

基于 Hough 变换的 PDF417 条码图像倾斜矫正

Skew angle detection and correction of PDF417 barcode image based on Hough transform

周利华 刘志 (浙江工业大学屏峰校区杭州 310023)

摘要:在对 PDF417 条码图像扫描输入的过程中,条码图像不可避免地发生倾斜,因此必需先对图像进行倾斜矫正,才能实现条码的准确识别。Hough 变换具有抗噪声能力强的特点,是一种检测图像倾斜角度的重要方法。为了克服 Hough 变换计算量大的缺点,先通过行差运算提取图像水平边缘,然后只对提取的边缘进行两级 Hough 变换,实现倾斜角度的检测。本文采用基于直线拟合的方法实现倾斜图像的矫正。与传统的 Hough 变换以及其它的倾斜图像矫正方法相比,本文实现了准确性与快速性的很好结合。

关键词:PDF417 条码 条码识别 Hough 变换

1 引言

二维条码可以在平面的纵横两个方向上记录信息,能够实现对物体的“描述”,而且二维条码具有很高的纠错率、编码范围广泛、制作简便且成本低廉,是不可多得的具有较强防伪和纠错性能的“便携数据库”^[1]。其中 PDF417 条码不仅具有错误侦测能力,而且可以从受损的条码中读回完整的信息,其错误复原率最高可达到 50%,在流通行业得到广泛的应用^[2]。PDF417 条码译码的关键问题是找到每个码字所对应的条空信息,即黑条白条的宽度比。由于在实际的识别过程中,扫描进来的条码图像或多或少出现倾斜,使得我们无法得到准确的条空信息,给快速、正确地识别条码带来了困难,因此,必需先对倾斜图像进行矫正,以便实现对条码的正确识别。

图像倾斜矫正的关键是检测图像的倾斜角,目前估算图像倾斜角的方法主要有以下几种: Hough 变换、投影特性法、傅立叶变换。投影特性法就是对图像进行不同角度的投影测试,提取最佳的投影结果,从而得到图像倾斜角,此方法存在的问题是计算量大,且精度取决于进行投影测试时的单位步长角度值。Postl 提出傅立叶变换法^[3],即傅立叶空间密度最高的方向就是要求的倾斜角度,此方法也存在计算的空间和时间的复杂度太高的问题。因此,本文提出了一种基于 Hough 变换的、更快捷的倾斜角检测算

法。该算法对条码图像进行两次 Hough 变换,其中 Hough 变换只对图像中的水平边缘进行,这样就大大减少了运算量,实现了倾斜角的快速检测。通常图像倾斜矫正都采用旋转变换的方法,此方法计算量较大,本文采用基于直线拟合的方法,可快速地实现倾斜图像的矫正^[4]。

2 Hough 变换

Hough 变换是图像空间和参数空间之间的一种变换, Hough 变换不仅可以检测图像中直线和连接处在同一条直线上的点,也可以用来检测圆、椭圆等曲线^[5]。

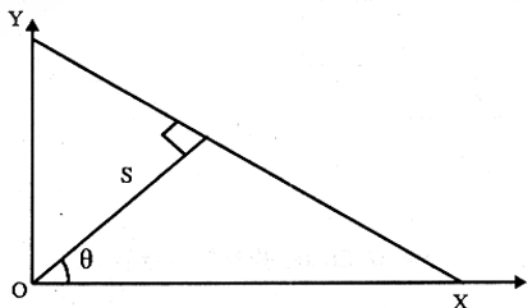


图 1 一条与原点距离为 S , 方向角为 θ 的一条直线

由于二维图像中与原点距离为 S , 方向角为 θ 的直线(如图 1 所示)可表示为:

$$S = x \cos \theta + y \sin \theta \quad (1)$$

其中 $S, \theta (0 \leq \theta \leq \pi)$ 为参数。直线上的每一点都满足(1)式,从而图像中的直线被映射到参数空间的一点 (S, θ) ^[4]。基于这一点,我们将直线检测问题转化为参数空间寻找局部最大值的问题,通过在参数空间里进行累加统计完成检测。Hough 变换是一个累加的变换过程,它的主要优点是利用了图像全局特性,所以受噪声和边界间断的影响较小,具有很好的鲁棒性。但如何解决精确度和运算复杂度的矛盾以及减少它对存储空间的要求是应用 Hough 变换的主要困难。参数空间量化得越精细,算法的精度越高,但算法所需的存储空间和运算量也越大。为了解决这些问题,人们提出了分级 Hough 变换的思想,通过多次 Hough 变换来降低算法的复杂度。

测

3 基于 Hough 变换的条码图像倾斜角检测

3.1 行差运算

PDF417 二维条码图像(如图 2 所示,已经经过二值化处理)是一种堆叠式条码(由若干一维条码堆叠而成),可将条码看成由一系列平行粗线段组成^[5]。在运用 Hough 变换进行角度测量时,如果对条码图像中每一个黑点(像素值为 0)都进行 Hough 变换,则计算量相当大,所需的存储空间也相当大,因此本文采用行差运算提取图像中的特征点——水平边缘点的方法。另外,通过行差运算后进行 Hough 变换,积累矩阵的峰值更加明显(见 2.2)。行差运算的内容如下:对图像进行列扫描,当像素值发生变化时,使该像素值为 0;若像素未发生变化,则使该像素值为 255。通过行差运算,得到如图 3 所示的由一系列平行细线段组成的图像。

3.2 Hough 变换

用 Hough 变换检测由一系列直线组成的行线性图像(如图 4 所示)的倾斜角的方法如下:

- (1) 找出 (θ, S) 矩阵中每个 S 向量中的最大值;
- (2) 对每个 S 向量中的元素,将不是最大值的清零;
- (3) 对所有 S 向量进行累加;
- (4) 在累加后的 S 向量中找出最大值。

最大 S 向量下标所对应的 θ 值即为这一系列直线的倾斜角。但对粗直线进行 Hough 变换时,在一



图 2 倾斜的 PDF417 条码图像



图 3 行差运算后的图像

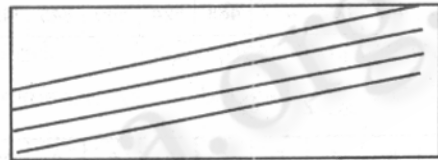


图 4 行线性图像



图 5 粗直线 Hough 变换分析

些 S 向量中,峰值并不出现在粗直线的实际倾斜角度上。如图 5 所示,直线 2 上的点要多于直线 1 上的点,因此对某一向量 S 来说,峰值出现在距原点为 S 的粗直线内且与线条边缘成最大角的直线上。对图

2(图像倾斜角为 3.45)进行 Hough 变换时,S 向量的积累量如表 1 所示。

表 1 粗直线 Hough 变换的积累量

θ 角	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1
积累量	4629	3674	5303	2647	2468	3575	1621	1801	737	254

表 1 续

0	1	2	3	4	5	6	7	8	9	10
0	771	1473	2050	2128	2861	2918	3855	3174	3918	6006

由表 1 可知,S 向量的最大积累值为 6006,峰值出现的角度为 10,并不在实际的倾斜角度范围之内。

而对细直线簇进行 Hough 变换时,虽然也有一些 S 向量峰值不出现在倾斜角处,但对所有 S 向量进行累加,最大的的积累值仍出现在倾斜角处。对图 3 进行 Hough 变换时,S 向量的积累量如表 2 所示。

表 2 细直线 Hough 变换的积累量

θ 角	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1
积累量	224	354	443	307	479	468	524	580	845	1458

表 2 续

0	1	2	3	4	5	6	7	8	9	10
252	1455	1216	2228	1651	647	328	335	514	331	333

由表 2 可知,S 向量的最大积累值为 2228,峰值出现的角度为 3,在实际的倾斜角范围内,我们可以通过多次 Hough 变换逼近实际的倾斜角。

因此,我们要先对图像进行行差运算,然后再进行 Hough 变换。为了解决 Hough 变换运算的复杂度大的问题,我们采取以下方法。首先,考虑到扫描图像的倾斜程度一般不可能太大,所以我们可以通过限制搜索范围来降低运算的复杂度:将搜索区间限制在 $[-\alpha, +\alpha]$ 内,这里 α 是我们假定的最大倾斜角。其次,为了得到精确的倾斜角,必需细化角度参数 θ ,这样就必然带来存储空间和运算量大的问题,这里我们采用两级 Hough 变换方法来解决:一次 Hough 变换得到倾斜角的区间;在第一次得到的角度区间再进

行二次 Hough 变换,求得精确的倾斜角度。

4 条码图像倾斜矫正

得到图像倾斜角度 θ 之后,便可对倾斜图像进行矫正。通常,倾斜校正采用旋转变换的方法,即:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (2)$$

该方法虽然简单,但计算量较大,长 $\times$ 宽为 $W \times H$ 的图像,采用旋转变换方法的计算包括 $4W \times H$ 次乘法运算和 $2W \times H$ 次加法运算。为了减少计算量,本文采用直线拟合的方法。 $W \times H$ 的图像在竖直方向的最大偏移值 $Y_{off} = [W \times \tan\theta]$, ($[x]$ 指不小于 x 的最小整数),则第 i 列的像素点的竖直方向偏移值 $\Delta Y_i = Y_{off} * i/W$;相应地,图像在水平方向的最大偏移值 $X_{off} = [H \times \tan\theta]$,则第 i 行的像素点的水平方向偏移值 $\Delta X_i = X_{off} * i/H$ 。则图像矫正可表示为: $x' = x + \Delta X$; $y' = y + \Delta Y$ (3) (其中 $\Delta X, \Delta Y$ 分别为像素点水平和竖直方向的偏移量)。该方法只需 $2W \times H$ 次加法运算,大大节省了矫正时间。

5 条码图像倾斜矫正的具体实验步骤

下面给出基于 Hough 变换的 PDF417 条码图像矫正的具体实验步骤,可供读者参考。

- (1) 原始条码图像读入。
- (2) 图像预处理:先进行灰度化,再进行二值化^[7]转化成二值图像。
- (3) 对二维条码图象进行行差变换。
- (4) 进行 Hough 变换(两级 Hough 变换)

由图象空间目标象素的坐标去计算参数空间中参考点的可能轨迹,并对给出的参考点进行累加计数。如果参数空间是 (S, θ) ,累加矩阵 HoughMat,按公式(1)进行计算。这里, x, y 是象素的直角坐标。由此, Hough 变换算法设计为:

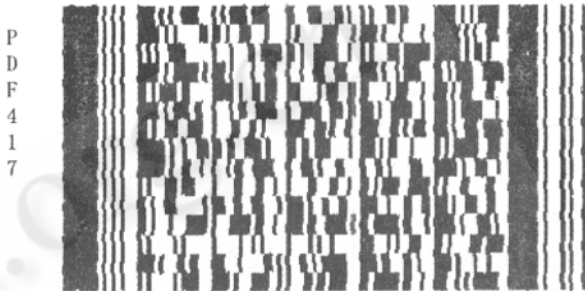
- 设定累加矩阵 HoughMat (S, θ) 的合适大小 (实验中我们设定倾斜角最大为 α ,图像长和宽的平方根为 L, S 取 $0 \sim L$,步长为 1。一次 Hough 变换, θ 取 $-\alpha \sim \alpha$,步长为 1,求得 θ_1 ;二次 Hough 变换, θ 取 $\theta_1 - 1 \sim \theta_1 + 1$,步长为 0.05);
- 为累加矩阵 HoughMat (S, θ) 分配内存空间,并置初值为 0;

• 遍历图像每一个目标像素点 (x', y') , 若像素值为 0 (黑点), 则对每一个 θ 的取值 θ' , 计算 $S' = x' \cos \theta' + y' \sin \theta'$, 并把矩阵 (S', θ') 相应元素累计值加 1, 即 $\text{HoughMat}(S', \theta') = \text{HoughMat}(S', \theta') + 1$;

0.05 度), 得到理想的图像矫正效果, 并且对扫描图像具有很好的鲁棒性。

二维条码在手机领域中的应用将带来增值业务的全新模式

二维条码在手机领域中的应用将带来增值业务的全新模式



拿出你的手机拍出以上二维条码, 即可登陆我公司的网上平台, 了解更多信息

拿出你的手机拍出以上二维条码, 即可登陆我公司的网上平台, 了解更多信息

(a) 倾斜条码图像

(b) 矫正图像

图 6 实验效果图

• 在累加过程中, 为了使峰值更加明显, 选取合适的 K 值 (在实验中我们选取的 K 值为 0.85), 在某一 S 向量中将小于 $K \cdot \text{Max}(i)$ 的元素设为零, 其中 $\text{Max}(i)$ 为每一 S 向量的最大值, i 从 0 到 $\text{Max}(S)$;

• 对每一个 θ 的取值 θ' , 累加 S 向量值。 $S_{\theta'} = \text{sum}(S(i)), i = 0 \dots \text{Max}(S)$;

• 求出 S 向量值取最大值时所对应的角度 θ' , θ' 角即为最后需要检测的图像倾斜角度。

(5) 图像倾斜矫正

利用公式 (3) 对图像进行矫正。

另外, 由于 Hough 变换是一个累加过程, 具有很强的抗噪声能力, 即使当扫描进来的条码图像旁边有文字等干扰信息, 也能够得到理想的效果, 实验结果如图 6 所示。

6 小结

本文针对 PDF417 二维条码图像的特点, 提出了行差运算和二次 HOUGH 变换相结合进行二维条码图像倾斜矫正的方法。实验结果表明, 该方法能够准确快速地检测出扫描题图像的倾斜角度 (精度达到

参考文献

- 1 Pavlidis T. A new paper/computer interface: two-dimensional symbologies [J]. IEEE Computer Magazine, 2000, 2: 145 - 151.
- 2 赛勤颖、姚青, 条码技术的发展及其应用 [J], 计算机工程与科学, 2003, 25 (5): 50 - 52.
- 3 章毓晋, 图像工程 (中册) 图像分析 [M], 清华大学出版社, 2005.
- 4 李政、杨扬、顾斌等, 一种基于 Hough 变换的文档图像倾斜纠正方法 [J], 计算机应用, 2005, 25 (3): 583 - 585.
- 5 Jiann - Yeou Rau, Liang - Chien Chen. Fast Straight Lines Detection Using Hough Transform with Principal Axis Analysis [J], 航测及遥测学刊, 2003, 8 (1): 15 - 34.
- 6 张玲、胡东红、孔华锋等, 二维条码码图结构特性分析 [J], 湖北大学学报 (自然科学版), 2004, 26 (3): 226 - 231.
- 7 杨淑莹, VC++ 图像处理程序设计 [M], 清华大学出版社, 2005.