

文章编号: 2095-2163(2020)02-0023-07

中图分类号: TP391.41

文献标志码: A

基于多特征融合的高鲁棒性视觉 SLAM 改进算法

朱鸣镛, 陈 婵

(上海理工大学 光电信息与计算机工程学院, 上海 200093)

摘要: 特征提取的有效性对视觉 SLAM 的性能具有重要影响。为了提高匹配精度, 本文提出了一种基于改进 oAGAST 算法和旋转 rLATCH 特征(OARL)的二进制字符串描述局部图像的新方法。首先, 在图像金字塔尺度空间上进行自适应通用加速分割(AGAST)检测; 然后, 采用灰度质心方法来进行方向补偿; 最后, 使用可学习的3个图像块的 LATCH 特征用于生成特征点描述符。在求取特征点主方向阶段太依赖像素邻域灰度质心法, 提出一种改进的匹配算法, 采用汉明距离匹配和余弦相似度相结合的方法进行特征点匹配。本文在特征点提取方法以视觉 SLAM 的 ORB 算法和其他算法为对照, 分别进行不同视角、光照和尺度的识别实验。实验结果显示所提出的 OARL 算法较 ORB 算法将精度提高了5%以上, 但速度却依旧可以达到实时运算。

关键词: 特征检测; 二进制描述子; AGAST; 实时性

An improved algorithm of high robust visual SLAM based on multi-feature fusion

ZHU Mingdi, CHEN Chan

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

[Abstract] The effectiveness of the feature extraction has an important influence on the performance of the visual SLAM. This paper proposes a new method for describing local images based on binary strings with Oriented AGAST and Rotated LATCH (OARL) method to improve the accuracy of visual SLAM. Firstly, Adaptive and Generic Corner Detection based on the Accelerated Segment Test(AGAST) is adopted to detect corner feature on the scale space of image pyramid. Then, the intensity centroid method is used to obtain orientation compensation. Finally, the Learned Arrangements of Three Patch Codes is used to describe the feature. An improved algorithm is proposed in which a combined measure of distance similarity matching with cosine similarity matching is adopted to the main direction of the feature point, while it is dependent on the pixel neighborhood's intensity centroid method. In this paper, the feature detection is based on the visual SLAM ORB algorithm and other algorithms, and the different perspective, illumination and scale recognition experiments are carried out. The experimental results show that the proposed OARL algorithm improves the accuracy by more than 5% compared with the ORB algorithm, also the speed can still achieve real-time operation.

[Key words] feature detection; binary descriptor; AGAST; real-time property

0 引言

同时定位与地图构建(SLAM)是计算机视觉和机器人领域的主流研究方法。这是指搭载有特定传感器的主体, 在没有环境先验信息的情况下, 于运动过程中建立环境的模型, 同时估计自己的运动。将相机作为唯一外部传感器的 SLAM 称为视觉 SLAM。由于相机价格低, 重量轻, 易于装配在其他硬件上, 并且图像包含丰富的信息, 近年来视觉 SLAM 技术取得了很大进展^[1]。

根据图像信息如何用于位姿估计, 视觉 SLAM 可以分为直接方法: 基于特征的方法和混合方法。本文主要讨论基于特征的 SLAM 方法。图像基本上是亮度和颜色的矩阵。直接从矩阵水平考虑姿态估

计是非常困难的。因此, 可以从称为特征的图像中选择更多角点, 这些角点在摄像机角度稍有变化后保持不变。因此, 可以在每个图像中找到相同的功能。然后就可以基于这些特征讨论姿势估计问题。每个特征点都有一个描述符, 可以定量测量与其他特征的相似性。

为了从图像中提取角点, Harris 等人^[2-3]改进并提出了基于 Moravec 角点检测器的 Harris 角点检测算法。并且 David^[4]提出的 SIFT 角点检测算法和 Bay 等人^[5]提出的 SURF 角点检测算法已经证明在许多应用中特别成功。一方面, 视觉 SLAM 中的特征点应该是独特的, 对于视点和光照变化是不变的, 同时对模糊和噪声具有弹性; 另一方面, 检测算法应

基金项目: 国家自然科学基金(61873169)。

作者简介: 朱鸣镛(1994-), 男, 硕士研究生, 主要研究方向: 基于单目相机的视觉 SLAM 图像匹配算法研究; 陈 婵(1995-), 女, 硕士研究生, 主要研究方向: 基于双目相机的视觉 SLAM 图像匹配算法研究。

收稿日期: 2019-10-11

该是计算上有效且快速的。然而, Harris、SIFT 和 SURF 计算成本都很高, 很难在视觉 SLAM 中实现实时速度。随着图像处理技术的发展, Rosten 等人^[6]提出了 FAST 特征检测算法, 大大提高了速度, 并应用于 PTAM^[7]等一些实时 SLAM 应用。Rublee 等人^[8]提出的 ORB 特征检测算法将灰度质心法应用于 FAST, 使其具有旋转不变性。Mur-Artal 等人^[9-10]在 2015 年提出的 ORB-SLAM 使用 ORB 提取特征, 这是目前最先进的视觉 SLAM 算法。然而, 越来越多的项目正在将 SLAM 应用于智能手机和嵌入式设备, 因此提高 SLAM 的速度仍然是一个亟待研究的重要问题^[11]。

本文提出了一种改进版的特征点提取算法, 能够很好地应用于视觉 SLAM 当中。在特征点检测中, 使用多尺度 AGAST 角点检测^[12]算法找出具有尺度不变性的兴趣点。然后, 结合重心法, 改进的 AGAST 算法可以获得很强的鲁棒性。在特征描述

中, 提取特征点的 LATCH 特征, 并且这两个特征组合形成新的二进制 AGAST-LATCH 特征检测算法。

1 OARL 算法

实时性能是视觉 SLAM 的重要前提, 因此提高算法的速度始终是一项重要的研究。ORB 算法本质上是 FAST 和 BRIEF 的组合。因为 Mair 等人^[12]提出的 AGAST 算法提升了 FAST 算法的速度与在光照变换下的鲁棒性, 但并未改善其对尺度变化与旋转变换下鲁棒性不高的缺陷。因此在本节中, 研究提出了一种更好的特征提取过程, OARL 算法 (Oriented AGAST and Rotated LATCH), 就是将 AGAST 算法与 BRIEF 算法相结合。OARL 算法对 AGAST 兴趣点构建尺度空间并加入灰度质心法, 使得 AGAST 算法对旋转变换与尺度变化有着良好的鲁棒性。OARL 加快角点的提取, 同时确保算法的性能。本文的方法是在灰度图像上进行的。图 1 显示了本次研究提出的 OARB 特征提取过程。

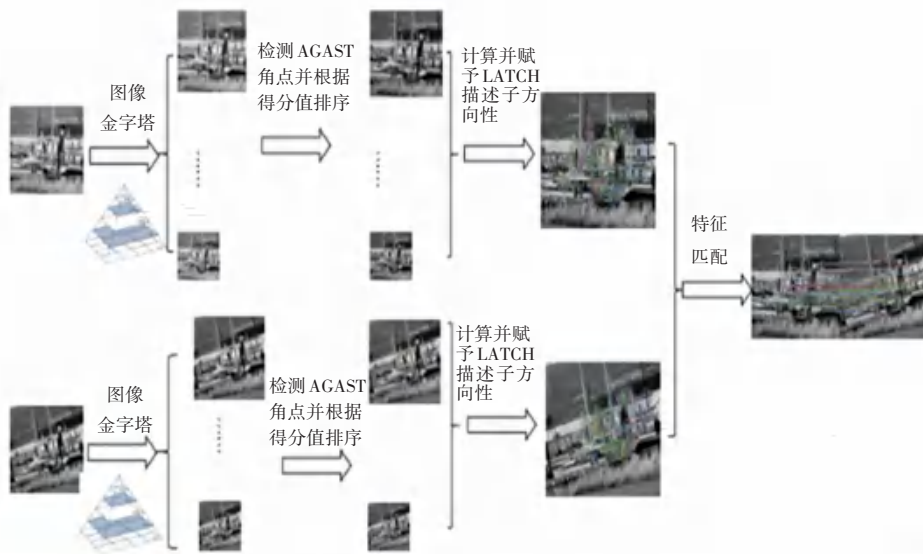


图 1 OARL 特征提取过程概述

Fig. 1 An overview of the OARL feature extraction process

1.1 AGAST 算法

FAST 算法是在视觉 SLAM 和其他实时系统中查找特征点的首选方法。AGAST 特征检测算法对 FAST 进行了改进, 使得其相比 FAST 来说, 运算速度更快, 耗时更少, 对复杂场景图片有更好的表现。该算法检测角点所用的判据与 FAST 一致, 如图 2 所示, FAST 算法将像素“ p ”与 Bresenham 圆上的 16 个像素进行比较, 其周围半径为 3。当存在亮度高于或低于中心点的连续 N 个像素时, 像素“ p ”被判断为角点。与 FAST 不同的是, AGAST 提供了一种更详细的配置空间, 采用“非较亮”与“非较暗”对原配

置空间进行扩展。本文采用了以下概念: $S_{n \rightarrow x}$ 表示为 $n \rightarrow x$ 的像素对于核 n 的状态, 分配如式 (1) 所示:



图 2 中心像素点 n 及其周围 16 个像素点 p

Fig. 2 Center pixel point n and 16 pixel points p around it

$$S_{n \rightarrow x} = \begin{cases} \uparrow d, & I_{n \rightarrow x} < I_n - t & (\text{较暗}); \\ \downarrow d, & I_{n \rightarrow x} \geq I_n - t \cap S'_{n \rightarrow x} = u & (\text{非较暗}); \\ \downarrow s, & I_{n \rightarrow x} \geq I_n - t \cap S'_{n \rightarrow x} = \bar{b} & (\text{相同}); \\ \downarrow s, & I_{n \rightarrow x} \leq I_n + t \cap S'_{n \rightarrow x} = \bar{d} & (\text{相同}); \\ \downarrow b, & I_{n \rightarrow x} \leq I_n + t \cap S'_{n \rightarrow x} = u & (\text{非较亮}); \\ \uparrow b, & I_{n \rightarrow x} > I_n - t & (\text{较亮}). \end{cases} \quad (1)$$

其中, $S'_{n \rightarrow x}$ 是前一个状态; I 是该像素的亮度; u 意味着状态仍然未知。该结果使得二叉树与三元树相比时能在每个节点上单独评估。此外配置空间大小也会增加到 $6N$, 而 $616 \approx 2 \times 1012$, 所以将可能的节点 N 设置为 16。阈值 t 越大, 检测到的角点越准确, 计算越快, 但角点数越少; 阈值 t 越小, 检测到的角点越多, 但计算越慢。

对于一幅待检测图像来说, 通常都具有表示均匀表面的同质化区域和(或)杂乱区域或具有纹理的结构化区域。所以, 不是从训练图像(如 FAST)学习像素配置的分布, 而是首先概括学习结构化和同质区域的概率并根据该分布优化决策树。图像均匀的概率可以通过像素状态与核(P_s)相似的概率来建模。“更亮”和“更暗”状态是镜像状态, 这意味着, 例如, 测试图案上的更亮像素将在其变为中心像素时将当前核像素评估为更暗。由于这种镜像, 状态“更亮”和“更暗”被假定具有相同的概率(P_{bd}), 选择其与 P_s ($P_s + 2P_{bd} = 1$) 总和为 1。因此, 像素配置 p_X 的概率可用如下公式进行计算:

$$p_X = \prod_{i=1}^N p_i \text{ with } p_i = \begin{cases} \uparrow 1, & \text{for } S_{n \rightarrow i} = u; \\ \downarrow p_s, & \text{for } S_{n \rightarrow i} = s; \\ \downarrow p_{bd}, & \text{for } S_{n \rightarrow i} = d \vee S_{n \rightarrow i} = b; \\ \downarrow p_{bd} + p_s, & \text{for } S_{n \rightarrow i} = \bar{d} \vee S_{n \rightarrow i} = \bar{b}. \end{cases} \quad (2)$$

根据式(2)的概率分布设定, 利用特定场景下的训练集来确定概率数值, 然后通过逆向归纳法构造出如图3所示的最优二叉决策树, 在进行决策树切换时, AGAST 采用了一种自适应的解决方案。先建立 2 棵树, 其中一个专门用于同质化, 另一个用于对图像进行结构化; 在每个决策树的末端, 对其执行基于该叶的像素配置的适当的专用树的跳转, 如图2所示。当像素的邻像素发生变化时, AGAST 会在 2 个专用树间进行切换。叶节点的灰度越浅, 则在

其配置中的像素越平等。AGAST 算法在生成专用树时, 离线完成的对叶节点的评估使得其在专用决策树之间的切换没有额外的成本。另外, 左树实现了更少的像素评估, 而右树则可以优化纹理区域。同时 AGT 在任意场景中的适应性能也得到了改善, 无须进行额外的训练。

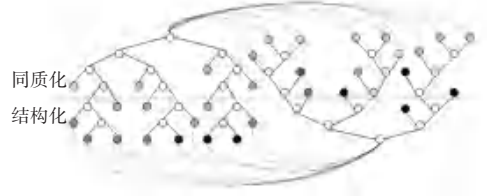


图3 AGAST 算法示意图

Fig. 3 Schematic of the adaptive and generic accelerated segment test

1.2 改进后的 AGAST 算法

1.2.1 尺度不变性 AGAST

在 SLAM 中, 当相机前后移动时捕获的相同对象将具有不同的比例, 因此比例不变性对于特征检测非常重要。由于 AGAST 特征检测算法不具有尺度不变性, 研究中设置了比例因子 F (默认取 1.2) 并构建 L (默认取 8) 层图像金字塔, 在此基础上提取角点和计算不同尺度图像上的描述符。这使得角点具有比例不变性。然后将 AGAST9-16 (圆周上共有 16 个像素, 阈值为 9) 算子应用于尺度空间每一层, 再记录下候选点所在尺度空间位置, 最终求出候选点及其 AGAST 分数 V 。

1.2.2 旋转不变性

AGAST 不包含特征点的方向信息, 因此难以构造旋转不变特征描述。在本节中, 研究添加了一个有效计算的方向方法。本节的方法使用灰度质心^[2]方法来测量角点的方向。与普通方向参数算法相比, 强度质心对随机噪声具有更高的鲁棒性。

本文采用灰度质心法来计算角点方向。首先定义一个图像块 B 的灰度矩为 M_{pq} , 数学计算公式为:

$$m_{pq} = \sum_{x,y \in B} x^p y^q I(x,y), \quad p, q = \{0, 1\}, \quad (3)$$

其中, $I(x,y)$ 表示此坐标处的灰度值。本次研究中采用矩来计算图像块 B 的质心 C , 由其运算求得从角中心 O 到质心 C 的矢量 OC , 这样就可以得到角点的方向 θ , 如图4所示。研究中给出的数学定义如下:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) = \tan^{-1} \frac{m_{10}}{m_{01}}. \quad (4)$$

改进前后的算法运行效果如图5所示。由图5

可看出,加入图像金字塔和灰度质心法的 AGAST 角点面对具有尺度不变性和旋转不变性的数据集具有了更好的鲁棒性,准确率更高。

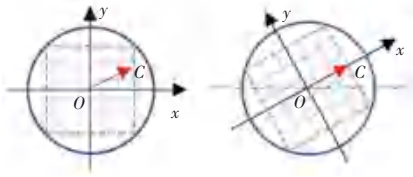


图4 图像块的质心 C 和点的方向 θ

Fig. 4 The centroid of the image block is C and the direction of the point is θ



(a) 改进前的 AGAST 算法

(a) The original AGAST algorithm



(b) 改进后的 AGAST 算法

(b) The improved AGAST algorithm

图5 改进前/后的 AGAST 算法

Fig. 5 The original/improved AGAST algorithm

1.3 LATCH 算法

在先前的二进制描述符的设计中,设检测到的图像关键点为中心选取检测窗口 W , W 是固定的预定尺寸的图像块大小,一个二进制描述子 b_w 由 T 对抽样坐标序列 $S = \{s_i\}_{i=1,2,\dots,T} = \{[p_{i,1}, p_{i,2}]\}_{i=1,2,\dots,T}$ 组成,其中 $p_{i,1} = (x_{i,1}, y_{i,1})$ 和 $p_{i,2} = (x_{i,1}, y_{i,2})$ 定义在 W 坐标系,索引 i 既与 W 中的一对坐标关联,又与高斯核 $\sigma_i = (\sigma_{i,1}, \sigma_{i,2})_{i=1,2,\dots,T}$ 相关联。对于每一抽样对 s_i ,比较 $p_{i,1}$ 和 $p_{i,2}$ 经过光滑后的灰度,从而由式(5)来设置二进制中相应位的值,即:

$$f(W, s_i, \sigma_i) = \begin{cases} 1, & \text{if } W(p_{i,1}, \sigma_{i,1}) > W(p_{i,2}, \sigma_{i,2}); \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

其中, $W(p_{i,1}, \sigma_{i,1})$ 和 $W(p_{i,2}, \sigma_{i,2})$ 是图像块 W 中坐标 $p_{i,1}$ 和 $p_{i,2}$ 经标准差 $\sigma_{i,1}, \sigma_{i,2}$ 高斯滤波后的值。最终的二进制串 b_w 由式(6)来定义:

$$b_w = \sum_{1 \leq i \leq T} 2^i f(W, s_i, \sigma_i), \quad (6)$$

这种对像素直接比对方法易受噪声和局部表观变化的影响。进行预高斯光滑可以在一定程度上使其得到缓解,但是这也意味着存在特征较多的高频区域信息的缺失。LATCH 引进了三元组像素块的比对方法,从而确定二进制中相应位的取值。考虑 $t = 1, 2, \dots, T$ 个三元组序列, S 转化为 $\hat{S} = \{\hat{s}_i\}_{i=1,2,\dots,T} = \{[p_{i,a}, p_{i,1}, p_{i,2}]\}_{i=1,2,\dots,T}$, $p_{i,a}, p_{i,1}$ 以及 $p_{i,2}$ 给出了大小为 $k * k$ 像素块中心像素的定位。将 $p_{i,a}$ 定义为主像素块, $p_{i,1}$ 和 $p_{i,2}$ 是其伴随像素块,通过计算主像素块和伴随像素块之间的 F 范数来评估相似程度,由式(7)确定二进制串中对应位的值。具体公式为:

$$g(W, \hat{s}_i) = \begin{cases} 1, & \text{if } \|P_{i,a} - P_{i,1}\|_F^2 > \|P_{i,a} - P_{i,2}\|_F^2; \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

将 g 替换式(6)中函数 f 就确定了基于三元组方法的二进制串 b_w 。

LATCH 算法描述子形成过程如图6所示。在运行时间上, LATCH 描述子保持了二进制描述子的优势,比基于直方图描述子快好几个数量级,在鲁棒性方面, LATCH 在大多数数据集上的效果优于其他二进制描述子,缩小了与基于直方图描述子的差距^[13]。



图6 LATCH 算法描述子形成过程

Fig. 6 LATCH algorithm describes the sub-forming process

1.4 特征匹配

特征点匹配方法采用了汉明距离匹配法,基本原理是计算2个等长的字符串中对应位置的不同字符的个数。判断2个向量相似程度一般有2种方式,即:距离测度法和相似性函数法。本文在汉明距离测度的基础上再用余弦相似度作为约束,通过设定相似性的函数的阈值来去除多余的匹配点对。

1.4.1 匹配算法的改进

OARL 算法匹配过程中,如果匹配图像局部点领域信息相近和图像视角不同,会引起2个不同特征点描述符的匹配程度超过同一点的特征描述符匹

配程度。因此在匹配的 2 幅图像中如果出现形状相似区域, 就会产生大量误匹配。本文使用余弦相似度进行二次匹配, 对误匹配对进行消除。

1.4.2 向量空间余弦相似度匹配

判断 2 个向量相似程度的准则一般有 2 种, 分别是: 距离测度法和相似性函数法。其中, 距离测度法是根据向量空间上存在的距离来判断向量间的差异程度。相似性函数是用函数值的大小来表明两向量间的差异程度。本文在汉明距离测度的基础上再用余弦相似度作为约束, 通过设定相似性函数的阈值来去除多余的匹配点对。对该方法过程可阐释如下。

先使用汉明距离选取初步特征点对, 然后再使用余弦相似度函数进一步筛选, 如果 2 个向量的余弦值大于阈值 K 则保留, 反之删除。 K 可以根据实验得到, 对于 2 个向量 \mathbf{x} 和 \mathbf{y} , 余弦相似度 $S(\mathbf{x}, \mathbf{y})$ 可用式 (8) 进行计算:

$$S(\mathbf{x}, \mathbf{y}) = \frac{(\mathbf{x}, \mathbf{y})}{\|\mathbf{x}\| \|\mathbf{y}\|} = \frac{\sum_{i=1}^n x_i y_i}{(\sum_{i=1}^n x_i^2 \sum_{i=1}^n y_i^2)^{1/2}}. \quad (8)$$

使用余弦相似度对产生缩放和旋转、亮度对比度改变、模糊、视角变化的 4 类图像进行测试, 选择平均阈值 $K = 0.977$, 测试结果见表 1。

表 1 测试结果

Tab. 1 Test results

| 测试图片 | 旋转和缩放 | 亮度 | 模糊 | 视角 |
|------|-------|------|------|------|
| 最佳阈值 | 0.98 | 0.95 | 0.99 | 0.99 |

2 算法流程设计

本文提出的 OARL 算法设计流程如图 7 所示, 以实现检测和描述的优势互补以及快速性和鲁棒性的有效结合。

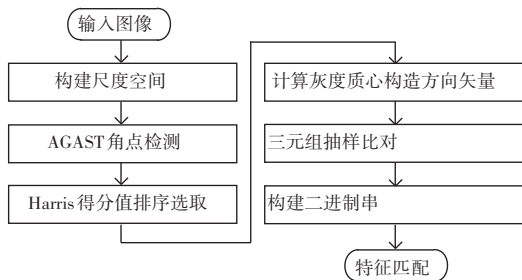


图 7 OARL 算法流程图

Fig. 7 OARL algorithm flow chart

在多尺度 AGAST 角点检测上, 对输入图像进行图像金字塔尺度空间的构建, 而后对每一图层进行 AGAST 角点检测和尺度评判与特征的选取。在二

进制描述阶段, 首先采用灰度质心方法对特征周围固定大小的图像块进行方向补偿, 而后对旋转后的图像块进行抽样和三元组的选取比对, 从而为特征点形成二进制描述子, 为后续的特征匹配做好基础准备工作。

3 实验环境及分析

研究用 Mikolajczyk 数据集^[14]作为测试用图, 以正确匹配率作为评价指标来评估 OARL 算法, 实验测试设备是 ubuntu16.04 操作系统的台式电脑, 在其上进行了所有实验, CPU3.5 GHz, 8 GB 内存, 采用 Microsoft Visual Studio code2018 编程实现, 本文使用 2 个性能指标来评价算法: 计算时间, 重复率^[15]。其中, 重复率中的距离阈值设定 $\epsilon = 1.5$, 映射到另一幅图像中重叠误差阈值设定为小于 0.2。

3.1 平均计算时间

这里, 将研究中的 OARL 算法与传统的 SIFT、SURF、ORB 算法各自选择 2 000 特征点进行比较, 结果见表 2, 可以看出本文的 OARL 算法与 ORB 算法的运算时间是同一数量级的, 稍快一些, 仍然保持实时性, 比 SIFT、SURF 的运算速度低好几个数量级。

表 2 各算法运算在不同数据集上的时间

Tab. 2 The time each algorithm operates on a different data set

| 图像集 | SIFT | SURF | ORB | OARL |
|----------|------------|---------|----------|--------|
| Bikes | 701.491 | 450.160 | 47.301 4 | 46.628 |
| Leuven | 554.412... | 330.921 | 45.456 3 | 44.635 |
| Graffiti | 538.188 | 329.128 | 46.000 0 | 45.322 |
| Boat | 635.606 | 340.560 | 63.677 2 | 62.073 |

3.2 匹配精度

首先, 研究在 Mikolajczyk 的数据集中选择了 4 组图像, 从图像模糊、不同光照、不同比例和不同视角等方面评估算法的性能。仿真在 4 组测试图片中的每一组上执行 OARL 原始方法 50 次。图 8、图 9 和表 3 显示了结果。表 3 中, 第三列和第四列显示从左侧和右侧的图像中提取的特征点的数量。第五列显示检测时间。最后一列显示了成功匹配的数量。可以看出, 在模糊、亮度变化、视点变化、变焦和旋转的情况下, 本文的 ORAL 算法平均匹配时间与 ORB 算法为相同数量级, 但是正确率却平均提升了 5% 左右。本文的 OARL 算法在不同环境变化下, 准确率均高于 ORB 算法, 这主要是因为, OARL 算法的三元组 F 范数有着良好的抗干扰性和应对局部表观变化的能力, 并且具有旋转机制, 从而使得改进算法具备较好的尺度和旋转不变性, 缩小了与传统的 SIFT 和 SURF 算法之间在鲁棒性的差距。



图8 OARL算法在Mikolajczyk数据集上的效果

Fig. 8 Effect of OARL algorithm on Mikolajczyk dataset



图9 ORB算法在Mikolajczyk数据集上的效果

Fig. 9 Effect of ORB algorithm on Mikolajczyk dataset

从图8、图9可以看出,与ORB算法相比,本文的OARL算法比ORB算法匹配要更加整齐,这也显示出匹配精度的优劣,表3的结果也显示,在相同检测角点情况下,匹配的正确率高于主流的ORB算法和传统的直方图的特征点描述法,速度却可以达到实时检测。

表3 各种算法在不同情况下重复率

Tab. 3 Repeatability of various algorithms in different situations

| 图像集 | 算法 | 同一图像 图1中检测 出的点数 | 同一图像 图2中检测 出的点数 | 匹配对的 数量 | 重复率/ % |
|----------|------|-----------------------|-----------------------|------------|-----------|
| Bikes | SIFT | 2 000 | 1 728 | 600 | 66.605 2 |
| | SURF | 2 270 | 1 593 | 552 | 70.069 4 |
| | ORB | 2 000 | 2 000 | 497 | 75.656 8 |
| | OARL | 2 000 | 2 000 | 490 | 76.161 8 |
| Leuven | SIFT | 2 000 | 2 000 | 589 | 76.312 1 |
| | SURF | 2 110 | 2 012 | 563 | 74.530 0 |
| | ORB | 2 000 | 2 000 | 443 | 75.694 7 |
| | OARL | 2 000 | 1 994 | 450 | 77.111 9 |
| Graffiti | SIFT | 2 000 | 2 000 | 660 | 60.377 4 |
| | SURF | 2 186 | 1 871 | 650 | 53.674 1 |
| | ORB | 2 000 | 2 000 | 399 | 67.261 9 |
| | OARL | 2 000 | 2 000 | 424 | 70.000 0 |
| Boat | SIFT | 2 000 | 2 000 | 599 | 73.533 5 |
| | SURF | 2 101 | 1 980 | 580 | 59.074 1 |
| | ORB | 2 000 | 2 000 | 300 | 72.155 3 |
| | OARL | 2 000 | 2 000 | 357 | 75.445 6 |

4 结束语

本文提出了一种新的特征提取过程,称为OARL算法,可为图像构建一个尺度金字塔,并检测每个金字塔层中的AGAST角点,再测量每个角点的方向。图像集和序列的实验结果表明,与OpenCV中使用的ORB算法相比,所提出的OARL算法将精度提高了5%以上。通过实验可以得知,本文提出的OARL算法同时在面对旋转、尺度变化以及亮度变化的情况下都具有良好的鲁棒性,且精确度优于ORB算法,但速度却依旧可以达到实时运算。

参考文献

- [1] CADENA C, CARLONE L, CARRILLO H, et al. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age[J]. IEEE Transactions on Robotics, 2016, 32(6): 1309.
- [2] HARRIS C, STEPHENS M. A combined corner and edge detector [C]//Proceedings of the 4th Alvey Vision Conference. Manchester: University of Manchester, 1988: 147.
- [3] MORAVEC H P. Obstacle avoidance and navigation in the real world by a seeing robot rover[D]. Stanford: Stanford University, 1980.
- [4] DAVID G L. Distinctive image features from scale-invariant keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91.
- [5] BAY H, ESS A, TUYTELAARS T, et al. SURF: Speeded up robust features [J]. Computer Vision and Image Understanding, 2008, 110(3): 346.
- [6] ROSTEN E, DRUMMOND T. Fusing points and lines for high performance tracking [C]//Tenth IEEE International Conference on Computer Vision (ICCV 2005). Beijing, China: IEEE, 2005: 1508.
- [7] KLEIN G, MURRAY D. Parallel tracking and mapping for small AR workspaces [C]//6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2007). Nara, Japan: IEEE, 2007: 225.
- [8] RUBLEE E, RABAU D, KONOLIGE K, et al. ORB: An efficient alternative to SIFT or SURF [C] // 2011 IEEE International Conference on Computer Vision (ICCV). Barcelona, Spain: IEEE, 2011: 2564.
- [9] MUR-ARTAL R, MONTIEL J M M, TARDOS J D. ORB-SLAM: A versatile and accurate monocular SLAM system [J]. IEEE Transactions on Robotics, 2015, 31(5): 1147.
- [10] MUR-ARTAL R, TARDOS J D. ORB-SLAM2: An open-source slam system for monocular, stereo, and rgb-d cameras [J]. IEEE Transactions on Robotics, 2017, 33(5): 1255.
- [11] YOUNES G, ASMAR D, SHAMMAS E, et al. Keyframe-based monocular SLAM: Design, survey, and future directions [J]. Robotics and Autonomous Systems, 2017, 98: 67.
- [12] MAIR E, HAGER G D, BURSCHKA D, et al. Adaptive and generic corner detection based on the accelerated segment test [C]//Proc. of 11th European Conference on Computer Vision. Crete, Greece: Springer, 2010: 183.

(下转第33页)