



简单的验证码识别

验证码又称全自动区分计算机和人类的图灵测试(Completely Automated Public Turing test to tell Computers and Humans Apart, 简称CAPTCHA)，俗称验证码，是一种区分用户是计算机和人的公共全自动程序。在CAPTCHA测试中，作为服务器的计算机会自动生成一个问题由用户来解答。这个问题可以由计算机生成并评判，但是必须只有人类才能解答。由于计算机无法解答CAPTCHA的问题，所以回答出问题的用户就可以被认为是人类。
(from wikipedia)

扩展

登录名:

用户名/邮箱

密码:

[忘记用户名/密码?](#)

验证码:

请点击下图中**所有的** **瓷砖**

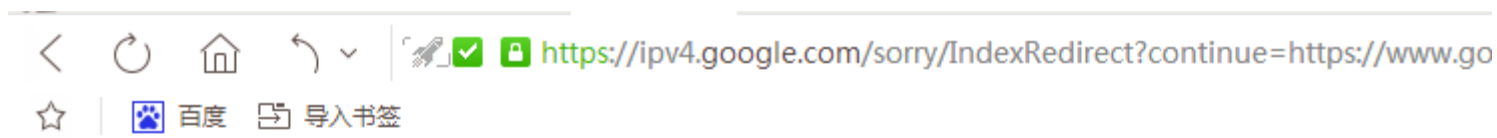
 刷新



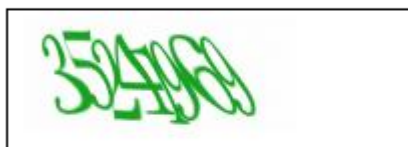
[验证码如何使用?](#)

登录

快速注册



请键入下图显示的字符以继续操作：



关于此网页

我们的系统检测到您的计算机网络中存在异常流量。此网页用于确认这些请求是由您而不是自动程序发出的。[为什么会这样？](#)

IP 地址: [REDACTED]

时间: 2015-11-25T10:44:11Z

网址: [https://www.google.com.hk/search?](https://www.google.com.hk/search?q=ip%20地址&sa=N&biw=1366&bih=608)

[q=inurl:%22php%3Fid%3D%22&safe=strict&ei=v5BVVq7KIYKloQSD14CABw&start=10&sa=N&biw=1366&bih=608](https://www.google.com.hk/search?q=ip%20地址&sa=N&biw=1366&bih=608)

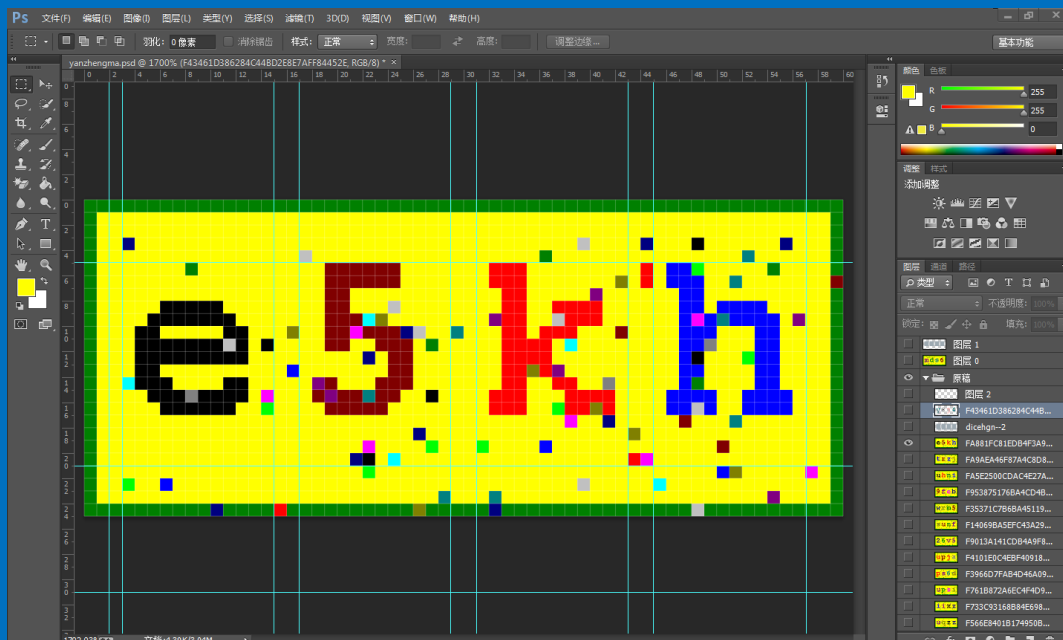


图片分析

--图片是不是有相同点？

工具

photoshop CC



1.找规律

图片尺寸大小？

字符位置？？

字符尺寸??

字符种类？？





图片分析

图片分析

1. 找规律

txj9 000F2AA8 3E664FC0 BF88F0870 96C5CB8...	7kj3 00A6C6CF B43343BB BDB5BF84 27AA8A1...	rc25 00A7CD4B DEFE4812 BE62FC4D 0056A29...	bfwm 00A10A8C 6C764A18 A2730288 42B8993...	anyv 00A23BFA F3CB44A8 AE7ABE0A 757473E...	4iuj 00A38FBA CD9549B1 BE506CF8 2EC19C0...	4n9m 00A191E1 28F44B7C 9CB17AA A036EBF...	ak8y 00A406E1 C79F4F96B 231977CC B22F7B9...	y78s 00A75626 659D4F2F8 7282FEF37 6D8D68...	jug6 00AB38BC 77324E04 A88E4263 04365E9...	i26a 00AD1D6E 45794E1E8 A6881137 A3B8373...	hxx2 00AE311E C2214B57 989127F90 1B7472E...	yz7u 00AFF2731 9234B3B8 2AC7D38D EDE4F1A...
x6gp 00B1C30C 69F04FF5B C9D1E4A4 581141A...	2w5d 00B8EB4E 54AE4655 88504D0FF 61F3B6B...	j529 00B23A33 16E14D05 88504D0FF 67FA6AA...	8neu 00B40CBF 149447E8 BFF3462E DBAC397...	4s7z 00B0627B BE4B441D ADD19406 A133914...	ziuf 00B08767 C51F46649 7039EB88 F8AB41E...	eux4 00BEE946 6E60423C BF02238BF 56E6890...	f27j 00C0B4F5 36074E748 B00F1C14 B632615...	ta36 00C2DADE F67D41CF B2252198 5AFF7CE...	74uk 00C3DB88 625B4E92 A8453E86 7C425A9...	r6xu 00C8E4CB 159B4C03 BE3158D5 7EC2220...	hxf7 00C9B708 4A4948CA AC7E45B2 DC144F1...	lrhi 00C479ED B7974CB1 9DF71878 9D02AC0...
7vvt 00C605A2 3CE0470C A1658B3B 6F17524...	54vq 00C72510 0A4246E4 ADFF8282 E56DFC5...	4giq 00C74602 27514E97 B150543B BCDD04...	n8zf 00C86028 4C164744 81401D38 F11B47C...	3jcg 00CF9C2A F1664933A E7A3EE18 CE43521...	hpd4 00CFF8EA C2BA4139 97F37054C 6598D1E...	i7xs 00D8C85F 27C7425F8 A2C2EFAC B5FFA5B...	akcr 00D30A77 C89A4368 A916C555 9AADD7...	u73j 00D35694 0D974329 8233A21F DBF4ECF...	5nb9 00D36672 3D844CF5 9788F817 C8301B8...	wyxn 00D62205 2916496E B82E2CA0 BF01001...	z85a 00D8D979 E0B0410C AE6C20FD 219C720...	7inr 00DC57A2 BD654ECA 8C38C7C6 71D75F7...
njjj 00DDE462 43F947279 BAB79198 FA38232...	nev4 00E1AED6 B2CD4837 9BD48DD 065F326...	p5w4 00E96CCB A9D94C83 9D9287E7 1F0CF3E...	dhb9 00E97AAA 3499433A A49E9F242 53C59D5...	xea2 00E535283 54246688 519AE840 A609330...	u8p6 00EAD544 FDC04364 934430091 912F40E...	wbia 00EB491C DB384C6A A0AAAFB4 88DE806...	g39z 00F0CCFF 96043C9B AB3DD1 2C29894...	f7m4 00F3AD5B B7B64D42 BDD75D85 EB47D7F...	bwqu 00F35F09C AF944858 C77AD74C E369550...	0kvr 00F9598B4 D3044709 1A598680 8F9452A...	8erw 00FA930D 83C54144 BC93FDC4 6A7C254...	89k6 00FAF68FA 40D48349 E2DA067F 19C2516...
vamj 00FB512CF 4C84529A C157C9A0 CF48BDE...	ms8u 00FBF0EA6 EDCA471A 99972775F D083A6...	mds6 0.bmp	eeaa 0A0AA249 AFCC45D6 A96D32C3 9530594...	n9tb 0A0AFE03 2C164C18 8E72D14F 38FED2F...	bwyt 0A0C69F8 7BF045F09 DD107852 50B6748...	kexy 0A0CCCA1 882E4DA8 BBD9C681 9F9369E...	msf1 0A0CEB6C 367C4729 A41F316D 9247D2A...	rj7c 0A0D52C8 987B4CB2 AFB8E529 7789BA5...	kep2 0A0EADD D4BA945D F938BD59 6064D51...	tjbjk 0A0F4D1E 871046FD B7312072 1C62AD9...	nenh 0A1B335E A37644C1 89E31A94 1505C88...	f1pe 0A1E748E 312D4283 BD655EBA 43DF1D...





2.分析



Handwritten white scribbles and symbols, including a small circle and some lines, located below the main grid.



2.分析



Hand-drawn white lines and a small drawing of a person are visible on the right side of the image.



2.分析

图片尺寸大小 25*60

字符位置—固定

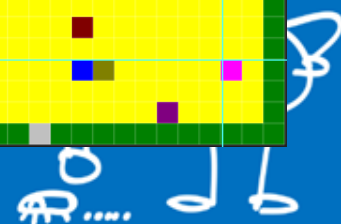
字符尺寸—固定

字符种类2-9a-z没有l和0（欧）





2.分析

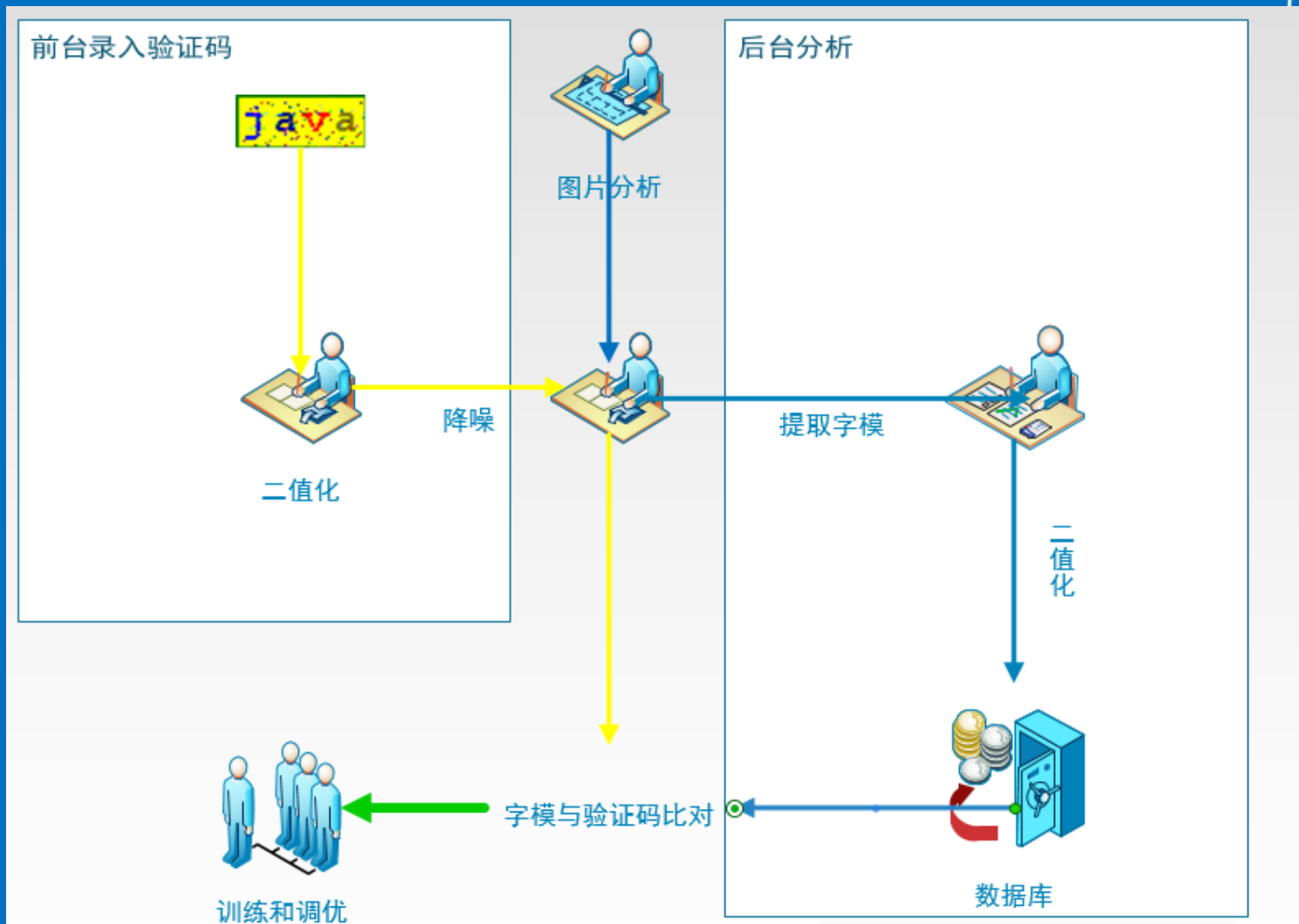




处理方法

--流程

图片分析





图片分析

处理方法

--图片降噪

去除无用点

工具:

1.photoshop CC

语言:

java or python

直接ps去噪点



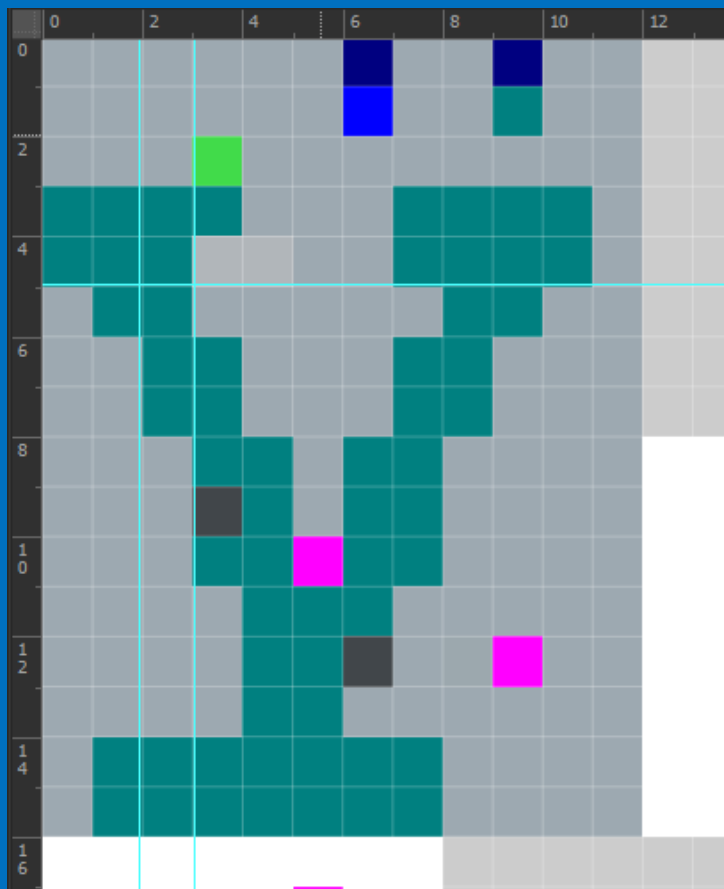


处理方法

--图片降噪

图片分析

去除无用点



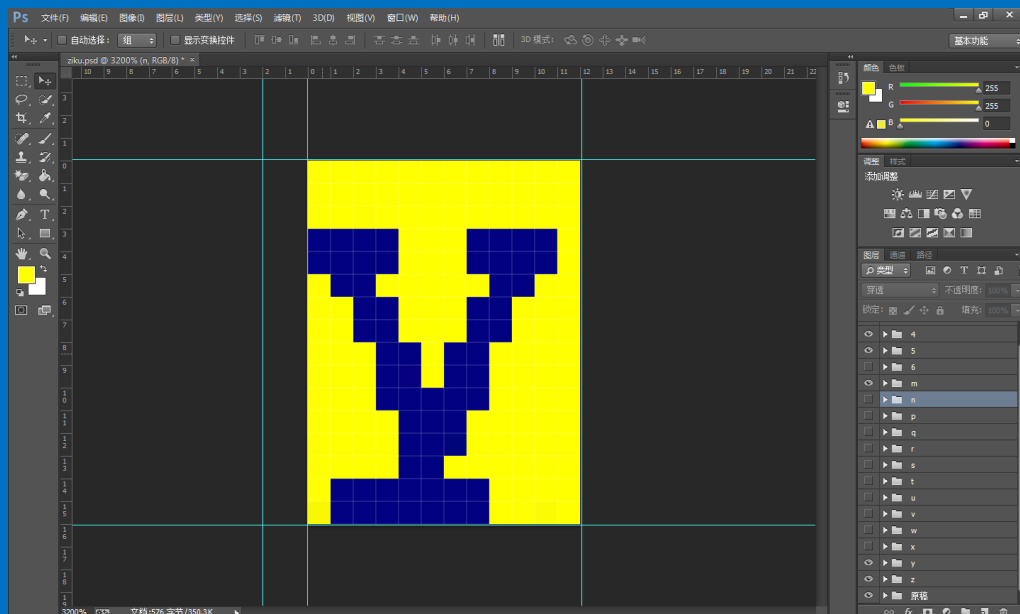


处理方法

--图片降噪

图片分析

去除无用点





处理方法

--提取字模

图片分析

去除无用点





处理方法

--提取字模

图片分析

去除无用点





图片分析

处理方法

--二值化

去除无用点

工具:

1.netbeans



2.Editplus



语言

1.java or python(PIL库)





处理方法

--二值化

把图像转换成二维数组

```
47 public static int[][] getErZhiHua( BufferedImage bi) throws IOException {  
48     /// File file = new File("E:\\png.png");  
49  
50     //   BufferedImage bi = ImageIO.read(file);  
51     int h = bi.getHeight();  
52     int w=bi.getWidth();  
53     int a[][] = new int[h][w];  
54     int rgb=bi.getRGB(0, 0);  
55     int x=0;  
56     // int arr[][]=new int[w][h];  
57     // int x = bi.getRGB(2, 4);  
58     // System.out.println(x);  
59     for(int i=0;i<h;i++){  
60         for(int j=0;j<w;j++){  
61             a[i][j]= bi.getRGB(j, i)==-256?0:1;  
62         }  
63     }  
64  
65     }  
66     return a;  
67  
68 }
```



处理方法

--二值化

图片分析

把图像转换成二维数组

```
000000000000
000000000000
001000000000
111100011110
111100011110
011000001100
001100011000
001100011001
000110110000
000110110000
000111110000
000011100000
000011100000
000011000000
011111110000
111111110010
```





把图像转换成二维数组

[illegible]



把图像转换成二维数组

[illegible]



图片分析

处理方法

--二值化

把图像转换成二维数组

```
148 public static final int ZI_7 [][] = {  
149     {0,1,1,1,1,1,1,1,1,0,0,0},  
150     {0,1,1,1,1,1,1,1,1,0,0,0},  
151     {0,1,1,0,0,0,0,1,1,0,0,0},  
152     {0,0,0,0,0,0,1,1,1,0,0,0},  
153     {0,0,0,0,0,0,1,1,0,0,0,0},  
154     {0,0,0,0,0,0,1,1,0,0,0,0},  
155     {0,0,0,0,0,1,1,1,0,0,0,0},  
156     {0,0,0,0,0,1,1,0,0,0,0,0},  
157     {0,0,0,0,0,1,1,0,0,0,0,0},  
158     {0,0,0,0,1,1,0,0,0,0,0,0},  
159     {0,0,0,0,1,1,0,0,0,0,0,0},  
160     {0,0,0,0,1,1,0,0,0,0,0,0},  
161     {0,0,0,0,0,0,0,0,0,0,0,0},  
162     {0,0,0,0,0,0,0,0,0,0,0,0},  
163     {0,0,0,0,0,0,0,0,0,1,0,0},  
164     {0,0,0,0,0,0,0,0,0,0,0,0}  
165 };
```





处理方法

--识别训练

图片分析

1.相似度





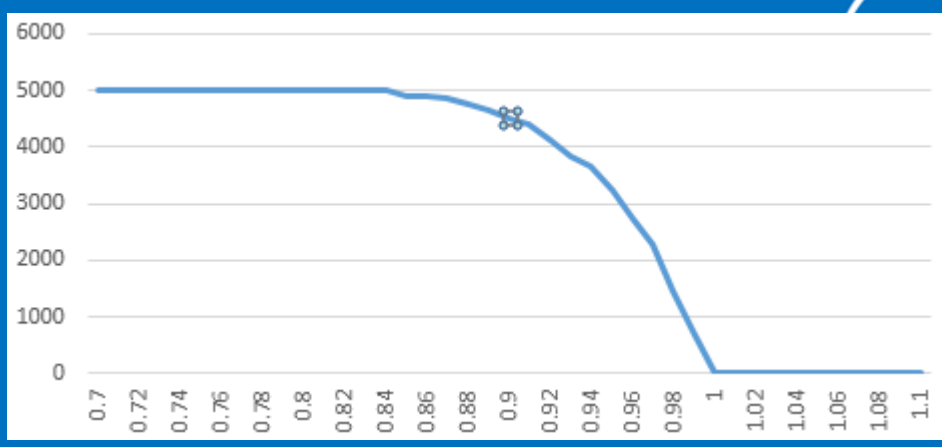
图片分析

处理方法

--识别训练

1.相似度

序号	相似度	未识别	错误率
	0.88	248	
	0.89	339	
	0.9	484	
	0.88	248	





图片分析

处理方法

--识别训练

1.相似度

0.89



1.相似度

找错





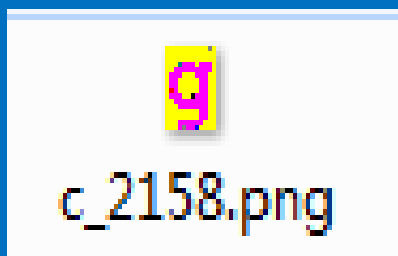
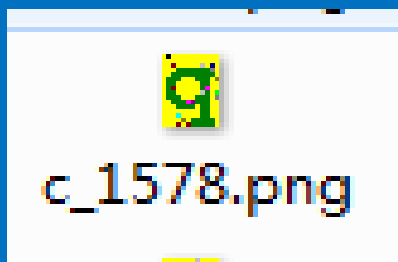
图片分析

处理方法

--识别训练

1.相似度

错误字符的处理



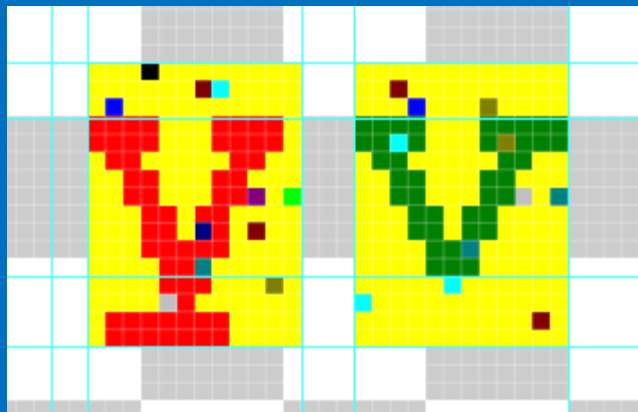
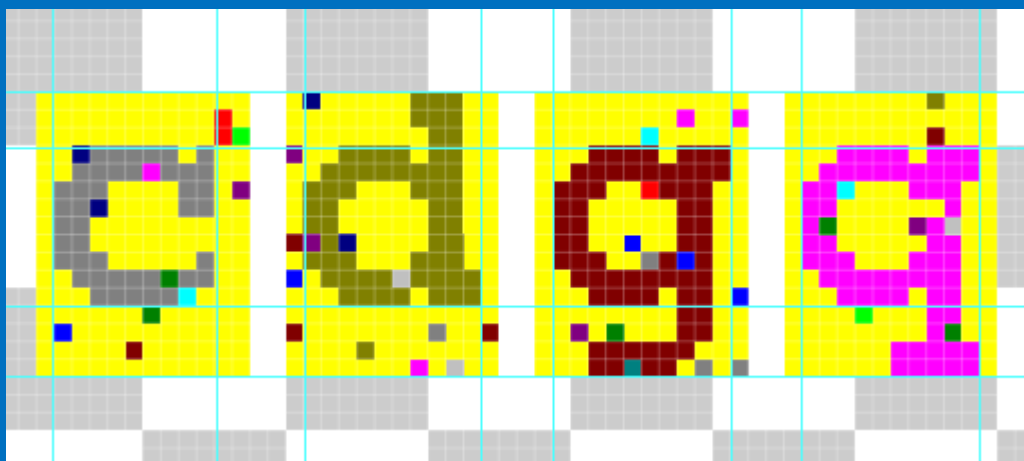


处理方法

--识别训练

图片分析

对相似的字符进行区分



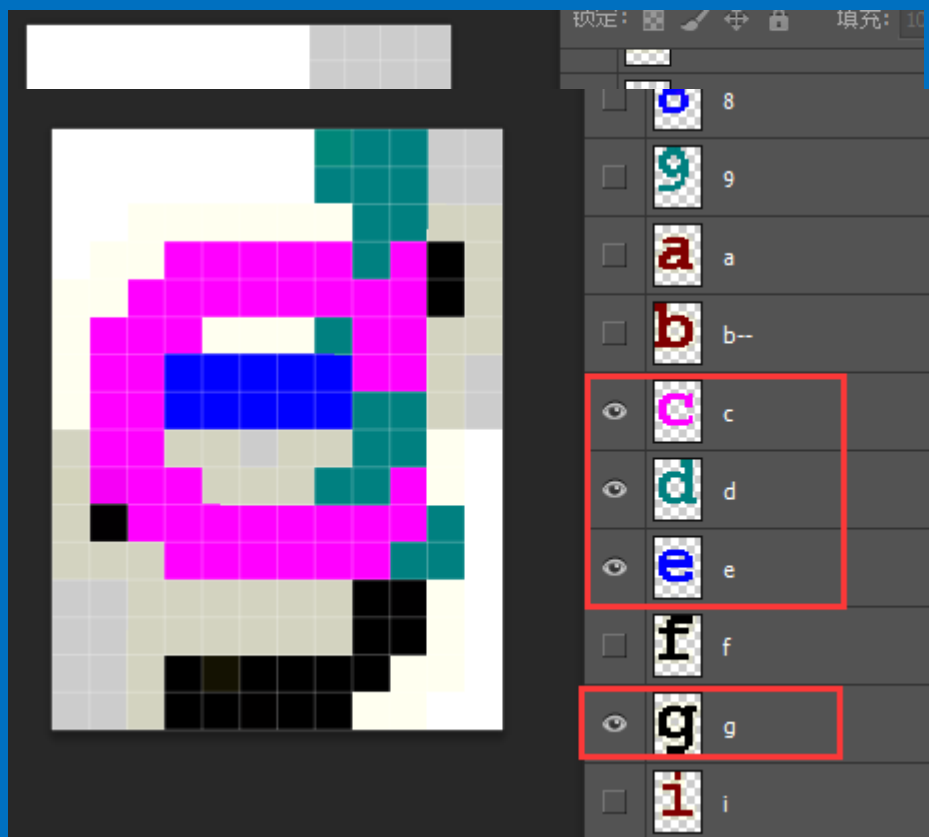


处理方法

--识别训练

图片分析

对相似的字符进行区分





图片分析

处理方法

--识别训练

对相似的字符进行区分

```
78         if("c".equals(next)){
79         if((tmp>0.97)){
80             //处理c d g 的d
81             {
82                 if(next.equals("c")){
83                     int m=0;
84                     for(int k=0;k<3;k++){
85                         for(int l =7;l<10;l++){
86                             if(start[k][l]==1)m++;
87                         }
88                     }
89                     if(m>5)return "d";
90                 }
```





图片分析

处理方法

--识别训练

对相似的字符进行区分

结果c d g 完全分开
YV 完全分开





图片分析

处理方法

--识别训练

对相似的字符进行区分

二次处理





图片分析

处理方法

--识别训练

对相似的字符进行区分

```
188 public static String ImageOCRTwoErCi(BufferedImage img) throws IOException{
189     int start[][]=imageTest.getErZhiHua(img);
190     Set<String>Zi=ziku.ZIs.keySet();
191     Iterator<String> iter= Zi.iterator();
192     while (iter.hasNext()) {
193         String next = iter.next();
194         System.out.println("-----"+next);
195         // ziku.Print_zi(ziku.ZIs.get(next));
196
197         int count=0;
198         int src[][]=ziku.ZIs.get(next);
199         for(int i=0;i<16;i++){
200             for(int j=0;j<12;j++){
201                 if(1==src[i][j]&&1==start[i][j]){
202                     count++;
203                 }
204             }
205         }
```





图片分析

处理方法

--识别训练

对相似的字符进行区分

C和e识别有误



处理方法

--识别训练



图片分析

对相似的字符进行区分

42207张验证码我们团队
15级的看了两个多小时没
发现两处错误





处理方法

--识别训练

图片分析

对相似的字符进行区分





处理方法

--识别训练

对相似的字符进行区分

2a2t 2a2t_257 93.png	2a3b 2a3b_24 53.png	2a3k 2a3k_16 234.png	2a7q 2a7q_87 24.png	2a7v 2a7v_25 73.png	2a25 2a25_35 97.png	2a36 2a36_10 894.png	2ab9 2ab9_87 8.png	2acf 2acf_180 24.png	2ad9 2ad9_12 214.png	2ad9 2ad9_16 480.png	2adc 2adc_14 446.png	2adm 2adm_21 90.png	2ag9 2ag9_14 194.png	2ah8 2ah8_12 726.png
2aj8 2aj8_122 4.png	2ajr 2ajr_987 1.png	2akr 2akr_753 6.png	2akz 2akz_11 554.png	2an8 2an8_97 27.png	2apq 2apq_26 558.png	2aqb 2aqb_15 467.png	2arc 2arc_135 78.png	2arx 2arx_268 42.png	2au5 2au5_20 177.png	2aup 2aup_25 466.png	2av3 2av3_16 186.png	2avt 2avt_179 39.png	2az7 2az7_43 63.png	2azj 2azj_240 37.png
2b3d 2b3d_43 95.png	2b3g 2b3g_87 80.png	2b7b 2b7b_17 02.png	2b52 2b52_43 98.png	2b59 2b59_14 22.png	2b67 2b67_41 06.png	2b69 2b69_25 492.png	2b95 2b95_19 349.png	2bc5 2bc5_21 770.png	2bcj 2bcj_207 38.png	2bdy 2bdy_20 533.png	2beb 2beb_13 266.png	2bip 2bip_587 1.png	2bja 2bja_251 92.png	2bjc 2bjc_101 96.png
2bjp 2bjp_227 71.png	2bpp 2bpp_33 48.png	2brn 2brn_16 416.png	2bwd 2bwd_64 22.png	2bx7 2bx7_16 920.png	2bxc 2bxc_11 590.png	2c2n 2c2n_22 477.png	2c3n 2c3n_76 13.png	2c4u 2c4u_16 753.png	2c5f 2c5f_623 2.png	2c6m 2c6m_15 119.png	2c7z 2c7z_224 38.png	2c8m 2c8m_21 513.png	2c8v 2c8v_740 7.png	2c9x 2c9x_147 02.png
2c87 2c87_26 409.png	2cb4 2cb4_80 77.png	2cck 2cck_120 65.png	2cdr 2cdr_946 9.png	2cg8 2cg8_14 521.png	2cgj 2cgj_251 21.png	2cgn 2cgn_14 538.png	2ch4 2ch4_26 181.png	2cik 2cik_259 2.png	2cim 2cim_57 17.png	2cka 2cka_17 920.png	2cn3 2cn3_23 211.png	2cqq 2cqq_26 300.png	2cs3 2cs3_919 .png	2cts 2cts_560 7.png
2cu5 2cu5_10 266.png	2cu8 2cu8_24 643.png	2cup 2cup_21 076.png	2cvy 2cvy_111 81.png	2cw8 2cw8_69 54.png	2cwu 2cwu_11 7.png	2cy9 2cy9_531 5.png	2cyh 2cyh_258 10.png	2cz4 2cz4_264 86.png	2d8x 2d8x_32 54.png	2d9n 2d9n_21 488.png	2d93 2d93_23 352.png	2dbx 2dbx_90 25.png	2dhn 2dhn_20 849.png	2dhz 2dhz_90 81.png
2dig 2dig_847 0.png	2dj4 2dj4_184 90.png	2dk6 2dk6_14 293.png	2dqy 2dqy_25 24.png	2dqy 2dqy_17 711.png	2dst 2dst_845 9.png	2dv7 2dv7_19 557.png	2dwj 2dwj_15 002.png	2dy7 2dy7_12 774.png	2dyq 2dyq_11 94.png	2dyz 2dyz_97 7.png	2dzf 2dzf_248 31.png	2e2k 2e2k_26 315.png	2e4i 2e4i_547 .png	2e8k 2e8k_21 135.png
2e9t 2e9t_838 7.png	2e9v 2e9v_41 46.png	2e52 2e52_58 49.png	2eaw 2eaw_56 54.png	2eaw 2eaw_10 878.png	2eb6 2eb6_15 391.png	2ecz 2ecz_207 59.png	2ee7 2ee7_16 351.png	2een 2een_63 86.png	2eip 2eip_140 04.png	2eit 2eit_681 2.png	2eit 2eit_160 92.png	2ep6 2ep6_61 72.png	2er7 2er7_150 1.png	2esj 2esj_121 47.png

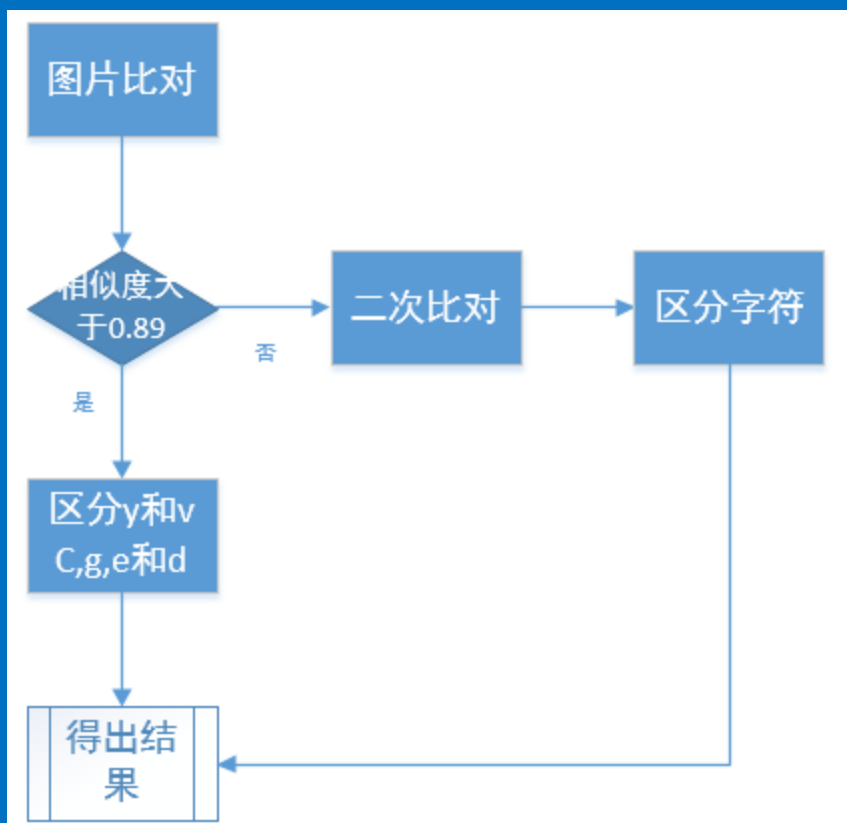


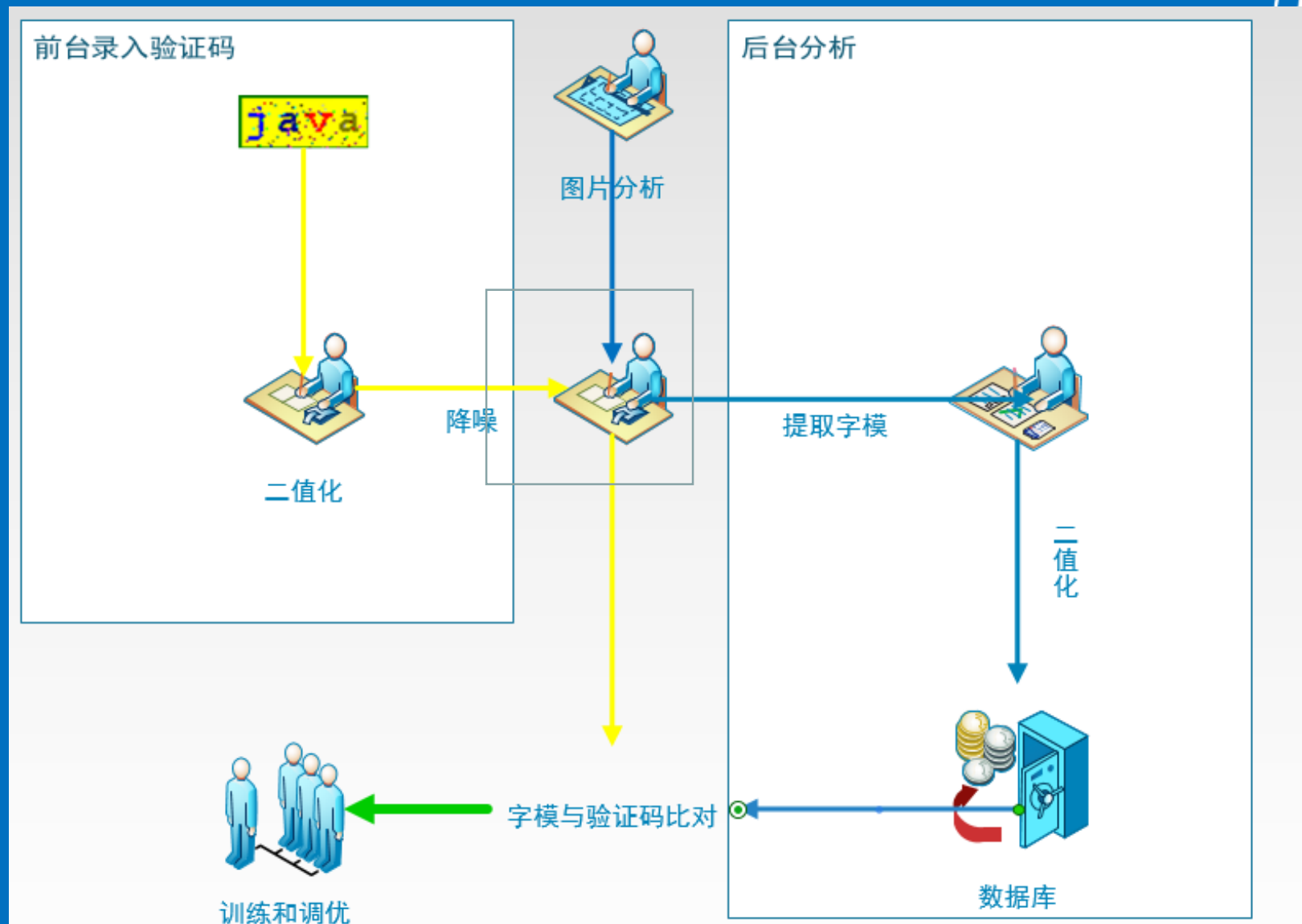
图片分析

处理方法

--识别训练

对相似的字符进行区分







图片分析

总结

--所用技术

图片处理

--photoshop
--java imageIO类

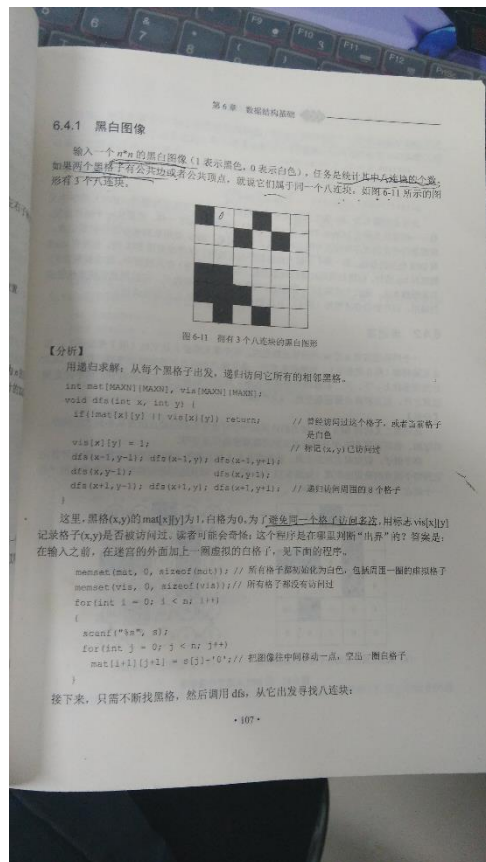
正则匹配

---editplus



1. 图片降噪的算法

数据结构中八连块问题



算法竞赛入门经典107页八
连块问题—深度优先搜索

2.字模矩阵压缩

压缩矩阵

-----参考数据结构严蔚敏版

3.*****

还有更好的优化方法，更好的验证码识别算法

以下摘自网络

分类

纯文本型验证码

图型验证码

纯文本型验证码

- $1+1=?$
- 本论坛的域名是?
- 今天是星期几?
- 复杂点的数学运算

图形验证码

识别图形验证码可以说是计算机科学里的一项重要课题，涉及到计算机图形学，机器学习，机器视觉，人工智能等等高深领域.....

简单地说，计算机图形学的主要研究内容就是研究如何在计算机中表示图形、以及利用计算机进行图形的计算、处理和显示的相关原理与算法。图形通常由点、线、面、体等几何元素和灰度、色彩、线型、线宽等非几何属性组成。计算机涉及到的几何图形处理一般有 2维到n维图形处理，边界区分，面积计算，体积计算，扭曲变形校正。对于颜色则有色彩空间的计算与转换，图形上色，阴影，色差处理等等

在识别验证码中需要用到的知识一般是 像素，线，面等基本 2 维图形元素的处理和色差分析。常见工具为：

- 支持向量机(SVM)
- OpenCV
- 图像处理软件(Photoshop,Gimp...)
- Python Image Library

后续知识请自行搜索

- 1.勤用搜索引擎（学好了可以装逼）
- 2.多和别人交流或者听听别人（大师）的解决问题的方法
- 3.请尽快把想法付诸实践

thanks

!

