

基于过线特征的车牌识别

余宗健

(四川大学 计算机学院图形图像研究所, 四川 成都 610064)

摘要: 车牌识别的速度和正确率直接地反映识别系统的好坏。为了提高车牌的识别速度, 提出了基于过线特征的车牌识别算法。算法分两步进行: 首先对分割的待识别字符进行过线特征提取, 每个字符都被映射成一个12维向量, 计算出待识别字符向量与库字符向量的距离; 然后, 对于置信度高的字符直接作为识别结果, 对于置信度不具有明显优势的待识别字符在缩小的范围下再采用模板匹配识别。实验结果表明: 应用该算法的实验系统识别速度较快、准确度较高、稳定性较好。

关键词: 过线特征; 线性滤波器; 形态学膨胀

中图分类号: TP39

文献标识码: A

文章编号: 1673-629X(2010)05-0043-04

License Plate Recognition Based on Through-Line-Feature

YU Zong-jian

(Inst. of Image & Graphics, Sch. of Computer Sci. & Eng., Sichuan Univ., Chengdu 610064, China)

Abstract: The speed and accuracy of license plate recognition directly reflects the recognition system. Propose through-line-feature recognition algorithm in order to improve the speed of license plate recognition. Recognition algorithm is in two steps: First, it carries out on the through-line feature extraction of license plate image, each character is mapped into a 12-dimension vector, calculates the distance between the vector of identifying character and the vector of identified character. Second, for those characters that have a high degree of recognizability, it takes the characters as recognition results directly, and for those characters that does not have distinct recognizability, it employs algorithm of template matching recognition after it commits the through-line-feature algorithm. The experimental results showed that this algorithm which applied in LPRS enhanced the system recognition speed, accuracy and stability.

Key words: through-line-feature; linear filter; morphological dilation

0 引言

近年来,随着高速公路的发展和交通智能化的需求,汽车牌照自动识别技术已经越来越受到人们的重视。汽车牌照自动识别技术是目标自动识别的一个典型问题,它在智能交通管理和公共安全等系统中有着广泛的应用前景。车牌自动识别系统中要求迅速准确地检测出车牌区域和正确的识别结果。前人已提出了许多车牌定位方法,如使用统计的投影直方图^[1]、使用Hough变换检测边缘直线^[2]、使用模糊边缘检测算法^[3]等。字符识别技术的原理主要有三种:相关匹配、文法分析和松弛匹配。相关匹配这类方法通过在特征空间中计算输入特征向量与各模板向量之间的距离(或相似度)进行分类判决,常用的距离有: m 阶

Minkowski度量,两向量夹角的余弦等。文法分析是典型的结构识别方法,但由于重点是在笔段形状、位置和笔段之间相互关系的分析上,所以它的优点是分辨相似字的能力较强,缺点是对局部噪声敏感。松弛匹配法首先通过迭代运算,找出输入向量与模板向量各分量间的对应关系,然后再根据这一对应关系进行匹配,由于需要进行大量的迭代才能达到稳定,计算量较大。

车牌识别过程一般分为图像预处理、车牌定位、字符分割和字符识别四个步骤。在文中主要是用 MATLAB 作为工具,根据汽车牌照的纹理特征,利用一种线性滤波器对水平边缘增强,然后运用形态学膨胀和跃变规律相结合来对车牌进行定位,用投影法来对汽车牌照进行分割,用改进的过线数特征的算法对车牌进行识别。

1 预处理

任何一幅未处理的原始车牌图片都存在噪声。这

收稿日期:2009-09-08;修回日期:2009-12-16

作者简介:余宗健(1983-),男,硕士研究生,研究方向为图像处理;
导师:刘怡光,教授,研究方向为图形图像、计算机视觉、视频三维重建等。

些噪声恶化了图像的质量,使图像模糊,甚至淹没了车牌的特征,给识别带来了困难。所以在对图像进行分析之前,需要进行减少或清除图像中的噪声的预处理。

有些彩色车牌图片灰度化后的灰度值比较均衡,不利于对车牌区域进行二值化处理。所以,首先对灰度图进行灰度拉伸。

在一幅有复杂背景的车辆图片中,车牌中字符串总是连续出现,在水平方向上形成一系列明暗交替的模式,就算牌照位置有所偏移,倾斜的角度也不会太大,所以牌照区域在水平方向形成了规则纹理。为突出车牌区域水平频繁变化的这种纹理特征,采用了一种线形滤波器^[4],将水平方向的梯度进行累加运算。字符区域边界密集的地方梯度得到增强,而背景中的近景部分被减弱,这样就使近景部分中的牌照区域明显比其他区域明亮。线性滤波器的数学表达式为:

$$G(i) = \frac{1}{\mu R} \sum_{k=-R/2}^{R/2} \text{Filter}(|F(i+k-d) - F(i+k+d)|, T)$$

其中: $\text{Filter}(x, T) = \begin{cases} x & x \geq T \\ 0 & x < T \end{cases}$, $F(i)$ 是原图的每列的灰度值, $G(i)$ 是经过线形滤波处理之后每列的灰度值; R, d 为常数,选取与图像中车牌的大小有关; T 为一阈值,用于对所选图像对比度的控制; μ 用于增大滤波器的响应输出。保证 $R/2 + d$ 大于车牌中字符笔画的宽度可起到加强车牌区域边缘特征的作用。

2 车牌的定位

2.1 形态学膨胀

为了将车牌作为一个整体而非几个字符来分析,采用数学形态学方法^[5]中的膨胀技术把车牌区域生成连通区域^[6]。结构元素 Se 的选择,对于候选牌照区域的形成与牌照区域提取至关重要。考虑到线性滤波可能会使得车牌字符有损失,采用矩形结构元素 $se = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$ 进行形态学膨胀。膨胀后,车牌区域构成了一个连通区域,如图 1 所示。



图 1 膨胀后的连通区域

2.2 粗定位

对膨胀后的图像进行标记^[7],得到每个连通区域的最左、最右、最上、最下位置像素点的坐标,分别记为 Left, Right, Top 和 Bottom。利用这些坐标计算出所标定区域的宽度、高度、宽高比例以及区域面积,分别记为 $xWidth$, $yHeight$, $Ration$ 和 $Area$ 。

通过连通区域大小、长宽、长宽比例和区域面积等几何特征得到候选牌照区域,即满足:

$$\begin{cases} w_1 \leq xWidth \leq w_2 \\ \lambda_1 \leq Ration \leq \lambda_2 \\ Area \geq A \end{cases}$$

这样可能提取到多个候选牌照区域,有的是噪声边缘,有的是纹理丰富的汽车及背景边缘图像经形态学膨胀后形成的虚假块,也有可能是牌照与附近的一些汽车纹理轮廓发生了粘连的区域等。因此,为提取正确区域,必须设法去除虚假候选牌照,分离出真正的牌照区域。

2.3 细定位

考虑到车牌可能存在一定程度的倾斜,车牌倾斜角 θ 可能从 0 到 10° 不等。若不加以校正,定位得到牌照的上下边界将有很大偏差,字符也有较严重的变形,严重影响到后续的字符分割和字符识别处理。文中采用 Hough 变换法检测倾斜角 θ ,将粗定位车牌旋转 θ ,得到矫正的车牌区域。

候选牌照区域内每扫描行的跳变次数相对稳定,即在汽车图象中有满足该纹理特征的区域来实现其边界的准确定位,该约束条件可表示为^[8]:

$$\text{Line}(\text{jump} \geq 7) \geq \alpha$$

其中, Line 是满足条件的连续扫描行数, jump 是扫描行内的跳变次数, α 为经验值。

当扫描行的跳变次数不小于 7 时,第一次和最后一次跳变的坐标值分别存入数组 $xLeft$ 和 $xRight$ 。最终得到最长的连续有效行的集合,该集合的上下边界作为车牌的上下边界,该集合中找到最小的 $xLeft$ 作为左边界,最大的 $xRight$ 作为右边界。在实际应用中,由于噪声等原因,满足上述条件的扫描行可能不连续,因此常采用扫描行的间断为两行即可解决该问题。

3 车牌的分割

利用车牌字符“峰、谷、峰”的纹理特征进行分割^[9,10]。将车牌区域图片重新二值化,垂直投影于 $Vsum(i)$,用水平分割线 $y = y_c$ 去截取曲线 $y = Vsum(i)$,即: $\begin{cases} y = Vsum(i) \\ y = y_c \end{cases}$

得到交点的水平坐标,然后取整存于数组 $Seg_{m \times 2}$,

同时对车牌左右边界进行判断。

车牌的每个字符都有一定的宽度,把 Seg 中小于字符最小宽度的噪声的坐标删除。如果 $m < 7$,把 Seg 中坐标间隔大于字符最大宽度的投影不断用 $y_c = y_c + 1$ 进行以上操作,直到 $m \geq 7$;如果 $m \geq 7$,利用宽度和像素面积限制条件得到字符的分割坐标,从而得到各个字符的分割图。

4 车牌的识别

字符识别方法很多,如 Rough 集的理论方法^[11]、BP 神经网络方法、svm 方法^[12]等。这都需要大量的待识别图片数据,进行多次训练,工作量大。为了提高车牌的识别速度和识别正确率,文中对基于过线特征的方法^[13]进行改进。识别过程分两步:首先采用过线特征进行识别,对于置信度高的结果直接作为识别结果。对于置信度不具有明显优势的待识别字符再采用模板匹配识别。过线特征识别运算量小,但是精度较低,而模板识别则精度高,运算量大。所以先采用过线特征进行初次识别,对于可能识别错误的字符再采用模板识别既保证算法的鲁棒性,也能保证实时性。

过线特征识别方法其实是模板匹配的一种方法,是对模板的算法进行的优化。通过提取待识别字符的横线数、竖线数、过线数三类特征来对其识别。

横线:

$$\begin{cases} hRatio = \frac{hBlackNumber}{width} \\ 0.6 \leq hRatio \leq 1 \end{cases}$$

竖线:

$$\begin{cases} vRatio = \frac{vBlackNumber}{height} \\ 0.4 \leq vRatio \leq 1 \end{cases}$$

其中:hBlackNumber 为字符水平

方向连续出现的黑色像素数,width 为字符图像的像素宽度。vBlackNumber 为字符竖直方向连续出现的黑色像素数,height 为字符图像的竖直高度。

水平过线数:用水平线从上至下扫描字符,该扫描线穿越字符的黑色像素点个数(相邻的黑色像素点算一个像素点)。同理可知竖直过线数。如果字符有断裂,过线数会受影响,为了避免这种现象的影响,采用最大过线数代替过线数。如字符“0”的水平最大过线数是 2。在实际中,在提取水平线数时,将字符从上至下等分为三份,分别提取它们的水平线数。在提取其他特征的时候也相应地把待识别图像分成三等份,分别提取了特征。任何一个带识别字符都映射成了 12 维向量:

字符→(b,c,d,e,f,g,h,i,j,k,l,m)

其中:

- b: 水平上部线特征

c: 水平中部线特征

d: 水平下部线特征

e: 竖直左部线特征

f: 竖直中部线特征

g: 竖直右部线特征

h: 水平上部过线特征

i: 水平中部过线特征

j: 水平下部过线特征

k: 垂直左部过线特征

l: 垂直中部过线特征

m: 垂直右部过线特征

先将分割处理好的字符进行归一化,主要是大小的归一化,然后进行二值化。对所得到的字符图像进行上述特征提取,然后根据提取的特征进行分类识别,最后输出结果。具体算法如下:

1) 建立模版及特征提取。

从待识别图片中提取数字,A-Z 和各省简称的模板,并根据线特征提取算法提取各模板对应的特征,保存在文件 feather.mat 中,在程序启动时载入。

2)待识别字符的特征提取。

对于分割后的每个字符同样进行线特征提取。

3)字符识别。

利用距离公式求出待识别向量和识别库向量的距离,置信度高的结果直接作为识别结果;否则,取距离最小的 N 个(文中取 $N = 3$)库字符进行模板识别,得到识别结果。整个识别过程的结构图如图 2 所示。

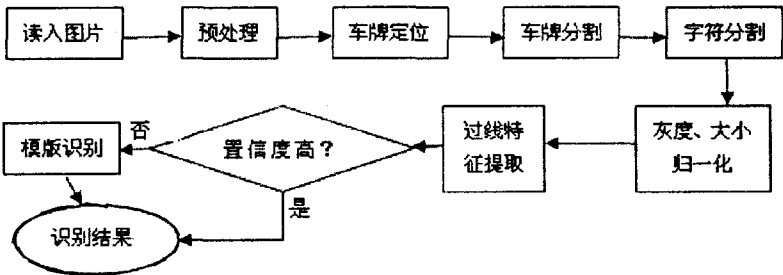


图 2 识别过程结构图

5 实验结果

文中利用 Matlab GUIDE 做了个 Demo 演示我们的算法(如图 3 所示)。黑线上面部分是对一张图片进行识别的算法流程:1、读取原始图片;2、车牌定位提取;3、车牌分割及归一化;4、提取特征;5、计算距离进行识别得到识别结果。下半部分显示对整个目录下所有图片进行连续识别。

对整个目录下的 296 幅 768×576 的车牌图像进行识别。文中改进的基于特征线的识别较快,准确率可达 92.8%。不能识别的车牌照字符出现的原因主要是车牌照照片质量差,用人眼识别有困难图片以及汉字模糊的图片等。

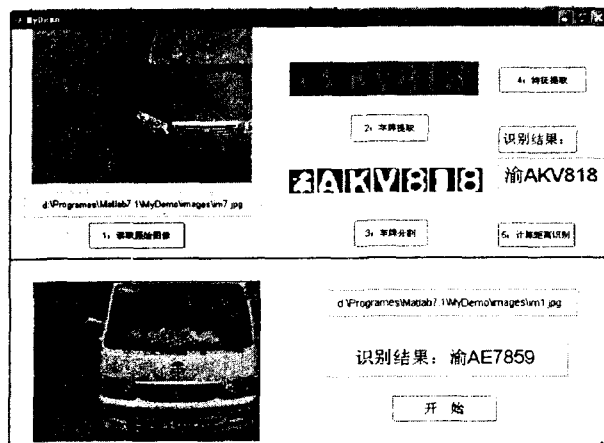


图 3 识别结果图

6 结束语

由此可以看出,文中所采用的水平边缘增强、形态学膨胀和投影法相结合的车牌照定位和分割,以及提出的基于线性特征的改进识别算法,复杂度低,能够满足实时性要求;而且该算法对复杂的背景并不敏感,具有较好的鲁棒性。

参考文献:

- [1] 韩永强,李世祥. 汽车牌照子图像的的定位算法[J]. 微型电脑应用,1999(3):14-16.
- [2] 郭索至. 基于灰度图的机动车牌照定位与识别[D]. 长春:

(上接第 42 页)

4 结束语

以上介绍并分析了计算机游戏中常采用的几种基于层次包围体的碰撞检测算法,需要指出的是:精度满足要求是关键,因为逼真的游戏体验是检验一个游戏好坏的最重要标准,当然,在这个前提下,包围体并不是越紧密越好,因为包围体越紧密,碰撞检测的时间开销会越大,当一个游戏场景中有大量运动物体时,由于碰撞检测导致的时延可能会让用户无法接受。因此,在 3D 游戏开发中,务必要权衡好精度与效率这一对矛盾。

参考文献:

- [1] 范昭伟. 实时碰撞检测技术研究[D]. 杭州:浙江大学图书馆,2003.
- [2] 王志强,洪嘉振,杨辉. 碰撞检测研究综述[J]. 软件学报,1999,10(5):545-551.
- [3] Ar S. Chazelle B. Self-customized BSP trees for collision[J]. Computational Geometry, Theory and Applications, 2000, 15

吉林大学,2000.

- [3] 李庆庆,张燕平. 基于模糊边缘检测算法的车牌定位[J]. 计算机技术与发展,2006,16(12):8-12.
- [4] 廖金周,宣国荣. 车辆牌照的自动分割[J]. 微型电脑应用,1999(7):32-34.
- [5] Faradji F, Rezaie A H, Ziaratban M. Morphological - Based License Plate Location[J]. IEEE Image Processing, 2007 (1):57-60.
- [6] 张禹,马驹良,韩笑,等. 车牌识别中的图像提取及分割算法[R]. 长春:吉林大学,2006:406-410.
- [7] 牛欣,沈兰荪. 基于特征的车辆牌照定位算法[J]. 交通与计算机,2000(1):31-33.
- [8] 范勇,蒋欣荣,游志胜,等. 汽车牌照快速定位算法[J]. 光学工程,2001(2):56-59.
- [9] 陈涛,杨晨晖,青波. 基于投影和固有特征结合的车牌字符分割方法[J]. 计算机技术与发展,2009,19(5):45-47.
- [10] He Xiangjian, Zheng Lihong, Wu Qiang, et al. Segmentation of characters on car license plates[J]. IEEE, Multimedia Signal Processing, 2008(10):399-402.
- [11] 王希雷. 基于 Rough 集理论的车牌汉字特征提取[J]. 计算机技术与发展,2007,17(6):26-28.
- [12] Abdullah, Sheikh S H, Khairuddin O, et al. License plate recognition based on support vector machine[J]. ICEEI, Electrical Engineering and Informatics, 2009(1):78-82.
- [13] 黄山. 车牌识别技术的研究和实现[D]. 成都:四川大学,2005.

(1):91-102.

- [4] Baraff D. Interactive simulation of solid rigid bodies[J]. IEEE Computer Graphics and Applications, 1995, 15(3):63-75.
- [5] Kamat V V. A survey of techniques for simulation of dynamic collision detection[J]. Computer & Graphics, 1993, 17(4):379-385.
- [6] Lin M, Manocha D, Cohen J, et al. Collision Detection: Algorithms and Applications[J]. Algorithms for Robotics Motion and Manipulation, 1996, 6(1):129-142.
- [7] Palmer L J, Grimsdale R L. Collision detection for animation using sphere - trees[J]. Computer Graphics Forum, 1995, 14(2):105-116.
- [8] Gottschalk S, Lin M, Manocha D. OBBTree: A Hierarchical Structure for Rapid Interference Detection[C]//Proceedings of SIGGRAPH. New Orleans, LA: [s. n.], 1996:171-180.
- [9] Jime Hnez P, Thomas F, Torras C. 3D collision detection: a survey[J]. Computers & Graphics, 2001, 25:269-285.
- [10] Moore M, Wilhelms J. Collision detection and response for computer animation[J]. ACM Computer Graphics, 1988, 22(4):289-298.