

# 基于 VC++ 的车牌识别系统关键技术研究<sup>\*</sup>

赵启升 李存华

(淮海工学院计算机科学系 连云港 222005)

**摘要** 车辆牌照识别(LPR)系统是一个专用的计算机视觉系统, LPR 系统的广泛应用将有助于加快我国交通管理自动化的进程。本文对车牌识别系统中的图像预处理、车牌定位、字符分割和字符识别等环节涉及到的技术、算法以及系统整体设计作了全面的论述, 并与已有方案进行了比较, 对部分关键算法进行了设计和改进。用采集到的 40 多幅汽车图片作为实验样本数据, 用纯软件的方法实现了车牌字符的自动识别。

**关键词** 车牌定位, 竖向纹理, 字符分割, 字符识别

## Research of Kernel Technology in License Plate Recognition System Based on VC++

ZHAO Qi-Sheng LI Cun-Hua

(Department of Computer Science, Huaihai Institute of Technology, Lianyungang 222005)

**Abstract** As a special computer vision system, the License Plate Recognition System (LPR) can capture a vehicle automatically and identify the plate numbers in the image. Contributing a lot to the automation of the transportation management. In this paper, some methods and algorithms are introduced in such stages as preprocessing of images, license plate location, character segmentation and character recognition. At the same time, compare the methods with the existing. Moreover, designing and improving parts of critical algorithm. Testing 40 representative vehicle license plates, I can achieve the good accuracy.

**Keywords** License plate location, Vertical texture, Character segmentation, Character recognition

## 1 引言

智能交通管理系统(Intelligent Transport Systems, ITS)是 21 世纪道路交通管理的发展趋势, 而车牌识别(License Plate Recognition, LPR)是实现交通管理智能化的重要环节。LPR 是一个以特定目标为对象的专用计算机视觉系统, 它运用模式识别、人工智能等技术, 对采集到的汽车图像进行处理, 实时准确地识别出车牌的数字、字母及汉字字符, 并以计算机可直接运行的数据形式给出识别结果, 使得车辆的电脑化监控和管理成为现实。

车牌自动识别系统中有三个部分在提高识别率上起着关键的作用: 即车身图像中车牌部分定位、定位后车牌字符的分割以及对单个字符的识别。本文在研究前人成果的基础上, 对车牌识别系统的设计进行了系统的阐述, 特别在车牌定位和字符分割方面, 进行了一些探索和创新。系统由 4 部分组成: 车辆图像预处理、车牌区域的定位、车牌字符识别的预处理、车牌字符的识别, 本文重点介绍车辆图像预处理、车牌区域的定位和车牌字符的识别部分。

## 2 车辆图像的预处理

### 2.1 图像的灰度化

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。由于图像的每个像素都具有 3 个不同的颜色分量, 存在许多与识别无关的信息, 不便于进一步的识别工作, 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。

将彩色图像转化为灰度图像常采用如下的经验公式:

$$\text{gray} = 0.30 \times R + 0.59 \times G + 0.11 \times B$$

式中, gray 为灰度值, R、G、B 分别为红色、绿色和蓝色分量。这是由于人眼对绿色的敏感度最高, 对红色的敏感度次之, 对蓝色的敏感度最低, 这样得到的灰度图像是较为合理的。

### 2.2 图像的增强

#### 2.2.1 对比度增强

由于所拍摄的车辆图像的质量受天气、光照、观察点等的影响, 车辆图像往往会产生对比度不足的弊端, 使图像细节分辨不清, 影响识别率。因此, 需要将车辆图像进行图像灰度扩展, 以便改善图像的观察质量, 提高字符识别率。

本文采用的增强对比度的方法是灰度概率统计法。灰度概率统计方法的实质是 Wallis 和 Jong Sen Lee 提出的用局部均值和方差进行对比度增强的方法的拓展。

对图像的灰度进行概率修正是从统计的角度出发, 使修改后图像的灰度具有给定数学期望和方差。其计算公式如下:

$$g = \frac{\text{percentage}}{100} (f - E_s) + E_d$$

式中, percentage 表示百分比分子, 它的值根据经验给出, 本文中取值为 192。

#### 2.2.2 图像的平滑

图像平滑是图像增强的一种手段, 它的目的是为了消除噪声。噪声并不限于人眼所能看到的失真或变形, 有些噪声

<sup>\*</sup>江苏省教育厅自然科学基金资助项目(NO. 02KJB520012)。赵启升 工程师, 主要从事计算机网络、图形图像处理方面的教学和科研工作; 李存华 教授, 主要从事数据挖掘及图形图像处理方面的教学与科研。

只有在进行图像处理时才能发现。一般来说,图像的能量主要集中在其低频部分,而车牌的信息主要在高频部分,为了去掉高频干扰,有必要进行图像平滑。

可以采用低通滤波的方法来去除噪声。具体的方法是设计一个空间域系统的单位冲激响应矩阵。

$$g(i, j) = \sum_m \sum_n f(m, n) H(i-m+1, j-n+1)$$

式中,  $g(i, j)$ ,  $f(m, n)$  为灰度值,  $H$  为  $L \times L$  矩阵, 又叫低通

$$\text{卷积模板。本文选用 } H = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}.$$

### 2.3 图像的边缘检测

在对车辆图像进行边缘检测,要经过两个步骤:初步边缘检测和水平梯度方向边缘检测。这样得到的图像中就留下了大量的竖向边缘特征,为后面准确的车牌定位提供了依据。

#### 2.3.1 初步边缘检测

本文采用的是 Sobel 算子边缘检测算法。Sobel 边缘检测用两个卷积核作用于同一幅图像,这两个卷积核分别用来计算水平和垂直方向的梯度,它们是:

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}, \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

设有两个卷积核分别进行卷积操作的中间图像为  $G_h[x]$   $[y]$  和  $G_v[x]$   $[y]$ , 其中  $x$  取值范围为  $0 \sim \text{width}-1$ ,  $y$  的取值范围为  $0 \sim \text{height}-1$ , 则最后的结果图像为:

$$M_{\text{sobel}}[x][y] = \sqrt{G_h^2[x][y] + G_v^2[x][y]} \text{ 或 } M_{\text{sobel}}[x][y] = |G_h[x][y]| + |G_v[x][y]| \text{ 或 } M_{\text{sobel}}[x][y] = \max(|G_h[x][y]|, |G_v[x][y]|)$$

这时检测的方向为:

$$\Phi_{\text{sobel}}[x][y] = \tan^{-1} \left( \frac{G_v[x][y]}{G_h[x][y]} \right)$$

很显然,上述算法的计算量相当大:第一,要做两次卷积操作;第二,要计算范数。本文介绍的系统使用的是改进后的 Sobel 边缘检测算法,它以当前像素为中心获取一个  $3 \times 3$  邻域,按行按列依次编号为 a, b, c, d, e, f, g, h, i, 这样,中心像素被编号为 e, 在这个邻域中,过中心像素 e 的直线共有 4 条,它们是 a-e-i, b-e-h, c-e-g, d-e-f。每条分割线均将邻域剩下的像素分成两部分。对于每一条分割线,分别计算其两个子部分的像素平均值之差的绝对值。最后在 4 个绝对值中选择最大者作为当前像素的最后值。实验证明, Sobel 边缘检测算法所生成的结果适度,细致,是令人满意的。

#### 2.3.2 水平梯度方向边缘检测

经过了 Sobel 边缘检测之后得到的图像具有大量的边缘特征(包括横向和竖向),为了获得车辆图像的竖向边缘特征,还需进行水平梯度方向边缘检测。事实上,水平梯度边缘检测与上述几种边缘检测方法一样都具有一个共同的算法思想:用原图减去经过变换后的图像。它所用的卷积算子为:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

### 2.4 图像的二值化

二值图像是指整幅图像画面内仅有黑、白二值的图像。在数字图像处理中,二值图像占有非常重要的地位。这是考虑到在实用的图像处理系统中,针对彩色或灰度图像进行速

度高、成本低、信息量大的处理所产生的花销太大。此外,二值化后的图像能够用几何学中的概念进行分析和特征描述,比灰度图像优势大得多。

本文采用的是基于像素值的全局迭代阈值选取方法。基本思想是:

(1) 求出图像中的最大和最小灰度值  $Z_0$  和  $Z_L$ , 并令初始

$$\text{阈值为: } T_0 = \frac{Z_L + Z_K}{2}$$

(2) 根据阈值  $T_k$  将图像分割成目标图像和背景图像两部分,再求出这两部分的平均灰度值  $Z_0$  和  $Z_B$ :

$$Z_0 = \frac{\sum_{Z(i,j) < T_k} Z(i,j) \times N(i,j)}{\sum_{Z(i,j) < T_k} N(i,j)}$$

$$Z_B = \frac{\sum_{Z(i,j) > T_k} Z(i,j) \times N(i,j)}{\sum_{Z(i,j) > T_k} N(i,j)}$$

式中,  $Z(i, j)$  是图像中  $(i, j)$  点的灰度值;  $N(i, j)$  是  $(i, j)$  的加权系数,一般为 1。

(3) 求出新的阈值。

$$T_{k+1} = \frac{Z_0 + Z_B}{2}$$

如果  $T_k = T_{k+1}$ , 则迭代结束, 否则  $k \leftarrow k+1$ , 转到第(2)步继续迭代。

## 3 车牌区域的定位

### 3.1 算法描述

在得到车辆图像的边缘点图之后,就可以依据该图对车牌区域进行定位。车牌区域的定位算法如下:

(1) 车牌区域列范围确定。主要的工作是对全图进行扫描,找到每行可能的车牌区域起始点和终止点,以及两者之间边缘点的数量。这是从竖向来确定车牌区域可能的列范围,并为下一步确定车牌区域行范围提供依据。设  $l_{\text{max}}$  为每行允许的最大边缘点间距,  $l_{\text{min}}$  为每行允许的最小边缘点间距,  $m$  为每行起始点和终止点之间允许的最小边缘点数,  $\text{col\_s}[L]$ ,  $\text{col\_e}[L]$  为某一段边缘点的起始和终止坐标,  $r[L]$  为起始点和终止点之间边缘点的数量,  $\text{lr\_proportion}$  为当前点到起始点的距离与边缘点数量的比值, 设  $L=1$ ;

① 对第  $L$  行扫描, “while (dot( $L, k$ )) != 边缘点,  $k++$ , black++”, 如果 “dot( $L, k$ ) == 边缘点”, 则 “col\_s[L] =  $k$ , col\_e[L] =  $k$ ,  $k++$ ”, 找到可能的行起始点。

② 对第  $L$  行继续扫描, 如果 “dot( $L, k$ ) == 边缘点”, 并且边缘点距离小于  $l_{\text{max}}$ , 如果  $\text{lr\_proportion}$  满足要求, 则 “col\_e[L] =  $k$ ,  $r[L]++$ ,  $k++$ ”, 否则就重新寻找, 执行 “col\_s[L] =  $k$ , col\_e[L] =  $k$ ,  $r[L]=0$ ,  $k++$ ”; 如果边缘点距离大于  $l_{\text{max}}$  并且 “ $r[L] \leq m$ ”, 则 “col\_s[L] =  $k$ , col\_e[L] =  $k$ ,  $r[L]=0$ ,  $k++$ ”; 如果边缘点距离大于  $l_{\text{max}}$  并且 “ $r[L] > m$ ”, 则 “ $L++$ ”。

③ 如果到了行尾则 “ $L++$ ”, 回到步骤①执行。

(2) 车牌区域行范围确定。根据每行可能的车牌区域的起始点和终止点, 以及起始点和终止点之间的边缘点数量, 来求出车牌区域可能的行范围。  $b$  为连续行中允许的不满足要求的连续行的最大行数,  $c$  为满足要求的连续行的最小行数,  $j_{\text{max}}$  为每行中满足要求的最大边缘点数量,  $j_{\text{min}}$  为每行中满足要求的最小边缘点数量,  $l_{\text{max}}$  为每行允许的最大边缘点间距,  $l_{\text{min}}$  为每行允许的最小边缘点间距。 设  $L=1, b=0, c$

=0;

①“while( r[L]<= jmin || r[L]>= jmax || col\_e[i]-col\_s[i]<=lmin || col\_e[i]-col\_s[i]>=lmax ), L++”,记下可能的起始行。

② 查看第 L 行是否满足“r[L]>=jmin && r[L]<=jmax && col\_e[i]-col\_s[i]>=lmin && col\_e[i]-col\_s[i]<=lmax”,如果满足则“c++, b=0”,如果不满足,则“b++”,如果“b>=B”且“c<C”,则从上一个可能区域尾到此行不存在牌照区域。

③ 如果“c>=C”且“b>=B”,则记下这个连续区域为候选牌照区域,回到步骤①继续执行。

④“L++”,回到步骤②继续执行。

(3)真正的车牌区域的确定。经过上面的两个步骤,可能会找到几个满足要求的图像区域。这时就需要分析,以排除伪车牌区域。由于车牌的长宽有一定的比例,且车牌的长宽在整幅车辆图像中有一定的比例,本文正是据此来找出真正的车牌。

(4)车牌左右边界的确定。经过了上面的三个步骤,已经找到了车牌可能所在的行范围,接下来的问题是定车牌的左右边界。虽然在第一个步骤中,已经得到了每行的起始点和终止点,但是在找到的车牌区域中,各行的起始点和终止点并不完全一样,所以要找出确切的车牌区域的左右边界。实验结果证明,车牌区域各行的起始点和终止点分布相当集中,只有个别行偏差较大,所以本文对车牌区域各行的起始点和终止点的分布进行了统计,选取出现次数最频繁的起始点和终止点作为车牌区域的左右边界。

### 3.2 算法特点

车牌主要在纹理上而不是平均亮度上或色彩上与其周围背景和其它物体有区别,所以,图像定位可以以纹理为基础。车牌区域主要体现在其结构特征上,即其纹理模式由纹理基元(文字的空间排列的特征)。由于牌照字符的笔划变化及笔划边缘相对于背景的对比度构成了牌照区域强烈的空间频率变化,因此充分利用牌照区域强烈的空间频率特征。特征提取就变为确定这些基元的空间频率变化。如果这些空间频率变化符合直观测量要求,则这块区域可初步定为车牌区域。

对车牌区域检测需要运用车牌区域所特有的属性。按照模式识别原理,应找到车牌区域图像的固有的且与图像其它区域不易混淆的属性,并且所使用的属性在各种环境下摄取的图像具有稳定性。通过以上分析可知,在各种条件下车牌所在的图像区域稳定可靠的信息是它具有丰富的边缘,因此设计了上面以边缘信息分析为基础的纹理定位算法。在设计具体算法的时候,不仅考虑了车牌区域每行的边缘点数量,还考虑了边缘点数量与车牌区域长度的比值以及非边缘点的连续数量,这使得车牌区域的定位率大大提高。

## 4 车牌字符的识别

目前常用的字符识别方法有多种,模板匹配是字符识别的基本方法之一。

$$D(i, j) = \sum_{m=1}^M \sum_{n=1}^N [I(m, n) - T(m, n)]^2 \quad (1)$$

此项越小,表示此时与模板的匹配程度越好, D(i, j) 为零则说明两者完全匹配。这种方法虽然算法简单,容易实现,但是有一些不足之处:(1)计算量大,公式(1)的计算量随着 Tn 所包含像素增加而迅速增大,而且执行匹配运算的次数随着

原图的增大而增多,匹配效率不高;(2)对于字符的方位旋转和尺度缩放的不适应性;(3)图像的平均灰度值的变化会影响匹配结果的正确性。针对上述缺点,本文采用了一种改进的模板匹配算法——13 特征模板加权匹配,可以在很大程度上克服传统模板匹配算法的不足。

### 4.1 字符的特征提取

字符的 13 特征具体取法如下:

(1)首先把字符平均分成 8 份,如图 1 所示(这里以字符“8”举例)。分别统计这 8 个区域中的黑像素的数目,可以得到 8 个特征。

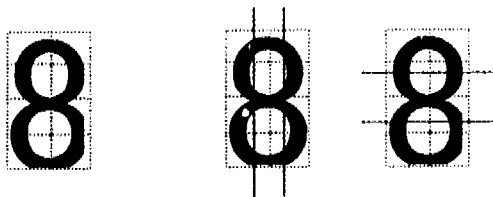


图 1 13 特征提取法

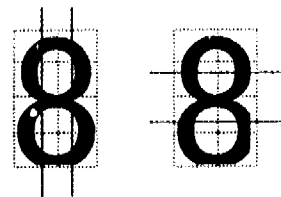


图 2 13 特征提取法

(2)然后统计水平方向中间两列和竖直方向中间两列的黑色像素点的个数作为 4 个特征,最后统计所有黑色像素点的个数作为第 13 个特征。也就是说,画 4 道线,统计线穿过的黑像素的数目,可以得到 4 个特征。示意图如图 2 所示。

(3)将字符图像的全部黑色像素的数目的总和作为一个特征,总共即得到 13 个特征。

### 4.2 特征模板的匹配

#### 4.2.1 算法描述

13 特征模板匹配的数学描述为:

$$D(i) = \sum_{k=1}^{13} A(k) \times |I(k) - T(k)| \quad (2)$$

式中, A(k) 为第 k 个特征的权值, I(k) 为待匹配字符的第 K 个特征, T(k) 为模板库中某个模板的第 k 个特征, D(i) 为匹配值,表示待匹配字符的特征与模板字符的特征的差异程度,匹配值越小,两个字符特征越一致。

算法描述如下:

设 i=0, j=0;

① 取第 i 个字符与模板库中第 j 个模板进行比较。

② 比较第 1-2 个特征, A(k) 取值为 1; 比较第 3-6 个特征, A(k) 取值为 2; 比较第 7-8 个特征, A(k) 取值为 0.5; 比较第 9 个特征, A(k) 取值为 1; 比较第 10-11 个特征, A(k) 取值为 1; 比较第 12-13 个特征, A(k) 取值为 2。“j++”, 如果全部模板匹配完毕, 进入步骤③执行, 否则回到步骤①执行。

③ 取出最小的模板匹配值, 以该模板字符作为与当前字符匹配成功的字符。“i++, j=0”, 回到步骤①执行。

#### 4.2.2 算法特点

本算法在提取了字符的 13 个特征的基础上, 用模板匹配的方法对字符进行了匹配, 大大地提高了匹配算法的执行速度。同时, 根据每个特征的重要程度的不同, 对 13 个特征进行加权匹配, 使得字符识别的准确率大大提高。

## 5 系统性能分析与结论

受客观条件限制, 本文介绍系统的测试是在 PC 上进行的(CPU 主频为 Pentium4 1.7GHz, 内存为 256M), 完成整个流程(从车牌定位到字符识别完成), 耗时大约为 7~8 秒。系统各处理环节成功率、所花费的时间及比率见表 1。从表中可以看出, 各环节及整体成功率比较高, 系统耗时也主要集中在

(下转第 213 页)

是局部连通的,并且每个捕获分支的边界是 Jordan 曲线。

从在双二次多项式族  $f_c(z) = z^4 - 2c^2 z^2$  动力系统分形图的逐级放大图(图 7)中可以看到,第三个放大图中包含了 Mandelbrot 集,事实上,由于这个分形图有自相似性,因此其中还包含着无数个 Mandelbrot 集。在逐级放大图 7 中,我们还可以观察到参数空间的每个捕获分支的边界是 Jordan 曲线。

### 5.5 双二次多项式 $f_{a,b}(z) = z^4 + az^2 + b$ 的 Julia 集的分形图绘制和性质

仍然采用本文 3.1 节阐述的“逃逸时间”算法,我们设计出绘制出双二次多项式  $f_c(z) = z^4 + az^2 + b$  的 Julia 集的分形图像的程序。在程序中选择一些特定的参数值  $a, b$  我们就可以得到一些特定的 Julia 集分形图像,这里列举一些 Julia 集的分形图和对应的参数(图 8)。

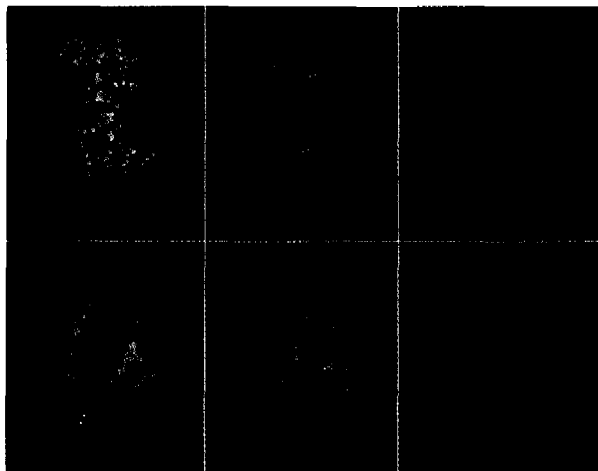


图 8 双二次动力系统的 Julia 集分形图举例

通过对图 8 中参数取值和 Julia 集分形图对应关系的观

(上接第 209 页)

在低通滤波、边缘检测等车辆图像预处理环节,本文所采用的核心算法的耗时非常低,达到了预期的效果。

表 1 系统各处理环节成功率、所花费的时间及比率

处理过程	样本数量	成功数量	失败数量	成功率	耗时(us)	耗时比率
车牌定位	42	37	5	88.0%	101	1.27%
字符分割	41	37	4	90.2%	387	4.88%
字符识别	38	30	8	78.9%	100	1.26%
整体状况	42	30	12	71.4%	7937	100%

**结论** 车牌的识别问题是现代交通工程领域中研究的重点和热点问题之一,本文就汽车牌照的定位与分割及自动识别等问题进行了研究,提出了汽车牌照自动识别系统的思路 and 总体设计方案。通过对大量实际拍摄的车牌图像的反复研究比较,对车牌识别系统的图像预处理、车牌定位、字符分割、字符识别等环节所涉及的技术和算法进行了设计、改进,完成了车牌识别系统的基于 VC++ 6.0 的实现。由于本文介绍的系统在车牌识别技术的研究中,主要是使用灰度图像,没有用到车牌图像的色彩特征,这在一定程度上影响了车牌自动识别系统的性能。车牌本身是彩色物体,其底色和字符的颜色是有限的几种,而且摄像设备和计算机性能的提高使彩色图像的实时处理变得容易。因此,充分利用车牌的色彩信息,并

察,可以观察到以下性质,而这些性质已经在我的导师吕菁的论文《双二次多项式动力学》中得到证明<sup>[1]</sup>:

性质 1  $f_{a,b}$  的 Julia 集是连通的是当且仅当它所有临界点的轨道是有界的。

性质 2  $f_{a,b}$  的 Julia 集是完全不连通的当且仅当它任何临界点的轨道不在填充 Julia 集的周期分支内。

性质 3 当  $f_{a,b}$  不属于上面两种情形, $f_{a,b}$  的 Julia 集有可数个多于一点的连通分支,其余为不可数个单点集。

**结论** 本文通过借鉴二次多项式动力系统的 Mandelbrot 集和 Julia 集的定性研究方法,在 Visual Basic 6.0 编译器上实现双二次动力系统的分形图像绘制程序,并为方便研究增加了逐级放大观察分形图细部结构的功能,采用了可调节的上色方案。利用这些功能我们对  $f_c(z) = z^4 - 2c^2 z^2$  动力系统的参数空间、Julia 集进行了数学试验,验证了我的导师吕菁在论文《双二次多项式动力学》中所用理论方法证明的双二次动力系统的性质。并通过假设不同的参数,观察  $f_c(z) = z^4 + az^2 + b$  动力系统的 Julia 集图像,归纳出一些性质,这些性质也已经被证明<sup>[1]</sup>,说明这些工作对于更好地研究双二次动力系统的性质有指导意义。

### 参 考 文 献

- 1 吕菁. 双二次多项式动力学:[上海:复旦大学数学系博士论文], 2004
- 2 Glaser F, Robles A J. A Hyperpolar Image of the Mandelbrot Set. GLASER, ROBLES, Fall 1998. 139~146
- 3 吕以攀. 复解析动力系统. 科学出版社, 2002. 10~96
- 4 [英]肯尼思·法尔科内. 分形几何-数学基础及应用. 东北大学出版社, 1991. 220~289
- 5 李水根. 分形. 高等教育出版社, 2004. 20~27
- 6 任福尧主编. 复解析动力系统. 复旦大学出版社, 1991. 4~35

结合计算机彩色视觉技术,则可以更有效地实现车牌字符的分割和识别。

### 参 考 文 献

- 1 Da Rocha Gesualdi A, De Albuquerque M P. MARCELO PORTES DE ALBUQUERQUE. Recognition of characters in plates of private Brazilian vehicles using neural networks [J]. IEEE, 2000
- 2 Ter M H, Brugge, Stevens J h, Nijhuis J A G. License Plate Recognition Using DTCNNs. In: Proceeding of the IEEE International Workshop on Cellular Neural Networks and their Application, 1998. 212~217
- 3 Wang Li, Pavlidis T. Direct gray-scale extraction of features for character recognition [J]. IEEE Trans on PAMI, 1993, 15(10): 1053~1067
- 4 左奇, 史忠科. 一种基于数学形态学的实时车牌图像分割方法. 中国图像图形学报, 2003, 8A(3): 281~285
- 5 刘伟铭, 赵雪平. 一种基于扫描行的汽车车牌定位算法. 计算机工程与应用, 2004. 223~225
- 6 宋加涛, 刘济林. 车辆牌照上英文和数字字符的结构特征分析及提取. 中国图像图形学报, 2002, 7A(3): 945~949
- 7 郭大波, 陈礼民, 卢朝阳, 等. 基于车牌底色识别的车牌定位方法. 计算机工程与设计, 2003, 4(5): 81~89