Vehicle License Plate Recognition

Can He
Department of Electrical and Electronic Engineering
Southern of University Science and Technology
Shenzhen, China
11711009@mail.sustech.edu.cn

***Abstract*—With the development of economy, the number of vehicles is increasing, and the phenomenon of violating traffic regulations will also increase, which brings greater pressure on the traffic police to enforce the law. In order to reduce the law enforcement pressure of traffic departments, we use license plate recognition technology instead of manual recognition to reduce the burden on traffic police officers. License plate recognition technology can be divided into four parts: license plate location, license plate segmentation, character segmentation and character recognition. Among them, the location and segmentation of the license plate directly determine whether the final license plate recognition is successful or not. In recent years, in the latest license plate recognition technology, in order to improve the accuracy of recognition, most of people use the neural network technology to achieve license plate recognition. In this project, due to time and technical issues, the character recognition method I adopted was template matching.**

Keywords—License Plate Recognition, license plate location, character segmentation, character recognition

I. INTRODUCTION

Nowadays, the license plate recognition system has been put into use in the parking lot and the entrance and exit of the community. In the realization of license plate recognition, the location of license plate, character segmentation and recognition occupy the main part of license plate recognition system. For the location of license plate, in order to facilitate the processing, we will generally take the color picture into gray image processing. There are several ways to transform grayscale images, including the average value method of proportional transformation of three colors of RGB, the maximum (minimum) value method of the maximum (minimum) value of the value of RGB and the weighted average method based on the difference of people's sensitivity to color. Among them, the maximum (minimum) value method in the license plate recognition process is not good. The average value method and weighted average method do not have a great difference on the recognition effect. Therefore, in the whole project, we adopted a relatively simple average method to realize the graying of pictures. In order to better promote the success of license plate positioning, we do a Gaussian filter for the gray image. This can not only have the effect of removing noise, but also can blur out some unnecessary objects such as shadows in the image or other non-essential objects affecting positioning.

For the edge detection of gray image, the first derivative and the second derivative are used to detect the brightness discontinuity to determine the image boundary. Detection operators include Sobel, Prewitt, Robert, Canny and other operators. Among them, The Sobel operator does not strictly distinguish the subject and background of the image, so sometimes the extracted image contour is not very good. Prewitt operator performs pixel averaging, which can suppress noise. However, pixel averaging is equivalent to low-pass filtering of the image, so the edge positioning is not very accurate. Although Robert operator is sensitive to noise, its edge positioning is correct, even if the edge is not very smooth. Canny operator is a powerful edge detection operator. However, compared with several other simple operators, it is more difficult to implement, complexity is larger and its final recognition result has little difference compared with other operators. Therefore, after comprehensive consideration, Robert operator is adopted as the edge detection operator. Although it is sensitive to noise, we will carry out image denoising before image processing, and Robert's simple principle and convenient implementation are also one of the reasons I chose.

There are many unrelated structures in the image after edge detection, so we need to etch the image to reduce the unrelated structures. We use “imerode” in MATLAB to corrode images, which requires input of an image data and a frame, in the image, the output value of the pixel whose background and struct completely coincide is 1, and the output value of pixels that do not coincide perfectly and that do not coincide completely is 0, and finally the function returns the image data eroded by this frame. The image structure after corrosion is mostly distributed and incoherent. In order to facilitate the confirmation of the license plate position, the image needs to be smoothed. We use the close operation to smooth the license plate and reduce noise. The close operation can be understood as expansion followed by corrosion. In the image after processing, in addition to the structure of the license plate part, there will also be some small objects. In order to better position the license plate, we can remove some small objects.

After the above processing, the original color image has been a binary image with the license plate as the main structure. We can scan the binary image with clear main structure to determine the location of the license plate. The main ideas are as follows:

i. Obtain the row number y and column number x of the image matrix;

ii. Create a matrix BLUE_y with row 1 column of y;

iii. Traverse each pixel of the binary image and record the number of pixels with a row value of 1 in the previously created matrix BLUE_y;

iv. Find the element with the largest value in the BLUE_y matrix and the row corresponding to it is the row with the most white pixels in the binary image, which can be considered as the row near the center of the license plate;

v. This line as a starting point, respectively up progressive scan down. When the number of white pixels in the scanned row is greater than the threshold, continue scanning upward (or downward) until the number of white pixels in a row is less than the threshold, stop scanning, and record the number of lines in this row, which is the upper (or lower) boundary of the license plate;

vi. Similarly, we can use the same method to determine the license plate left and right boundary.

vii. The license plate is cut from the original color image by the four boundary values obtained from the above positioning

For the next better segmentation character, we will divide the color license plate image into binary image. In order to reduce the interference elements in the image, we perform mean filtering to reduce the noise in the image and smooth the image. Due to the differences of different original images, the characters may be discontinuous or connected after processing here. At this time, we need to conduct corrosion or expansion processing on the filtered image again. In this project, we use the judgment structure to determine corrosion or expansion based on the area of the white part in the figure.

Since the plate frame and rivets will affect the character recognition effect, we need to remove the plate frame and rivets. In practice, for the upper and lower borders and rivets, the starting and ending lines of characters are located by counting the number of pixel jumps in the horizontal direction of the license plate. The line of rivet has six pixels jump (2 rivets 4 times + Left and right border 2 times), the line of the characters has at least 16 times jump (seven characters at least 14 times + Left and right border 2 times). We can scan up 1/3 of the row and down 2/3 of the row, set certain jump threshold, and determine the character of the starting line and the end of the line; For the left and right borders, it is theoretically feasible to remove the left and right borders in the same way, but in practice, different images may produce different removal effects, and incomplete removal may occur, affecting the subsequent character segmentation. Therefore, I decided to remove only the top and bottom borders, not the left and right, and instead split the overlay to achieve a similar effect.

After cutting, we can separate the characters in the image. The main ideas are as follows:

i. From left to right to scan images per column, and sum the values of all pixels in each column;

ii. If the sum is not zero (the column that makes up the character), the next column is scanned to the right until the sum of the columns is zero (the black space between the two white characters);

iii. The first character is between the left-most part of the image and the column, cut it out;

iv. Assign all pixels in the region to 0 (black out) and repeat until all characters are separated.

For character recognition, there are mainly template matching and neural network recognition. In this project, template matching is used. The main principle is that the normalized binary character image is matched with the binary character image in the template library, and the similarity degree is used to calculate the matching degree between license plate characters and each template character. Finally, the most similar image is the matching result.

II. Review

In recent years, the automobile license plate recognition technology has developed rapidly, which can be divided into direct recognition and indirect recognition. Indirect recognition is based on IC card recognition or bar code recognition. Instead, direct recognition is based on the image of the license plate recognition [4].

Indirect recognition is to identify the license plate and related information by identifying the license plate information stored in the IC card or bar code installed on the car. IC card technology identification accuracy is high, reliable operation, can work all day. However, the device is expensive and the hardware identification is very complex, so this technology cannot adapt to work in different places. Bar code technology has the advantages of fast identification, high accuracy, high reliability, and low cost, but it requires a high demand for the scanner. In addition, both require a uniform national standard, and there is no way to check whether the car and the bar code match. These are technical disadvantages, which makes it difficult to promote.

There are mainly three methods of direct recognition, including image processing technology, traditional pattern recognition technology and artificial neural network technology [4].

First of all, the use of image processing technology to solve the problem of license plate recognition began in the 1980s.The process involves using a camera to take a picture of the car in front of a computer for simple processing, which ultimately requires human intervention.

Secondly, Traditional pattern recognition techniques include structural feature method, statistical feature method, etc. In the 1990s, due to the development of computer vision technology, license plate recognition technology has been systematically studied. In 1990, Johnson et al. realized the automatic identification system of vehicle license plates by using computer vision technology and image processing technology [1]. The system can be divided into four parts: image segmentation, feature extraction, template construction and character recognition. It uses different histogram of different threshold values to determine the threshold range of image histogram of license plate position through a large number of statistical experiments, so as to separate license plates according to the histogram of specific threshold value pairs, and then uses the pre-set standard character template for pattern matching to identify characters.

In recent years, with the development of computer equipment and technology and the rise of neural network, people use neural network for automatic license plate recognition. But as early as 1994, M. M. M. Fahmy et al. successfully applied the method of BAM neural network to automatically recognize the characters on the license plate [2]. BAM neural network is a bi-directional joint monolayer network composed of the same neurons. Each character template corresponds to this unique BAM matrix, and the correct license plate number can be recognized by comparing with the characters on the license plate. At present, the commonly used neural networks mainly include BP neural network, Hopfield neural network, etc. Due to the complexity of neural network, there is no optimal way to choose which type of network is selected, which is mainly determined by the type and quantity of samples classified for neural network. Out of consideration for motor vehicle license plate characteristics, people mainly adopt the BP neural network for motor vehicle license plate for training and recognition. BP neural network adopts error back propagation learning algorithm, which can transform the input and output problems of a group of samples into a nonlinear optimization problem, and can approximate realize the arbitrary continuous nonlinear mapping from input to output. For

example, Y. P. Feng et al. used BP neural network to identify tilted license plate photos [3]. By recognizing 100 groups of vehicle videos and images, the method achieved a license plate recognition rate of more than 99%.

In recent years, with the improvement of license plate recognition system technology, the accuracy is getting higher and higher, it began to be applied to people's lives. Auto license plate recognition technology can be applied to traffic flow monitoring, traffic accident site investigation, traffic violation automatic record, expressway automatic toll collection system, parking lot automatic safety management, intelligent park management and other aspects. Specifically, for the management of parking lots, license plate recognition is mainly used for intelligent parking arrangements and fees for taking photos of vehicles entering and leaving. Automatic license plate recognition plays an important role in modern traffic monitoring and management system.

III. THE PROPOSED ALGORITHM

Since I don't particularly understand the role of each layer of neural network, I will use the traditional pattern recognition method to conduct experiments in this project. The license plate recognition system mainly consists of the following parts [5]:

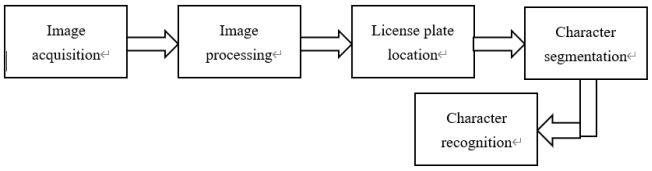

FIG.1: Design flow chart

Image preprocessing and license plate location:

```
function [OutputImage] = primary_location(InputImage)
    GrayImage=rgb2gray(InputImage);  %convert to grayimage
    H = fspecial('gaussian',5,3);  % 高斯模糊
    blurred = imfilter(GrayImage,H,'replicate');
    Gray = edge(blurred, 'roberts','vertical');
    operater1 = [1 ; 1 ; 1];
    pictureErode = imerode(Gray, operater1); %腐蚀操作
    operater2 = strel('rectangle', [24, 24]);
    pictureclose = imclose(pictureErode, operater2); %图像聚类，填充图像
    pictureCut = bwareaopen(pictureclose, 2000);  %remove area <2000
    [y,x]=size(pictureCut);
    picture=double(pictureCut);
    blue_y=zeros(y,1);
    for i=1:y
      for j=1:x
        if(picture(i,j,1)==1) %如果 myI 图像坐标为（i，j）点值为 1，即背景颜色为蓝色，blue_y 加 1
          blue_y(i,1)= blue_y(i,1)+1;
        end
      end
    end
    [~, MaxY]=max(blue_y); %temp 为向量 blue_y 的元素中的最大值，MaxY 为该值得索引
    %Y 方向车牌区域确定
    PY1=MaxY;
    while ((blue_y(PY1,1)>=5)&&(PY1>1))
      PY1=PY1-1;
    end
    PY2=MaxY;
    while ((blue_y(PY2,1)>=5)&&(PY2<y))
      PY2=PY2+1;
    end
    InputImage=InputImage(PY1:PY2,:,:);
    %X 方向车牌区域确定
    blue_x=zeros(1,x);
    for j=1:x
      for i=PY1:PY2
        if(picture(i,j,1)==1)
          blue_x(1,j)= blue_x(1,j)+1;
        end
      end
    end
    PX1=1;
    while ((blue_x(1,PX1)<3)&&(PX1<x))
      PX1=PX1+1;
    end
    PX2=x;
    while ((blue_x(1,PX2)<3)&&(PX2>PX1))
      PX2=PX2-1;
    end
    if (PX1<=1)
      PX1=1;
    else
      PX1=PX1-1;
    end
    PX2=PX2+1;
    OutputImage=InputImage(:,PX1:PX2,:);
    figure
    imshow(OutputImage);
    title('Cropped license plate image');
end
```

Character segmentation:

```
function [word,result] = char_seg(d)
    %获取字符
    word=[];
    flag=0;
    y1=8;
    y2=0.5;
```

```
    while flag==0
      [m,n]=size(d);
      wide=0;
      while sum(d(:,wide+1))~=0 && wide<=n-2
        wide=wide+1;
      end
      temp=remove_extra(imcrop(d,[1 1 wide m]));
    %用于返回图像的一个裁剪区域
      [m1,n1]=size(temp);
      if wide<y1 && n1/m1>y2
        d(:,[1:wide])=0;
        if sum(sum(d))~=0
          d=remove_extra(d);%切割出最小范围
        else word=[];flag=1;
        end
      else
        word=remove_extra(imcrop(d,[1 1 wide m]));
        d(:,[1:wide])=0;
        if sum(sum(d))~=0
          d=remove_extra(d);
          flag=1;
        else d=[];
        end
      end
    end
    result=d;
end
```

Remove license plate boundary:

```
function I4=remove_border(I2)
    projection_h = sum(I2,1);
    projection_v = sum(I2,2);
    for i=1:size(projection_v,1)
      if projection_v(i,1) >= 1
        new.rowa = i;
        break;
      end
    end
    for i=1:size(projection_v,1)
      j = size(projection_v,1) - i+1;
      if projection_v(j,1) >= 1
        new.rowb = j;
        break;
      end
    end
    for i=1:size(projection_h ,2)
      if projection_h(1,i) >= 1
        new.cola = i;
        break;
      end
    end
    for i=1:size(projection_h ,2)
      j = size(projection_h ,2)-i+1;
      if projection_h(1,j) >= 1
        new.colb = j;
        break;
      end
    end
    I3 = I2(new.rowa:new.rowb, new.cola:new.colb);
    diff_row = diff(I3,1,2);  % 前一行减后一行
    diff_row_sum = sum(abs(diff_row), 2);
    [rows, columns] = size(I3);
    trows1 = ceil(rows*(1/3));
    j = trows1;
    plate.rowa=1;
    for i=1:trows1
      if diff_row_sum(j,1)<10
        plate.rowa = j;
        break;
      end
      j = trows1-i;
    end
    plate.rowb=size(diff_row_sum,1) ;
    for i=2*trows1:size(diff_row_sum,1)
      if diff_row_sum(i,1)<10
        plate.rowb = i;
        break;
      end
    end
    I4 = I3(plate.rowa:plate.rowb, :);

end
```

Chinese character recognition:

```
function result = identityChinese(model, word)

    [height,width, num] = size(model);
    for n=1:num
      sums = 0;
      for h=1:height
        for w=1:width
          if word(h,w) == model(h,w,n)
            sums = sums + 1;
          end
        end
      end
      ratio(n) = sums/(height*width);
```

```
    end
    result_index = find(ratio>=max(ratio));
    c = {'湘','鄂','津','京','鲁','粤','浙','黑','辽','陕','豫'};
    result = cell2mat(c(result_index));
end
```

Number and letter recognition:

```
function result = identityDigitLetter(model, word)
    [height,width, num] = size(model);
    for n=1:num
        sums = 0;
        for h=1:height
            for w=1:width
                if word(h,w) == model(h,w,n)
                    sums = sums + 1;
                end
            end
        end
        ratio(n) = sums/(height*width);
    end
    result_index = find(ratio>=max(ratio));
        if result_index<=10
            base = 47;
            result = char(base + result_index);
        else
            base = 64;
            result = char(base + result_index-10);
        end
end
```

IV. Results and Analysis

FIG.2: original photo

FIG.3: Image of edge detection

FIG.4: Cut out license plate pictures

FIG.5: Binary license plate image

FIG.6: Remove the top and bottom bezels and rivets of the binarized license plate image

FIG.7: The result of identification

We can find that there is almost no noise in FIG.3, and the whole license plate is perfectly detected. Therefore, it is reasonable to perform Gaussian filtering before edge detection. At the same time, from the license plate cutting effect shown in FIG.4, the license plate positioning method mentioned in the introduction is relatively reasonable. However, although this positioning method can play a good role in the picture shown in FIG.2, if there are many objects in a picture, or the license plate has occlusion, the positioning effect is not very satisfactory.

From the comparison of FIG. 4 and FIG. 5, we find that the effect of the two colors of rivets converted to binary images is completely opposite. Therefore, we can understand that if the border of the license plate is black, then the license plate recognition system we designed will not be able to

perform good recognition effect. The reason is as follows: As we can see from FIG.6, we will only remove the top and bottom borders and rivets, and then deal with the left and right borders by means of overlay. Once the border of the license plate is black, there will be no white border on the license plate in the final binary image. Therefore, when we take out the first character, we overwrite it, which will lead to the overwrite of the first Chinese character. Finally, only 6 characters can be taken out.

In general, we designed the license plate recognition system only for license plates with no occlusion, no tilt, and white border.

V. Conclusion

With the development of license plate recognition technology and the improvement of accuracy, compared with manual recognition, it has higher efficiency. The license plate recognition system has been integrated into our life. In this project of license plate recognition system design, we designed a simple license plate recognition system by using the knowledge of image processing learned in class, which not only consolidated the knowledge of edge detection and denoising learned in class, but also exercised my skills of searching literature and learning new knowledge by myself.

In this license plate recognition system design, there are still some areas to be improved. First of all, the system is only suitable for small memory image, once the image memory is too large such as ultra-high definition image, the effect of license plate positioning will be poor. Therefore, after the high-definition pictures taken by the mobile phone, the compression of the pictures is also needed. In addition, the recognition effect of the system on slanted license plate photos is poor. Likewise, if the license plate is black bordered, the system cannot recognize it.

References

[1] R. A. Lotufo, A. D. Morgan, and A. S. Johnson, "Automatic number-plate recognition," in Image Analysis for Transport Applications, IEE Colloquium on, 1990.

[2] M. M. M. Fahmy, "Automatic number-plate recognition: neural network approach," in Vehicle Navigation & Information Systems Conference, 1994.

[3] Y. P. Feng, S. G. Li, and T. F. Pang, "Research and System Design of Intelligent License Plate Recognition Algorithm," in 2018 37th Chinese Control Conference, X. Chen and Q. C. Zhao, Eds. (Chinese Control Conference, 2018, pp. 9209-9213.

[4] W. J. Jia, X. J. He, and M. Piccardi, Automatic license plate recognition: A review (Cisst '04: Proceedings of the International Conference on Imaging Science, Systems, and Technology). 2004, pp. 43-49.

[5] A. Budianto, "Automatic License Plate Recognition: A Review with Indonesian Case Study," Scientific Journal of Informatics, vol. 5, pp. 258-270, 11/29 2018.