



(12)发明专利

(10)授权公告号 CN 106295642 B

(45)授权公告日 2019.12.17

(21)申请号 201610651988.7

(22)申请日 2016.08.11

(65)同一申请的已公布的文献号

申请公布号 CN 106295642 A

(43)申请公布日 2017.01.04

(73)专利权人 中防通用河北电信技术有限公司

地址 076150 河北省张家口市怀安县柴沟

堡镇新区工业园区

(72)发明人 刘黎明 魏为

(51)Int.Cl.

G06K 9/32(2006.01)

G06K 9/46(2006.01)

(56)对比文件

CN 102799882 A, 2012.11.28, 全文.

CN 102693423 A, 2012.09.26, 全文.

李文锋 等. 基于纹理特征的车牌定位方法.
《图形、图像与多媒体》. 2014,

郭延祥 等. 基于边缘检测和颜色纹理直方
图的车牌定位方法.《计算机科学与探索》. 2014,

石贵民. 基于扫描线的新型车牌定位方法研
究.《科学技术与工程》. 2014,

审查员 韦佳黎

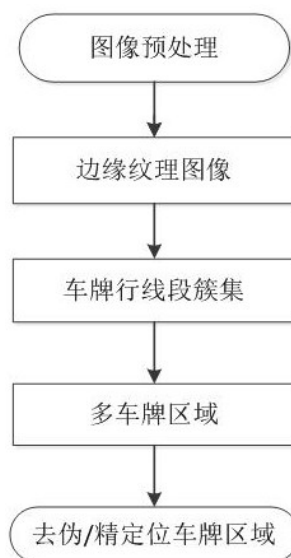
权利要求书3页 说明书9页 附图2页

(54)发明名称

一种基于容错率和纹理特征的车牌定位方
法

(57)摘要

本发明公开了一种基于容错率和纹理特征的车牌定位方法,属于图像处理技术领域,整体步骤如下:首先对图像进行预处理,利用受限自适应边缘直方图结合图像亮度均值得到边缘纹理图像;然后依据车牌的纹理特征并按车牌行特征提取算法对图像进行扫描,得到车牌行线段簇集;再对车牌行线段按设计的定位算法原理进行归并,在遍历扫描归并过程中通过融合容错率方法定位出车牌的所有可能区域;最后构造了纹理密度函数来精定位左右边缘、去除伪车牌。本发明通过在车牌定位算法中融合容错率,能有效提高在污染、磨损、对比度差等多种复杂情况下车牌的定位率;通过构建密度函数,在精定位车牌左右边界的同时还去掉了大部分伪车牌,降低了耗时和误检率。



1. 一种基于容错率和纹理特征的车牌定位方法,其特征在于:所述方法的具体步骤如下:

A、图像预处理:对抓拍的图像进行图像预处理,得到图像亮度均值;

B、求取边缘纹理图像:通过设计一种受限自适应边缘直方图,得到垂直边缘纹理图,其特征为:

(i)、利用公式I得到垂直边缘图像,其中, $e(i, j)$ 为边缘图像矩阵; $g(i, j)$ 为灰度图像矩阵;

$$e(i, j) = g(i, j+1) - g(i, j) \quad \text{公式I}$$

(ii)、对垂直边缘图像做增强处理,然后结合图像的亮度,设置自适应阈值过滤边缘强度较小的边缘点;通过一个尺寸为 $1*L$ 窗口选取边缘图像矩阵中每行的连续边缘点平均强度最大的窗口中心点作为边缘点;

(iii)、利用受限自适应边缘直方图得到其灰度图像;为了突出车牌字符的边缘,抑制或者减少非车牌边缘对车牌定位的影响,利用公式II求得边缘灰度直方图的均值,并判断图像是否为低对比度;其中: $hist(v)$ 为边缘灰度直方图、 $ave0$ 为图像亮度均值;通过判断是否为低对比度图像,然后分别利用公式III求得对比度低图像的锐化阈值、利用公式IV求得正常对比度图像的锐化阈值;利用 $thre$ 通过边缘灰度直方图锐化求得纹理图像;

$$\begin{cases} aveHist = \sum_{v=0}^{255} (v * hist(v)) / \sum_{v=0}^{255} hist(v) \\ isLowConstruct = \begin{cases} true & ave0 \leq threshold0 \\ false & ave0 > threshold0 \end{cases} \end{cases} \quad \text{公式II}$$

$$\begin{cases} thre = \begin{cases} pos0 & hist(pos0) \leq aveHist \\ v & hist(pos0) > aveHist \& hist[v] \leq threshold, v = pos0, \dots, 255 \end{cases} \\ threshold = w0 * hist[pos] \end{cases} \quad \text{公式III}$$

$$\begin{cases} threshold = w1 * aveHist \\ pos1 = v, \text{ when } \sum_{v=255}^{pos0} hist(v) \geq threshold \\ pos2 = v, \text{ when } threshold = \max_{v \in (pos0, \dots, 255)} \left(\frac{\Delta hist(v)}{\Delta v} \right) \\ thre = \begin{cases} pos1 & pos1 \leq pos2 \\ pos2 & pos1 > pos2 \end{cases} \end{cases} \quad \text{公式IV}$$

C、提取车牌行特征,得到车牌行簇集:根据实际图片中的最大车牌的尺寸设计字符间距的最大阈值和车牌行长度的最小最大阈值,得到车牌行簇集,其特征为:

(i)、根据字符间距设置最大宽阈值、最小被检测车牌宽阈值、最大车牌宽阈值,三者分别表示为 $maxCharW$, $minPlateW$, $maxPlateW$;

(ii)、遍历纹理图片,根据公式V得到符合车牌特征的车牌行线段;

$$\begin{cases} pLine(i,n) = line(n) \\ line(n) = \begin{cases} n++ & (e(i,j+m) - e(i,j)) \leq \maxCharW \& \\ p(i,n,s,e) & \minPlateW < (q(i,n,e) - q(i,n,s)) \\ & \leq \maxPlateW \end{cases} \end{cases} \quad \text{公式 V}$$

其中*i*是图像矩阵中的行号；*line(n)*中*n*为第*i*行中的第*n*根车牌行线段；*p(i,n,s,e)*为*line(n)*中该线段的首尾点的距离；*q(i,n,s)*、*q(i,n,e)*分别为线段起始端、结束端位置；

根据上述步骤中的阈值对每行的边缘点进行限定，当遇到边缘点间距小于 $\maxCharW$ 时，记录为该行的起始点；当间距大于 $\maxCharW$ 时，记录为该行的第一个结束点，并判断该线段大小是否大于最小检测的车牌宽 $\minPlateW$ ，且小于最大检测的车牌宽 $\maxPlateW$ ；如满足则记录为该行的第一个车牌行 $pLine(i,n)$ ，并记录各边缘点的位置，否则不记录；重复本步骤直至图片遍历完；

D、车牌定位：利用垂直边缘图像、相关阈值，以及容错率方案，对边缘图像进行车牌定位，其特征为：

(i)、得到车牌区域：根据车牌行集中的原则，利用上述步骤中设置的阈值，按公式VI、公式VII对边缘图像中的车牌行进行归属判断：

$$Rect(c) = (pLine(0,n), \dots, pLine(i,n) \dots) \quad \text{if} \begin{cases} p(i,n,s,e) \in (\min(Dis(0, \dots, t, \dots))) \\ \& p(i,n,s,e) < \maxCharW \end{cases} \quad \text{公式 VI}$$

$$Dis(t) = \text{abs}(q(i,n,s) - \text{aveXrect}(c)) + \text{abs}(q(i,n,e) - \text{aveYrect}(c)) \quad \text{公式 VII}$$

其中， $\text{aveXrect}(c)$ 、 $\text{aveYrect}(c)$ 为某一车牌区域所有线段左右端点的位置均值；

先从图像的第一行进行扫描，如果检测到车牌行线段尺寸满足大于 $\minPlateW$ 且小于 $\maxPlateW$ ，则把该线段归入到车牌区域的第一个车牌行线段，记为 $\text{rect}(0) = (pLine(0, 0), \dots)$ ；如果同一行中还有类似的车牌行线段，则新增一个区域，并把它作为第一个车牌行线段，记为 $\text{rect}(1) = (pLine(0, 1), \dots)$ ，如此一直到行尾；

紧接着扫描图像第*i*行，如果检测到了车牌行，则求得该车牌行的两端与前面已得到的所有车牌区域中对应的所有车牌行的两端位置坐标均值的欧式距离值，得到距离集 $Dis(t)$ ；从该欧式距离值集中取最小值，如果该最小值满足小于 $\maxCharW$ 预设条件，则该行并入到与最小距离值所对应的那个区域 $Rect(c) = (pLine(0, 0), pLine(i, n) \dots)$ 中；否则把该车牌行作为一个新的区域的第一行，继续检测车牌行线段到行尾；

(ii)、按上述步骤逐行扫描，一直到图像结束，形成多个车牌区域；为提高复杂环境下车牌的定位率，构造基于连续非零车牌行容错率 $th1$ ，其最大容错率表示为 $\maxTh1$ ；总非车牌线段行的容错率 $th2$ ，其最大容错率表示为 $\maxTh2$ ，并将它们融合在定位算法中；

在步骤D(i)中，将新检测到的行标记与之前所有检测到的区域中的最后一行做比较；若其差值大于 $\maxTh1$ ，则该车牌行线段不参与欧式距离计算；若新检测到的行标记与距离值集中的最小欧式距离值所在区域内最后一行的行标记之差大于1而小于 $\maxTh1$ ，则把该车牌线段所在行标记与该区域最后一行的行标记之差累积到所属区域的连续非车牌行线段容错率 $th1$ 上；当该差值大于 $\maxTh1$ 时，新建一个区域，把该车牌行线段的两端点计入到新的区域中，否则判断总非车牌线段行的容错率 $th2$ 是否大于阈值 $\maxTh2$ ，小于则并入到最

小欧式距离值所在的那个区域,同时该差值累积到总非车牌线段行的容错率th2上;

(iii)、对车牌区域中的伪车牌进行初步排除:设置车牌行数最小阈值、宽高比置信度区间来初步排除伪车牌;宽高比置信度区间指的是车牌宽高比应基本满足交通部制定的长宽比比例,设置一个最大值、最小值作为车牌置信度区域,不在该置信度区间内的区域将被删除;

E、车牌精定位:通过设计两种基于灰度边缘的自适应密度函数对定位出来的车牌进行去伪存真,并精定位车牌左右边缘,其特征为:

(i)、获得灰度密度函数:首先在灰度图像中利用定位得到的车牌区域把车牌截取出来,利用公式VIII对该边缘车牌图像 $e(i, j)$ 进行积分,得到密度函数 $den(x)$;

$$den(x) = \sum_w \iint_{i=0, j=0}^{i=h, j=u} e(i, j) \partial i \partial j \quad \text{公式VIII}$$

其中, w 是一个 $1 \times K$ 的窗口;取窗口内的数据样本的累加值作为离散点的密度幅值 $den(x)$, 其中 K 为车牌区域高 h 乘以一权值系数 w 且 $w < 1$, 此窗口大小的设置应使累加区间的长度尽量与车牌字符之间间隔的水平像素个数在数值上一致;

(ii)、对密度函数 $den(x)$ 求一阶导数, 得到函数 $der(x)$; 然后对该导数函数的离散值求其绝对值均值 $aveDer$; 用 $der(x)$ 中的各个绝对值从左到右与 $aveDer$ 比较, 若等于 $aveDer$ 则用数组 $array$ 记录该点的位置, 如公式IX、X、XI所示;

$$array(x) = y \text{ when } abs(der(y)) = aveDer \quad \text{公式IX}$$

$$\begin{cases} \maxArray(z, y) = \max_{\substack{0 < i < n-1 \\ i < x \leq i+1}} (den(array(x))) & z = 0, 1, \dots \\ y = array(x) \end{cases} \quad \text{公式X}$$

$$\begin{cases} \minArray(z, y) = \min_{\substack{0 < i < n-1 \\ i < x \leq i+1}} ((den(array(x)))) & z = 0, 1, \dots \\ y = array(x) \end{cases} \quad \text{公式XI}$$

取出 $array$ 数组元素, 以相邻元素之间的距离作为区间, 求得密度函数曲线的最大最小值及其相应位置的数组 $\maxArray$ 、 $\minArray$, 进而求得最大最小值的均值 $aveMax$ 、 $aveMin$; 根据车牌字符的排列规则, 密度函数中字符区域的密度值与字符之间的密度值有较大的落差, 如果 $aveMax$ 与 $aveMin$ 的比值小于某一值, 则判断为伪车牌, 算法转至步骤E(i), 继续对下一个车牌进行真伪判别, 否则转至下一步骤;

(iii)、结合最大最小值数组 $\maxArray$ 、 $\minArray$ 中的位置坐标, 求出峰峰值、谷谷值之间的距离度量数组 $array1$, 取其均值 $aveArray1$ 作为字符宽度阈值; 然后判断 $array1$ 第一个数的大小是否小于 $0.5 \times aveArray1$ 或者大于 $2 \times aveArray1$, 满足该条件则判断 $\maxArray$ 中第一个数值是否小于 $aveMin$ 或者大于 $aveMax$, 若满足则车牌区域的左边界向右移动至 $\minArray$ 或者 $\maxArray$ 的第二个值的位置, 同时数组 $array1$ 、 $\maxArray$ 、 $\minArray$ 删除第一个字符, 否则保留, 如此得到了车牌的左边界; 以此类推, 对 $array1$ 中的尾字符判断可得车牌区域的右边界;

(iv)、对上述步骤得到的结果进行分析, 如果 $array1$ 中的个数小于四或者大于某一值, 则该车牌是伪车牌, 应该去掉; 至此, 车牌定位完成。

一种基于容错率和纹理特征的车牌定位方法

技术领域

[0001] 本发明涉及一种定位方法,尤其涉及一种基于容错率和纹理特征的车牌定位方法,属于图像处理技术领域。

背景技术

[0002] 车牌识别技术作为交通管理自动化的重要手段和车辆检测系统的一个重要环节,经过车牌定位、字符分割和字符识别等一系列算法运算,识别出车辆牌照号码,从而使车辆的电脑化监控和管理成为现实。其中,车牌定位是车牌自动识别技术中的一个关键问题,目前许多学者都在研究发展多种车牌定位方法。

[0003] 从根本上讲,目前车牌定位的算法主要分为三类,一类是基于边缘的定位方法,一类是基于颜色的定位方法,一类是基于机器学习的定位方法。

[0004] 基于边缘的定位方法,主要利用了车牌边缘纹理的致密性纹理特征和图像二值化算法。该方法在车牌对比度较好,且车牌高度大于某一数量像素行时,车牌定位准确度较高,其计算简单,能很好地满足实时性要求。但在车牌对比度低的复杂环境下,因边缘损失以及二值化算法鲁棒性不够等造成了车牌区域特征消失,造成该种情况下车牌的定位率不高,同时对倾斜车牌定位不全现象的概率较高。

[0005] 基于颜色的车牌定位方法,主要运用了颜色模型和纹理分析来进行车牌定位。该方法利用了HIS、YIQ或者HSV等颜色模型来定位车牌,但对处于对比度很低环境下的车牌、与车身颜色相近的车牌、污染、颜色严重损失的车牌定位率并不高。

[0006] 基于机器学习的定位方法,一般具有定位率高、抗噪声能力强的优点,但该方法需要数十万的训练样本来训练数据文件,其训练样本需尽可能的多样化,且运算量非常大。在I5的电脑上对1080P图片全图车牌检测耗时有时竟达到秒级,其实时性不好造成了实用性不强。

[0007] 以上定位方法的侧重点都不同,或利用图像二值化寻找车牌纹理,或利用车牌的底色,或通过机器学习,或多种方法相结合。但在车牌自身污染和磨损以及对比度很低的复杂环境下,以上各种定位算法均存在车牌区域的细节特征丢失,这对车牌定位带来了一定程度的困难。

发明内容

[0008] 为了解决上述技术所存在的不足之处,本发明提供了一种基于容错率和纹理特征的车牌定位方法,其技术方案的具体步骤如下:

[0009] 一种基于容错率和纹理特征的车牌定位方法,具体步骤如下:

[0010] A、图像预处理:对抓拍的图像进行图像预处理,得到图像亮度均值;

[0011] B、求取边缘纹理图像:通过设计一种受限自适应直方图,得到垂直边缘纹理图,其特征为:

[0012] (i)、利用公式I得到垂直边缘图像,其中, $e(i, j)$ 为边缘图像矩阵; $g(i, j)$ 为灰度

图像矩阵；

[0013] $e(i, j) = g(i, j+1) - g(i, j)$ 公式I

[0014] (ii)、对垂直边缘图像做增强处理,然后结合图像的亮度,设置自适应阈值过滤边缘强度较小的边缘点;通过一个尺寸为 $1*n$ 窗口选取边缘图像矩阵中每行的连续边缘点平均强度最大的窗口中心点作为边缘点;

[0015] (iii)、利用受限边缘直方图得到其灰度图像;为了突出车牌字符的边缘,抑制或者减少非车牌边缘对车牌定位的影响,利用公式II求得边缘直方图的均值,并判断图像是否为低对比度;其中: $hist(v)$ 为边缘灰度直方图、 $ave0$ 为图像亮度均值;通过判断是否为低对比度图像,然后分别利用公式III求得对比度低图像的锐化阈值、利用公式IV求得正常对比度图像的锐化阈值;利用 $thre$ 通过灰度边缘直方图锐化求得纹理图像;

$$[0016] \quad \begin{cases} aveHist = \sum_{v=0}^{255} (v * hist(v)) / \sum_{v=0}^{255} hist(v) \\ isLowConstract = \begin{cases} ture & ave0 \leq threshold0 \\ false & ave0 > threshold0 \end{cases} \end{cases} \quad \text{公式 II}$$

$$[0017] \quad \begin{cases} thre = \begin{cases} pos0 & hist(pos0) \leq aveHist \\ v & hist(pos0) > aveHist \& hist[v] \leq threshold, v = pos0, ..., 255 \end{cases} \\ threshold = w0 * hist[pos] \end{cases} \quad \text{公式 III}$$

$$[0018] \quad \begin{cases} threshold = w1 * aveHist \\ pos1 = v, \text{ when } \sum_{v=255}^{pos0} hist(v) \geq threshold \\ pos2 = v, \text{ when } threshold = \max_{v \in (pos0, ..., 255)} \left(\frac{\Delta hist(v)}{\Delta v} \right) \\ thre = \begin{cases} pos1 & pos1 \leq pos2 \\ pos2 & pos1 > pos2 \end{cases} \end{cases} \quad \text{公式 IV}$$

[0019] C、提取车牌行特征,得到车牌行簇集:根据实际图片中的最大车牌的尺寸设计字符间距的最大阈值和车牌行长度的最小最大阈值,得到车牌行簇集,其特征为:

[0020] (i)、根据字符间距设置最大宽阈值、最小被检测车牌宽阈值、最大车牌宽阈值,三者分别表示为 $maxCharW$, $minPlateW$, $maxPlateW$;

[0021] (ii)、遍历纹理图片,根据公式V得到符合车牌特征的车牌行线段;

$$[0022] \quad \begin{cases} pLine(i, n) = line(n) \\ line(n) = \begin{cases} n++ & (e(i, j+m) - e(i, j)) \leq maxCharW \& \\ p(i, n, s, e) & minPlateW < (q(i, n, e) - q(i, n, s)) \\ & \leq maxPlateW \end{cases} \end{cases} \quad \text{公式 V}$$

[0023] 其中 i 是图像矩阵中的行号; $line(n)$ 中 n 为第 i 行中的第 n 根车牌行线段; $p(i, n, s, e)$ 为 $line(n)$ 中该线段的首尾点的距离; $q(i, n, s)$ 、 $q(i, n, e)$ 分别为线段起始端、结束端位置;

[0024] 根据上述步骤中的阈值对每行的边缘点进行限定,当遇到边缘点间距小于

maxCharW时,记录为该行的起始点;当间距大于maxCharW时,记录为该行的第一个结束点,并判断该线段大小是否大于最小检测的车牌宽minPlateW,且小于最大检测的车牌宽maxPlateW;如满足则记录为该行的第一个车牌行pLine(i,n),并记录各边缘点的位置,否则不记录;重复本步骤直至图片遍历完;

[0025] D、车牌定位:利用垂直边缘图像、相关阈值,以及容错率方案,对边缘图像进行车牌定位,其特征为:

[0026] (i)、得到车牌区域:根据车牌行集中的原则,利用上述步骤中设置的阈值,按公式VI、公式VII对边缘图像中的车牌行进行归属判断:

$$[0027] \quad \text{Rect}(c) = (\text{pLine}(0,n), \dots, \text{pLine}(i,n) \dots) \quad \text{if} \begin{cases} \text{p}(i,n,s,e) \in (\min(\text{Dis}(0, \dots, t, \dots))) \\ \& \text{p}(i,n,s,e) < \text{maxCharW} \end{cases} \quad \text{公式VI}$$

$$[0028] \quad \text{Dis}(t) = \text{abs}(q(i,n,s) - \text{avxXrect}(c)) + \text{abs}(q(i,n,e) - \text{avxYrect}(c)) \quad \text{公式VII}$$

[0029] 其中,aveXrect(c)、aveYrect(c)为某一车牌区域所有线段左右端点的位置均值;

[0030] 先从图像的第一行进行扫描,如果检测到车牌行线段尺寸满足大于minPlateW且小于maxPlateW,则把该线段归入到车牌区域的第一个车牌行线段,记为rect(0) = (pLine(0,0),...);如果同一行中还有类似的车牌行线段,则新增一个区域,并把它作为第一个车牌行线段,记为rect(1) = (pLine(0,1),...),如此一直到行尾;

[0031] 紧接着扫描图像第i行,如果检测到了车牌行,则求得该车牌行的两端与前面已得到的所有车牌区域中对应的所有车牌行的两端位置坐标均值的欧式距离值,得到距离集Dis(t);从该欧式距离值集中取最小值,如果该最小值满足小于maxCharW预设条件,则该行并入到与最小距离值所对应的那个区域Rect(c) = (pLine(0,0),pline(i,n)...)中;否则把该车牌行作为一个新的区域的第一行,继续检测车牌行线段到行尾;

[0032] (ii)、按上述步骤逐行扫描,一直到图像结束,形成多个车牌区域;为提高复杂环境下车牌的定位率,构造基于连续非零车牌行容错率th1,其最大容错率表示为maxTh1;总非车牌线段行的容错率th2,其最大容错率表示为maxTh2,并将它们融合在定位算法中;

[0033] 在步骤D(i)中,将新检测到的行标记与之前所有检测到的区域中的最后一行做比较;若其差值大于maxTh1,则该车牌行线段不参与欧式距离计算;若新检测到的行标记与距离值集中的最小欧式距离值所在区域内最后一行的行标记之差大于1而小于maxTh1,则把该车牌线段所在行标记与该区域最后一行的行标记之差累积到所属区域的连续非车牌行线段容错率th1上;当该差值大于maxTh1时,新建一个区域,把该车牌行线段的两端点计入到新的区域中,否则判断总非车牌线段行的容错率th2是否大于阈值maxTh2,小于则并入到最小欧式距离值所在的那个区域,同时该差值累积到总非车牌线段行的容错率th2上;

[0034] (iii)、对车牌区域中的伪车牌进行初步排除:设置车牌行数最小阈值、宽高比置信度区间来初步排除伪车牌;宽高比置信度区间指的是车牌宽高比应基本满足交通部制定的长宽比比列,设置一个最大值、最小值作为车牌置信度区域,不在该置信度区间内的区域将被删除;

[0035] E、车牌精定位:通过设计两种基于灰度边缘的自适应密度函数对定位出来的车牌进行去伪存真,并精定位车牌左右边缘,其特征为:

[0036] (i)、获得灰度密度函数:首先在灰度图像中利用定位得到的车牌区域把车牌截取

出来,利用公式VIII对该边缘车牌图像 $e(i, j)$ 进行积分,得到密度函数 $den(x)$;

$$[0037] \quad den(x) = \sum_W \iint_{i=0, j=0}^{i=h, j=u} e(i, j) \partial i \partial j \quad \text{公式VIII}$$

[0038] 其中, W 是一个 $1*n$ 的窗口;取窗口内的数据样本的累加值作为离散点的密度幅值 $den(x)$,其中 n 为车牌区域高 h 乘以一权值系数 w 且 $w < 1$,此窗口大小的设置应使累加区间的长度尽量与车牌字符之间间隔的水平像素个数在数值上一致;

[0039] (ii)、对密度函数 $den(x)$ 求一阶导数,得到函数 $der(x)$;然后对该导数函数的离散值求其绝对值均值 $aveDer$;用 $der(x)$ 中的各个绝对值从左到右与 $aveDer$ 比较,若等于 $aveDer$ 则用数组 $array$ 记录该点的位置,如公式IX、X、XI所示;

$$[0040] \quad array(x) = y \text{ when } abs(der(y)) = aveDer \quad \text{公式IX}$$

$$[0041] \quad \begin{cases} \maxArray(z, y) = \max_{\substack{0 < b < L-1 \\ b < k \leq b+1}} (den(array(x))) & z = 0, 1, \dots \\ y = array(x) \end{cases} \quad \text{公式X}$$

$$[0042] \quad \begin{cases} \minArray(z, y) = \min_{\substack{0 < b < L-1 \\ b < k \leq b+1}} ((den(array(x)))) & z = 0, 1, \dots \\ y = array(x) \end{cases} \quad \text{公式XI}$$

[0043] 取出 $array$ 数组元素,以相邻元素之间的距离作为区间,求得密度函数曲线的最大最小值及其相应位置的数组 $\maxArray$ 、 $\minArray$,进而求得最大最小值的均值 $aveMax$ 、 $aveMin$;根据车牌字符的排列规则,密度函数中字符区域的密度值与字符之间的密度值有较大的落差,如果 $aveMax$ 与 $aveMin$ 的比值小于某一值,则判断为伪车牌,算法转至步骤E(i),继续对下一个车牌进行真伪判别,否则转至下一步骤;

[0044] (iii)、结合最大最小值数组 $\maxArray$ 、 $\minArray$ 中的位置坐标,求出峰峰值、谷谷值之间的距离度量数组 $array1$,取其均值 $aveArray1$ 作为字符宽度阈值;然后判断 $array1$ 第一个数的大小是否小于 $0.5*aveArray1$ 或者大于 $2*aveArray1$,满足该条件则判断 $\maxArray$ 中第一个数值是否小于 $aveMin$ 或者大于 $aveMax$,若满足则车牌区域的左边界向右移动至 $\minArray$ 或者 $\maxArray$ 的第二个值的位置,同时数组 $array1$ 、 $\maxArray$ 、 $\minArray$ 删除第一个字符,否则保留,如此得到了车牌的左边界;以此类推,对 $array1$ 中的尾字符判断可得车牌区域的右边界;

[0045] (iv)、对上述步骤得到的结果进行分析,如果 $array1$ 中的个数小于四或者大于某一值,则该车牌是伪车牌,应该去掉;至此,车牌定位完成。

[0046] 本发明通过在车牌定位算法中融合容错率,提高了车牌定位率;同时通过构建密度函数,在精定位车牌左右边界的同时还去掉了大部分伪车牌,因而避免了大量的伪车牌进入到矫正、分割、识别等模块中,既降低了耗时又降低了误检率,具有很高的算法鲁棒性和较低的时间复杂度。

[0047] 本发明的有益效果是:实现在光照不均、车牌磨损、对比度低下等复杂环境下图片的车牌定位,根据本发明中的车牌纹理提取方法,并结合容错率方法依然可以很好地将车牌定位出来;利用密度函数分析法,较好地解决车牌左右精定位的难题;同时该密度函数具有鉴别、去除伪车牌的功能,从而避免了大量伪车牌进入到车牌矫正、字符分割模块和识别

模块,在减少车牌识别系统耗时的同时,进一步降低了车牌的误检率,提升了车牌的正确识别率,具有很强的实用性。

附图说明

[0048] 图1为本发明的整体步骤流程图。

[0049] 图2为本发明的详细步骤流程图。

具体实施方式

[0050] 下面结合附图和具体实施方式对本发明作进一步详细的说明。

[0051] 如图1、图2所示,本发明包括以下步骤:

[0052] A、图像预处理:对抓拍的图像依次进行高斯滤波、中值滤波和图像增强处理,得到图像整体亮度均值ave0;

[0053] B、求取边缘纹理图像:通过设计一种受限自适应直方图,得到垂直边缘纹理图,具体步骤如下:

[0054] (i)、利用公式I得到垂直边缘图像,其中, $e(i, j)$ 为边缘图像矩阵; $g(i, j)$ 为灰度图像矩阵;

[0055] $e(i, j) = g(i, j+1) - g(i, j)$ 公式I

[0056] 遍历图像求得垂直边缘图像,求垂直边缘图像的优势是能排除车牌框的影响、精确定位上下边框。

[0057] (ii)、对垂直边缘图像做增强处理后,结合图像的亮度,设置自适应阈值过滤边缘强度较小的边缘点;然后根据前景目标边缘处所在位置具有一阶导数最大值或二阶导数的过零点的特性,通过一个尺寸为1*n窗口选取边缘图像矩阵中每行的连续边缘点平均强度最大的窗口中心点作为边缘点;

[0058] (iii)、利用受限边缘直方图得到其灰度图像;受限边缘直方图是一条呈整体下降趋势的曲线,在灰度极小值的附近值很大,越靠近右端值越小,这符合图像中目标边缘较少的规则;为了突出车牌字符的边缘,抑制或者减少非车牌边缘对车牌定位的影响,利用公式II求得边缘直方图的均值;其中, $hist(v)$ 为边缘灰度直方图;

[0059]
$$aveHist = \sum_{v=0}^{255} (v * hist(v)) / \sum_{v=0}^{255} hist(v)$$
 公式II

[0060] 边缘纹理图像的具体求取方法如下:

[0061] (1)、根据公式III、IV、V对图片的对比度进行简单判断并得到低对比度图片的锐化阈值thre:

[0062]
$$isLowConstract = \begin{cases} true & ave0 \leq threshold0 \\ false & ave0 > threshold0 \end{cases}$$
 公式III

[0063]
$$\begin{cases} thre = \begin{cases} pos0 & hist(pos0) \leq aveHist \\ v & hist(pos0) > aveHist \& hist[v] \leq threshold, v = pos0, ..., 255 \end{cases} \\ threshold = w0 * hist[pos] \end{cases}$$
 公式IV

[0064] $threshold = w0 * hist[pos]$ 公式V

[0065] 首先设置一阈值threshold0,判断步骤A中所取得的灰度均值ave0是否大于该阈值;若小于则说明为对比度低的图片,此时判断直方图序列中的第一个非零点pos0出的值是否大于avehist,若小于则直接以该非零点为阈值thre,大于则以该非零点的we*hist[pos0]为初始阈值threshold,此时该初始阈值的位置一定大于pos0,向右搜索到hist[v] ≤ threshold止,以v为阈值thre进行锐化,否则转入下一步骤;

[0066] (2)、根据公式VI、VII、VIII、IX获得正常对比度图像的锐化阈值thre:

[0067] threshold=w1*aveHist 公式VI

[0068]
$$\text{pos2} = v, \text{ when threshold1} = \max_{v \in (\text{pos0}, \dots, 255)} \left(\frac{\Delta \text{hist}(v)}{\Delta v} \right)$$
 公式VII

[0069]
$$\text{pos1} = v, \text{ when } \sum_{v=255}^{\text{pos0}} \text{his}(v) \geq \text{threshold}$$
 公式VIII

[0070]
$$\text{thre} = \begin{cases} \text{pos1} & \text{pos1} \leq \text{pos2} \\ \text{pos2} & \text{pos1} > \text{pos2} \end{cases}$$
 公式IX

[0071] 以w1*aveHist作为边缘点初始阈值threshold,同时以非零值pos0的右端数据的一阶导数的最大值为经验阈值threshold1,且记录其所在位置pos2;从直方图的右边开始向左统计hist[v],如果大于threshold则停止,得到该点的位置pos1;如果该点位置处的直方图值小于等于阈值threshold1,则取pos1为阈值thre;否则取pos2的点作为阈值thre;

[0072] (3)、利用阈值thre,通过直方图映射来锐化求得纹理图像。

[0073] C、提取车牌行特征,得到车牌行簇集:根据实际图片中的最大车牌的尺寸设计字符间距的最大阈值和车牌行长度的最小、最大阈值,得到车牌行簇集,具体操作步骤如下:

[0074] (i)、通过公安部制定的公共安全标准GA36-2007,根据字符间距设置最大宽阈值、最小被检测车牌宽阈值、最大车牌宽阈值,三者分别表示为maxCharW,minPlateW,maxPlateW;

[0075] (ii)、遍历纹理图片,根据公式X得到符合车牌特征的车牌行线段;

[0076]
$$\begin{cases} \text{pLine}(i,n) = \text{line}(n) \\ \text{line}(n) = \begin{cases} n++ & (e(i,j+m) - e(i,j)) \leq \text{maxCharW} \& \\ \text{p}(i,n,s,e) & \text{minPlateW} < (q(i,n,e) - q(i,n,s)) \\ & \leq \text{maxPlateW} \end{cases} \end{cases}$$
 公式X

[0077] 其中i是图像矩阵中的行号;line(n)中n为第i行中的第n根车牌行线段;p(i,n,s,e)为line(n)中该线段的首尾点的距离;q(i,n,s)、q(i,n,e)分别为线段起始端、结束端位置。

[0078] 根据上述步骤中的阈值对每行的边缘点进行限定,当遇到边缘点间距小于maxCharW时,记录为该行的起始点;当间距大于maxCharW时,记录为该行的第一个结束点,并判断该线段大小是否大于最小检测的车牌宽minPlateW,且小于最大检测的车牌宽maxPlateW;如满足则记录为该行的第一个车牌行pLine(i,n),并记录各边缘点的位置,否则不记录;重复本步骤直至图片遍历完;

[0079] D、根据容错率进行车牌定位:利用垂直边缘图像、相关阈值构造容错率方案,对边缘图像进行车牌定位,具体步骤如下:

[0080] (i)、得到车牌区域：根据车牌行集中的原则，利用上述步骤设置的阈值，按公式XI、公式XII对边缘图像中的车牌行进行归属判断：

$$[0081] \quad \text{Rect}(c) = (\text{pLine}(0,n), \dots, \text{pLine}(i,n) \dots) \quad \text{if} \begin{cases} \text{p}(i,n,s,e) \in (\min(\text{Dis}(0, \dots, t, \dots))) \\ \& \text{p}(i,n,s,e) < \text{maxCharW} \end{cases} \quad \text{公式XI}$$

$$[0082] \quad \text{Dis}(t) = \text{abs}(q(i,n,s) - \text{avxXrect}(c)) + \text{abs}(q(i,n,e) - \text{avxYrect}(c)) \quad \text{公式XII}$$

[0083] 其中aveXrect(c)、aveYrect(c)为某一车牌区域所有线段左右端点的位置均值；

[0084] 先从图像的第一行进行扫描，如果检测到车牌行线段尺寸满足大于minPlateW且小于maxPlateW，则把该线段归入到车牌区域的第一个车牌行线段，记为rect(0) = (pLine(0,0), ...)；如果同一行中还有类似的车牌行线段，则新增一个区域，并把它作为第一个车牌行线段，记为rect(1) = (pLine(0,1), ...)，如此一直到行尾；

[0085] 紧接着扫描图像第i行，如果检测到了车牌行，则求得该车牌行的两端与前面已得到的所有车牌区域中对应的所有车牌行的两端位置坐标均值的欧式距离值，得到距离集Dis(i)；从该欧式距离值集中取最小值，如果该最小值满足小于maxCharW预设条件，则该行并入到与最小距离值所对应的那个区域Rect(c) = (pLine(0,0), pLine(i,n)...)中；否则把该车牌行作为一个新的区域的第一行，继续检测车牌行线段到行尾；

[0086] (ii)、按上述步骤逐行扫描，一直到图像结束，形成多个车牌区域；因任何算法均存在无法满足所有复杂环境下车牌定位的局限性，本发明中的定位算法也不可能对各种复杂环境下的车牌定位都有理想的效果，如在某些环境中如磨损、污染、对比度低车牌会损失一些边缘细节，从而导致车牌区域中部分行线段丢失，进而导致车牌定位不全或者根本定位不出车牌。因此，本发明构造基于连续非零车牌行容错率th1，其最大容错率表示为maxTh1；总非车牌线段行的容错率th2，其最大容错率表示为maxTh2，并将它们融合在定位算法中；

[0087] 在步骤D(i)中，将新检测到的行标记与之前所有检测到的区域中的最后一行做比较；若其差值大于maxTh1，则该车牌行线段不参与欧式距离计算；若新检测到的行标记与距离值集中的最小欧式距离值所在区域内最后一行的行标记之差大于1而小于maxTh1，则把该车牌线段所在行标记与该区域最后一行的行标记之差累积到所属区域的连续非车牌行线段容错率th1上；当该差值大于maxTh1时，新建一个区域，把该车牌行线段的两端点计入到新的区域中，否则判断总非车牌线段行的容错率th2是否大于阈值maxTh2，小于则并入到最小欧式距离值所在的那个区域，同时该差值累积到总非车牌线段行的容错率th2上；通过这两个容错率的设计，极大地提升了车牌定位率，同时定位出来的车牌上下边框也较精确。

[0088] (iii)、对车牌区域中的伪车牌进行初步排除：设置车牌行数最小阈值、宽高比置信度区间来初步排除伪车牌；宽高比置信度区间指的是车牌宽高比应基本满足交通部制定的长宽比比例，设置一个最大值、最小值作为车牌置信度区域，不在该置信度区间内的区域将被删除；

[0089] 由于图像中与车牌相近的纹理区域存在，任何一种定位方法都不可避免会带来一些伪车牌。本发明设计分别基于灰度图像、边缘图像的自适应密度函数对定位出来的车牌进行辨别、精定位车牌。在具体实施方式中只阐述基于灰度图像的密度函数的设计方法，基于边缘图的密度函数设计方法与其类似，不再详述。

[0090] E、车牌精定位:通过设计两种基于灰度边缘的自适应密度函数对定位出来的车牌进行去伪存真,并精定位车牌左右边缘,具体步骤如下:

[0091] (i)、获得灰度密度函数:首先在灰度图像中利用定位得到的车牌区域把车牌截取出来,利用公式VIII对该边缘车牌图像 $e(i, j)$ 进行积分,得到密度函数 $den(x)$;

$$[0092] \quad den(x) = \sum_w \iint_{i=0, j=0}^{i=h, j=u} e(i, j) \partial i \partial j \quad \text{公式 XIII}$$

[0093] 其中, w 是一个 $1*n$ 的窗口;取窗口内的数据样本的累加值作为离散点的密度幅值 $den(x)$,其中 n 为车牌区域高 h 乘以一权值系数 w 且 $w < 1$,此窗口大小的设置应使累加区间的长度尽量与车牌字符之间间隔的水平像素个数在数值上一致;

[0094] (ii)、对密度函数 $den(x)$ 求一阶导数,得到函数 $der(x)$;然后对该导数函数的离散值求其绝对值均值 $aveDer$;用 $der(x)$ 中的各个绝对值从左到右与 $aveDer$ 比较,若等于 $aveDer$ 则用数组 $array$ 记录该点的位置,如公式XIV、XV、XVI所示;

$$[0095] \quad array(x) = y \text{ when } abs(der(y)) < aveDer \quad \text{公式 XIV}$$

$$[0096] \quad \begin{cases} \maxArray(z, y) = \max_{\substack{0 < b < L-1 \\ b < k \leq b+1}} (den(array(x))) & z = 0, 1, \dots \\ y = array(x) \end{cases} \quad \text{公式 XV}$$

$$[0097] \quad \begin{cases} \minArray(z, y) = \min_{\substack{0 < b < L-1 \\ b < k \leq b+1}} ((den(array(x)))) & z = 0, 1, \dots \\ y = array(x) \end{cases} \quad \text{公式 XVI}$$

[0098] 取出 $array$ 数组元素,以相邻元素之间的距离作为区间,求得密度函数曲线的最大最小值及其相应位置的数组 $\maxArray$ 、 $\minArray$,进而求得最大最小值的均值 $aveMax$ 、 $aveMin$;根据车牌字符的排列规则,密度函数中字符区域的密度值与字符之间的密度值有较大的落差,如果 $aveMax$ 与 $aveMin$ 的比值小于某一值,则判断为伪车牌,算法转至步骤E(i),继续对下一个车牌进行真伪判别,否则转至下一步骤;

[0099] (iii)、结合最大最小值数组 $\maxArray$ 、 $\minArray$ 中的位置坐标,求出峰峰值、谷谷值之间的距离度量数组 $array1$,取其均值 $aveArray1$ 作为字符宽度阈值;然后判断 $array1$ 第一个数的大小是否小于 $0.5*aveArray1$ 或者大于 $2*aveArray1$,满足该条件则判断 $\maxArray$ 中第一个数值是否小于 $aveMin$ 或者大于 $aveMax$,若满足则车牌区域的左边界向右移动至 $\minArray$ 或者 $\maxArray$ 的第二个值的位置,同时数组 $array1$ 、 $\maxArray$ 、 $\minArray$ 删除第一个字符,否则保留,如此得到了车牌的左边界;以此类推,对 $array1$ 中的尾字符判断可得车牌区域的右边界;

[0100] (iv)、对上述步骤得到的结果进行分析,如果 $array1$ 中的个数小于4或者大于某一值(如图片中车身上的广告),则该车牌是伪车牌,应该去掉;至此,车牌定位完成。

[0101] 本发明的整体流程为:先对要处理的图像进行滤波,增强,求得图像整体亮度均值,然后利用垂直边缘算子得到其灰度边缘;结合图像的亮度,设置自适应阈值去掉强度较小的边缘点后,再通过设计的受限自适应边缘直方图结合图像区域亮度均值得到边缘纹理图像;紧接着依据车牌的纹理特征并按车牌行特征提取算法对图像进行扫描,得到车牌行线段簇集;再对车牌行线段按设计的定位算法原理进行归并,在遍历扫描车牌行的过程中

通过融合两种容错率方法找到车牌的所有可能区域;最后构造了纹理密度函数来精定位左右边缘、除去伪车牌。

[0102] 本发明通过在车牌定位算法中融合容错率,提高了在对光照不均、车牌磨损、对比度低下等复杂环境中的车牌定位率;同时通过构建密度函数,在精定位车牌左右边界的同时还去掉了大部分伪车牌,因而避免了大量的伪车牌进入到车牌矫正、分割、识别等模块中,既降低了耗时又降低了误检率,具有很高的算法鲁棒性和较低的时间复杂度。

[0103] 上述实施方式并非是对本发明的限制,本发明也并不仅限于上述举例,本技术领域的技术人员在本发明的技术方案范围内所做出的变化、改型、添加或替换,也均属于本发明的保护范围。

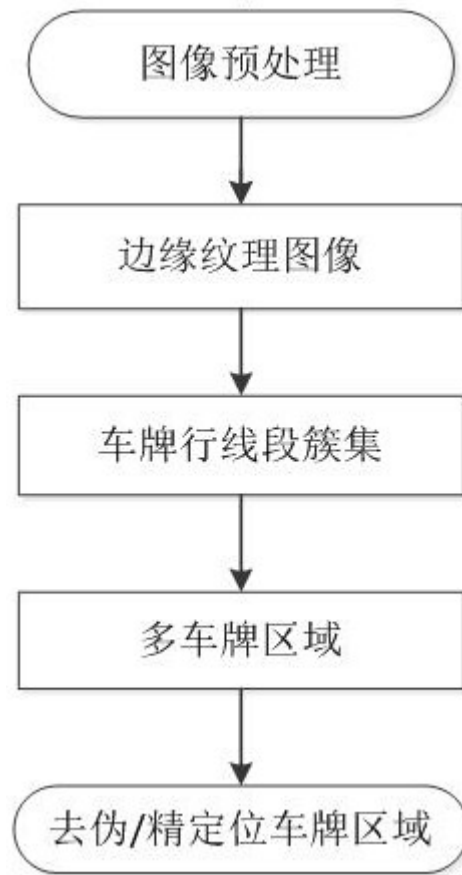


图1

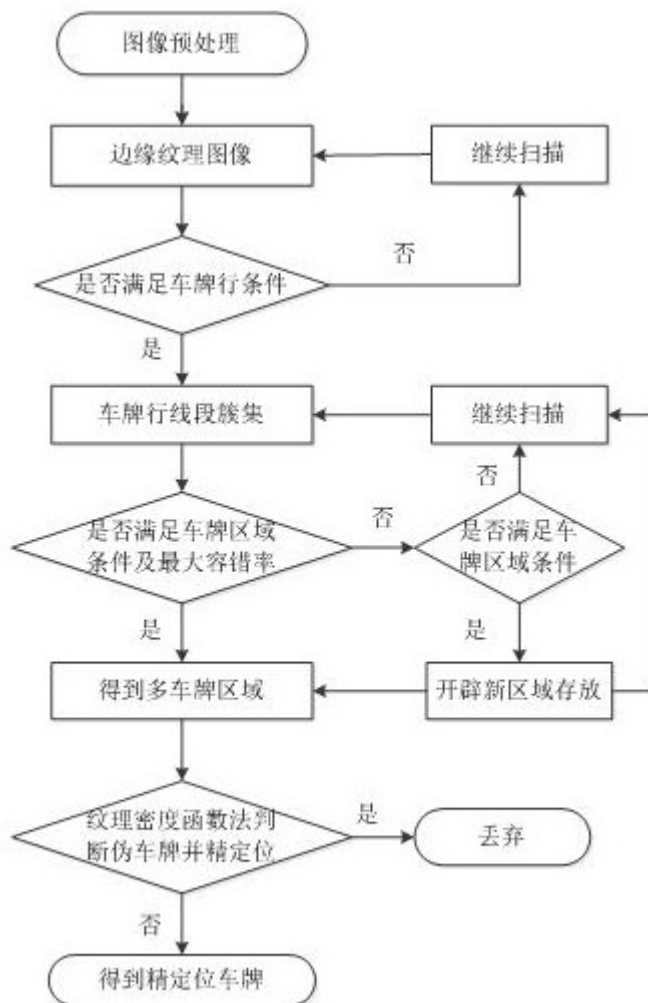


图2