International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.3 (2016), pp.445-456
http://dx.doi.org/10.14257/ijsip.2016.9.3.38

The Research and Design of Vehicle License Plate Recognition System in Traffic Management System

Ronghui Fu

College of Computer Science, Neijiang Normal University, Neijiang, SiChuan, China, 64110
411586637@qq.com

Abstract

In this article, we research and design a new vehicle license plate recognition system in traffic management system, this new system will solve a lot of problems about vehicle. The system include two parts, "Plate Detect" and "Chars Recognize", we did pretreatment in the first part, and using our improved fusion kernel function SVM to detect the plate, and in the second part, we segmented every single char batch, and used the new deep learning model CNN model to implement recognizing alphabets and numbers, this system recognition accuracy rate will over 97%, can be a part of traffic management system in Smart City.

Keywords*: vehicle license plate, Support Vector Machine, kernel function, Convolution Neural Network, Smart City*

1. Introduction

With the rapid development of computer science and technology, more and more industry are using scientific computing to complete the calculation and recognition of their own data. For traffic system, the intelligent traffic system was proposed in 1990s, via 10 years development, this system integrate advanced information processing technology, navigation and location technology, wireless communication technology, automatic control technology, image processing and recognition technology, these advanced new technologies have played a good effect in traffic management system[1]. Not only implement automatic manage traffic and vehicles, but also enhanced traffic safety, reduced the traffic jams and environmental pollution, played a good role in promoting the social development. As the years, traditional traffic management system has a tendency to intelligent traffic management system.

Vehicle license plate recognition is one of the most core technology in intelligent management system[2]. This technology used advanced computer image information processing and patterns recognition technology, to process the video data of traffic intersection and parking. In general, the VLPR system has three parts[3], the license plate extraction, the license plate segmentation and the character recognition. There are a lot of researchers do the research of three parts.

In the field of license plate extraction, some researchers used edge information to extract the needed region of license plate[4-6], they used the Sobel features and Canny features to get the most important information, others used global image information[7,8] to get the whole license plate for the following research. After the color image appeared, some algorithms of license plate extraction are using texture and color features[9-11], these information can be regarded as auxiliary information for license plate extraction.

In the field of license plate segmentation, some researchers used pixel connectivity to extract the local information that can get the single character image[12]. Others turn to the field of projection profiles and character contours[13-14]. The best effect of license plate

ISSN: 2005-4254 IJSIP

segmentation is the combined features[15,16]. These features can extract the single information image in an efficient way.

In the field of character recognition, some researchers using the traditional statistical learning algorithms[17], such as LDA, KNN. Others used the up to date algorithms for the deep learning algorithms[18], but this method is still immature. In this article, we will propose the deep learning algorithm for character recognition.

In the processed results, the VLPR technology can extract the plate information for the corresponding vehicle. This will give the basis for the vehicle's traffic management, charge, scheduling and statistics. Judging from the present condition, there still exist many inconformity of vehicle and license plate, in order to solve these problems, some methods using video camera to record the wrong situations, and adopt manual work to recognition the wrong situations, this method needs many manual auxiliary work to complete recognition, the cost is larger. So, the VLPR can perfect solve these situations, if we can get high accuracy rate license plate, we can using them to implement many intelligent traffic management system without manual service, such as public security law enforcement system, Highway automatic fare collection system, city road monitor system and intelligent parking lots management system. This strategy will simplify the manual labor, and eliminate man-made jamming, reduce the error opportunity. Against the RFID technology and bar code technology, VLPR have the convenient that no need to install device on the vehicle, and the technology based on the image processing recognition will be more convenient to implement the license plate recognition.

The VLPR system has a lot of significance:

(1) promote the traffic management level

(2) promote the traffic management efficiency

(3) in favor of vehicle anti-theft and recovery

(4) make automatic manage vehicle possible

In this article, we give a research and design of vehicle license plate recognition in intelligent management system. This system is a important part of intelligent traffic management system that can recognition the license plate number from a vehicle image in real-time, and the recognition procedure is automatic in full-time.

2. Overview Method

In this chapter, we give the overview method of our VLPR system. Our VLPR system have two main part, the "Plate Detect" and the "Chars Recognize" are described in Figure 1, and (a) is the procedure of "Plate Detect" (b) is the procedure of "Chars Recognize" respectively.

The "Plate Detect" modeling has three steps:

(1) The system read an image with vehicle, and using basic image processing technology to cut out some image batches, and these batches need including the license plate. In this step, we handle the distortion and rotate of the license plate image batches.

(2) There is a trained SVM model before, we cut out many images which is a license plate image batch or not, and then marked everyone to train a SVM to classify if a image batch is license plate or not.

(3) The batches we extract in step (1) are test data for the SVM model, we put every image batch in SVM model to get the classify result, retain the license plate image batch and discard the others.

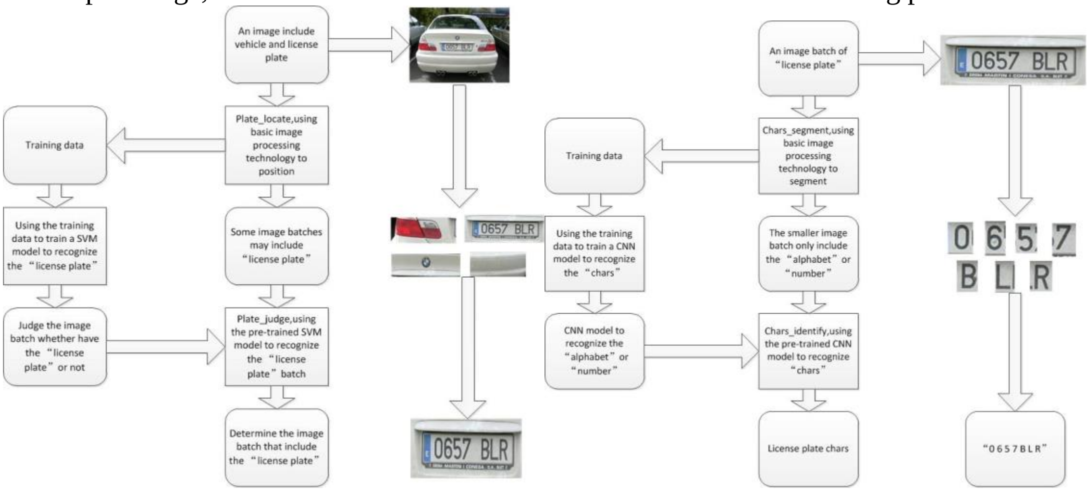
After three steps, we can extract the image batches which are including the license plate to do the next processing.

The "Chars Recognize" modeling has three steps:

(1) After get the license plate image batch, first, we using basic image processing technology in the batches, and segment the chars of the license plate, get the single char respectively.

(2) Under normal circumstance, the chars in a license plate often include 26 alphabets and 10 numbers. We using Convolution Neural Network to recognize the license plate chars, we train the 26 alphabets and 10 numbers for a CNN, and the CNN have better performance to recognize the non-rigid object than traditional ANN, so we use CNN to train a model for chars recognition.

(3) After chars segmentation and a trained CNN model, we need recognize the corresponding chars by CNN model. After chars recognition, we can get the chars from an input image, then we can store the chars into database for the following procedure.



(a)Plate Detect (b)Chars Recognize

Figure 1. The Overview of Our method

The VLPR system using machine learning algorithm to train the SVM and CNN model, so this procedure need major train data to support the model training. As we know, common machine learning algorithm need mass data to learn more robust feature, the robust feature make a contribution to the recognition rate. In addition, mass data will avoid the over-fit of neural network. So, in this article, we use the popular datasets of "ImageNet"[19] in our VLPR system, most of the Image Recognition system also using this datasets for their training, in this datasets, there are a lot of data for research. Our system aim at license plate recognition, so we extract 10000 common vehicle images for training, each images have the vehicle and the corresponding license plate. In common situation, CNN or SVM model need the input images maintain the same size, so we cut out each vehicle image to the size of 224 * 224 with Height * Width. In the Figure 2, we exhibit some vehicle and license plate samples.

Figure 2. Some Vehicle Image Extracted from ImageNet

3. The Detail Algorithm for "Plate Detect"

The procedure of "Plate Detect" has two processes including the pretreatment process and the SVM recognition process. We will introduction the detail of these two processes respectively.

3.1. Pretreatment process

For a new vehicle image, the scale and the pixels are different for everyone. The common method first using Gaussian Blur to fuzzy the images to make any image get the similar scale and pixels[20]. Then, we use the graying method to put the RGB image into Gray image, reducing the channel numbers will make the image robust for processing. After graying process, the image only have one channel to process, in order to detect the location of license plate, we need using the Sobel operator to get the edge of the whole vehicle image. If we want to get the edge of license plate, first we will get the essence feature of the plate, the edge feature is the most discrimination feature, so we use the fastest method Sobel operator to get the whole edge.

As we all know, any plate have the basis edge feature is that it has bounding rectangle. So, after get the whole edge, we calculate any edges which have the bounding rectangle, then we will classify these "Plate region" into two classes, the right "Plate" and the wrong "Plate", then we will mark every region, and put these "Plate region" into SVM for the training data.

Of course, before putting these regions into SVM, we will eliminate the distortion and rotation of these regions. The rotation operator is a key operator for the following process. We will first get the rotate angle of the bounding rectangle, and then rotate the rectangle to the horizontal direction. After rotation, we can put these region data into SVM for training. Figure 3 shows the technological process and the corresponding results.

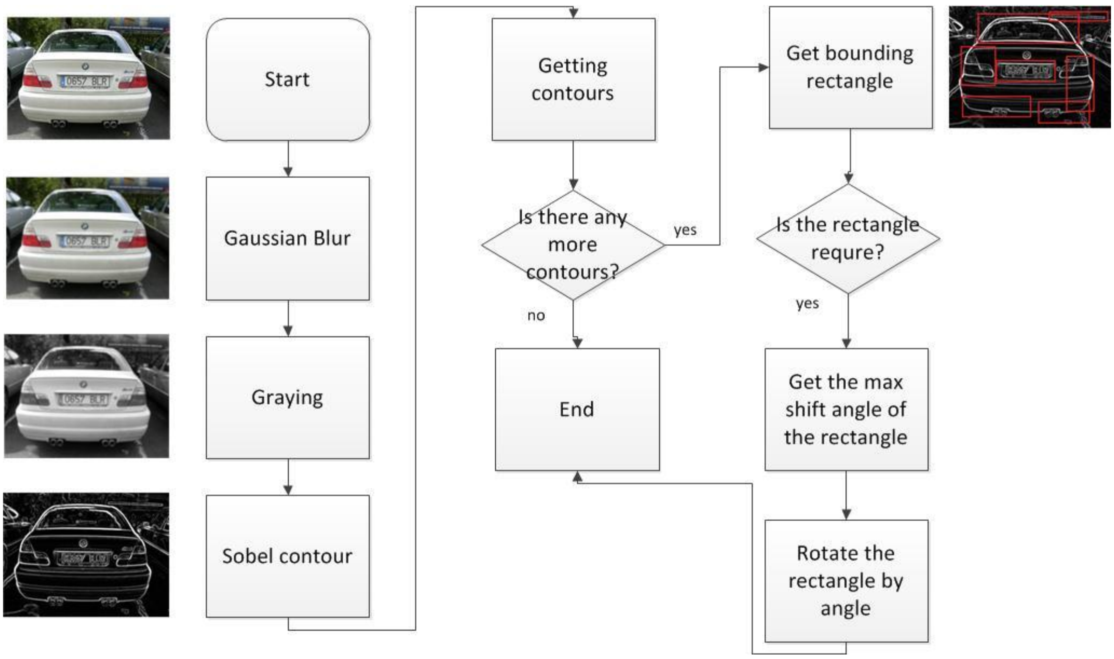


Figure 3. The Procedure of Pretreatment and the Corresponding Results

3.2. Train SVM Model and Recognize the License Plate

After pretreatment process, we need to put the training data into SVM to train a model. From now on, we have two class images, the one is "License Plate", the other is not, we marked the corresponding image batches for their label.

3.2.1. Support Vector Machine: Support Vector Machine is developed by the Perceptron[21]. By calculate the linear classification hyper-planes to distinguish the two class data. In Perceotron, we will get some orders to distinguish our data, but the SVM technology is going to make the optimal hyper-plane. In this step, we can use the SVM algorithm to complete the binary classification problem.

Assume that a training dataset {x_i,y_i, i=1,2,3...n}, the data "x" express the training data, and the data "y" express the training label. SVM will find a optimal hyper-plane "w*x+b=0" to as many as possible distinguish the data "x". In the hyper-plane "w*x+b=0", "w" means weight, "x" means support vector, b means bias. In order to solve out the optimal hyper-plane, we will solve the quadratic program:

$$\min_{\omega,b} \frac{1}{2}\|\omega\|^2 \tag{4-1}$$

$$s.t. y_i(\omega * x_i + b) \ge 1, i = 1, 2, ..., n \tag{4-2}$$

3.2.2. Our new Method for Kernel Function Choosing: As we know, this linear optimal hyper-plane will distinguish the two linear separable data, if the data is not linear separable, this scheme doesn't work. Like Figure 4 shows, the linear inseparable data need a curve hyper-plane to distinguished. In SVM, these problems used the kernel function to transform linear inseparable data to linear separable data.

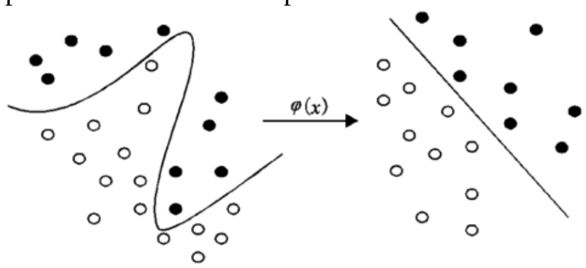


Figure 4. Non-linear Classifying Samples Convert to Linear Classifying Samples

We used φ_x express the kernel function, by the change of the kernel function. We can convert the linear inseparable model to linear separable model, and the changed quadratic program:

$$\min_{\omega,b} \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{m}\xi_i \tag{4-3}$$

$$s.t. y_i(\omega * \varphi(x_i) + b) \ge 1 - \xi_i \tag{4-4}$$

$$\xi_i \ge 0, i = 1, 2, 3, ..., n \tag{4-5}$$

where ξ_i means relaxing factor, φ_x means kernel function, C means penalty coefficient. Solving this quadratic program we can use the Lagrange multiplier and KKT-condition, convert the primal problem to a linear optimization problem, the results show:

$$f(x) = \text{sgn}[\sum_{i,j=1}^{m} a_i y_i K(x_i, x_j) + b] \tag{4-6}$$

We can choose some kernel function for $K(x_i, x_j)$ to complete the recognition task for training.

There are four common kernel can be used in the SVM[22]:

(1)Linear Kernel Function: $$f(x)=\mathrm{sgn}[\sum_{i=1}^{n}a_i y_i (x_i * x)+b]$$

(2)Polynomial Kernel Function: $$f(x)=\mathrm{sgn}\{\sum_{i=1}^{n}a_i y_i [(x_i * x)+1]^d +b\}$$

(3)Radial basis Kernel Function： $$f(x)=\mathrm{sgn}\{\sum_{i=1}^{n}a_i \exp\{-\frac{x_i - x}{\sigma^2}\}+b\}$$

(4)Sigmoid Kernel Function： $$f(x)=\mathrm{sgn}\{\sum_{i=1}^{n}a_i \tanh[\gamma *(x_i * x)+\theta]+b\}$$

These kernel functions have two characteristic, the locality characteristic (polynomial kernel function) and the whole characteristic (Radial kernel function). For the recognition of vehicle license plate, there are not only has the locality characteristic, but also have the whole characteristic. So, in this article, we propose a new kernel strategy: fusion kernel function, we use the weight ρ to express the relationship between the whole kernel function and the locality kernel function:

$$K_{mix}=\rho K_{RBF}+(1-\rho)K_{poly} \tag{4-7}$$

where K_{RBF} and K_{poly} express the whole kernel function and the locality kernel function respectively, and weight ρ to control the relationship.

For the weight ρ , we do a lot experiments on the SVM of vehicle license plate recognition, aim at plate recognition, we choose a lot plate images and other rectangle but non-plate images for training and recognition. The results show that if someone want to use the fusion function in the SVM to complete the recognition of vehicle license plate, the fusion weight ρ =0.619. This means the vehicle license plate recognition with the help of RBF kernel function more.

By the trained model, we can input all region batches image into the SVM model to get the distinguish task result. The result will tell the label of any region batches, and retain the "License Plate", discard the non-"License Plate". Then, the "Plate Detect" will be completed. We can select the "License Plate" region from a vehicle image, Figure 6 shows some result of "Plate Detect".

3.3. The results of "Plate Detect"

We give the fusion kernel function SVM a lot "Plate" samples and non-"Plate" samples, then lead SVM to train the recognized model. From a single image, we extract 100 candidate image batches as the training data for the SVM. The vehicle image choose from ImageNet have a lot of material data that we can train our model well, in the following Figure 5 we give the results of "Plate Detect" of samples in Figure 2.

Figure 5. The Corresponding Plate Detect Results of Figure 2

4. The Detail Algorithm for "Chars Recognize"

After we select the "License Plate" region, the next procedure is "Chars Recognize" procedure. This step also have two steps, first we must segment the "License Plate" region batch to a series single "Char" batch. And then, we can using the pre-trained Convolution Neural Network to recognize the corresponding alphabets and numbers.

4.1. Segment the "License Plate" Region Batch

The segmentation step is described in Figure 6, from it we can see that the step is similar than the section 3.1. First, we will graying the RGB license plate image, after graying we can calculate the R/G/B pixel numbers to compare which color is maximum, then the license plate color is this color. After determine the color of the license plate, we binary the gray image into binary image. Then, find the contours in the image, as we know, the plate number is separate, so we can use the contours information to distinguish every char. After get contours, we also use the bounding rectangle to get the separate "alphabet" or "number". We can use the bounding rectangle to segment the corresponding "Chars". These already segmented "chars" will put in the pre-trained CNN model to recognize the corresponding "alphabet" and "number". In Figure 6, we also give the results of segmenting "License Plate" region batch.

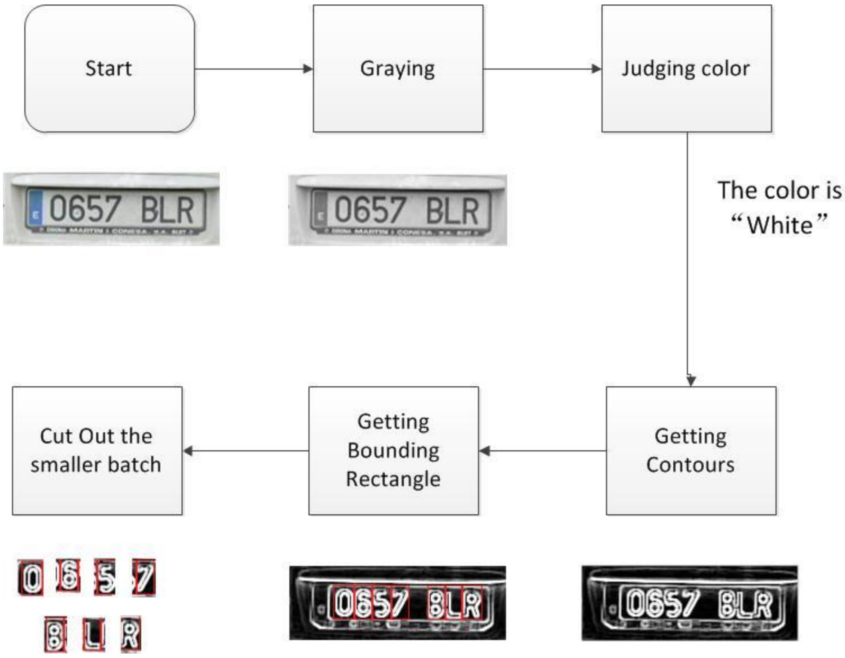


Figure 6. The Procedure of "Plate Chars" Segmentation and the Corresponding Results

4.2. Recognize the "Chars" by Pre-trained CNN Model

In this article, we design a vehicle license plate recognition system for traffic management system, we collect our plate dataset in ImageNet, so our samples aim at only "alphabet' and "number". We propose a new neural network in deep learning-----Convolution Neural Network(CNN). The Convolution operations and the Pooling operations are the key technologies in our new CNN architecture, we will introduce them respectively.

4.2.1. CNN Architecture: CNN is a progress from Artificial Neural Network(ANN). The traditional ANN include a input layer, a hidden layer and a output layer, CNN adds many hidden layers based on this architecture, multi-hidden layer will enhance the mass of the image feature. In the article, we will train 26 alphabets and 10 numbers for recognition. So, we adopt the architecture of five convolution hidden layers and three full connected layers. We trained our CNN model on a two cores GPU computer, so it has the high performance for recognition. From the input images, we will get 4096-dimensions feature, and using the softmax function to give the recognized results, the results will include 36 classes for the alphabet and the number recognition. In the Figure 7, we give the actual architecture CNN and their training mode on the two cores GPU.

In the Figure 7, we can find that the CNN model architecture is made up with input layer, five convolution layer, three full connected layer and the output layer. The input image we normalization to the size of 224*224 and the output feature is 4096-dimensions. We used the pooling operator in the first, second and fifth layer, this operation can help the CNN model reducing the scale of the parameters. And in order to speed up the training and recognition procedure, we put the parallel computing in two GPUs, including the Convolution operations and the Pooling operations. Besides, the parallel computing need data exchange, so we exchange the two GPUs' data between the second and third convolution layer. Above all, these operations will reduce the time and computational complexity.

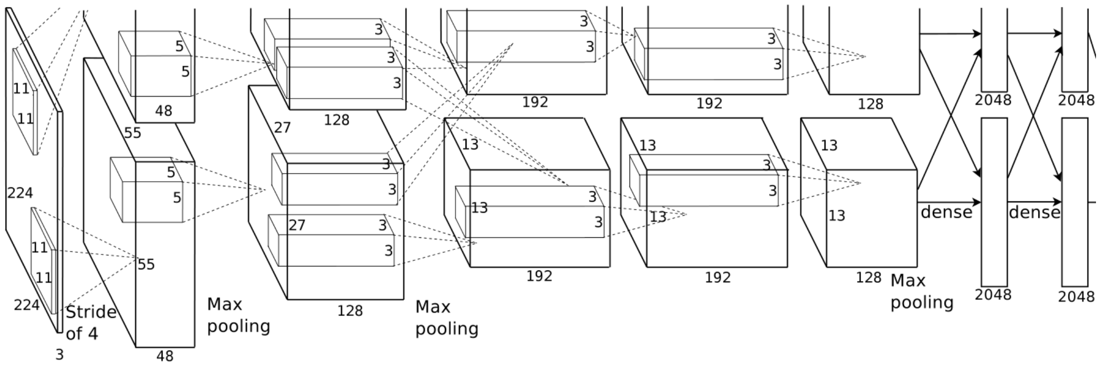
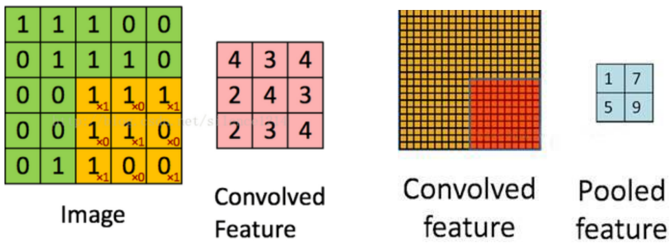


Figure 7. The Architecture of Convolution Neural Network Model of in our Article

4.2.2. Convolution Operations: Convolution operation is the most important operation in the Convolution Neural Network[23], this operation using the convolution kernel to recombine the extracted feature before, and it will help to obtain more complicated feature. As we all know, in an image, the complicated image feature can be combined by some simple features. In the CNN model, the convolution operation will make the feature more complicated, so it will express more useful in the task of classification. In the common situations, the image is stored by two-dimensional discrete data, and we assume that the image is f(x, y), and the convolution kernel is g(x, y). Then we can define the convolution operation:

$$f(x,y)*g(x,y)=\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} f(m,n)g(x-m,y-n)x=0,1,2...M-1,y=0,1,2...,N-1 \quad (5\text{-}1)$$

We will use the formula (4-1) to complete the operations of convolution, and the computational domain is sliding window, so we can using sliding window to compute the whole image for convolution features. In the Figure 8, we show that the procedure from image to convolved feature.



(a)convolution (b)pooling

Figure 8. The Convolution Operations and the Pooling Operations

From Figure 8, the convolution kernel is g(x, y) = [1,0,1;0,1,0;1,0,1], in the training procedure, in every convolution layer there are many convolution kernel to compute the convolved feature. With the purpose of improving the recognition rate of alphabet and numbers, we use many convolution kernels to get more and abroad feature combination. For instance, we use 256 convolution kernels in the first convolution layer, and the second convolution layer will produce 256 convolved feature that provided more feature to the following layer to use.

4.2.3. Pooling Operations: The share of convolution kernels will effectively reduce the parameters in the training stage, but the actual situation is that with the number of the convolution layer increased, the corresponding CNN model parameters growth by exponential. For example, in the fifth convolution layer, there are 1 million parameters in the CNN model, too many redundant parameters not only increased the time of training, but also cause the problem of over-fit[24], in the last, this problem will reduce the recognition rate of the "Chars". So, in this section, we propose a new operation "Pooling" to reduce a lot useless parameters.

Pooling operation is a statistical operation[25], it is by computing the statistical feature in the region to represent this region, this will reduce the number of parameters in the region. Because of the static image have the character of aggregation. The features in this region will play a role in the near regions. So with the regards of the superfluous parameters of the neural network, we can aggregate the whole parameters into tiny parameters, and this tiny parameters will represent the whole region.

The pooling operations have two way, the mean pooling and max pooling. The mean pooling is computing the mean of the region parameters to represent the whole region, and the max pooling is computing the maximum parameters. As well as the convolution operation, the pooling operation is also doing on the sliding window, and we can see the pooling operation in the Figure 8. We can see from the figure, the pooling operation computes the mean value to reduce the 10*10 region into a 1*1 region, this will utmost extent reduce the number of parameters. In our CNN model, we used mean pooling in whole layers, and reducing the parameters from 1 million to 6 hundred thousand, and we do a lot experiments that this reducing parameters will do not occur the problem of over-fit.

4.3. The Results of "Chars Recognize"

We used the 26 alphabets and 10 numbers 500 samples crawling from the Internet, and put these samples into CNN model to train a model that can recognize alphabets and numbers, in the CNN model, we used the convolution operations to increase the complex of the feature, and the pooling operations to reduce the number of parameters in the CNN model.

After pre-trained an alphabets and numbers CNN model, we normalize the segmented "Chars" region batch image into CNN model, and then recognized the corresponding alphabets or numbers, we do a lot experiments on the data extracted from ImageNet, and the Figure 9 give the results of "Chars Recognize" of the plate image batch from Figure 5.

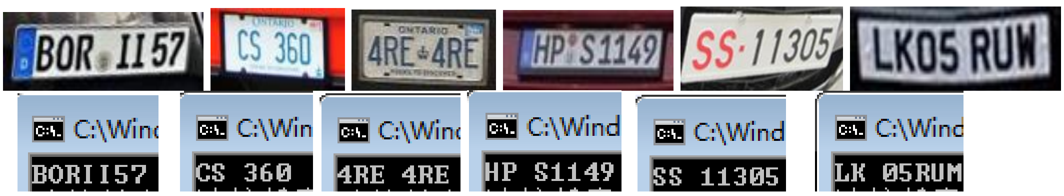


Figure 9. The Results of "Chars Recognize" of the Plate Image in Figure 5

5. Conclusion

In this article, we research and design a vehicle license plate recognition system to help perfect the traffic management system. As we all know, the intelligent traffic management system is one of the most important part of Smart City.

The vehicle license plate recognition system is composed by two key procedures, the "Plate Detect" procedure and the "Chars Recognize" procedure. For each procedure, we give the detail of the design by words and figures.

In the procedure of "Plate Detect", we used pretreatment process to extract candidate plate region, and use improved fusion kernel function SVM to complete recognizing "Plate" region from all candidate regions. The fusion kernel function between RBF kernel and Polynomial kernel is our innovative point. This will improve the recognition rate about 7%.

In the procedure of "Chars Recognize", we used chars segmentation process to extract candidate chars region, and use pre-trained CNN model to complete "alphabets" and "numbers" recognition. The Convolution Neural Network is a new deep learning model of recognition, we used the improved convolution operations to get more complicated feature, and pooling operations to reduce parameters to avoid the problem of over-fit, this two optimizing operations will let our character recognition accuracy rate over 97%.

Our vehicle license plate recognition system will solve a lot problems about vehicles in Smart City, and in order to build a perfect traffic management system in Smart City, we will do more research and design..

References

[1] Q.-J. Kong, L. Li and B. Yan, “Developing Parallel Control and Management for Urban Traffic Systems”, IEEE intelligent systems, vol. 28, no. 3, **(2013)**, pp. 66-69.

[2] A. H. Ashtari, M. J. Nordin and M. Fathy, “An Iranian License Plate Recognition System Based on Color Features”, IEEE transactions on intelligent transportation systems, vol. 15, no. 4, **(2014)**, pp. 1690-1705.

[3] G. Liu, Z. Ma, Z. Du and C. Wen, “The calculation method of road travel time based on license plate recognition technology”, Proc. Adv.Inform. Tech. Educ. Commun. Comput. Inform. Sci., **(2011)**.

[4] M. Sarfraz, M. J. Ahmed and S. A. Ghazi, “Saudi Arabian license plate recognition system”, Proc. Int. Conf. Geom. Model. Graph., **(2003)**.

[5] D. Zheng, Y. Zhao and J. Wang, “An efficient method of license plate location”, Pattern Recognit. Lett., vol. 26, no. 15, **(2005)**, pp. 2431–2438.

[6] H.-J. Lee, S.-Y. Chen and S.-Z. Wang, “Extraction and recognition of license plates of motorcycles and vehicles on highways”, Proc. Int. Conf. Pattern Recognit., **(2004)**.

[7] M. M. I. Chacon and S. A. Zimmerman, “License plate location based on a dynamic PCNN scheme”, Proc. Int. Joint Conf. Neural Netw., **(2003)**.

[8] K. Miyamoto, K. Nagano, M. Tamagawa, I. Fujita and M. Yamamoto, “Vehicle license-plate recognition by image analysis”, Proc. Int.Conf. Ind. Electron. Control Instrum., **(1991)**.

[9] R. Zunino and S. Rovetta, “Vector quantization for license-plate location and image coding”, IEEE Trans. Ind. Electron., vol. 47, no. 1, **(2000)**, pp. 159–167.

[10] Y.-R. Wang, W.-H. Lin and S.-J. Horng, “A sliding window techniquefor efficient license plate localization based on discrete wavelet transform”, Expert Syst. Applicat., vol. 38, **(2010)**, pp. 3142–3146.

[11] X. Shi, W. Zhao and Y. Shen, “Automatic license plate recognition system based on color image processing”, Lecture Notes Comput. Sci., vol. 3483, **(2005)**, pp. 1159–1168.

[12] B.-F. Wu, S.-P. Lin and C.-C. Chiu, “Extracting characters from real vehicle license plates out-of-doors”, IET Comput. Vision, vol. 1, no. 1, **(2007)**, pp. 2–10.

[13] C. A. Rahman, W. Badawy and A. Radmanesh, “A real time vehicle’s license plate recognition system”, Proc. IEEE Conf. Adv. Video Signal Based Surveillance, **(2003)**.

[14] A. Capar and M. Gokmen, “Concurrent segmentation and recognition with shape-driven fast marching methods”, Proc. Int. Conf. Pattern Recognit., **(2006)**.

[15] S. Nomura, K. Yamanaka, O. Katai, H. Kawakami and T. Shiose, “A novel adaptive morphological approach for degraded character image segmentation”, Pattern Recognit., vol. 38, no. 11, **(2005)**, pp. 1961–1975.

[16] D.-J. Kang, “Dynamic programming-based method for extraction of license plate numbers of speeding vehicle on the highway”, Int. Journal of Automotive Tech., vol. 10, no. 2, **(2009)**, pp. 205–210.

[17] Y. Dia, N. Zheng, X. Zhang and G. Xuan, "Automatic recognition of province name on the license plate of moving vehicle", Proc. Int.Conf. Pattern Recognit., **(1988)**.
[18] J. Jiao, Q. Ye and Q. Huang, "A configurable method for multistyle license plate recognition", Pattern Recognit., vol. 42, no. 3, **(2009)**, pp. 358–369.
[19] Q. Zhang, H. Yang and Y. Wei, "Selection of Destination Ports of Inland-Port-Transferring RHCTS Based on Sea-Rail Combined Container Transportation", Innovation and sustainability of modern railway: Third international symposium on Innovation and sustainability of modern railway (ISMR), **(2012)**; Nanchang, China.
[20] E. Pasolli, F. Melgani and D. Tuia, "SVM Active Learning Approach for Image Classification Using Spatial Information", IEEE Transactions on Geoscience and Remote Sensing, vol. 52, no. 4, **(2014)**, pp. 2217-2233.
[21] L.-P. Wang, L.-P. Zhang and M. Zhao, "An algorithm for moving target detection in IR image based on grayscale distribution and kernel function", Journal of Central South University, vol. 21, no. 11, **(2014)**, pp. 4270-4278.
[22] C. Arteaga and I. Marrero, "Approximation in weighted p-mean by RBF networks of Delsarte translates", Journal of Mathematical Analysis and Applications, vol. 414, no. 1, **(2014)**, pp. 450-460.
[23] K. Fergani, D. Lui and C. Scharfenberger, "Hybrid structural and texture distinctiveness vector field convolution for region segmentation", Computer vision and image understanding: CVIU, **(2014)**, pp. 125:85-96.
[24] E. Claxton, "How to prevent prolapse in overfit pregnant ewes", Farmers Weekly, vol. 161, no. 2, **(2014)**, pp. 31.
[25] J. Guo, K. Wang and C.-W. Weng, "Pooling semilattices and non-adaptive pooling designs", Discrete mathematics, vol. 320, **(2014)**, pp. 64-72.

Author

Ronghui Fu, Master degree in 2008, she is now a lecturer in computer science at Neijiang Normal University. Her research interests include Software engineering,Databse and Computer application,mainly in the field of software theory and technology.