

改进的棋盘格自生长亚像素角点检测算法

昌海亮^{1,2,3}, 席 隆^{1,2,3}

(1. 中国科学院空间应用工程与技术中心, 北京 100094;

2. 中国科学院大学, 北京 100049;

3. 中国科学院太空应用重点实验室, 北京 100094)

摘 要:针对当前棋盘格内角点检测算法需要提前指定棋盘格尺寸大小以及无法处理一张图片包含多张棋盘格情况的问题,提出了一种改进的棋盘格自生长亚像素角点检测算法。在分析现有算法的基础上,引入棋盘格的能量函数使棋盘格能够自适应生长。该算法的原理是首先定位棋盘格角点位置,接着亚像素化角点,提高精度。最后,优化引入的能量函数,使棋盘格生长到最大化。该算法鲁棒性好,因为棋盘格是自适应生长的,所以如果出现干扰,就会绕过干扰,生长出最大的棋盘格。因为该算法不需要提前指定棋盘格尺寸,可以满足很多自动化应用。相比较 Matlab 集成工具箱中的算法,该算法可以检测一个图片里包含多张棋盘格的情况。实验分析发现,该算法抗干扰强,能在复杂背景中识别出棋盘格角点。同时,在不同倾斜角度下对棋盘格的检测性能也比较良好。

关键词:亚像素角点;棋盘格;自生长;能量函数;角点检测

中图分类号: TP301.6

文献标识码: A

文章编号: 1673-629X(2022)0040-04

An Improved Checkerboard Self-growing Subpixel Corner Detection Algorithm

CHANG Hai-liang^{1,2,3}, XI Long^{1,2,3}

(1. Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing 100094, China;

2. University of Chinese Academy of Sciences, Beijing 100049, China;

3. Key Laboratory of Space Utilization, Chinese Academy of Sciences, Beijing 100094, China)

Abstract: The current checkerboard corner detection algorithm needs to specify the size of checkerboard in advance and cannot deal with the situation that a picture contains multiple checkerboards. An improved checkerboard corner detection algorithm based on self-growing sub-pixel is proposed. On the basis of analyzing the existing algorithms, the energy function of checkerboard is introduced to make the checkerboard grow adaptively. The principle of the algorithm is that Position the checkerboard corners, then subpixelate the corners to improve the accuracy, and finally, optimize the introduced energy function to maximize the growth of the checkerboard. The algorithm is robust because the checkerboard is adaptive, so if there is interference, it will bypass the interference and grow the largest checkerboard. Because this algorithm does not need to specify the checkerboard size in advance, it can satisfy many automation applications. Compared with the algorithm in Matlab integrated toolbox, this algorithm can detect the situation that a picture contains more than one checkerboard. Experimental analysis shows that the algorithm has strong anti-interference and can recognize the corner points of checkerboard in complex background. At the same time, the detection performance of checkerboard under different tilt angles is also good.

Key words: sub-pixel corners; checkerboard; self growing; energy function; corner detection

0 引 言

角点检测是图像视觉领域中的经典研究方向。棋盘格标定板由于制作简单,成本低廉而被广泛应用于摄像机标定过程中^[1],通过检测与提取模板图像中角点的位置坐标,确定图像平面坐标与空间坐标的关系,进而计算摄像机的内外参数。可以看出,角点检测在

摄像机标定中的重要作用^[2]。

目前,针对棋盘格角点检测已有较多方法,根据对角点的描述不同,可以分为三类:基于灰度检测^[3-6]、基于边缘检测^[7-8]、基于模板匹配检测^[9-10]。基于灰度检测的思想是:使用一个固定窗口在图像上进行任意方向上的滑动,比较滑动前和滑动后窗口中的像素

收稿日期: 2021-08-16

作者简介: 昌海亮 (1996-), 男, 硕士研究生, 从事计算机视觉测量相关的研究; 席 隆, 正高级工程师, 从事空间站控制柜、计算机相关的研究。

变化程度。如果存在任意方向上的滑动,都有着较大灰度变化,那么认为该窗口中存在角点^[11]。目前运用该方法比较成熟的有 SUSAN 算法、Harris 算法、FAST 算法^[12-14]等。基于灰度检测的算法的精度依赖于参数设置,并且对图像噪声比较敏感,因此检测结果的鲁棒性不高^[15]。基于边缘检测的思想是:先对图像进行边缘检测,然后通过 Hough 变换等手段求取直线,两条及以上直线交点即是角点^[13]。这种算法简单易行但计算量比较大。基于模板匹配的检测思想是:使用角点特征已知的模版,在目标图像种执行遍历操作,查询并确定角点^[1]。该算法模版设计工作量比较繁重。

此三类棋盘格角点检测算法目前都需要提前指定棋盘格尺寸大小,无法满足自动化应用的场景中,并且无法处理一张图片中包含多张棋盘格情况的问题。为此,在结合前人研究成果的基础上,提出一种改进的棋盘格自生长角点角点检测算法,检测精度达到亚像素级别。

1 算法原理

算法分为三部分,如图 1 所示。首先定位棋盘格角点位置,接着亚像素化角点,使定位更精确,最后优化能量函数,生长棋盘格到最大化。

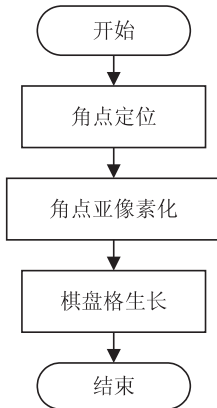


图 1 算法流程

1.1 角点定位

角点定位采用类似 Harris 角点检测的方法,其总的流程如图 2 所示。

首先计算图像 $I(x,y)$ 在 x 和 y 两个方向的梯度 I_x, I_y :

$$I_x = \frac{\partial I}{\partial x} = I \times [-1 \ 0 \ 1] \quad (1)$$

$$I_y = \frac{\partial I}{\partial y} = I \times [-1 \ 0 \ 1]^T \quad (2)$$

接着计算图像两个方向梯度的乘积:

$$I_x^2 = I_x \bullet I_x \quad (3)$$

$$I_y^2 = I_y \bullet I_y \quad (4)$$

$$I_x I_y = I_x \bullet I_y \quad (5)$$

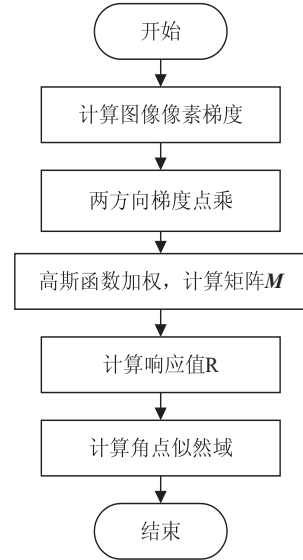


图 2 角点定位流程

之后使用高斯函数对所求的 $I_x^2, I_y^2, I_x I_y$ 进行高斯加权,计算中心点为 (x,y) 的窗口 W 对应的矩阵 M :

$$A = \sum_{(x,y) \in W} g(I_x^2) = \sum_{(x,y) \in W} I_x^2 \bullet w(x,y) \quad (6)$$

$$B = \sum_{(x,y) \in W} g(I_y^2) = \sum_{(x,y) \in W} I_y^2 \bullet w(x,y) \quad (7)$$

$$C = \sum_{(x,y) \in W} g(I_x I_y) = \sum_{(x,y) \in W} I_x I_y \bullet w(x,y) \quad (8)$$

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} = \sum_{(x,y) \in W} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (9)$$

得到 M 的值后,计算每个像素点 (x,y) 处的 Harris 响应值 R :

$$R = \det(M) - K (\text{trace}(M))^2 \quad (10)$$

其中 $\det(M) = \lambda_1 \lambda_2$, $\text{trace}(M) = \lambda_1 + \lambda_2$, k 是常量,一般取值为 0.04 ~ 0.06。

最后过滤大于某一阈值的 R 值得到角点似然域,再进行非极大值抑制即可得到像素精度的角点值。

1.2 角点亚像素化

在前文检测出的像素级坐标的基础上进行进一步分析可以得到亚像素级坐标。对于棋盘格图像角点附近的点分为在边缘上的点和不在边缘上的点两类^[1],如图 3 所示。

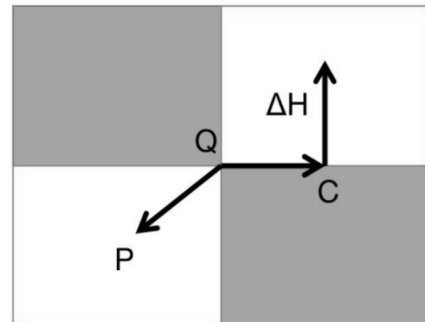


图 3 棋盘格图像

假设点 Q 为真实的角点位置,点 P 和点 C 分别表示角点平坦区域的点和边缘上的点。 Q 点上的梯度方向与 QC 垂直,而 P 点处于平坦区域,所以 P 点处灰度梯度为零。由此可知从中央点到其邻域点的向量和该邻域点处的图像梯度正交,其表达式如下:

$$\nabla H_i^T \bullet (\vec{\alpha} - \vec{\beta}) = 0 \quad (11)$$

其中, ΔH_i^T 为灰度梯度向量, $\vec{\alpha}$ 为图像原点指向 O 点的向量, $\vec{\beta}$ 为图像原点指向第 i 点的向量。

在现实情况下,图像通常会受到噪声点的影响,故上文中的式子(11)左端一般不为零,设误差为 θ ,则有:

$$\nabla H_i^T \bullet (\vec{\alpha} - \vec{\beta}) = \theta \quad (12)$$

在以角点为中心的某一邻域内,对所有的点,按照式(12)计算误差,误差和以 S 表示,则有:

$$S = \sum_i \theta = \sum_i \nabla H_i^T \bullet (\vec{\alpha} - \vec{\beta}) = \theta \quad (13)$$

由此,只要求使误差和 S 最小的点即可。此问题可以使用迭代的方法求解,进而可以对所有的角点进行优化,得到亚像素级坐标。

1.3 棋盘格生长

得到角点之后,需要从角点中重构出棋盘格结构。首先对每一个角点初始化一个以其为中心的,包含 3×3 个角点的棋盘,并定义一个能量函数来描述棋盘的合理性:

$$E(x, y) = E_{\text{corners}}(y) + E_{\text{struct}}(x, y) \quad (14)$$

$$E_{\text{corners}}(y) = -|\{y | y \neq 0\}| \quad (15)$$

$$E_{\text{struct}}(x, y) = |\{y | y \neq 0\}| \max_{(i,j,k) \in \tau} \frac{\|c_i + c_k - 2c_j\|^2}{\|c_i - c_k\|^2} \quad (16)$$

其中, E_{corners} 是当前棋盘角点总数的负值, E_{struct} 描述了用两个相邻角点来预测 3 个角点的匹配程度,分别对棋盘的每行和每列相邻的 3 个角点计算其结构能量,取其中的最大值作为该棋盘的结构能量。对于每一个角点,取其左右或者上下的相邻角点构成一个 triple,如图 4 所示。



图 4 triple

公式中 i 和 k 分别代表两端的角点, j 代表中心的角点,显然,如果三个角点处于一条直线上,则局部线形约束的能量就越小。棋盘生长的过程如图 5 所示。

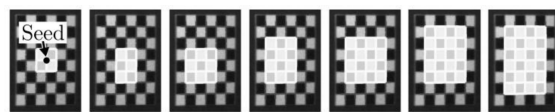


图 5 棋盘格生长过程

给定一个 seed 角点,沿着其边缘方向搜索产生一个初始化的种子棋盘格。该棋盘格有 3×3 个角点,有 2×2 个棋盘。然后以此种子棋盘格为基础,分别从最外沿的 4 个边缘去生长棋盘格,生成 4 个新的棋盘格。如果其中能量最小的那个棋盘格的能量值比棋盘格扩展前的能量更减少了,就说明生长成功,用这个新的棋盘格代替原来棋盘格。继续生长,直到 4 个方向新棋盘格能量不再减少为止。

2 实验结果与分析

图 6 所示是在图像中存在其他干扰物体的复杂背景中的实验结果。可以看出本算法的抗干扰性比较好。



图 6 复杂背景中棋盘格检测

图 7 所示是在不同倾斜方向下的实验结果。可以看出本算法的鲁棒性比较好。

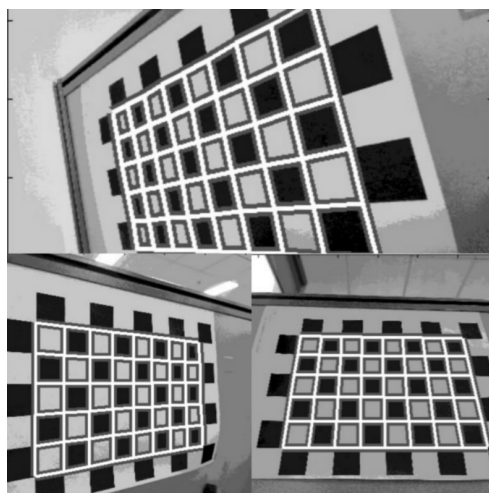


图 7 不同倾斜角度棋盘格检测

如图 8 所示,是在一张图片中包含多张棋盘格情况的几次实验结果,可以看出本算法能同时检测出一张图片中的多张棋盘格。

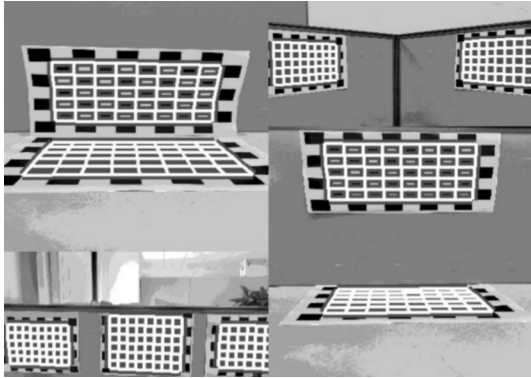


图 8 多张棋盘格检测

3 结束语

首先采用类似 Harris 角点检测的方法对角点进行定位,接着将角点坐标亚像素化,使精度更高,最后通过定义的能量函数使棋盘格自适应生长,直到 4 个方向新棋盘格能量不再减少为止,从而检测出完整的棋盘格。该算法鲁棒性好,抗干扰能力强。因为该算法不需要提前指定棋盘格尺寸,可以满足很多自动化应用。相比较 Matlab 集成工具箱中的算法,该算法可以检测一个图片里包含多张棋盘格的情况。

参考文献:

- [1] 刘飞飞,吴志刚,任舒琪,等.复杂背景下改进的棋盘格内角点检测算法[J].计算机工程与设计,2019,40(12):3474-3478.
- [2] 王晓辉,程健庆,韩瑜.基于 Harris 的棋盘格角点检测改进算法[J].电子测量技术,2013,36(10):51-54.
- [3] 夏海英,刘伟涛,朱勇建.一种改进的快速 SUSAN 棋盘格角点检测算法[J].广西师范大学学报:自然科学版,2018,36(1):44-52.
- [4] 吴鹏,徐洪玲,李雯霖,等.基于区域检测的多尺度 Harris 角点检测算法[J].哈尔滨工程大学学报,2016,37(7):969-973.
- [5] 刘俊杰,谢春利,王娟.棋盘格图像角点坐标亚像素提取方法[J].华中科技大学学报:自然科学版,2015,43(S1):170-172.
- [6] 范新峰,程远增,付强.采用黑白检测算子的棋盘格角点检测[C]//中科院长春光机所、《光学精密工程》编辑部.2015 光学精密工程论坛论文集.中科院长春光机所、《光学精密工程》编辑部,2015:5.
- [7] 龙思源,张葆,宋策,等.基于改进的加速鲁棒特征的目标识别[J].中国光学,2017,10(6):719-725.
- [8] 王海涛,洪亮,谭成国.一种用于相机检校的黑白棋盘格角点提取算法[J].测绘通报,2016(12):33-38.
- [9] 洪磊,嵇保健,凌超.一种棋盘格靶标不完整角点识别的修正算法[J].计算机辅助设计与图形学学报,2016,28(9):1521-1526.
- [10] 柏猛,李敏花,吕英俊.基于对称性分析的棋盘图像角点检测方法[J].信息与控制,2015,44(3):276-283.
- [11] 蔡欣展,刘艳艳.基于自适应筛选 Harris 角点检测的快速图像配准算法[J].半导体光电,2020,41(6):875-878.
- [12] 周伟,巢渊,李龔,等.基于改进 Harris 角点检测的芯片图像快速校正[J].电子测量与仪器学报,2020,34(10):74-81.
- [13] 艾裕丰,赵敏,张琪,等.基于亚像素边缘的棋盘格的角点检测[J].西安理工大学学报,2019,35(3):333-337.
- [14] 刘亮,王平,孙亮.基于区域灰度变化的自适应 FAST 角点检测算法[J].微电子学与计算机,2017,34(3):20-24.
- [15] 杨幸芳,黄玉美,李艳,等.基于改进的 SUSAN 算子的棋盘格亚像素角点检测算法[J].中国机械工程,2010,21(21):2541-2545.
- [16] GEIGER A, MOOSMANN F, CAR Ö, et al. Automatic camera and range sensor calibration using a single shot[C]//IEEE international conference on robotics and automation (ICRA). [s.l.]:IEEE,2012:3936-3943.
- [17] HUANG G, SUN Y, LIU Z, et al. Deep networks with stochastic depth[C]//European conference on computer vision. [s.l.]:Springer,2016:646-661.
- [18] HOWARD A G, ANDREW G. Mobilenets: efficient convolutional neural networks for mobile vision applications[J]. arXiv:1704.04861,2017.
- [19] REDMON, JOSEPH, ALI FARHADI. Yolov3: An incremental improvement[J]. arXiv:1804.02767,2018.

(上接第 39 页)

ty in deep neural networks[J]. Advances in Neural Information Processing Systems,2016,29:2074-2082.

- [15] LI H, KADAV A, DURDANOVIC I, et al. Pruning filters for efficient convnets[J]. arXiv:1608.08710,2016.
- [16] HAN S, MAO H, DALLY W J. Deep compression: compressing deep neural networks with pruning, trained quantization and huffman coding[J]. arXiv:1510.00149,2015.