section, we first introduce the datasets and some training details. Then we compare FAFNet with LeNet and AlexNet to verify the performance in precision and speed on the datasets. Finally, we discuss the influence of license plate region, car color and contour binary image on FAFNet model. The experimental environment is Ubuntu 16.04 system, CPU Intel(R) Xeon(R) CPU E5-2603 v4 @ 1.70GHz, GPU NVIDIA GTX 1080Ti. The running speed of different models are tested simultaneously on the embedded device HISI3516 AV200.

3.1. Datasets and preprocess
All the data of our datasets are collected from different security and surveillance cameras. The positive samples cover different license plate recognition scenes during the day and night, as well as scenes under some special circumstances, such as high noise, low light, rain and fog, etc. The negative sample data consists of three parts: (1) some car-free natural scenes, including green belts, road speed bumps and road railings; (2) false-alarm data collected by the camera, including the text of the car body and natural scenes; (3) part of scene text pictures of COCO-Text datasets [11]. In order to further expand the data volume of negative samples and improve the diversity of our datasets, we carried out data augmentation for negative samples, mainly including random rotation, scaling and shearing. Now there are more than 200 thousand images in the datasets, including more than 100 thousand positive samples and more than 90 thousand negative samples. As the input to FAFNet, all images should be resized to $112 \times 112$.

3.2. Training details
The paper selects 90% of the datasets as the training set and 10% as the test set. All the networks are trained using stochastic gradient descent method (SGD). On our datasets, we train using batch 128 for 1000 epochs. The initial learning rate is set to $10^{-3}$ and is lowered by 10 times at epoch 200 and 600. We use a momentum of 0.9 [12] and a weight decay of $5 \times 10^{-4}$ [13]. All the experiments in this paper are trained on Darknet [18].

3.3. Compared with LeNet and AlexNet
In order to demonstrate the performance of our algorithm and the feasibility of its deployment on embedded devices, the results of FAFNet are compared with LeNet [16] and AlexNet [17] on the datasets. The results are shown in Table 2.

Table 2. Results compared with LeNet and AlexNet

| Network | Model Size (MB) | Accuracy (%) | Computation (GFLops) | Speed (ms/image) |
|---------|----------------|--------------|---------------------|-----------------|
|         |                |              |                     | GPU | CPU | AV200 |
| LeNet   | 59.7           | 99.75        | 0.16                | 1.5 | 60  | 250   |
| AlexNet | 114            | 99.98        | 0.479               | 2.2 | 170 | 1000  |
| FAFNet  | 0.11           | 99.99        | 0.024               | 2.7 | 14  | 55    |

Table 2 shows the powerful performance of FAFNet. The model size of FAFNet model is only 0.11 MB, which is 542 times and 1036 times lower than that of LeNet and AlexNet respectively. Meanwhile, the accuracy of FAFNet is slightly higher than AlexNet and LeNet. We also compared the recognition speed on CPU, GPU and AV200. LeNet is fastest on GPU because of its simple structure. But on CPU and AV200, FAFNet is the best, which can realize real-time false-alarm processing on AV200.

On the whole, FAFNet can effectively avoids the redundancy of network parameters and greatly improves the parameter utilization. The FAFNet model is can be applied to the embedded devices because its size and speed perfectly meet the memory requirement and the real-time requirement of embedded devices.
3.4. Discussion
In order to demonstrate the robustness and the major feature of FAFNet model, three experiments are designed to further explore the influence of license plate region, car color and contour binary image. The training data of this experiment is a subset of the datasets mentioned in 3.1. The Figure 5 shows one input example of FAFNet and its license plate filling image, gray image and contour binary image. The experimental results are shown in Table 3.

![Image](a) Color Image (b) LP Filling Image (c) Gray Image (d) Contour Binary Image

**Figure 5.** Input example and its license plate filling image, gray image and contour binary image

**License plate region.** In the paper, we know that the input image of FAFNet is cropped according to license plate coordinates and cropping parameters. Since the input image contains the information of license plate region, so we consider whether the information of license plate region is also one of the main features of FAFNet model. If the information of license plate region has a direct impact on FAFNet model, it will be difficult for FAFNet model to be applied to the real world because of the diversity of the license plate.

In the process of data preprocessing, we fill the region of license plate with pixel zero to shield the region completely and train the network. By comparing the training results before and after filling, we analyze the influence of the license plate region on false-alarm filtering. According to the Table 3, we can see that the accuracy before and after filling is almost the same. Therefore, we can conclude that FAFNet mainly extracts the feature of car head, and the license plate region nearly has no influence on false-alarm filtering.

**The car color.** As we know, the colors of cars are numerous and complicated, which brings a big challenge to the generalization ability of the FAFNet model. In order to verify the influence of the car color, we convert the color image into gray image to block the color information. From Table 3 we can see that the accuracy of gray image is equal to the accuracy of color image. It is obvious that FAFNet has a strong robustness for the color information.

**Contour binary image.** We consider that the main feature extracted by FAFNet is the contour feature of the car head. To demonstrate our guess, the contour binary image is extracted and used as the training data [14, 15]. In Table 3, the training result of contour binary image is slightly lower than the original result. Therefore, we can confirm that the main feature extracted by FAFNet is the contour feature of car head.

| Training Data              | Accuracy (%) | Computation (GFlops) |
|----------------------------|--------------|----------------------|
| Color Image                | 99.98        | 0.024                |
| Gray Image                 | 99.98        | 0.021                |
| Contour Binary Image       | 99.94        | 0.021                |
| License Plate Filling Image| 99.97        | 0.024                |

**Table 3.** Results of the discussion

4. Model internationalization
In the paper, the FAFNet model achieves excellent performance in the task of false-alarm filtering for Chinese license plates. In the same way, FAFNet can also solve the false-alarm problem of license plate in other countries. Here are two solutions: (1) collect the positive and negative sample data for different countries to train the false-alarm filtering model; (2) apply the false-alarm filtering model for
Chinese license plates to other countries if it is possible. Although solution 1 seems simple and effective, it takes amounts of data collection costs and training costs. Therefore, the paper considers solution 2 to achieve model internationalization.

When we applied the FAFNet model of Chinese license plate to other countries, it did not work well. According to our analysis, the main reason is that the license plate size of different countries is different, so when the input image is cropped in accordance with the Cropping Rule mentioned in Section 2.1, the cropped image cannot be available to the current FAFNet model. Therefore, the key to solve the problem is how to crop a reasonable input image for the Chinese FAFNet model according to the license plate coordinates of other countries.

We perform a statistical analysis of license plate size in different countries (excluding special plates, such as double layer plate, etc.) and find that they are nearly identical in height. Therefore, this paper considers calculating an area similar in size to Chinese license plate based on the license plate coordinates of other countries, and then cropping the input image based on this area. We call the area "virtual license plate".

For the license plate coordinates of other countries, the paper set the height as $h_a$, width as $w_a$ and the ratio of $w_a$ to $h_a$ as $\text{ratio}_a$. The height of virtual license plate $h_v$ can be calculated as follows:

$$h_v = \begin{cases} h_a & \text{ratio}_a > 1.5 \\ \theta \times h_a & \text{ratio}_a \leq 1.5 \end{cases}$$  \hspace{1cm} (5)

where the license plate is a single-layer license plate when $\text{ratio}_a > 1.5$ and the virtual license plate height is equal to the actual license plate height. When $\text{ratio}_a \leq 1.5$, the license plate can be considered as a double-layer license plate, whose height is equal to $\theta(\theta < 1)$ times of the actual license plate height. Based on the empirical test, we set $\theta = 0.6$.

The width of virtual license plate $w_v$ can be calculated as follows:

$$w_v = \text{ratio}_{\text{base}} \times h_v$$  \hspace{1cm} (6)

where $\text{ratio}_{\text{base}} = 4$, which is the statistical value of the width-height ratio of the Chinese license plate.

According to the formula (5) and (6), we can calculate the coordinates of virtual license plate and then crop the reasonable input image, which is available for the FAFNet model of Chinese license plate.

Based on solution 2, we propose the virtual license plate to solve the problem of model internationalization. It has the following two advantages compared with solution 1:

1. It reduces the cost of data collection, so there is no need to collect data of each country.
2. It reduces the cost of training FAFNet model, so there is no need to train the corresponding model for each country.

Therefore, we only need to train the FAFNet model of one country, and then make it applicable to other countries through the method of virtual license plate.

5. Conclusion

The paper introduces a powerful false-alarm filtering algorithm for license plate recognition: FAFNet, which is constructed based on deep convolutional neural network. FAFNet is a lightweight neural network and can be trained end to end. In the paper, we first show the design details of FAFNet. Its unique network architecture not only greatly reduces the number of parameters and calculation speed, but also deepens the network depth, enabling the model to achieve a high precision recognition. Then we verify the excellent false-alarm filtering performance of FAFNet and the recognition speed on different devices through the experiments, and recognition speed on different hardware devices (GPU, CPU, AV200). Finally, we propose a solution of virtual license plate to solve the problem of model internationalization, while the validity of the virtual license plate still needs to be verified in the future. And other model internationalization methods is also a worthy of exploration in the future.
Acknowledgments

This work is Supported by Natural Science Foundation of Zhejiang Province (No. LY18F02002).

References

[1] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, I. D. Psoroulas, V. Loumos, and E. Kayafas. License Plate Recognition From Still Images and Video Sequences: A Survey. IEEE Transactions on Intelligent Transportation Systems, vol. 9, no. 3, pp. 377–391, Sep. 2008.

[2] H. Li and C. Shen. Reading Car License Plates Using Deep Convolutional Neural Networks and LSTMs. arXiv:1601.05610 [cs], Jan. 2016, arXiv: 1601.05610.

[3] V. Jain, Z. Sasindran, A. Rajagopal, S. Biswas, H. S. Bharadwaj, and K. R. Ramakrishnan. Deep Automatic License Plate Recognition System. In Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing, ser. ICVGIP ’16. New York, NY, USA: ACM, 2016, pp. 6:1–6:8.

[4] Sergey. Z and Alexey G. License Plate Recognition via Deep Neural Networks. arXiv: 1806.10447 [cs], Jun. 2018, arXiv: 1806.10447.

[5] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. IJCV.

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In CVPR, 2015.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In CVPR, 2016.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR, 2016.

[9] Huang G, Liu Z, Der Maaten L V, et al. Densely Connected Convolutional Networks[J]. computer vision and pattern recognition, 2017: 2261-2269.

[10] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv:1502.03167 [cs], Feb. 2015, arXiv: 1502.03167.

[11] Veit A, Matera T, Neumann L, et al. COCO-Text: Dataset and Benchmark for Text Detection and Recognition in Natural Images[J]. arXiv: Computer Vision and Pattern Recognition, 2016.

[12] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In ICML, 2013.

[13] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[J]. computer vision and pattern recognition, 2016: 770-778.

[14] Canny J F. A Computational Approach to Edge Detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986, 8(6): 679-698.

[15] Otsu N. A Threshold Selection Method from Gray-Level Histograms[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1979, 9(1): 62-66.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.

[17] Krizhevsky A, Sutskever I, Hinton G E, et al. ImageNet Classification with Deep Convolutional Neural Networks[J]. neural information processing systems, 2012, 141(5): 1097-1105.

[18] Redmon J, Divvala S K, Girshick R B, et al. You Only Look Once: Unified, Real-Time Object Detection[J]. computer vision and pattern recognition, 2016: 779-788.