Incomplete character recognition technology in the license plate recognition system

Kai Song, Caihong Niu*

Information Science and Engineering College, Shenyang Ligong University, Shenyang, China

Received 1 May 2014, www.tsi.lv

Abstract

The incomplete characters recognition in the license plate, characters can be divided into linear character and curve character. For the curve characters, used a method of statistic the number of character holes as the character characteristics for feature extraction, which extended seed filling algorithm in the Computer Graphics .For linear characters, we proposed a method of extracting a character conversion slope of the line feature by Hough transform, which had a good effect on linear character recognition.

Keywords: license plate recognition; extraction of character feature; incomplete character recognition

1 Introduction

An important sign for license plate recognition is the recognition accuracy, under normal circumstances ,the accuracy can reach 90% or even above 95%,but in the practical applications, There are many factors caused accuracy reduced, including artificial factors and natural factors, such as some drivers deliberately paint brush in license plate, with tape paste the license plate, place the preparation tire cover license plate, these are artificial factors, or rainy day, muddy, pollution, the shortage of parked position is wrong, and insufficient light can also result license plate fuzzy, In this case, it is difficult to accurately identify a license plate number [1].

Therefore, extracting effective and easy to implement feature is particularly important for incomplete characters. We presented a character feature extraction method, which can effectively increase the accuracy of fuzzy license plate recognition. The method by extracting characters of part features and build character database, compare the part characteristics of the incomplete characters with the database, and thus achieve the goal of improve the incomplete character recognition.

The commonly used character feature extraction include statistical feature extraction and topological feature extraction, not all these methods of feature extraction for incomplete characters have high effective degree, so bearing this in mind, extracting the number of character holes and character strokes straight slope characteristics as a key object of study. Firstly, through the license plate location, tilt correction and segmentation process, extracting independently the license plate character, and then through the character of thinning algorithm processing to extract characters, at this point, we can extract the feature of character [2, 3]. The number of character holes and Hough transform algorithm for line slope detailed below instructions. It obtained better effect on the incomplete character recognition by these two features [4].

2 The number of holes of character for feature extraction

This article presents a method that search the number of circles and straight lines as characteristics, the main idea of this method is that since many of the characters have circles or holes, we use a recursive seed filling algorithm to fill these holes statistically, [5] determines the number of characters circles as a feature of the character.

This article uses the recursive seed-filling algorithm in computer graphics to calculate the number of character circles [6]. This algorithm firstly assumes that some points in the closed contour line is known, and then search the adjacent points around the seed point, and these points must be in the contour lines. If it is found that adjacent point is not in the contour line that is the outline of the edge; if it is found that the adjacent point in the contour line, this point will become a new seed point, and continue to search as above method.

4-connected algorithm and 8-connected algorithm: seed filling algorithm is essentially a recursive algorithm, It specify a seed point firstly and use the seed point as the benchmark to search in all directions, each pixel is processed, until it meet the boundary, the differences among all sorts of seed filling algorithm are in the colour and edge processing mode. We introduced two concepts, 4-connected algorithm and 8-connected algorithm. We start from any point in the region, if it was just for up, down, left and right to search the four directions and reach any pixel of the region, so the region is filled by this method is called the 4-connected regions, the filling method is called 4-connected algorithm. If we trigger

* ***Corresponding author*** e-mail: niucaihong1010@163.com

from any one point in the region, and search the up, down, left, right, bottom-left, top- left, bottom- right and top- right under the eight directions and reach any pixel of the region, the region is 8-connected regions, this algorithm is called 8-connected algorithm.

Recursive seed filling algorithm process is shown in Figure1 below.

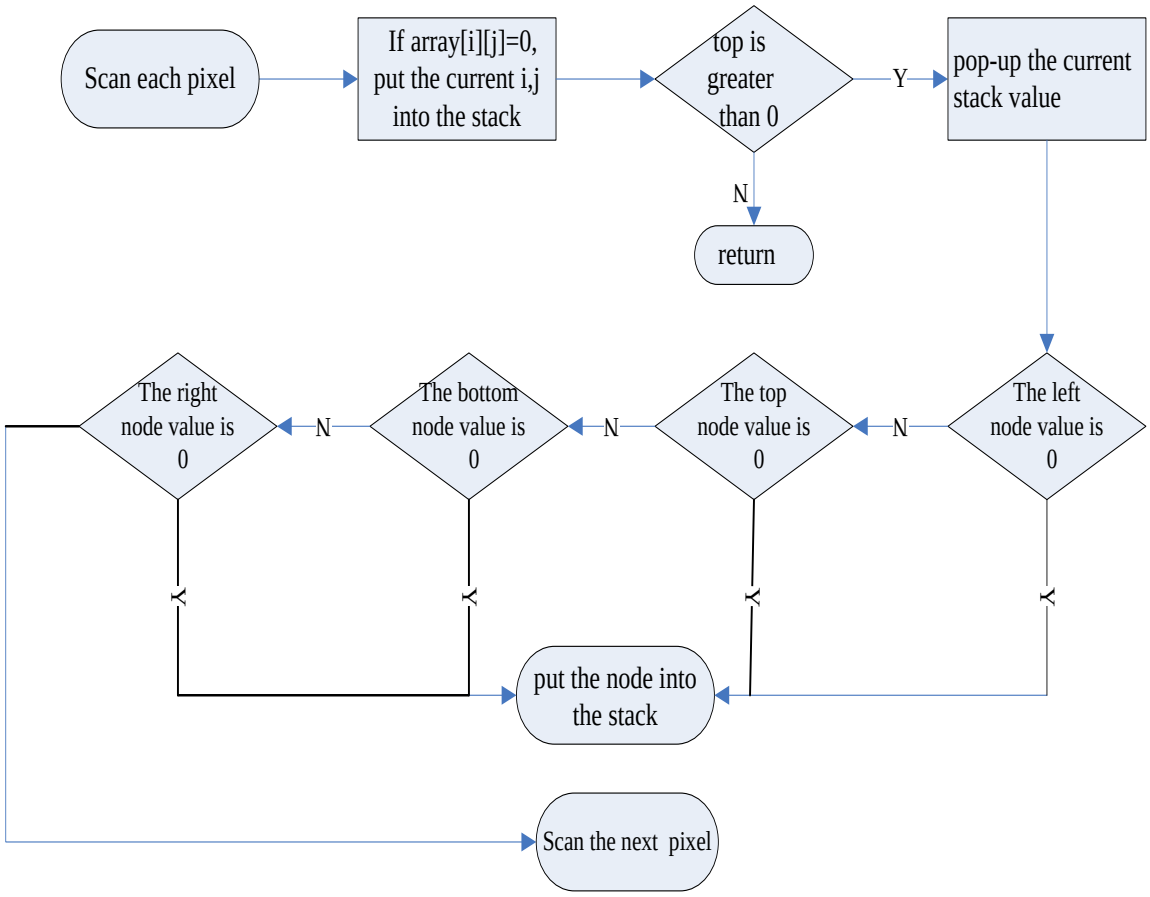

FIGURE 1 The algorithm process of extracting the characters' feature

We use the 4-connected filling algorithm, Specific steps of the algorithm are:

1) Selecting seed , push the seed into the stack. Scan each pixel value and set up two stack stack1 and stack2, if the image data array [i] [j] = 0, then the pixel is a white pixel, select it as a seed, the horizontal and vertical coordinates were pushed into stack1 and stack2, so stack1 [+ + top] = i, stack2 [top] = j.

2) If the stack is empty or the stack top value is less than 0, then jump out.

3) If the stack is not empty, then the pixels into a filling colour, namely the array [i][j] = flag, the flag value represents a kind of colour , and judge whether the four connected pixels adjacent to the pixel are boundary colour or have set into a polygon fill colour, if not, then push the pixel coordinates into stack, as a new seed. If array [i]] [j-1] = 0, then push the top node into the stack, stack1 [++top] = I, stack 2[top] =j. If array[i-1][j]=0,then push the left node into the stack, if array[i+1][j]=0, then push the right node into the stack , If array[i][j+1]=0,then push the bottom node into the stack, when a hole filling is completed, the flag automatically add 1, as a new colour, repeat the first two steps of the algorithm.

4) End of the algorithm.

The Figures 2-4 show the results that deal with seed filing algorithm , we can see the different circles are dyed different colours, we take advantage of this to calculate the number of circles in a character, and calculate where these circles in the character according to the colour.

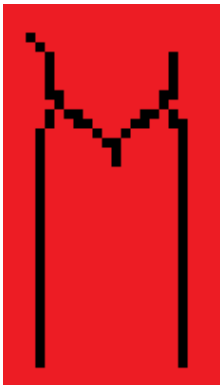

FIGURE 2 The character with no circle

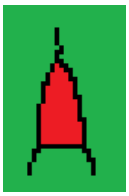

FIGURE 3 The character with one circle

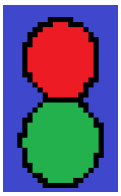

FIGURE 4 The character with two circles

The algorithm process of extracting the feature of the characters is shown in Figure 5.

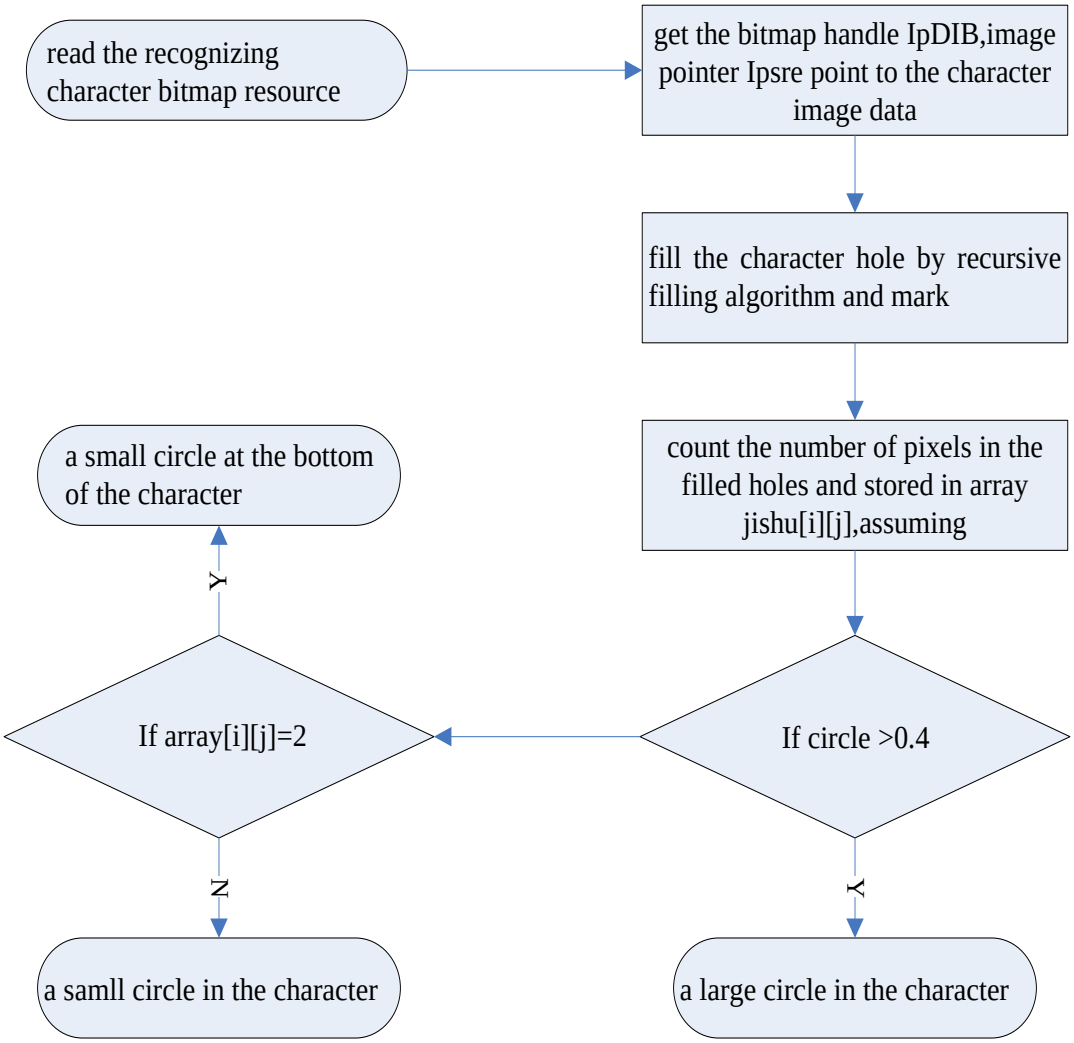

FIGURE 5 The algorithm process of extracting the characters' feature

3 Feature extraction of the character slope

According to the above discussion on character, we know its ability of recognition for the complete character is good, but for the incomplete character is rather poor, in order to improve the system's ability to recognize, this section is put forward four character features in view of the incomplete characters, which are the character stroke slope feature, the angle feature of the stroke intersection, the central angle feature and the curvature feature [5, 6]. For incomplete characters, we need to extract the local features, because we cannot estimate the damaged parts and degree of disability of the characters. Local features will have a better recognition effect. For example, the slope feature of the stroke, as long as we have a bit remnants of the straight line in the direction can we find the slope and recognize. But this kind of method have a large amount of calculation, it has influence on the system speed, so we should combine this feature with the previous feature to recognize the characters, so that you can balance the ability of recognition and the speed of recognition.

We use the method of Hough transforms to detect the straight line in the character and find the slope of the line

[7, 8]. Due to the principle of Hough transform has been elaborated in the past, no longer described here, the reader can refer to relevant reference material [18]. Usually the process of detecting lines by Hough transform is as shown in Figure 6 below.

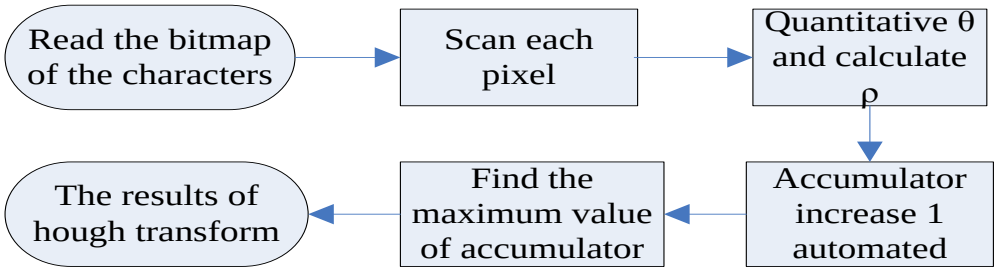

FIGURE 6 The process of detecting lines by Hough transform

1) Constructing an accumulator A (ρ,θ).Each element of the array represents the transform space location of the point and the number of straight lines through this point, initialize A, each element is zero. Here we apply for a pointer to the transform region, and apply large enough memory space for accumulator through the pointer, int* lpTransArea=(int *)malloc(lwidth * lheight * sizeof (int), then we can save pointer address of the transform region lpTrans = lpTransArea, finally the accumulator reset memset(lpTrans,0,lwidth * lheight * sizeof(int)).

2) If the linear feature points were detected ,then corresponding to each feature point (x, y) calculated the value of θat 0 degree to 180 degrees ,loop in each pixel scanning line , if the value of the data pointer lpsrc is 0 , it indicates that the pixel is a black pixel , we use variable I AngleNumber represents the quantitative value of ρ and calculated the value of ρ from 0 degree and 180 degrees using $\rho = x\sin\theta + y\cos\theta$.

3) The value of the corresponding accumulator adds one. We will make the pointer point to transform region as the base pointer and make * (lpTransArea + iDist * iMaxAngleNumber + iAngleNumber) add one. This expression represents the corresponding memory space of the accumulator.

4) We find the maximum value in the accumulator, the value represents the linear parameter what we're detecting. Using the parameters to find the slope of the straight line $k = -\tan\theta$ and put the value of slope into the feature database.

In order to improve the precision and speed, we adopt an improved linear detection method, algorithm is as follows:

1) Searching in the image, find out a non-zero value point p as a seed.

2) Make p as the starting point of a rectangle, the size of the rectangle is M*N, search the non-zero points in the rectangle, then calculate linear parameters (ρ, θ) of each non-zero point.

3) Make the deviation value of ρ for $\Delta\rho$ and θ for $\Delta\theta$, we vote selection in the rectangle, then statistics the number of parameters within $(\rho_i + \Delta\rho, \theta_i + \Delta\theta)$, we get the value n_i .

4) Find the vote maximum $n_{\max}$ range in this region and average to the parameter which belongs to the range, so it can reduce the quantization error, the average value of this parameter is the line parameter which through a point p in this region.

5) Set T_1 to line length threshold value ,if $n_{\max}$ is bigger than T_1, go to step 6, search other points of the line ;Otherwise thinking about the line which through point p does not exist ,reset ,go to step 1and search again.

6) Search point p in the image and calculate the corresponding value of ρ ,if $\rho - \bar{\rho} < \Delta\rho$,then count the number of cumulative voting and set the grey value of this point to 0.

7) After the full search ,if the number of line vote is bigger than the setting threshold value ,then we think this line exist ,then find the parameters of the line and put them into the array line1 [n], line2 [n], these two arrays store the value ρ and θ .N is the number of detected line.

According to this algorithm, we can calculate the number of all lines in the binary images. In this way, we counted the number of straight lines, [9-11] and found the slope of the straight lines, recorded the slope of each character one by one and put them into the feature database [12-15].

For a complete or an incomplete character, we can still extract the feature of slope by Hough transform.

Experimental results are as follows:

1) Searching in the image, find out a non-zero value point p as a seed.

2) Make p as the starting point of a rectangle, the size of the rectangle is M*N, search the non-zero points in the rectangle, then calculate linear parameters (ρ, θ) of each non-zero point.

3) Make the deviation value of ρ for $\Delta\rho$ and θ for $\Delta\theta$, we vote selection in the rectangle, then statistics the number of parameters within $(\rho_i + \Delta\rho, \theta_i + \Delta\theta)$, we get the value n_i .

1) Find the vote maximum $n_{\max}$ range in this region and average to the parameter which belongs to the range, so it can reduce the quantization error, the average value of this parameter is the line parameter which through a point p in this region.

2) Set T_1 to line length threshold value, if $n_{\max}$ is bigger than T_1, go to step 6,search other points of the

line; otherwise thinking about the line which through point *p* does not exist, reset, go to step 1 and search again.

3) Search point p in the image and calculate the corresponding value of ρ, if $\rho - \bar{\rho} < \Delta\rho$, then count the number of cumulative voting and set the grey value of this point to 0.

4) After the full search, if the number of line vote is bigger than the setting threshold value ,then we think this line exist ,then find the parameters of the line and put them into the array line1 [n], line2 [n], these two arrays store the value ρ and θ .N is the number of detected line.

According to this algorithm, we can calculate the number of all lines in the binary images. In this way, we counted the number of straight lines, [9-11] and found the slope of the straight lines, recorded the slope of each character one by one and put them into the feature database [12-15].

For a complete or an incomplete character, we can still extract the feature of slope by Hough transform.

Experimental results are as follows:

FIGURE 7 The Slope feature extraction of incomplete "A"

Because of the quantization error, it will lead to some errors, but this is entirely result from the quality of the refinement algorithm, if the processing effect for the part of character refinement algorithm is better, it will greatly reduce the problem. FIGURE 8. The Slope feature extraction of incomplete "N" reduces the problem.

FIGURE 8 The Slope feature extraction of incomplete "N"

FIGURE 9 The Slope feature extraction of incomplete "4"

Because the origin of coordinates in the image is starting from the up-left corner, we will use this coordinate system, so the calculated value of slope may be different from common coordinate system, but there is no effect on the slope of the character. The number 4 with two special lines is respectively the horizontal and vertical straight line, we calculated the slope of a straight line, respectively k_1 = 2.14286, k_2 = 2.18182, k3 = 2.05. We calculated the average value k = 2.1249 of the three. The algorithm has to meet the requirements of extracting its feature of slope for linear character.

Finally, we give some characters slope characteristics and the range of error, the slope of the straight line, horizontal and vertical not considered here. [13, 14] Selecting (k-0.3, k+0.3) for the error interval, if the slope of the straight line fall in this range, judging from the value of K, we can know the character ,the value of K represents the value of slope for the character.

1) The slope of a slash in the number 2 is k = 1.4;
2) The slope of a line in the number 4 is k = 2.1249;
3) The slope of two lines in the letter A is k1 = 5.25, k2 = 5.25;
4) The slope of two lines in the letter K respectively is k1 = - 1.85, k2 = 4.4;
5) The slope of two lines in the middle of the letter M respectively is k1 = 1.107, k2 = 1.18;
6) The slope of a line in the letter N is k1 = 2.64;
7) The slope of a line in the bottom- right side of the letter Q is k = 1.66;
8) The slope of a line in the bottom- right side of the letter R is k = 2.52;
9) The slope of two lines in the letter V respectively is k1 = 4.8, k2 = 5.14;
10) The slope of four lines in the letter W respectively is k1=-5.68, k2 = 16.25, k3= 5.77, k4 = -17;
11) The slope of two lines in the letter X respectively is k1=2.51, k2 = -2.49;
12) The slope of two lines in the letter Y respectively is k1=1.96, k2 = -1.96;
13) The slope of a line in the letter Z is k = 2.38.

4 Conclusion

Extraction the number of holes in the character, extends the application of seed filling algorithm, this method is easy to implement, with high accuracy, in addition, it is an innovation to calculate the slope of the character stroke by the method of Hough transform, we obtained a relatively standard form about the slope of characters through calculation, in the incomplete character recognition, the slope has a great difference between the characters, so it has a good degree of differentiation, we achieved better results through experiment.

Acknowledgments

My sincere thanks should go to Liaoning province technology hall plan projects (2012217005) and Liaoning province Science of public research funds (2012004002). It is because of their sponsorship, this project can be completed so smoothly. Any progress that I have made is the result of their profound concern and selfless devotion.

Reference

[1] Wang Xiaoxue 2010 Application of Digital Image Processing in License Plate Recognition *Process Automation Instrumentation* **31**(7) 22-8

[2] Han Liming 2010 Research and implementation on key technology in license plate recognition system *Computer Engineering and Design* **31**(17) 3919-23

[3] Tao Xun 2011 License Plate Recognition System Design and Realization, *Electrical Automation* **33**(4) 77-80

[4] Wang Yulei, Jiang, Lixing 2009 Digital Character Recognition Algorithm Based on Character Feature *Marine Charting* **29**(1) 56-8

[5] Dong Lingjiao 2008 Character feature extraction for car plate recognition *Electrical Engineering* **25**(9) 106-8

[6] Rui Ting 2004 License plate character recognizing under high noise using sam *Pattern recognition and artificial intelligence* **17**(2) 467-70

[7] Ye Chenzhou 2000 Number-Plate Character Recognition *Journal of Shanghai Jiaotong University* **34** (5) 672-5

[8] Zhang Jian 2011 Research on character recognition of license plate recognition *Information Technology* **9**(4) 109-20

[9] Yu Lasheng, Shen 2004 Deyao A Refinement of the Scan Line Seed Fill Algorithm *Computer Engineering* **29**(10) 70-2

[10] Wen Y Lu, Y Yan J Zhou, Z von Deneen K M, Shi P 2011 An Analog Feedback Associative Memory *IEEE transactions on intelligent transportation systems* **12**(3) 117-26

[11] Zou Mingming, Lu Di 2010 Recognition algorithm of car license plate characters based on modified template match *Foreign Electronic Measurement Technology* **29**(1) 59-61

[12] Zhao Kun 2010 An Improved Preprocessing Algorithm of Vehicle License Plate *Henan Sciences* **28**(3) 329-32

[13] Wang Jianyong 2006 An Improved Algorithm for Line Detection *Computer Engineering* **32**(16) 172-4

[14] Pasi Franti, Alexey Mednonogov, Ville Kyrki, Heikki Kalviainen 2000 Content-based matching of line-drawing images using the Hough transform *International Journal on Document Analysis and Recognition December* **3**(2) 117-24

[15] Zheng L, He X, Samali B, Yang L T 2013 Accuracy enhancement for license plate recognition *Journal of Computer and System Sciences* **79**(2) 511-6

[16] Hsu G-S, Chen J-C, ChungY-Z 2013 Application-Oriented License Plate Recognition *IEEE Transactions on Vehicular Technology* **62**(2) 552-61

[17] Jiao J B, Ye Q X, Huang Q M 2009 A configurable method of multi-style license plate recognition *Pattern Recognition* **42**(3) 358-69

[18] Khader Mohammad, Sos Agaian, Hani Saleh 2011 Practical automatic Arabic license plate recognition system SPIE *Conference on Multimedia on Mobile Devices* 2011.

Authors

Kai Song, born in 1964, Liaozhong, China

Current position, grades: professor and a PhD supervisor, Party committee secretary of the College of Information and Engineering in Shen Yang Ligong University now
University studies B.S. degree in computer control theory from Xiamen University, Xiamen, China, in 1986, M.S. degree in network and communication from Northeastern University, Shenyang, China, in 1989 and Ph.D. degrees in biological environmental monitoring and control engineering from Shenyang Agricultural University, Shenyang, China, in 2008.
Scientific interest: computer vision, image processing and analysis.
Publications: about 20 academic papers about image processing in core journals in recent three years

Caihong Niu, born in 1987, Xinxiang, China

Current position, grades: M.S. degree student in Shenyang Ligong University
University studies: B.S. degree in Communication engineering from Henan University, Kaifeng, China, in 2011, M.S. degree in Shenyang Ligong University now
Scientific interest: image processing and analysis.