Captcha Detection

Midterm Report

Name: Anrui Liang

UIN: 726007240

Class: ECEN 765

1. Introduction

Image processing may be one of the most important part in machine learning. With the experience in childhood, I want to use the knowledge in class to deal with captcha detection problem.

The project mainly contains three parts --- data collection, data processing and using machine learning knowledge to solve detection problem. Before the midterm, my goal is to finish data collection part and data processing part.

2. Data collection

There may be lines, characters and numbers in one captcha. Some regions may use special characters in captcha.(Shown in figure 1)

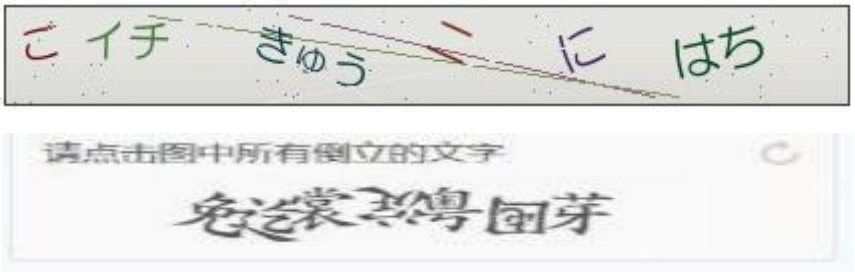

Figure 1. Different characters in captcha

To solve the problem more efficiently, before the midterm, I will choose capital characters and numbers with lines in captcha.

Captcha detection is different from handwriting detection. Captcha picture can be generated by computer easily. It also means that I can program to get a large number of training data and test data in this project.

For increasing the complexity of the captcha, I add some lines as noise and change the shape of captcha.

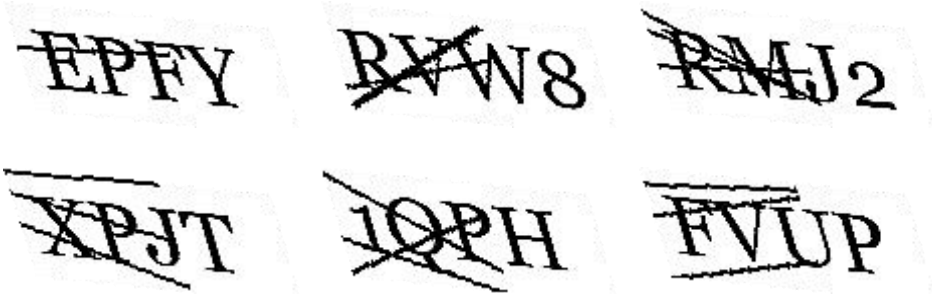

Figure 2. Captcha

3. Data Processing

Captcha detection is more difficult than hand writing detection. In homework 1, hand writing data is easily to get either from MNIST or from existing data set. However, I did not know how to get these data.

I think data processing in captcha detection should contain five parts. 1) Transfer picture to black white picture. 2) Delete frame. 3) Eliminate point noise. 4) Transfer picture to 0-1 matrix. 5) Divide picture to every digit.

3.1 Transfer Picture to Black White Picture

Color is useless for us to classify captcha in this project. However, color may increase complexity of project. I use OpenCV to deal with captchas. I use the function cvtColor to change the color and the result is shown as:

Figure 3. Capture with White and Black

And the picture can be also shown as a matrix which only contains 0 (means black) and 255 (means white). And then, the matrix can be used to transfer the picture to 0-1 matrix easily.

```
[[255 255 255 ... 255 255 255]
 [255 255 255 ... 255 255 255]
 [  0 255 255 ... 255 255 255]
 ...
 [255 255 255 ...   0 255 255]
 [255 255 255 ...   0 255 255]
 [255 255 255 ...   0 255 255]]
```

Figure 4. Capture Shown as Matrix

3.2 Delete Frame and Gain 0-1 Matrix

Because the matrix is gained, I can transfer it to matrix with 0 and 1. Why this part is needed is that matrix with 0 and 1 can be more easily calculated and I can use logic calculation in the program which will be shown in next step. The process of this part is easy. I divide the matrix by 255 and transfer result from floating data to intiger data.

Figure 5. Captcha Without Frame

The same as the color, frame is useless for us to classified captchas. So frame should be eliminated. Normally, captcha may have square frame which is easy to eliminate or

not have frame. In the project, I generate a captcha without text and change it to 0-1 matrix. And then, I use captcha without frame to xor other captchas. After calculation, the frame can be eliminated.

3.3 Eliminate Point Noise

Point noise may be generated because of changing colors or original picture. There are many methods to solve this problem and I directly use a filter function to solve this problem.

3.4 Divide Picture to Digits

I want to divide picture to every digits, so I can solve this problem as MNIST problem and make it easy. For this part, I directly divide the picture which contains 4 digits to 4 parts. However, one character's size is different from one number's size. So I may update this part in my next step.

4 Next Steps

In my next step, I want to find a good way to divide captcha to decrease error rate of my model.

I may start to use the data to train models. With the knowledge in ECEN 765, I may use Naive Bayes and KNN as my models. If I have more time, I may use some deep learning method to solve this problem like CNN.

In Naive Bayes and KNN models, I want to solve some problems such as what's the best number of neighbors in my project and other details in these two models.