A Car License Plate Recognition System Based on Dual-Core DSP OMAPL138

Wang Yibo[a,*], Zhou Yujun[b]

College of Electronic Science and Technology, Hunan Institute of Information Technology, Changsha, China

[a] 2728800690@qq.com, [b] 914232157@qq.com

Keywords: Car license plate recognition, neural networks, Cainny edge detection operator

Abstract: In this paper, a car license plate recognition system based on dual-core DSP OMPL138, is proposed. The core algorithm consists of four parts: image pre-processing, license plate localization, character segmentation, and character recognition. An adaptive filtering method is used to solve some image blurring and unclear problems caused by external environment. The Roberts operator is used to edge detection; A minimum threshold selection optimization genetic algorithm is used to get a complete character segmentation; Finally, a neural networks algorithm based on three different training models, is used to achieve license plate character recognition. Such program is developed in a cross-compiling Qt platform and runs in the ARM-Linux environment on the TL138F-TEB experiment box, in which the DSP core executes the main calculation tasks while the ARM core outputs the results of recognition, and these dual cores make data exchange through TI's inter-core communication framework (syslink). The final test results show the differences in the accuracy of license plate recognition achieved by three different training models.

1. Construction of the Car License Plate Recognition System Based on OMAPL138

The hardware foundation is based on TL138F-TEB (Tronlong LLC) experiment box[1], which contains a Xilinx Spartan-6 FPGA and an OMAPL138 dual-core DSP, as shown in Figure 1.

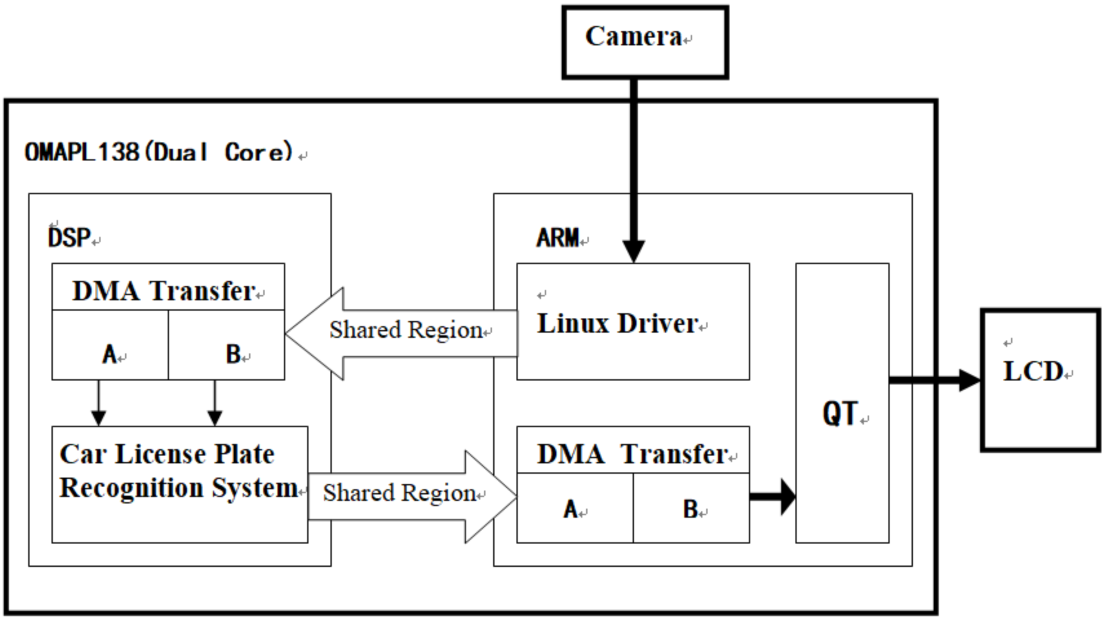

Figure 1 Hardware foundation for the car license plate recognition system

The kernel algorithm of car license recognition system, is mainly embedded into the C674X-DSP rather than into the ARM core, because that the DSP core contains two parallel float multipliers which support various types of computation in the car license recognition algorithm.

Usually, a car license recognition algorithm, can be divided to five succeeding parts as shown in Figure2.

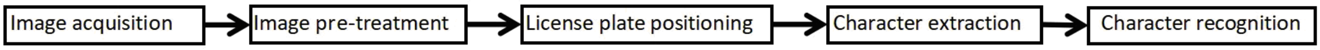

Figure 2 Main algorithm process of the car license plate recognition system

As the first step, the camera is connected to an onboard USB port which can be easily handled in the ARM-Linux environment. Then, the image data is transferred to DSP core by the “shared region” mechanism, which is a part of TI inter-core communication framework (syslink). DMA and A/B dual buffer are used to make a coordination between the image data I/O process and the car license recognition process. On this basis, the left four parts of above algorithm is running in the C674X core and finally gives a recognition result which is send back to ARM for display.

2. Pre-treatment of car license recognition

The first step of car license plate recognition system is to obtain the image of the vehicle to be processed through professional cameras and collection equipment, and then locate the license plate area. Before locating the license plate area, the vehicle image obtained is often of low quality due to external environmental factors, and there is a large amount of noise. It is necessary to accurately and quickly locate the license plate area. The image acquired needs to be pre-processed as necessary, mainly graying the colour in acquired image, then enhancing the image, and transforming it into a binary image. The filter operator is used for filtering pre-processing to eliminate or reduce noise interference in the image.

2.1. Gray scale transformation

The vehicle images collected through various collection devices are generally color images. The storage cost of color images is high, and the processing speed is slow, which affects the recognition speed. Therefore, before positioning the images, the images should be converted into grayscale images to speed up the processing speed. According to the sensitivity of the human eye to different colors, the following formula (1) is often used to convert color images into grayscale images:

$$Y = R*0.2989 + G*0.5870 + B*0.1141 \tag{1}$$

Usually, the license plate area is located in the middle of the lower part of the vehicle, which is generally in the middle grayscale range. However, some areas above the vehicle body and in the background are generally brighter, belonging to the high grayscale range, and the edges are darker, belonging to the low grayscale range. To highlight the grayscale details of the license plate and suppress high and low grayscale areas, formula (2) is used to perform intermediate grayscale stretching on the license plate image.

$$G(\mathrm{i,j}) = \begin{cases} \dfrac{u'}{u} T(i,j) & 0 \le G(i,j) \le u \\ \dfrac{v'\text{-}u'}{v\text{-}u}[G(i,j) - u] + u, & u \le G(i,j) \le v \\ \dfrac{w' - v'}{w - v}[G(i,j) - w] + w, & v \le G(i,j) \le w \end{cases} \tag{2}$$

2.2. Image filtering and denoising

During the collection process of automotive image information, it is often interfered by various noise sources, such as lighting, damage, and stains. These noises often appear as isolated pixel points on the image, and the grayscale of pixels in grayscale images is spatially correlated, meaning that the grayscale of noisy pixels is significantly different from that of their neighboring pixels. To reduce the impact of noise on subsequent image processing, it is necessary to smooth and denoise the image. Common image smoothing methods include image averaging, mean filtering, and median filtering.

This system adopts an adaptive filtering method[2] for image filtering processing, which achieves adaptive filtering based on changes in local pixel variance of the image. The filtering probability model is:

$$\begin{cases} \mu = \dfrac{1}{MN}\sum_{x,y} T\ (x,y) \\ \sigma^2 = \dfrac{1}{MN}\sum_{x,y} T^2(x,y) - \mu^2 \\ T'(x,y) = \mu + \dfrac{\sigma^2 - \upsilon^2}{\sigma^2}[T(x,y) - \mu] \end{cases} \tag{3}$$

2.3. Image binarization

In digital image processing, binarization of images is a very important step. Usually, an image needs to be processed and analyzed, and the grayscale image needs to be binarized firstly to obtain a new image, which is conducive to further processing of the image. The key to binarization of images is to find a suitable threshold to divide the processed image area into two parts, which is conducive to further analysis and processing.

In this article, the classic Otsu algorithm[3] is used to binarize vehicle images. The Otsu algorithm belongs to the global dynamic binarization method. The algorithm mainly calculates and classifies the grayscale values of each pixel in the image, and then uses the least squares method for derivation. During the derivation process, the ratio of inter class variance to intra class variance is calculated, and the maximum value in the ratio is used as the threshold to effectively distinguish the target area and background of the image. This method is simple to implement and has strong adaptability.

3. Character extraction

The contour detection of digital images is an indispensable step in specific shape segmentation, which includes important edge features such as specific area recognition and shape differentiation. The understanding of pixel regions and the accuracy of analysis and tracking are mostly related to the selection of starting points. Improper selection can lead to increased computational complexity and difficulty in querying. Additionally, the selection of search rules must comply with the design analysis, Also consider the design of program algorithms.

In the process of license plate localization, the simplest method is to first search for pixels on the contour of the target license plate based on the contour of the license plate. Because the edge of the license plate is the part of the image where the local pixel value changes the most significantly, for example, starting from the upper left corner of the image and gradually scanning the pixel points, searching for grayscale values, texture structures, colour value mutations, and other parts, until it returns to the starting point and closes. The direction and size of image edge detection are executed in an orderly manner, with some transformations tending to be smooth and others experiencing significant changes. At the same time, corresponding parameter adjustments are made during the analysis process.

3.1. Edge detection

Edge detection is one of the important contents in image processing. Edge is the most fundamental feature of an image. The so-called edge refers to the collection of pixels whose surrounding pixels have a step change in grayscale or a roof change. Edge detection is based on the original image, to investigate the grayscale step change in a certain field of each pixel of the image, and to detect the edge by using the change rule of the first or second order directional derivative adjacent to the edge. The commonly used edge detection methods include Roberts operator, Sobel operator, Prewitt operator, Laplace operator, etc., each of which has its own advantages and disadvantages in application.

Based on the reference of previous researchers, John F. Canny first completed the improvement work of this algorithm. The Cainny edge detection operator[4] has the best algorithm implementation in edge search, and its new feature is to encapsulate candidate pixels with parallel edges into outer bounding frames.

The detailed calculation process of the Canny algorithm is as follows:

Firstly, perform convolutional array calculations (acting in the x and y directions) according to the implementation steps of the Sobel filter:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{4}$$

Then use Gaussian smoothing filter for convolution denoising to further eliminate noise. The following is an example of a 5 * 5 Gaussian kernel:

$$K = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \tag{5}$$

Finally, use the following formula to calculate the amplitude and direction of the grayscale gradient:

$$G = \sqrt{G_x^2 + G_y^2} \qquad \theta = \arg\tan\left(\frac{G_x}{G_y}\right) \tag{6}$$

The Roberts operator is an operator that uses local difference operators to find edges, which is given by the following equation:

$$g(x,y) = \{[\sqrt{f(x,y)} - \sqrt{f(x+1),(y+1)}]^2 + [\sqrt{f(x+1,y)} - \sqrt{f(x,y+1)}]^2\}^{\frac{1}{2}} \tag{7}$$

The Roberts operator has high accuracy in edge localization, but is sensitive to noise and is prone to losing some edges. This operator performs well on images with obvious edges and less noise. The matrix template is:

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{8}$$

3.2. Image segmentation based on genetic algorithm

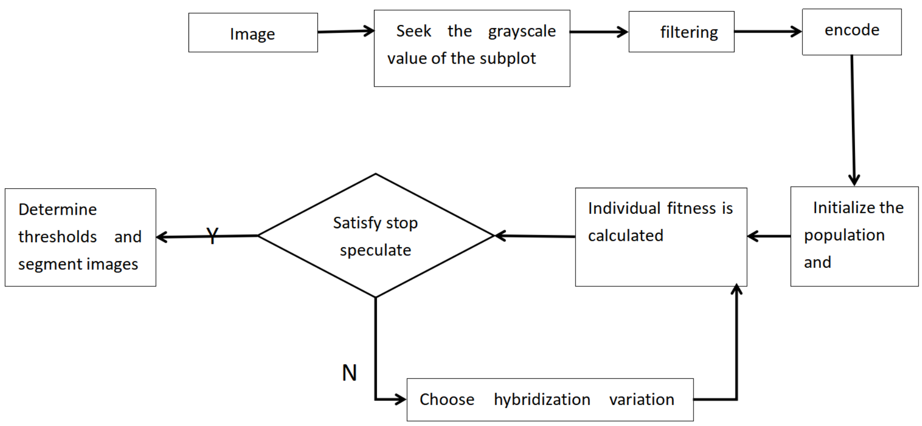

Figure 3 Main process of image segmentation based on genetic algorithm

Extracting image segmentation thresholds based on genetic optimization principles is easy to reduce the impact of image target size and noise. The algorithm is simple, stable, and has good robustness. The following is an optimization of the minimum threshold selection based on genetic algorithm[5], as shown in Figure 3.

3.3. Character tilt correction

After determining the area where the license plate is located, we need to separate individual characters from the license plate. Due to the influence of the angle between the camera and the vehicle, the license plate image we obtain generally has a certain degree of tilt, which brings difficulties to character segmentation and leads to segmentation errors. Therefore, it is necessary to correct the tilt of the license plate before character segmentation. In practical applications, the tilt angle of the license plate is generally not too large, as different shooting angles may result in different tilt angles, mainly including horizontal tilt, vertical tilt, and horizontal vertical tilt.

The main methods used for correcting tilted license plates[6] are: (1) Hough change method (2) template matching method; (3) Connected domain method. This article uses a combination of Hough transform and template matching to correct tilted vehicle images.

In this paper, Hough transform is used to obtain the border information of the license plate in the tilted image, and template matching is used to determine the coordinates of the four corners of the license plate image area. Finally, bilinear space transformation is used to complete the image correction.

4. Character Recognition Based on Neural Networks

In license plate number recognition, the data that the computer can process is no longer a lightweight structured data form, but the quality of pixel data in the sample library and other images, and the amount of data processed in a very short time is large.

The information of license plate numbers in the sample library and the information that is of great concern in recognition all imply great utilization value in the pixels of these vehicle image information. Using this method to recognize the information of characters in the license plate image, the generation of the target output is propagated forward. A learning neural network loads the target and extracts license plate characters using the function. This is mainly applied to the Back Propagation algorithm. BPNN is a three-layer network that connects multiple units, which can be used to use the weights of the learning network. The goal is to minimize the square difference between the actual output of the network and the target output. The training process involves creating a network with hidden layers, which can randomly assign these neurons a small value between -0.05 and 0.05. Then, the input sample vector is fed into the input of the network, where the network output value is calculated. To make the results close to the expected output and minimize the error between the two, it is necessary to assign weights to the output layer units[7], so that when inputted into the network again, the results can be closer . During this process, all different sample vectors need to be repeated multiple times until the error drops to an acceptable value before the training network is considered complete, as shown in Figure 4.

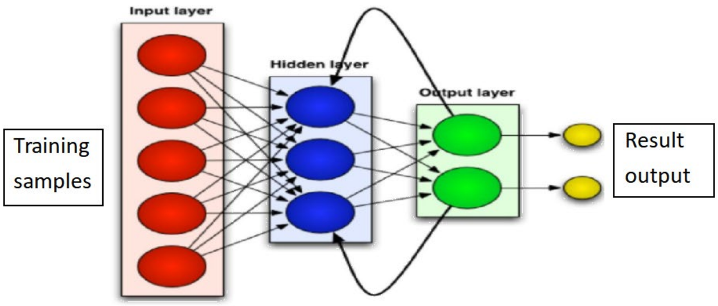

Figure 4 Structure of License Plate Character Recognition Network

By using inputs and weights, the activation of a given node can be calculated. It is very convenient to directly link the hidden layer to the actual input layer. The forward transmission and error reverse interaction process are the training methods of the BP algorithm. Generally, there are three standards that can be used as the termination judgment for network training[8]: for example, stopping when a separate test sample set meets the standard classification error; Stop when the training sample error has dropped to a certain parameter value; Stop when the number of iterations for a given value reaches a reasonable value.

5. Results of test for car license recognition

Above car license plate recognition algorithm finally runs in the DSP core, and leads to a output on the LCD of TL138F-TEB, as shown in Figure 5.

Figure 5 LCD outputs for recognition results on OMAPL138 Platform

Three standards for judgment of the termination of network training, leads to different recognition success rate, as shown in Table 1.

Table 1 Recognition success rate under three termination patterns for neural network training

Termination patterns for neural network training	Recognition success rate
Stop when the classification error matches the target	88%
Stop when the sample error has decreased to the specified value	91%
Stop when the number of iterations reaches the specified value	94%

6. Conclusion

According to the test results, the dual-core DSP, OMAPL138, which is proposed in 2010, is still enough to sustain above recognition algorithm. However, the still recognition algorithm discussed in this paper gives a high mis-recognition rate, which means there is still huge room for improvement.

Acknowledgements

This study was supported by the *Enterprise-University Collaborative Education Program* from Ministry of Education of the People's Republic of China (grant no. 202102196002: Construction of an experiment teaching system aimed at ‘high speed AD-IO and data processing’ for the application of multi-core DSP), and the *Education Research Study Program* from Hunan Institute of Information Technology (grant no. XXY021ZXYB08: Construction of a New Multi Core DSP

Application Technology Innovation Platform Based on Research Incentive Experimental Teaching).

References

[1] Tronlong LLC, Introduction to the TL138F-TEB Experiment box, [online] Available: https://www.tronlong.com/Product/show/65.html.

[2] Fu Z, Embedded image and video coding algorithm based on adaptive filtering equation. Advances in Mathematical Physics, vol.2021, pt3, pp.7953993-1~7953993-11, 2021.

[3] Kumari K M, Christian O, Karthe D, et al. Automatic extraction of large-scale aquaculture encroachment areas using Canny Edge Otsu algorithm in Google earth engine – the case study of Kolleru Lake, South India . Geocarto International, vol.37, no.26, pp.11173-11189, 2022.

[4] Canny J. A computational approach to edge detection. IEEE Trans. on Pattern Analysis & Macliine Intelligence, vol.8, no.6, pp.679-698, 1986.

[5] Silva R D, Amanda L, de S Maíra A, et al. Features selection study for breast cancer diagnosis using thermo-graphic images, genetic algorithms, and particle swarm optimization. International Journal of Artificial Intelligence and Machine Learning, vol.11, no.2, pp.1-18, 2021.

[6] Berganski C, Hoffmann A, Möller R, et al. Tilt correction of panoramic images for a holistic visual homing method with planar-motion assumption[J]. Robotics, vol.12, no.1, pp.20-26, 2023

[7] Li H, Qin T. "A license plate location algorithm based on multi-component edge combination of the HSI colour space". 2011 4th International Congress on, Image and Signal Processing (CISP), pp.1127-1129, July, 2011.

[8] Yang, Muthukumar B,Wang Y. Comparative evaluation of convolution neural networks, training image sizes, and deep learning optimizers for weed detection in alfalfa. Weed Technology. vol.36, no.4, pp.512-522, 2022.