

## A Kind of De-noising and Segmentation Method for CAPTCHA with Colored Interference Lines

Zhao Wang<sup>1, a</sup>, Zitong Cheng<sup>1, b</sup>

<sup>1</sup> Institute of Software, School of EECS, Key Laboratory of HCST, MoE, Peking University, Beijing, 100871, China

<sup>a</sup>email: wangzhao@pku.edu.cn, <sup>b</sup>email: cz0616@sina.com

**Keywords:** CAPTCHA Recognition; Colored Interference Lines; Characters Segmentation

**Abstract.** The research of CAPTCHA recognition can discover the security vulnerabilities of CAPTCHA in time. It has great significance in improving the CAPTCHA design method and promoting the design level of CAPTCHA. Text-based CAPTCHA recognition mostly uses the method of characters segmentation and recognition. It is not very ideal for the CAPTCHAs containing colored interference lines, characters adhesion, rotation, distortion and scaling interference. A kind of de-noising and segmentation algorithm which is suited to the CAPTCHA with colored interference lines is presented in this paper. Finally, it is verified through a large number of real data.

### Introduction

CAPTCHA mechanism have been widely used in a large number of sites. It is used to distinguish the user's identity is computer malware or human in the process of user registration, login and personal information changes, its role is to prevent the machine attacking, the specific performance is: to prevent malicious registration, prevent password violence, to prevent the release of advertising and spam, guarantee the authenticity of online voting.

The research of CAPTCHA recognition is similar to the research of cryptanalysis. It has important significance, it can discover the security flaws of the verification code and promote the design level of the verification code. In addition, verification code recognition is a comprehensive problem, which requires the technology of image processing, pattern recognition, artificial intelligence and other fields. Verification code recognition can promote the progress of other areas at the same time.

CAPTCHAs are divided into text, image, sound and question four types. The text-based CAPTCHA is the verification code which is used to verify the password only by the letters and numbers. It is the most widely use. Text-based CAPTCHA recognition involves the segmentation and recognition of characters, and the character recognition technology is mainly based on the optical character recognition technology [1], which has been developed at home and abroad. In this paper, a kind of de-noising and segmentation method for CAPTCHA with colored interference lines is presented.







### Application Status of CAPTCHA

Although there are a lot of new CAPTCHAs, text-based CAPTCHA is widely used, mainly because the text-based CAPTCHA has a lot of advantages: first, the text-based CAPTCHA is easy to generate, the cost is not high; second, the correct answer is not only unique but also simple and easy to be tested. It will not be ambiguity because of the differences in the knowledge and culture background of the tested objects, it is applicable to a wide range of application. Many of the life-services web sites that are closely related to the daily life have users with much difference in the knowledge background, are mostly using text-based CAPTCHAs.

At present, the text-based CAPTCHAs recognition is based on the method of first segmentation and then recognition [2], and has achieved good recognition results in the CAPTCHA

with a small amount of discrete noise, no adhesion characters, no rotation and distortion. For the colored interference lines, adhesion, rotation, distortion and scaling interferences, CAPTCHA recognition effect is not ideal [3], so now most of the sites are using this kind of CAPTCHAs. Table 1 lists the CAPTCHAs used by some of the common web sites in China in May 2015.

Table 1. CAPTCHAs used by some of the common web sites in China (May 2015)

| Web sites           | CAPTCHA examples   |
|---------------------|--|
| www.bankcomm.com    |  |
| www.abchina.com/cn/ |  |
| www.boc.cn          |  |
| www.ganji.com       |  |
| www.58.com          |  |
| www.youku.com       |  |

### Status of Characters Segmentation Technology for CAPTCHA

The traditional CAPTCHA recognition method includes preprocessing, characters segmentation, feature extraction and recognition after the picture is obtained. Every step has important effect on the final recognition rate.

In order to fight against the attack, CAPTCHAs tend to be added to the interference dots and lines, some also include color changes. The preprocessing is removing interference before segmentation to get a black-and-white picture, so to improve the success rate and speed of the subsequent segmentation. Generally, the preprocessing steps of a picture are as follows: to obtain the initial image, graying & binaryzation, de-noising and characters segmentation.

There are many kinds of graying algorithms, such as weighted average algorithm, maximum value method, and average value method. Using maximum value method can get a high brightness image. Using average value of the method will form a relatively soft gray image. Using weighted average method can get the most appropriate. In practice, the color image is converted into gray image using formula (1),  $F$  is gray value, and the image is stored in RGB model.[4]

$$F = 0.229 * R + 0.587 * G + 0.114 * B \quad (1)$$

The gray level of the 24 bit RGB images are 256 gray levels of 8 bits. The gray level is reduced to 2 gray levels, a two value image is obtained. If the 0 represents the target pixels, 1 is on behalf of the background color, then the binaryzation of a gray image can be expressed by the following:

$$g(x, y) = \begin{cases} 0, & f(x, y) > T \\ 1, & f(x, y) \leq T \end{cases} \quad (2)$$

Where  $T$  is a determined threshold. The methods of determining the threshold value are the following: manual setting threshold, p-quantile method, Otsu method, the optimal threshold method. Manually set the threshold is direct threshold input, the threshold range is 0 to 255. It is characterized by very fast calculation speed. It can save time in large quantities of CAPTCHAs attacking, get a higher attacking success rate. Usually,  $T$  taken as 150 is more appropriate.

Noise removal algorithms include connected domain de-noising method, Hough transform de-noising method, spatial domain filtering based algorithm, etc. Connected domain de-noising method is widely used in the identification. Its basic step is to detect all the connected domains in the image, and then the number of black pixels in each connected domain is then counted. Then a threshold is determined, the entire connected domain whose number of black pixels is less than the threshold is removed. Hough transform can detect the specific geometry in the image. Straight interference lines can be eliminated by Hough transform based detection. The spatial domain filtering based algorithm is one of the most widely used techniques in digital image processing. It

can improve the image quality, including smoothing image, removing noise, sharpening image and so on.[3]

Conventional characters segmentation methods of CAPTCHA include pixel projection, connected domain segmentation and upper and lower profile projection method. Pixel projection segmentation can only deal with the relatively simple image, it is difficult to segment the overlapping image. Connected domain de-noising method can get the size of each connected domain in the stage of de-noising, after the removal of small area, the remaining parts are separated characters. The advantage of this algorithm is it will not be affected by the distortion, inclination, and deformation of the characters. The upper and lower profile projection method is only suitable for the case of minor adhesion. In addition, there are also segmentation method based on the character width and the number of characters.

### The Preprocessing and Characters Segmentation of CAPTCHAs

In this section, a kind of de-noising and characters segmentation method for CAPTCHA with colored interference lines is designed. The CAPTCHAs from [www.58.com](http://www.58.com) are taken as examples to illustrate and verify the effectiveness of the algorithm.

First, the pictures from the site are download and saved as BMP format. Gray images are obtained using formula (1). Then the threshold is set as 150 to binaryzation. Suppose the gray value of each pixel is  $F$ , and the following processing is carried out.

$$\begin{cases} \text{if } F > 150; & F = 255 \\ \text{if } F \leq 150; & F = 0 \end{cases} \quad (3)$$

The results are showed as Table 2.

There is an obvious border after the binaryzation in some of the samples obtained. Considering this feature is very obvious, it is a single pixel wide black line around the picture, it can be removed in advance. The simplest method is adopted, the gray value of a pixel wide region around the image is directly set to 255. For an image whose size is  $a*b$ , the gray value  $inbmp[i][j]$  of pixel  $[i][j]$  on the image is set to 255, show as formula (4):

$$\begin{cases} \text{if } i == 1 & inbmp[i][j] = 255 \\ \text{if } i == a & inbmp[i][j] = 255 \\ \text{if } j == 1 & inbmp[i][j] = 255 \\ \text{if } j == b & inbmp[i][j] = 255 \end{cases} \quad (4)$$

There are no borders in all of the pictures after the above processing, the effects are showed as Table 3.

Table 2. Comparison before and after binaryzation













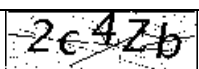
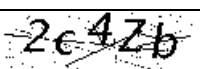


| Before binaryzation   | After binaryzation  |
|---|---|
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Table 3. The effects of border removal

| Before border removal  | After border removal  |
|--|---|
|  |  |
|  |  |
|  |  |

The CAPTCHA with multiple colored interference lines is studied. This type of CAPTCHAs usually has a plurality of interference lines and a large number of noises. In order to enable users to quickly distinguish between characters and interference lines, and the user experience will not be affected, the width of the interference line is set thinner than the character strokes, and because of the limited size of the CAPTCHA image, the interference lines are often set to one pixel wide. Finally, the algorithm is designed based on this observation to eliminate the interference lines.

The core of the algorithm is as following, traverse each pixel of the binary diagram, each pixel whose transverse width or longitudinal width is one pixel is set to white. After the traversal, most of the line interferences and noises can be deleted. For an image whose length is “ $a$ ” pixels and width is “ $b$ ” pixels, the gray value of pixel point  $(i,j)$  is  $inbmp[i][j]$ . When formula (5) and (6) or formula (5) and (7) are simultaneously satisfied, set  $inbmp[i][j]=255$ .

$$inbmp[i][j] = 0 \quad (5)$$

$$\begin{cases} inbmp[i-1][j] == 255 \\ inbmp[i+1][j] == 255 \end{cases} \quad (6)$$

$$\begin{cases} inbmp[i][j-1] == 255 \\ inbmp[i][j+1] == 255 \end{cases} \quad (7)$$

After using the above method, most interference lines and noises are removed successively, the picture has become relatively "clean", but there are still a minority of interference lines or noises left, by comparison and analysis, most of these are regions interference lines cross. The results are showed as Table 4.

Because the goal of this method is to eliminate the interference lines of one pixel wide, so when multiple interference lines cross, the width of interference lines will exceed one pixel. But there are not too much such regions, the reason is as mentioned earlier, it is easy to distinguish between interference lines and character strokes. So the remained interference regions can be further removed using connected domain de-noising method. Then the image is processed by connected domain method, threshold is set to 10, the size of less than 10 pixels connected domains are all set white. Table 5 is the result of the treatment.

Table 4. Comparison before and after interference lines removal

| Before interference lines removal | after interference lines removal |
|-----------------------------------|----------------------------------|
|                                   |                                  |
|                                   |                                  |
|                                   |                                  |
|                                   |                                  |
|                                   |                                  |

Table 5. Comparison before and after connected domain de-noising

| before connected domain de-noising | after connected domain de-noising |
|------------------------------------|-----------------------------------|
|                                    |                                   |
|                                    |                                   |
|                                    |                                   |
|                                    |                                   |
|                                    |                                   |

On the basis of above, 200 samples are processed. Most of them can be well treated same as in Table 5. But there are a small number of samples have some problems. The problems are divided into two major categories:

- CAPTCHA breaking.
- characters are not successfully separated but are connected with the interference lines that are not removed successfully.

The distribution of these cases in the samples showed as Table 6:

Although the number of breaking samples is relatively high. But after analysis, they are all character “h” and “n”, showed as Fig. 1. The breaking happened at the strokes connections. Because there will be a small number of pixels whose left and right or up and down width is 1. The effect of the two kinds of faults are as Fig. 1.

Table 6. The de-noising result of the method in this paper

| total samples | normal samples | breaking samples | connected samples |
|---------------|----------------|------------------|-------------------|
| 200           | 157            | 41               | 2                 |

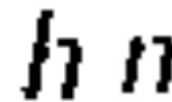


Fig. 1. Breaking characters

## Characters Segmentation of CAPTCHAs

In the process of preprocessing, because connected domain method is used, each connected domain in the picture is given different marks. For normal samples, the number of connected domains in which it contains is exactly 5, that is, 5 verification characters. And a sample of more than 5 connected domains is due to the inclusion of the letter h or n. Taking 6 connected domain as an example. For each connected domain, the starting and ending points in the horizontal ordinate are given as  $s_1, s_2, s_3, s_4, s_5, s_6, e_1, e_2, e_3, e_4, e_5, e_6$ . The projection width  $w_i = e_i - s_i$  ( $i=1,2,3,4,5,6$ ) is calculated. Then, the two connected domains with the narrowest width can be merged into one. For

a minority of connected samples, the algorithm is difficult to separate the parts of adhesion.

## The Analysis of Segmentation Results

The success rate of this method is 78.5%. Comparing with conventional connected domain de-noising and projection segmentation, it can deal with more complicated images.

Using CAPTCHAs from [www.58.com](http://www.58.com), the validity of the proposed method is verified. On the basis of above job, the CAPTCHAs from Youku, Bank of communications, Bank of China are processed with this method. The results are showed as Table 7, 8, 9 respectively.

From the job above, For CAPTCHAs with multiple colored interference lines, the algorithm presented in this paper can get good effect. Because in order to let the user can easily distinguish between the interference lines and the strokes of characters, and subject to the size of CAPTCHA image, the interference lines is usually set to a pixel width, so the algorithm presented in this paper has good generality.

Table 7. Processing effect of CAPTCHAs from youku.com




| Before processing   | After processing  |
|---|---|
|   |   |
|  |  |
|  |  |

Table 8. Processing effect of CAPTCHAs from bankcomm.com


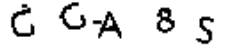

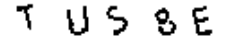

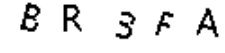
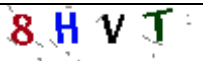
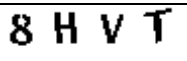

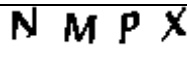
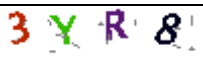
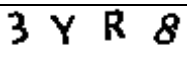
| Before processing  | After processing  |
|--|---|
|    |    |
|  |  |
|  |  |

Table 9. Processing effect of CAPTCHAs from abchina.com/cn/

| Before processing   | After processing  |
|---|---|
|  |  |
|  |  |
|  |  |

## Conclusion

Text-based CAPTCHA is currently the type of CAPTCHAs the most studied, the most widely used. CAPTCHA design flaws and shortcomings can be found by the research on CAPTCHAs attacking, thereby improving the design method.

In this paper, a de-noising and characters segmentation method based on the width difference

between the interference lines and the character strokes is presented. Take CAPTCHAs from 58.com as an example, the effectiveness of the proposed algorithm is verified. A simple method to solve the character breaking is also given.

Its shortcoming is also found, a minority of adhesions are left and difficult to deal with. This is the job to be done further.

### **Acknowledgement**

Zhao Wang is the corresponding author of this paper. We would like to thank the anonymous reviewers for their helpful suggestions. This work was sponsored by the NSFC under grant No. 61371131 and China Scholarship Council (CSC).

### **References**

- [1] Li Qiujie, Mao Yaobin, Wang Zhiquan, A Survey of CAPTCHA Technology, Journal of Computer Research and Development[J].2012 49(3):469-480.(in Chinese)
- [2] Gao H C, Wang W, Fan Y. Divide and Conquer: An Efficient Attack on Yahoo! CAPTCHA . Proc of the 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications[C]. 2012: 9-16
- [3] Yefei, Research on General Attack on Text-Based CAPTCHA[D]. Xi Dian University.2014. (in Chinese)
- [4] Yang Sifa, The Research and Implementation of CAPTCHA [D]. Nanjing University of science and technology.2014. (in Chinese)