摘 要

智能交通系统利用先进的信息技术改善交通状况，使交通更畅通、更安全、更绿色。车牌识别系统是的核心技术之一，它主要包括车牌定位、字符分割和字符识别三个核心模块。随着安防视频步入高清时代，视频的分辨率越来越高，智能交通系统对车牌识别技术有了更高的要求：处理速度更快、环境适应性更强、识别率更高。

本文从预处理、边缘检测、车牌定位、字符分割、字符识别五个方面，具体介绍了车牌自动识别的原理。并用MATLAB软件编程来实现每一个部分,最后识别出汽车牌照。

关键词：牌照识别； Matlab；边缘检测

**Abstract**

Intelligent Transportation System use the advanced information technology, it make the traffic more currency and more security .The automatic license plate recognition system is the key of ITS.It include license location, character segmentation, character recognition. With the development of the technology, high-definition of the picture, ITS need more reliable recognition of the license plate, faster, environmental adaptability, recognition rate.

This paper includes pre-processing, edge detection, license plate recognition, character segmentation, character recognition. Introduce the principle of the automatic license plate recognition system. Achieve every part by Matlab, and recognize the numbers in the end.

**Keywords:**License Plate Recognition; Matlab; Edge Detection

# 1 绪 论

1.1 车牌识别技术研究背景

近年来，车辆数量和交通设施随着经济的快速增长而增长，但是交通设施的增长速度远远落后于车辆数量的增长速度，这引发了交通拥堵、交通事故、环境污染等难题。为了在根本上解决交通难题，世界各国纷纷利用先进的信息技术研究智能交通系统。

智能交通系统是一种充分利用各种先进的高新技术来实现实吋、准确、高效的交通管理系统，使交通更畅通更安全；它也是一种交通信息服务系统，使人们出行更方便更快捷。随着智能交通系统的快速发展，智能交通系统已经融入人们的日常生活，使人们的生活越来越方便。车辆是智能交通系统中的重点研究对象，每辆车都有自身唯一的车牌号码，车牌号码反映了车辆信息以及关联着车主信息，通过车牌号码可以记录对应车辆的交通行为，因此，车牌识别技术是智能交通系统中最核心最基础的技术之一，决定着智能交通系统的发展速度和技术水平。

1.2 国内外发展现状

自从车牌识别技术提出以来，各国研究员己经进行了大量的深入研究，如今已经出现了很多的车牌识别技术方法，有的已经很成熟，已经投入实际使用，但是由于室外自然环境光线变化以及车牌常常被污染磨损等不利条件的影响，使得应用在实际复杂环境下的车牌识别系统的识别率往往不尽如人意。国外现有的优秀的车牌识别系统产品主要有：新加坡Opstasia公司的VLPRS产品，德国西门子公司的ARTEM7S系统，以色列Hi-Tech公司的See/Car System系列产品等。国内做得较好的车牌识别系统产品主要有：亚洲视觉科技有限公司的系列产品，昆明利普机器视觉工程有限公司的LPR产品。

2 设计原理

牌照自动识别是一项利用车辆的动态视频或静态图像进行牌照号码、牌照颜色自动识别的模式识别技术。其硬件基础一般包括触发设备、摄像设备、照明设备、图像采集设备、识别车牌号码的处理机等，其软件核心包括车牌定位算法、车牌字符分割算法和光学字符识别算法等。某些牌照识别系统还具有通过视频图像判断车辆驶入视野的功能称之为视频车辆检测。一个完整的牌照识别系统应包括车辆检测、图像采集、牌照识别等几部分。当车辆检测部分检测到车辆到达时触发图像采集单元，采集当前的视频图像。牌照识别单元对图像进行处理，定位出牌照位置，再将牌照中的字符分割出来进行识别，然后组成牌照号码输出。

2.1 牌照定位

自然环境下，汽车图像背景复杂、光照不均匀，如何在自然背景中准确地确定牌照区域是整个识别过程的关键。首先对采集到的视频图像进行大范围相关搜索，找到符合汽车牌照特征的若干区域作为候选区，然后对这些侯选区域做进一步分析、评判，最后选定一个最佳的区域作为牌照区域，并将其从图象中分割出来。

流程图:



2.2 牌照字符分割

完成牌照区域的定位后，再将牌照区域分割成单个字符，然后进行识别。字符分割一般采用垂直投影法。由于字符在垂直方向上的投影必然在字符间或字符内的间隙处取得局部最小值的附近，并且这个位置应满足牌照的字符书写格式、字符、尺寸限制和一些其他条件。利用垂直投影法对复杂环境下的汽车图像中的字符分割有较好的效果。

1. 计算水平投影进行车牌水平矫正。
2. 去掉车牌的框架。
3. 分析垂直投影找到每个字符中心的位置。
4. 按左右宽度切割出字符。



2.3 牌照字符识别

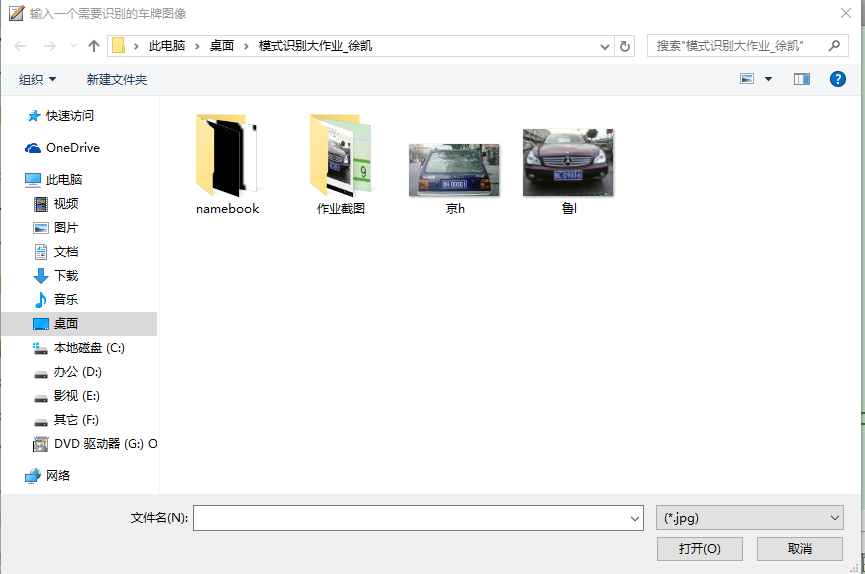
字符识别方法目前主要有基于模板匹配算法和基于人工神经网络算法。基于模板匹配算法首先将分割后的字符二值化,并将其尺寸大小缩放为字符数据库中模板的大小，然后与所有的模板进行匹配，最后选最佳匹配作为结果。基于人工神经元网络的算法有两种：一种是先对待识别字符进行特征提取，然后用所获得特征来训练神经网络分配器；另一种方法是直接把待处理图像输入网络，由网络自动实现特征提取直至识别出结果。实际应用中，牌照识别系统的识别率与牌照质量和拍摄质量密切相关。牌照质量会受到各种因素的影响，如生锈、污损、油漆剥落、字体褪色、牌照被遮挡、牌照倾斜、高亮反光、多牌照、假牌照等等；实际拍摄过程也会受到环境亮度、拍摄亮度、车辆速度等等因素的影响。这些影响因素不同程度上降低了牌照识别的识别率，也正是牌照识别系统的困难和挑战所在。为了提高识别率，除了不断的完善识别算法，还应该想办法克服各种光照条件，使采集到的图像最利于识别。



3 各个部分功能实现与分析

支持从导入原始图像，到最后的图像中的最后牌照输出到excel表格，其中经历了灰度处理，中值滤波，边缘检测，腐蚀，平滑等各种处理手段，之后产生了定位切割后的牌照图像，再对此牌照进行字符识别处理，最后输出在excel表格上。

3.1 前期的预处理阶段

图1

首先，载入原始图片，这里采用的是[filename,filepath]，可用于直接弹出对话框的选择图片，比较方便。如图1所示。

%自动弹出提示框读入图像

[filename,filepath]=uigetfile('.jpg','输入一个需要识别的车牌图像');% 直接自动读入%

file=strcat(filepath,filename); %strcat函数：连接字符串；把filepath的字符串与filename的连接，即路径/文件名

I=imread(file);

figure('name','原图'),imshow(I);title('原图')

载入原图阶段产生figure1图，如图2所示；



图2

接下来就是对图像进行处理阶段，主要经过灰度变换，中值滤波等手段。

彩色图像包含着大量的颜色信息，不但在存储上开销很大，而且在处理上也会降低系统的执行速度，因此在对图像进行识别等处理中经常将彩色图像转变为灰度图像，以加快处理速度。由彩色转换为灰度的过程叫做灰度化处理。选择的标准是经过灰度变换后，像素的动态范围增加，图像的对比度扩展，使图像变得更加清晰、细腻、容易识别。

代码如下所示；

I1=rgb2gray(I); % RGB图像转灰度图像

% %figure('name','灰度处理前'),subplot(1,2,1),imshow(I1);title('灰度处理前的灰度图');

% % subplot(1,2,2),imhist(I1);title('灰度处理前的灰度图直方图');

%线性灰度变换

I1=imadjust(I1,[0.3,0.7],[]);

figure('name','灰度处理后'),subplot(1,2,1),imshow(I1);title('灰度处理后的灰度图');

subplot(1,2,2),imhist(I1);title('灰度处理后的灰度图直方图');

%进行中值滤波

I1=medfilt2(I1);

figure,imshow(I1);title('中值滤波');

对原图进行灰度处理后的图像如图3所示；

对原图进行中值滤波后的图像如图4所示；

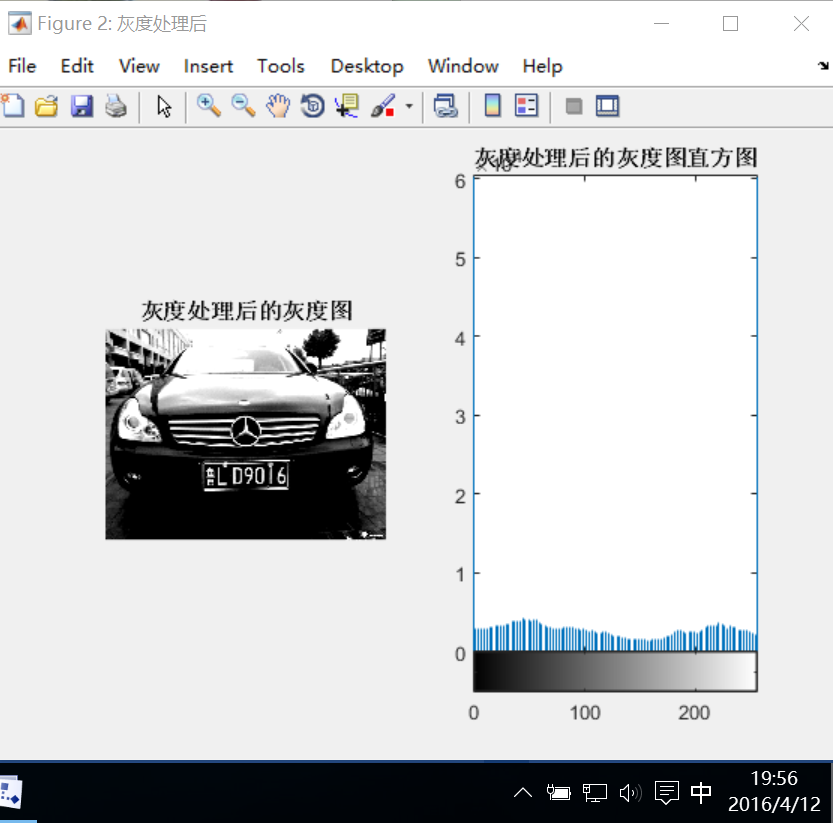
图3



图4

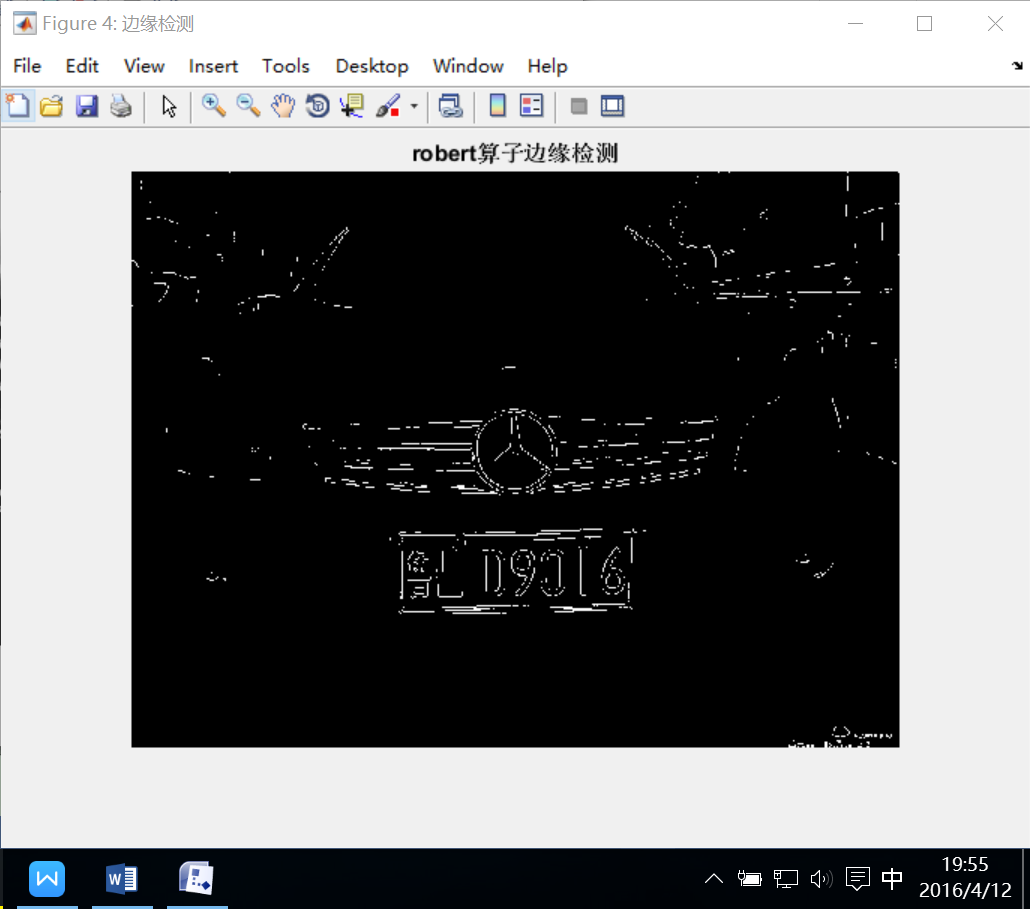
3.2边缘检测

两个具有不同灰度值的相邻区域之间总存在边缘，边缘就是灰度值不连续的结果，是图像分割、纹理特征提取和形状特征提取等图像分析的基础。为了对有意义的边缘点进行分类，与这个点相联系的灰度级必须比在这一点的背景上变换更有效，我们通过门限方法来决定一个值是否有效。所以，如果一个点的二维一阶导数比指定的门限大，我们就定义图像中的次点是一个边缘点，一组这样的依据事先定好的连接准则相连的边缘点就定义为一条边缘。经过一阶的导数的边缘检测，所求的一阶导数高于某个阈值，则确定该点为边缘点，这样会导致检测的边缘点太多。可以通过求梯度局部最大值对应的点，并认定为边缘点，去除非局部最大值，可以检测出精确的边缘。一阶导数的局部最大值对应二阶导数的零交叉点，这样通过找图像强度的二阶导数饿的零交叉点就能找到精确边缘点。如图-5所示

%边缘检测：sobel,roberts,canny,prewitt等

I2=edge(I1,'roberts',0.25,'both'); %边缘检测算法，强度小于阈值0.15的边缘被省略掉,'both'两个方向检测（缺省默认）

figure('name','边缘检测'),imshow(I2);title('robert算子边缘检测')

图5

3.4对牌照进行定位和切割

这里采用的是对蓝色像素点的采集判断，因为就中国地区而言绝大部分车牌的底色的蓝色的，虽然算法很快很识别出大区域蓝色，但缺点在于对蓝色车辆的车牌识别不是很好。

Blue\_y=zeros(y,1);% zeros(M,N) 表示的是M行\*N列的全0矩阵

for i=1:y

for j=1:x

if(myI(i,j,1)==1) %% 判断蓝色像素

Blue\_y(i,1)= Blue\_y(i,1)+1;% 蓝色像素点统计

end

end

end

[temp MaxY]=max(Blue\_y);% Y方向车牌区域确定 [temp MaxY]临时变量MaxY

PY1=MaxY; % 以下为找车牌Y方向最小值

while ((Blue\_y(PY1,1)>=5)&&(PY1>1))%% 为什么判断蓝色像素点>=5（才算蓝色）？？？？

PY1=PY1-1;

end

PY2=MaxY; % 以下为找车牌Y方向最大值 ???难道最大值不是MaxY????

while ((Blue\_y(PY2,1)>=5)&&(PY2<y))

PY2=PY2+1;

end

% IY=I(PY1:PY2,:,:);

%%%%%%%%%%%%%%%%% X方向 %%%%%%%%%

Blue\_x=zeros(1,x);% 进一步确定x方向的车牌区域

for j=1:x

for i=PY1:PY2 % 只需扫描的行

if(myI(i,j,1)==1) %% 判断蓝色像素

Blue\_x(1,j)= Blue\_x(1,j)+1; % 蓝色像素点统计

end

end

end

PX1=1;% 以下为找车牌X方向最小值

while ((Blue\_x(1,PX1)<5)&&(PX1<x))%% 为什么判断蓝色像素点<3（不算蓝色？）？？？？

PX1=PX1+1;

end

PX2=x;% 以下为找车牌X方向最大值

while ((Blue\_x(1,PX2)<3)&&(PX2>PX1))

PX2=PX2-1;

end

PY1=PY1-2;% 对车牌区域的校正

PX1=PX1-2;

PX2=PX2+3;

PY2=PY2+10;

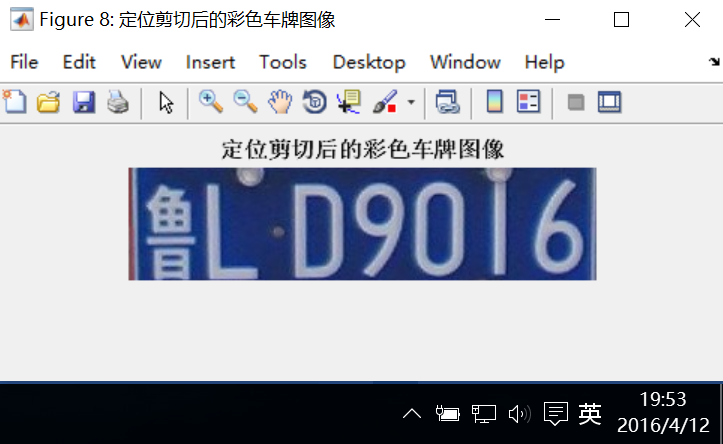
对图像进行裁剪，切割出牌照的彩色图像，如图7(1)所示

图7（1）

在测试过程中，因为对车牌的定位采用的是对蓝色区域的定位，所以对于蓝色车体来说，切割很容易产生错误。

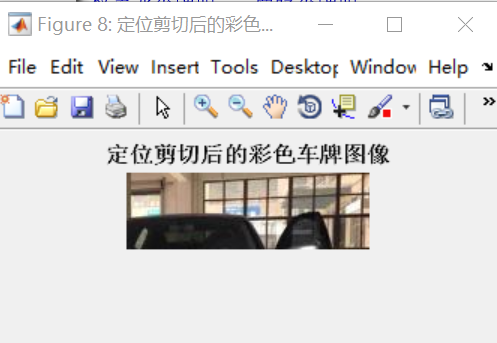
实验采用了蓝色车体进行对照试验，实验结果说明，算法对蓝色车体的车牌识别比

图7（2）

较不准，导致后来的车牌定位错误，最后导致后续的车牌分割字符分割也相应出错，具体错误如图-7（2）所示；

3.5对牌照字符进行预处理

和对整体图像的处理相同，进行一系列灰度，中值滤波和膨胀腐蚀处理，使车牌的字符更加明显，有利于下一步的分割处理。

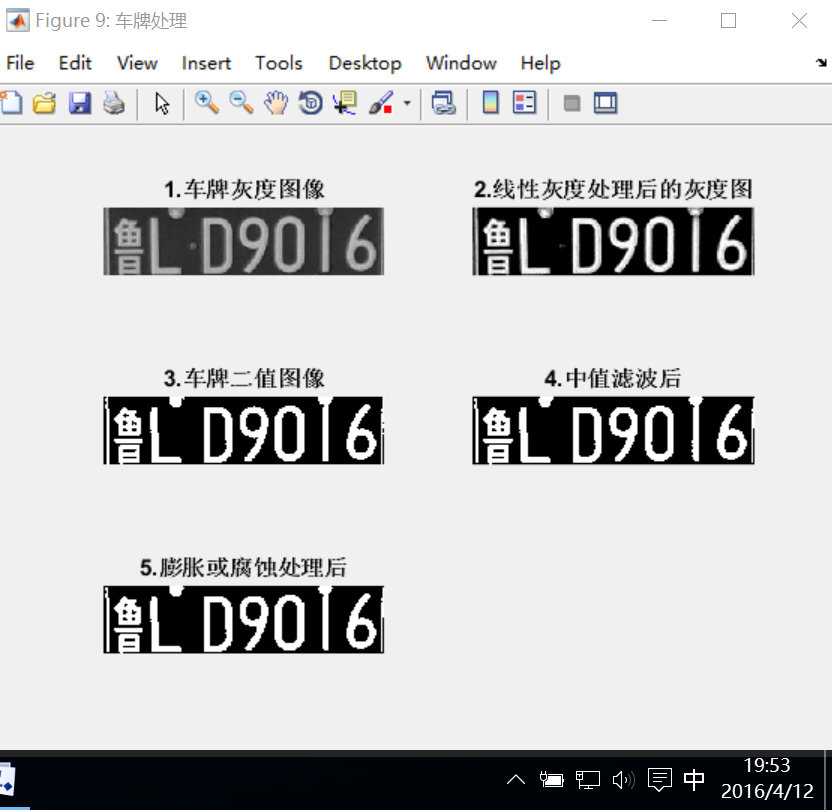
代码不再赘述，和上述对图像处理一致；效果如图8所示；

图8

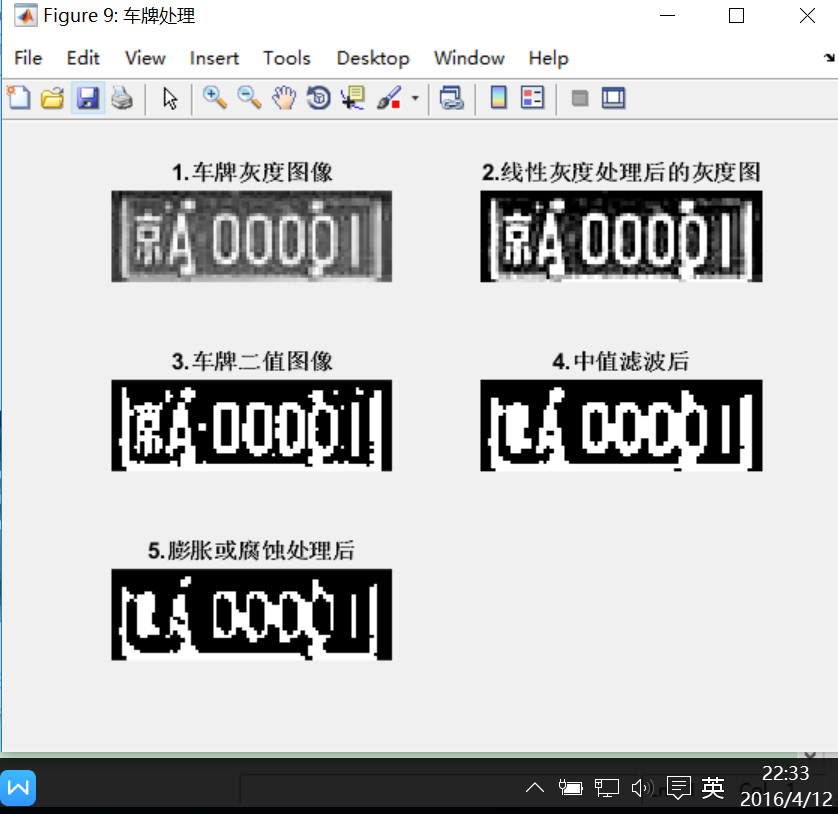
但是在实验过程中，如果车牌距离很远，经过这些操作可能导致车牌更加模糊的产生，

图9（1）

很不容易对车牌字符进行分割。如下图9（1）的灰度处理效果，图9（2）车牌切割效果。切割出的彩色图像虽然有些模糊，但仍然肉眼可分，但对于机器视觉来说，就很难了，采用灰度化，反而会得到不好的效果。

图9（2）

3.6对牌照字符进行分割

其中的分割子程序的原理，寻找连续有文字的字块，若长度大于阈值，则认为有两个字符组成，需要切割，反之，则认为只有一个字符。

function e=qiege(d)

[m,n]=size(d);

top=1;bottom=m;left=1;right=n; % init

while sum(d(top,:))==0 && top<=m

top=top+1;

end

while sum(d(bottom,:))==0 && bottom>=1

bottom=bottom-1;

end

while sum(d(:,left))==0 && left<=n

left=left+1;

end

while sum(d(:,right))==0 && right>=1

right=right-1;

end

dd=right-left;

hh=bottom-top;

e=imcrop(d,[left top dd hh]);

在主函数中调用切割子程序如下所示。

d=qiege(d);% 调用qiege()子程序

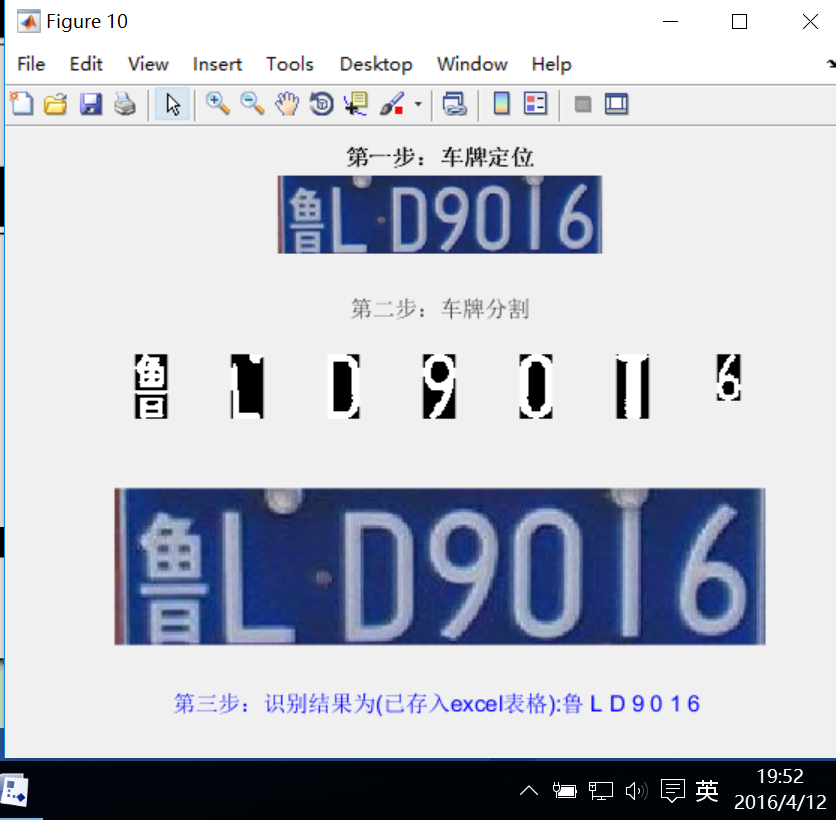
[m,n]=size(d);

figure,subplot(2,1,1),imshow(d),title(n)

k1=1;k2=1;

j=1;

s=sum(d);%sum(x)

效果如图9所示；

3.7 对字符的识别

这里对字符的识别采用了模板相减的方法；对分割出的字符挨个与字符库中的字符进行模板对比，选择差别最小的进行确认，这种方法优点是简单快捷，缺点是识别率比较差。识别流程图如图10所示；



图10

关键代码展示；

Dmax=0; % 与模板不同的点个数

for k1=1:40

for l1=1:20

if ( SubBw2(k1,l1) > 5 | SubBw2(k1,l1) < -5 ) % "|"/"||" 或操作 （>2 15）20以上无区别

Dmax=Dmax+1;

end

end

end

Error(k2)=Dmax; % 记录下字符与模板k2不同的点个数

end

Error1=Error(kmin:kmax);

MinError=min(Error1); % 差别最小的

findc=find(Error1==MinError); % 找出差别最小的模板

效果图如上图-9所示；

3.8 车牌存储；

最后将所得的车牌号码以文本形式写入excel表格中待后续的操作步骤；

关键代码；

xlswrite('C:\Users\MrLevo\Desktop\模式识别大作业\_徐凯\车牌记录.xls',Code,'A1:M1');

如图11所示；

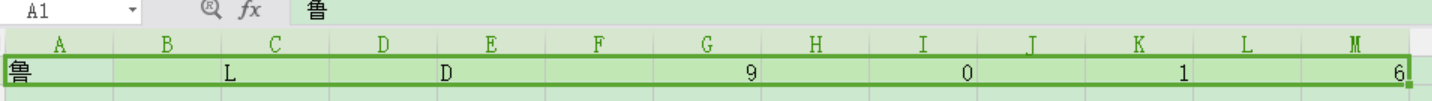
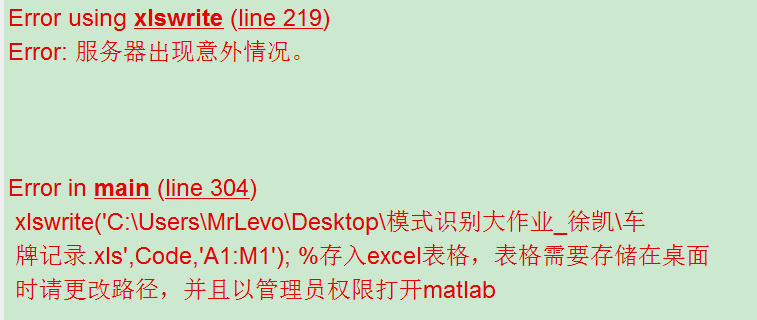


图11

值得注意的是，当文件夹在 C盘操作时候，Matlab需要用管理员权限打开，因为涉及到对系统盘的写入文件操作，否则可能会出现以下错误；如图12所示；

图12

4 总结与展望

本文对车牌识别系统进行了一次次模拟和改进，去除了对图像增强效果不太明显的色彩增强函数，采用对图像和车牌的两次灰度化，二值化来处理加强图片识别效果，在字符分割上，采用了连续字符的长度进行阈值判断的方法，效果随车牌的方位有很大的关系，而且和距离也有一定的关系。而最后的字符比较输出，则采用了模板相减的方法，效果同样受限于对字符预处理的效果，就像模糊字体的判断，在标准字符库中，这将变得非常不明显和误判，而且对省份的字符模板相减，可能存在误差点相似的情况，对结果也有很大的影响。

车牌识别还是一个比较热门的行业，而且光字符识别来说，以前的模板相减已经不能满足现在的高精准要求了，现在基于神经网络的训练字符模板方式是一个很不错的发展方向，经过神经网络的训练，能够得到较好的实际应用的识别率。因为车牌识别很多都已经商业化，所以公开的代码大多以模板相减来做，本人将网络上的代码进行整理，参数进行调节和测试，对各个版本的测试代码进行拼接调试，最后撰以此文，不足之处仍然较多。

参考文献

[1] 沈美明、温东蝉．IBM-PC汇编语言程序设计（第二版）．清华大学出版社，2001．8

[2] 李红．浅谈计算机病毒.山西大学财经学报，2002.12：527-530

[3] 赵均宇．强化科学管理机制．光明日报，1999-3-24（4）

[4] [刘佐濂](http://service.ilib.cn/Search/Search.aspx?Query=%e5%88%98%e4%bd%90%e6%bf%82" \o "刘佐濂) , [邓荣标](http://service.ilib.cn/Search/Search.aspx?Query=%e9%82%93%e8%8d%a3%e6%a0%87" \o "邓荣标) , [孔嘉圆](http://service.ilib.cn/Search/Search.aspx?Query=%e5%ad%94%e5%98%89%e5%9c%86" \o "孔嘉圆).[中国科技信息](http://www.ilib.cn/P-zgkjxx.html" \o "中国科技信息) [J].2005（23期）9～12.

[5] 宋建才.汽车牌照识别技术研究[J].工业控制计算机,2004,44～45.

[6] 韩勇强、李世祥.汽车牌照子图像的定位算法[M].微型电脑运用，1999.60～65.

[7] 王枚、王国宏.基于伴生与互补颜色特征的车牌字符分割技术[J].山东大学学报，2007。第37卷

[8] 贺兴华、周媛媛、王继阳等.MATLAB 图像处理[M].人民邮电出版社,2006.96～100.

[9] 龚声蓉、刘纯平、王强. 数字图象处理与分析[M]. 清华出版社 ,2006.24～29.

[10]刘阳，伊铁源等.数字图象处理应用于车辆牌照的识别.辽宁大学学报.2004，65～68.

[11] 张兴会, 刘玲, 杜升之.车牌照定位及倾斜校正方法研究[J].系统工程与电子技术, 2004, 26(2): 237～239.

[12] 叶晨洲，杨杰，宣国荣.车辆牌照字符识别[J].上海交通大学学报，2000，5(34)： 672～675.

[13] 魏武, 黄心汉, 张起森, 等.一种基于垂直字符边界特征的车牌定位方法,中国公路学报, 2000, (4) : 88-90

物理实现

验证(仿真)

电路级设计

系统逻辑描述

系统模块划分

确定算法

系统任务分析

附录

Main.m主程序

% function [d]=main()

close all

clc % 清空命令窗口的所有输入和输出，类似于清屏

%自动弹出提示框读入图像

[filename,filepath]=uigetfile('.jpg','输入一个需要识别的车牌图像');% 直接自动读入%

file=strcat(filepath,filename); %strcat函数：连接字符串；把filepath的字符串与filename的连接，即路径/文件名

I=imread(file);

figure('name','原图'),imshow(I);title('原图')

%图像增强

% h=ones(5,5)/25; %过滤器h

% I=imfilter(I,h);%真彩色增强

% figure('name','真彩色增强');imshow(I);title('真彩色增强');

I1=rgb2gray(I); % RGB图像转灰度图像

% %figure('name','灰度处理前'),subplot(1,2,1),imshow(I1);title('灰度处理前的灰度图');

% % subplot(1,2,2),imhist(I1);title('灰度处理前的灰度图直方图');

%线性灰度变换

I1=imadjust(I1,[0.3,0.7],[]);

figure('name','灰度处理后'),subplot(1,2,1),imshow(I1);title('灰度处理后的灰度图');

subplot(1,2,2),imhist(I1);title('灰度处理后的灰度图直方图');

%进行中值滤波

I1=medfilt2(I1);

figure,imshow(I1);title('中值滤波');

%边缘检测：sobel,roberts,canny,prewitt等

I2=edge(I1,'roberts',0.25,'both'); %边缘检测算法，强度小于阈值0.15的边缘被省略掉,'both'两个方向检测（缺省默认）

figure('name','边缘检测'),imshow(I2);title('robert算子边缘检测')

se=[1;1;1];

I3=imerode(I2,se);% 腐蚀Imerode(X,SE).其中X是待处理的图像，SE是结构元素对象

figure('name','腐蚀后图像'),imshow(I3);title('腐蚀后的图像');

se=strel('rectangle',[20,20]);% 25X25的矩形 strel???

I4=imclose(I3,se);% 用25\*25的矩形对图像进行闭运算(先膨胀后腐蚀)有平滑边界作用

figure('name','平滑处理'),imshow(I4);title('平滑图像的轮廓');

I5=bwareaopen(I4,1000);% 从二进制图像中移除所有少于2000像素的连接对象，消失的是连续的白色像素数量少于2000的字符

figure('name','移除小对象'),imshow(I5);title('从对象中移除小对象');

[y,x,z]=size(I5);% y是行数，x是列数，z是维数

myI=double(I5);% 转成双精度型

tic % 开始计时

Blue\_y=zeros(y,1);% zeros(M,N) 表示的是M行\*N列的全0矩阵

for i=1:y

for j=1:x

if(myI(i,j,1)==1) %% 判断蓝色像素

Blue\_y(i,1)= Blue\_y(i,1)+1;% 蓝色像素点统计

end

end

end

[temp MaxY]=max(Blue\_y);% Y方向车牌区域确定 [temp MaxY]临时变量MaxY

PY1=MaxY; % 以下为找车牌Y方向最小值

while ((Blue\_y(PY1,1)>=5)&&(PY1>1))%% 为什么判断蓝色像素点>=5（才算蓝色）？？？？

PY1=PY1-1;

end

PY2=MaxY; % 以下为找车牌Y方向最大值 ???难道最大值不是MaxY????

while ((Blue\_y(PY2,1)>=5)&&(PY2<y))

PY2=PY2+1;

end

% IY=I(PY1:PY2,:,:);

%%%%%%%%%%%%%%%%% X方向 %%%%%%%%%

Blue\_x=zeros(1,x);% 进一步确定x方向的车牌区域

for j=1:x

for i=PY1:PY2 % 只需扫描的行

if(myI(i,j,1)==1) %% 判断蓝色像素

Blue\_x(1,j)= Blue\_x(1,j)+1; % 蓝色像素点统计

end

end

end

PX1=1;% 以下为找车牌X方向最小值

while ((Blue\_x(1,PX1)<5)&&(PX1<x))%% 为什么判断蓝色像素点<3（不算蓝色？）？？？？

PX1=PX1+1;

end

PX2=x;% 以下为找车牌X方向最大值

while ((Blue\_x(1,PX2)<3)&&(PX2>PX1))

PX2=PX2-1;

end

PY1=PY1-2;% 对车牌区域的校正 为什么要这么+－？？？

PX1=PX1-2;

PX2=PX2+3;

PY2=PY2+10;

dw=I(PY1:PY2-8,PX1:PX2,:);% 裁剪图像

toc %t=toc; % 停止计时

%figure(7),subplot(1,2,1),imshow(IY),title('行方向合理区域');

figure('name','定位剪切后的彩色车牌图像'),%subplot(1,2,2),

imshow(dw),title('定位剪切后的彩色车牌图像')

imwrite(dw,'dw.jpg');

% 直接自动读入%[filename,filepath]=uigetfile('dw.jpg','输入一个定位裁剪后的车牌图像');

%jpg=strcat(filepath,filename); % strcat函数：连接字符串；把filepath的字符串与filename的连接，即路径/文件名

a=imread('dw.jpg');

b=rgb2gray(a);

imwrite(b,'1.车牌灰度图像.jpg');

figure('name','车牌处理');subplot(3,2,1),imshow(b),title('1.车牌灰度图像')

%g\_max=double(max(max(b)));% 以下作阈值化（灰度图转二值图）

%g\_min=double(min(min(b)));% max(a)求的每列的最大值，是一维数据；max(max(a)) 是求这一维数据的最大值。

%T=round(g\_max-(g\_max-g\_min)/2); % T 为二值化的阈值 round：取整为最近的整数

%[m,n]=size(b);% m:b的行向量数 n:b的列向量数

%d=(double(b)>=T); % d:二值图像

%imwrite(d,'2.车牌二值图像.jpg');

%线性灰度变换

b=imadjust(b,[0.3,0.7],[]);

subplot(3,2,2),imshow(b);title('2.线性灰度处理后的灰度图');

%进行二值化处理

d=im2bw(b,0.4);%将灰度图像进行二值化处理

imwrite(d,'2.车牌二值图像.jpg');

subplot(3,2,3),imshow(d),title('3.车牌二值图像');%显示二值化图像

%进行中值滤波

d=medfilt2(d);

imwrite(d,'4.均值滤波后.jpg');

subplot(3,2,4),imshow(d);title('4.中值滤波后');

% 均值滤波

%h=fspecial('average',3);

%d=im2bw(round(filter2(h,d)));% 滤波后，im2bw()：将图像转成二值图像 (可以不用round函数 也是一样的)

%imwrite(d,'4.均值滤波后.jpg');

%subplot(3,2,4),imshow(d),title('4.均值滤波后')

% 某些图像进行操作

% 膨胀或腐蚀 ？？？感觉没什么效果咧？？？

% se=strel('square',3); % 使用一个3X3的正方形结果元素对象对创建的图像进行膨胀

% 'line'/'diamond'/'ball'/'square'/'dish'... 线/菱形/球/正方形/圆

se=eye(2); % eye(n) 返回n乘n单一矩阵； 单位矩阵

[m,n]=size(d);

if bwarea(d)/m/n>=0.365 % 函数bwarea 计算目标物的面积，单位是像素；bwarea/m/n即为单个像素？？

d=imerode(d,se);% 腐蚀

elseif bwarea(d)/m/n<=0.235

d=imdilate(d,se);% 膨胀

end

imopen(d,se);

%se=eye(7);

%imopen(d,se);

imwrite(d,'5.膨胀或腐蚀处理后.jpg');

subplot(3,2,5),imshow(d),title('5.膨胀或腐蚀处理后')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%进行字符识别%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% 寻找连续有文字的块，若长度大于某阈值，则认为该块有两个字符组成，需要分割

d=qiege(d);% 调用qiege()子程序

[m,n]=size(d);

figure,subplot(2,1,1),imshow(d),title(n)

k1=1;k2=1;

j=1;

s=sum(d);%sum(x)就是竖向相加，求每列的和，结果是行向量;sum(x,2)表示矩阵x的横向相加，求每行的和，结果是列向量。sum(X(:))表示矩阵求和

while j~=n %%%%% 什么原理？？？

while s(j)==0 %% 无文字？？？为什么？？？

j=j+1;

end

k1=j;

while s(j)~=0 && j<=n-1

j=j+1;

end

k2=j-1;

if k2-k1>=round(n/6.5)

[val,num]=min(sum(d(:,[k1+5:k2-5])));

d(:,k1+num+5)=0; % 分割

end

end

% 再切割

%d=qiege(d);

% 切割出 7 个字符

y1=10;y2=0.25;flag=0;word1=[];

while flag==0 % flag为自定义，以便标记循环用

[m,n]=size(d);

left=1;

wide=0;

while sum(d(:,wide+1))~=0 % 二值图像：黑色像素代表感兴趣的对象而白色像素代表背景。逻辑矩阵只包括0(显示为黑色)和1(显示为白色)

wide=wide+1;% ？wide的意义？

end

if wide<y1 % 认为是左侧干扰 为什么是10？

d(:,[1:wide])=0; % 将白色汉字前的白色弄成黑色

% figure,imshow(d);

d=qiege(d); % 处理干扰后再次调用切割子程序

else

temp=qiege(imcrop(d,[1 1 wide m]));% imcrop函数截取图像[xmin ymin width height]

[m,n]=size(temp);

all=sum(sum(temp));

two\_thirds=sum(sum(temp([round(m/3):2\*round(m/3)],:)));

if two\_thirds/all>y2 %？？什么意思？？

flag=1;word1=temp; %第一个字符

end

d(:,[1:wide])=0;d=qiege(d); %？为什么又处理一次？

end

end

% 分割出第二个字符

[word2,d]=getword(d);

% 分割出第三个字符

[word3,d]=getword(d);

% 分割出第四个字符

[word4,d]=getword(d);

% 分割出第五个字符

[word5,d]=getword(d);

% 分割出第六个字符

[word6,d]=getword(d);

% 分割出第七个字符

[word7,d]=getword(d);

subplot(5,7,1),imshow(word1),title('1');

subplot(5,7,2),imshow(word2),title('2');

subplot(5,7,3),imshow(word3),title('3');

subplot(5,7,4),imshow(word4),title('4');

subplot(5,7,5),imshow(word5),title('5');

subplot(5,7,6),imshow(word6),title('6');

subplot(5,7,7),imshow(word7),title('7');

[m,n]=size(word1);

% 商用系统程序中归一化大小为 40\*20,此处演示

word1=imresize(word1,[40 20]);%imresize对图像做缩放处理，常用调用格式为：B=imresize(A,ntimes,method)；其中method可选nearest,bilinear（双线性）,bicubic,box,lanczors2,lanczors3等

word2=imresize(word2,[40 20]);

word3=imresize(word3,[40 20]);

word4=imresize(word4,[40 20]);

word5=imresize(word5,[40 20]);

word6=imresize(word6,[40 20]);

word7=imresize(word7,[40 20]);

subplot(5,7,15),imshow(word1),title('11');

subplot(5,7,16),imshow(word2),title('22');

subplot(5,7,17),imshow(word3),title('33');

subplot(5,7,18),imshow(word4),title('44');

subplot(5,7,19),imshow(word5),title('55');

subplot(5,7,20),imshow(word6),title('66');

subplot(5,7,21),imshow(word7),title('77');

imwrite(word1,'1.jpg'); % 创建七位车牌字符图像

imwrite(word2,'2.jpg');

imwrite(word3,'3.jpg');

imwrite(word4,'4.jpg');

imwrite(word5,'5.jpg');

imwrite(word6,'6.jpg');

imwrite(word7,'7.jpg');

liccode=char(['0':'9' 'A':'Z' '京辽陕苏鲁浙']); %建立自动识别字符代码表；'京津沪渝港澳吉辽鲁豫冀鄂湘晋青皖苏赣浙闽粤琼台陕甘云川贵黑藏蒙桂新宁'

% 编号：0-9分别为 1-10;A-Z分别为 11-36;

% 京 津 沪 渝 港 澳 吉 辽 鲁 豫 冀 鄂 湘 晋 青 皖 苏

% 赣 浙 闽 粤 琼 台 陕 甘 云 川 贵 黑 藏 蒙 桂 新 宁

% 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59

% 60 61 62 63 64 65 66 67 68 69 70

SubBw2=zeros(40,20); % 创建一个40行20列的0矩阵

l=1;

for I=1:7

ii=int2str(I); % 将整型数据转换为字符串型数据

t=imread([ii,'.jpg']);% 依次读入七位车牌字符

SegBw2=imresize(t,[40 20],'nearest'); % 对读入的字符进行缩放

if I==1 % 第一位汉字识别

kmin=37;

kmax=42;

elseif I==2 % 第二位 A~Z 字母识别

kmin=11;

kmax=36;

else I>=3 % 第三位以后是字母或数字识别 ;即I>=3

kmin=1;

kmax=36;

end

for k2=kmin:kmax

fname=strcat('namebook\',liccode(k2),'.jpg'); % strcat函数：连接字符串

SamBw2 = imread(fname);

for i=1:40

for j=1:20

SubBw2(i,j)=SegBw2(i,j)-SamBw2(i,j);

end

end

% 以上相当于两幅图相减得到第三幅图 进行匹配

Dmax=0; % 与模板不同的点个数

for k1=1:40

for l1=1:20

if ( SubBw2(k1,l1) > 5 | SubBw2(k1,l1) < -5 ) % "|"/"||" 或操作 （>2 15）20以上无区别

Dmax=Dmax+1;

end

end

end

Error(k2)=Dmax; % 记录下字符与模板k2不同的点个数

end

Error1=Error(kmin:kmax);

MinError=min(Error1); % 差别最小的

findc=find(Error1==MinError); % 找出差别最小的模板

% Code(l)=liccode(findc(1)+kmin-1); % 此处用2\*l-1且后面的2\*l=' ',第隔一空格输出一个字符

% Code(3)=' ';Code(4)=' ';

% l=l+1;

% if l==3;% 后五位与前两位字符隔开

% l=l+2;

% end

Code(l\*2-1)=liccode(findc(1)+kmin-1);

Code(l\*2)=' ';

l=l+1;

end

figure(10),subplot(5,7,1:7),imshow(dw),title('第一步：车牌定位'),

xlabel({'第二步：车牌分割'}); %'',

subplot(6,7,15),imshow(word1);

subplot(6,7,16),imshow(word2);

subplot(6,7,17),imshow(word3);

subplot(6,7,18),imshow(word4);

subplot(6,7,19),imshow(word5);

subplot(6,7,20),imshow(word6);

subplot(6,7,21),imshow(word7);

subplot(6,7,22:42),imshow('dw.jpg');%

xlabel(['第三步：识别结果为(已存入excel表格):', Code],'Color','b');

xlswrite('C:\Users\MrLevo\Desktop\模式识别大作业\车牌记录.xls',Code,'A1:M1'); %存入excel表格，表格需要存储在桌面时请更改路径，并且以管理员权限打开matlab

% % handles = guihandles(gcf); %若没有excel则创立

% %set(handles.text1,'string',num2str(Code)); %本来用于gui可视化的。

===============================================================

Qiege.m切割子程序

function e=qiege(d)

[m,n]=size(d);

top=1;bottom=m;left=1;right=n; % init

while sum(d(top,:))==0 && top<=m

top=top+1;

end

while sum(d(bottom,:))==0 && bottom>=1

bottom=bottom-1;

end

while sum(d(:,left))==0 && left<=n

left=left+1;

end

while sum(d(:,right))==0 && right>=1

right=right-1;

end

dd=right-left;

hh=bottom-top;

e=imcrop(d,[left top dd hh]);

===============================================================

Getword.m取出字符子程序

function [word,result]=getword(d)

word=[];flag=0;y1=8;y2=0.5;

while flag==0

[m,n]=size(d);

wide=0;

while sum(d(:,wide+1))~=0 && wide<=n-2

wide=wide+1;

end

temp=qiege(imcrop(d,[1 1 wide m]));

[m1,n1]=size(temp);

if wide<y1 && n1/m1>y2

d(:,[1:wide])=0;

if sum(sum(d))~=0

d=qiege(d); % 切割出最小范围

else word=[];flag=1;

end

else

word=qiege(imcrop(d,[1 1 wide m]));

d(:,[1:wide])=0;

if sum(sum(d))~=0;

d=qiege(d);flag=1;

else d=[];

end

end

end

%end

result=d;