

一种基于几何特征参数的圆检测方法

陈爱军, 李金宗

(哈尔滨工业大学图像信息技术与工程研究所, 哈尔滨 150001)

摘要: 提出了一种利用一组几何特征参数对图像中的圆形进行检测的方法。对提取出的边缘进行多边形拟合, 并对拟合后的多边形进行归一化处理, 提取平移、旋转和尺度变换不变性的几何特征, 计算相应的几何特征参数, 将几何特征参数满足一定条件的形状识别为圆形。描述了该方法的具体步骤, 并与随机 Hough 变换进行了对比实验。对合成图像和真实图像进行的实验结果表明, 该方法具有较高的效率和实用性。

关键词: 圆检测; 几何特征参数; 特征点; 图像分割; 边缘提取

Method for Circle Detection Using Geometric Feature Parameters

CHEN Aijun, LI Jinzong

(Institute of Image Information Technology and Engineering, Harbin Institute of Technology, Harbin 150001)

【Abstract】 A circle detection method using a group of geometric feature parameters is proposed. Polygon fitting to the extracted edges is implemented and the fitted polygons are normalized to obtain invariant geometric features to translation, rotation, and scale change. The parameters corresponding to the geometric features are calculated. Those shapes with the geometric feature parameters satisfying certain conditions are recognized as circles. The algorithm is described, and comparison experiments with the randomized hough transformation are made. The experimental results on synthetic and real images show that the proposed method is efficient and practical.

【Key words】 Circle detection; Geometric feature parameter; Feature point; Image segmentation; Edge extraction

快速而准确地检测圆在计算机视觉领域, 特别是自动化检验和装配领域有着广泛的应用前景。Hough变换是检测圆的有效方法, 但其复杂的运算、巨大的数据量严重限制了它的应用和发展。为了克服上述缺陷, Xu等^[1]提出了随机Hough变换(Randomized Hough Transform, RHT), 但利用这种方法在复杂图像中进行圆形识别时, 由于随机采样引入大量的无效单元, 因此造成大量无效积累。多年来, 研究者们提出了多种圆检测方法。如利用圆的几何对称性对图像中不完整的圆形进行识别^[2,3], 但这种方法不能识别发生部分变形, 即不完全对称的圆形; 模糊识别方法^[4,5]克服了上述方法的不足, 但这种方法要求待识别的圆形个数为已知。

1 算法描述

本文算法的基本思想是: 首先采用迭代阈值法对原图像进行二值化分割, 然后采用边界跟踪方法进行边缘提取, 得到一系列有序的、表示目标完整轮廓的边缘点集, 对每一个边缘点集构成的曲线进行多边形拟合, 并对拟合后的多边形进行归一化处理, 以归一化后的多边形的顶点作为特征点求取具有旋转、平移及尺度变化不变性的几何特征, 并获取几何特征与单位圆相应特征之比的参数, 从而得到最终几何特征参数。如果待检测形状的几何特征参数大于预先给定的相应阈值, 则此形状被识别为圆。具体检测流程如图1所示。

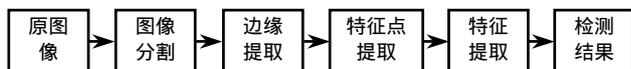


图1 圆检测过程框图

1.1 图像分割

图像分割的方法较多, 其中阈值化是一个最常用的工具。

本文利用一种阈值法——迭代阈值方法^[6]对图像进行分割。迭代阈值方法如下:

- (1) 选择灰度阈值 T 的初始估计值;
- (2) 用 T 将图像像素分成两组: G_1 和 G_2 ;
- (3) 分别计算 G_1 和 G_2 中的所有像素的平均灰度值 μ_1 和 μ_2 ;
- (4) 计算新的门限值:
$$T = (\mu_1 + \mu_2) / 2$$
- (5) 重复(2)~(4), 直到逐次迭代所得的 T 值之差小于事先定义的参数 T_0 。

本文将图像的最大灰度值和最小灰度值的均值作为灰度阈值 T 的初始估计值。迭代得到灰度阈值 T 后, 将大于 T 的像素的灰度值设为 255, 其余像素的灰度值设为 0, 这样便得到了二值图像。

1.2 边缘提取

经典的边缘提取方法是对整幅图像按像素在其邻域内构造边缘检测算子, 常用的有 Sobel、Gauss 和 Canny 等算子, 这些传统的边缘检测算法虽然应用范围较广, 但实际计算比较费时, 实时性较差。为了更为快速地实现边缘提取, 本文选用了建立在二值图像上的边界跟踪的方法^[7]。

边界跟踪算法是通过跟踪构成目标闭区域边界线的每个像素, 记录二值图像的坐标序列来完成的。具体算法是, 先用大步距的方法找到起始点, 然后再采用小步距顺时针搜索的方法寻找其余的边界点。在搜索过程中可将背景区域看作

作者简介: 陈爱军(1976-), 男, 博士生, 主研方向: 图像处理, 模式识别; 李金宗, 教授、博导

收稿日期: 2006-03-27 **E-mail:** jschenaj@163.com

平面, 目标区域视为建筑物, 一边让右手接触建筑物的壁面一边前进, 最后返回出发点, 这样就可完成整个目标的边缘跟踪, 即实现了边缘提取。

1.3 特征点提取

图像经过分割和边缘提取后, 可以得到一系列有序的、表示目标完整轮廓的边缘点集, 这些点集描述了目标边缘的形状特征。如果以此点集作为识别目标的特征点, 则识别算法的计算量十分巨大。为了达到压缩点集的目的, 本文通过对边缘点集构成的曲线进行多边形拟合, 以多边形的顶点作为特征点。

多边形拟合的思想如下:

将每一个边缘点集P中的第1点 $S(x_s, y_s)$ 和最后一点 $E(x_e, y_e)$ 作为一条虚拟线段的两个端点, 计算该点集中其它点 $K(x_k, y_k)$ 到该线段的距离 d_k 。

$$d_k = \frac{|x_k(y_s - y_e) + y_k(x_e - x_s) + y_e x_s - y_s x_e|}{\sqrt{(x_s - x_e)^2 + (y_s - y_e)^2}} \quad k = 2, 3, \dots, n-1 \quad (1)$$

式中, n 为虚拟线段间的边缘点个数(包含两个端点)。设 d_{\max} 为虚拟线段两端点间的边缘点集中点到此虚拟线段的距离, 其对应的点为 $M(x_{\max}, y_{\max})$ 。 d_{\max} 由下式给出。

$$d_{\max} = \max(d_k) \quad k = 2, 3, \dots, n-1 \quad (2)$$

如果 d_{\max} 小于给定的阈值 d_{th} , 则将该虚拟线段定为多边形的一条边; 否则, 将点 M 设为此虚拟线段的一个端点, 即 (x_e, y_e) (x_{\max}, y_{\max}) , 另一端点保持不变, 利用式(1)计算点 S 和点 E 间的边缘点集中的点到虚拟线段的距离 d_k , 然后再判断 d_{\max} 与 d_{th} 的大小关系。如果 $d_{\max} < d_{th}$, 则新生成的 M 和此边缘点集中的第1点 S 为多边形一条边的两个端点。将点 M 作为虚拟线段的起点, 即 (x_s, y_s) (x_{\max}, y_{\max}) , 将边缘点集中的最后1点作为虚拟线段的终点, 继续进行上述过程; 否则, 新生成的点 M 作为虚拟线段的终点, 而起点保持不变, 利用式(2)计算 d_{\max} , 直到 $d_{\max} < d_{th}$ 为止。

具体算法如下:

- (1) 令有序边缘点集个数为 n , 当前边缘点集序号 $Listnum=1$, 给定一个距离阈值 d_{th} , 并创建一个空的链表, 以便存储拟合多边形的顶点坐标;
- (2) 如果 $Listnum > n$, 结束; 否则继续;
- (3) 令当前坐标链中像素个数为 N , 虚拟线段起点的坐标和终点坐标在当前边缘集中的序号分别为 Fst 和 Lst , 并令 $Fst=1$ $Lst=N$;
- (4) 如果 $Fst > Lst$, 令 $Listnum = Listnum + 1$, 转(2); 否则继续;
- (5) 利用式(2)计算 d_{\max} , 令取 d_{\max} 时的点与虚拟线段起点的序号之差为 i_{\max} 。如果 $d_{\max} > d_{th}$, 则令 $Lst = Lst + i_{\max}$, 转(5); 否则继续;
- (6) 将虚拟线段起点坐标和终点坐标存入链表, 并令 $Fst = Lst$, $Lst = N$, 转(4)。

一个边缘点集经过上述处理后, 可以得到拟合后的多边形顶点的有序集合 V , 如图2所示。但由于成像距离、方位以及位置等因素的变化, 图像可能会发生旋转、平移及尺度变化, 为了满足平移、旋转和尺度变换不变性的要求, 必须对拟合的多边形进行归一化。本文以边界点集的形心到多边形顶点的最大距离作为归一化因子。

令边界点集形心 C 坐标为 (\bar{x}, \bar{y}) , 则

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i, \quad \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i \quad (3)$$

其中, N 为边界点集的像素数目, (x_i, y_i) 为边缘像素点 p_i 的坐标。于是归一化因子 D 可表示为

$$D = \max |v_i - C| = \max \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2} \quad i = 1, 2, \dots, M \quad (4)$$

其中, M 为多边形顶点数。

归一化后的特征点 v'_i 的坐标 (x'_i, y'_i) 与归一化前的特征点 v_i 的坐标 (x_i, y_i) 的对应关系为

$$x'_i = (x_i - \bar{x})/D, \quad y'_i = (y_i - \bar{y})/D \quad i = 1, 2, \dots, M \quad (5)$$

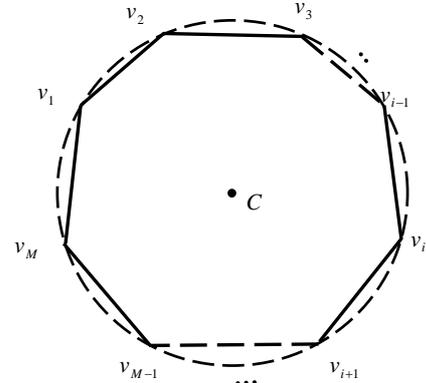


图2 拟合多边形

1.4 特征提取

经过归一化处理后, 下一步的工作是提取归一化多边形的几何特征, 即周长、平均极距离和面积, 见图3。

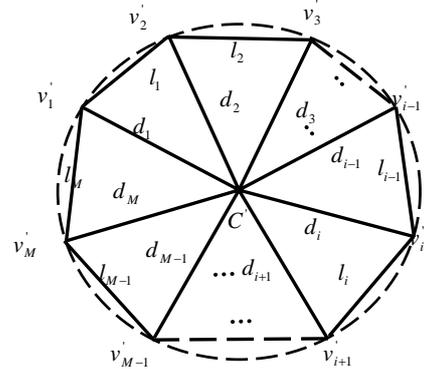


图3 归一化的多边形及其几何特征

(1) 周长

多边形的周长定义为

$$L = \sum_{i=1}^M l_i = \sum_{i=1}^M \sqrt{(x'_i - x'_{i+1})^2 + (y'_i - y'_{i+1})^2} \quad i = 1, 2, \dots, M \quad (6)$$

其中, l_i 为特征点 v'_i 与特征点 v'_{i+1} 间的欧氏距离, (x'_i, y'_i) 、 (x'_{i+1}, y'_{i+1}) 分别为 v'_i 和 v'_{i+1} 的坐标, 且当 $i=M$ 时, 取 $i+1=1$ 。

(2) 平均极距离

$$\bar{d} = \frac{1}{M} \sum_{i=1}^M d_i = \frac{1}{M} \sum_{i=1}^M \sqrt{x_i'^2 + y_i'^2} \quad (7)$$

其中, d_i 为特征点 v'_i 与形心 C 间的欧氏距离, C 的坐标为 $(0,0)$ 。

(3) 面积

$$A = \sum_{i=1}^M S_{\Delta i} = \sum_{i=1}^M \sqrt{s(s-d_i)(s-d_{i+1})(s-l_i)} \quad i = 1, 2, \dots, M \quad (8)$$

其中 $s = (d_i + d_{i+1} + l_i)/2$, $S_{\Delta i}$ 为由特征点 v'_i 、特征点 v'_{i+1} 和形心 C 这三个点为顶点的三角形的面积, 当 $i=M$ 时取 $i+1=1$ 。

若目标轮廓为一个标准圆, 则归一化后应为一个单位圆。本文将归一化后的目标几何特征与其相应的单位圆的几何特征的比值作为特征参数, 即

(1) 周长比

$$c_1 = L/(2\pi r) = L/(2\pi) \quad (9)$$

(2)距离比

$$c_2 = \bar{d}/r = \bar{d} \quad (10)$$

(3)面积比

$$c_3 = A/(\pi r^2) = A/\pi \quad (11)$$

如果对一个目标提取出的以上 3 个特征参数同时满足式(12)~式(14), 则认为此目标为一个圆形目标。

$$c_1 > T_1 \quad (12)$$

$$c_2 > T_2 \quad (13)$$

$$c_3 > T_3 \quad (14)$$

其中, T_1 、 T_2 、 T_3 分别为 c_1 、 c_2 和 c_3 的门限。

2 实验结果

实验用 1.60GHz Pentium微机, 在Windows XP平台上用 Visual C++ 6.0 编程实现。实验分两组进行: 一组是合成图像; 另一组是真实图像。实验中, $d_{th} = 3$, $T_0 = 0$, $T_1 = 0.93$, $T_2 = 0.91$, $T_3 = 0.8$, 并将本文提出的方法与 RHT法进行对比。由于 RHT采用随机采样, 因此每次运行结果并不完全相同, 本文通过考察 50 次运行结果来评价这两种算法的性能。

第 1 组实验的合成图像原图如图 4(a)所示, 其大小均为 256×256 像素, 其中, 第 1 幅图中有一个完整的圆形目标、一个近似于圆形的椭圆目标、一个边缘残缺的圆形目标和一个外边缘残缺的圆环目标, 第 2 幅图像中除了具有形状类似于第 1 幅图像中的目标外, 添加了一个正方形目标、一个圆角矩形目标和一个与圆相差较大的椭圆形目标。由于合成图像的边缘并不复杂, 因此本文算法和 RHT 法均能将原图中的圆正确检测出来。

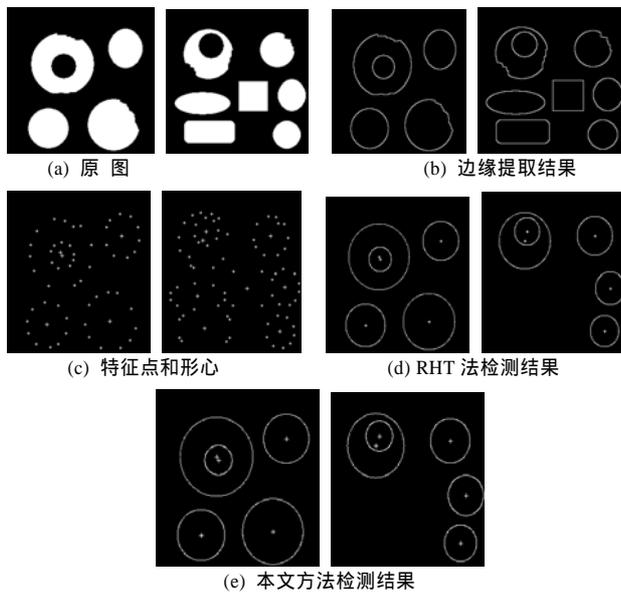


图 4 合成图像及其检测结果

第 2 组实验的真实图像的原图如图 5(a)所示, 其中第 1 幅图像的大小为 308×242 像素, 第 2 幅图像的大小为 256×256 像素, 它们均有 4 个圆形目标。图 5(f)为用 RHT 法检测圆的结果, 在第 1 幅图中正确检测出 4 个圆目标, 但有 8 个虚假目标, 在第 2 幅图像中正确检测出 4 个圆目标, 但有 3 个虚假目标。图 5(f)为用本文方法检测圆的结果, 在两幅图像中均能正确检测出圆目标, 且没有虚假目标。从实验结果可以看出, 对于真实图像而言, 当图像的边缘变得复杂时, RHT 法的虚警率大大上升, 而本文提出的方法仍然能保持较高的检测率且无虚警或较低的虚警率。

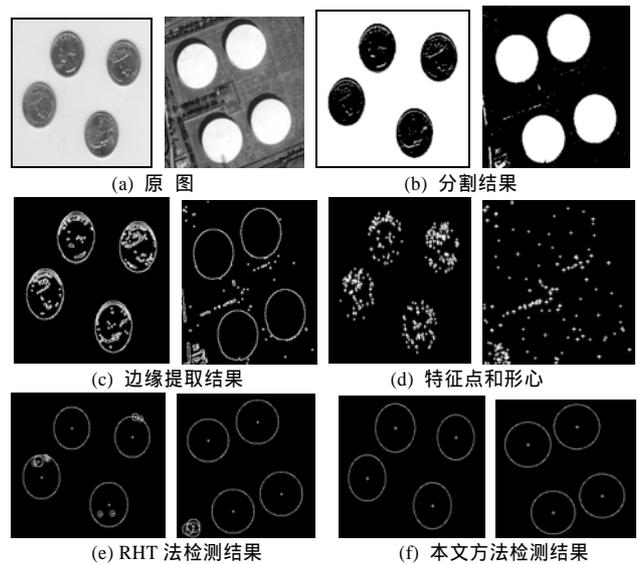


图 5 真实图像及其检测结果

为了比较计算量, 本文不管检测结果正确与否, 对两组实验图像进行 50 次实验, 平均运行时间列于表 1, 其中加“+”者为相应图号的第 2 幅图像。由表 1 可知, 对于图 4 所示的合成图像, 本文提出的算法比 RHT 法的计算速度至少约快 220 倍; 对于图 5 所示的真实图像, 本文提出的算法比 RHT 法的计算速度至少约快 857 倍。

表 1 运算时间对比

图 像	RHT 法 (s)	本文方法 (s)	(RHT法-本文方法)/ 本文方法
图 4 (a)	3.978	0.018	220
图 4 (a)+	15.578	0.021	740.8
图 5 (a)	39.778	0.028	1 419.6
图 5 (a)+	20.598	0.024	857.2

3 结论

本文提出了一种非基于 Hough 变换的圆检测方法, 它具有所需数据量少、内存需求小以及提取的特征具有平移、旋转和尺度变换不变性的特点。对合成图像和真实图像的实验结果表明, 本文方法具有较高的计算效率, 对于局部变形或局部残缺的圆形也能较快的检测, 具有一定的实用价值。

参考文献

- 1 Lei Xu, Oja E, Kultanen P. A New Curve Detection Method: Randomized Hough Transform (RHT)[J]. Pattern Recognition Letters, 1990, 11(5): 331-338.
- 2 Duco tte C. Localization of Objects with Circular Symmetry in a Noisy Image Using Wavelet Transforms and Adapted Correlation[J]. Pattern Recognition, 1994, 27(3): 351-364.
- 3 Ho Chunta, Chen Linghwei. A Fast Ellipse /Circle Detector Using Geometric Symmetry[J]. Pattern Recognition, 1995, 28 (1): 117-124.
- 4 Dave R N. Generalized Fuzzy C-shells Clustering and Detection of Circular and Elliptical Boundaries[J]. Pattern Recognition, 1992, 25 (7): 713-721.
- 5 Man Y, Gath I. Detection and Separation of Ring-shaped Clusters Using Fuzzy Clustering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994, 16 (8): 855-861.
- 6 冈萨雷斯. 数字图像处理[M]. 北京: 电子工业出版社, 2003: 485-486.
- 7 冯春环, 涂建平, 郭 健. 基于离散余弦变换的红外目标识别算法[J]. 系统仿真学报, 2005, 17(6): 1363-1365, 1369.