

文章编号: 1001-0920(2010)02-0251-04

基于单目视觉的机器人 Monte Carlo 自定位方法

刘洞波^{1,2}, 刘国荣^{1,2}, 喻妙华²

(1. 湖南大学 电气与信息工程学院, 长沙 410082; 2. 湖南工程学院 计算机与通信学院, 湖南 湘潭 411104)

摘要: 针对单目视觉机器人定位问题, 提出一种基于改进的尺度不变特征变换(SIFT)的 Monte Carlo 自定位方法. 应用改进的 SIFT 方法提取特征, 既能保证对图像光强变化、尺度缩放、三维视角和噪声具有不变性, 又能减少 SIFT 算法产生的特征点及其抽取和匹配的时间. 在机器人移动过程中, 环境特征点的观测信息和里程计信息通过粒子滤波相融合, 获得了更准确的环境标志点坐标. 仿真实验结果验证了该方法的有效性.

关键词: 移动机器人; Monte Carlo 定位; 单目视觉; 尺度不变特征变换

中图分类号: TP24

文献标识码: A

Monocular vision based Monte Carlo localization for mobile robot

LIU Dong-bo^{1,2}, LIU Guo-rong^{1,2}, YU Miao-hua²

(1. College of Electrical and Information Engineering, Hu'nan University, Changsha 410082, China; 2. Department of Computer and Communication, Hu'nan Institute of Engineering, Xiangtan 411104, China. Correspondent: LIU Dong-bo, E-mail: ldbymh@163.com)

Abstract: To deal with the localization problem of robot equipped with monocular camera, a Monte Carlo method based on scale invariant feature transform (SIFT) is proposed. The features are extracted by modified SIFT to make the features invariant to changes in illumination, scale, 3D viewpoint and noise, and to reduce the number of features generated by SIFT as well as their extraction and matching time. During robot motion, the information from feature observations is fused with that from the odometry by particle filter, so more exact coordinates of the features are gotten. Experimental results show the effectiveness of the approach.

Key words: Mobile robot; Monte Carlo localization; Monocular vision; Scale invariant feature transform

1 引言

移动机器人定位是机器人最基本、最重要的功能之一. 视觉传感器能感知丰富的环境信息, 因此基于视觉传感器的机器人定位受到了普遍关注. 对于不允许人为改变的环境, 特征提取方法十分重要, 它决定了定位方法.

本文采用改进的尺度不变特征变换(SIFT)算法^[1]提取特征点. 为了更好地适应实时性要求高的场合, 提出一种迭代的 SIFT 算法, 以减少特征抽取和匹配过程中的计算量, 并保持定位的精确性. 基于视觉的机器人定位时, 采集的图像不仅与当前位置的图像匹配, 而且会与类似位置的场景图像匹配, 从而造成定位混淆. 为了解决这一问题, 本文进一步提出一种基于单目视觉的 Monte Carlo 定位方法, 将机器人定位问题转化为状态空间中机器人位置概率

分布的计算和更新. 仿真实验验证了本文方法能以较低的计算开销、较高的准确率实现全局定位.

2 SIFT 算法

2.1 图像尺度空间的生成

尺度空间理论最早出现在计算机视觉中, 是为了模拟图像数据的多尺度特征. 一幅二维图像的尺度空间^[2]定义为

$$L(x, y, \sigma) = G(x, y, \sigma)I(x, y), \quad (1)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \quad (2)$$

其中: $G(x, y, \sigma)$ 为尺度可变的高斯函数, (x, y) 为空间坐标, σ 为尺度空间因子, L 为尺度空间. 为了有效地在尺度空间检测到稳定的特征点, 采用高斯差分尺度空间检测局部极值来作为特征点, 即 DoG (Difference of Gaussian) 金字塔. DoG 算子如下:

$$D(x, y, \sigma) =$$

收稿日期: 2009-02-16; 修回日期: 2009-04-22.

基金项目: 湖南省教育厅科研基金项目(08C224); 湖南省自然科学基金项目(05JJ40093).

作者简介: 刘洞波(1974—), 男, 湖南宁乡人, 讲师, 博士生, 从事移动机器人定位和导航等研究; 刘国荣(1957—), 男, 湖南华容人, 教授, 博士生导师, 从事智能控制、不确定系统控制等研究.

$$(G(x, y, k\sigma) - G(x, y, \sigma))I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma). \quad (3)$$

2.2 空间极值点检测

为了寻找尺度空间的极值点, 每个采样点要与所有的相邻点比较, 看是否比它的图像域和尺度域的相邻点大或小. 检测点与它同尺度的 8 个相邻点及上下相邻尺度对应的 9×2 个点(共 26 个点)比较, 以确保在尺度空间和二维图像空间都检测到局部极值.

2.3 特征点方向分配

利用特征点邻域像素的梯度方向分布特性, 为每个特征点指定方向参数, 使算子具备旋转不变性. (x, y) 处梯度的模值和方向可表示为

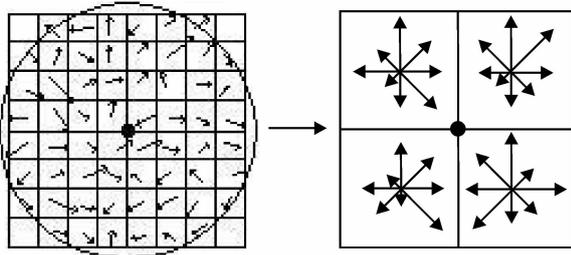
$$m(x, y) = ((L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2)^{1/2}, \quad (4)$$

$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y))). \quad (5)$$

其中 L 所用的尺度为每个特征点各自所在的尺度. 实际计算时, 在以特征点为中心的邻域窗口内采样, 并用直方图统计邻域像素的梯度方向. 直方图的峰值代表了该特征点处邻域梯度的主方向, 即作为该特征点的方向.

2.4 特征点描述子生成

首先将坐标轴旋转为特征点的方向, 以确保旋转不变性. 以特征点为中心取 8×8 的窗口.



(a) 邻域梯度方向

(b) 关键点特征向量

图 1 特征点邻域梯度方向生成特征向量

图 1(a) 的中央黑点为当前特征点的位置, 每个小格代表特征点邻域所在尺度空间的一个像素, 箭头方向代表该像素的梯度方向, 箭头长度代表梯度模值. 在每 4×4 的小块上计算 8 个方向的梯度方向直方图, 绘制每个梯度方向的累加值, 即可形成一个 8 维向量, 如图 1(b) 所示.

Lowe^[1] 在实际应用中采用 4×4 个小窗口, 这样每个特征点就用 128 维向量来表征. 最后对特征向量规一化, 去除光照影响.

3 迭代 SIFT

在 SIFT 算法中, 通过每个尺度空间的极值检

测来确定特征点位置和所在尺度. 有些特征点对机器人的定位并不重要, 本文将这种详尽的搜索改进成基于采样的迭代 SIFT(简称 ISIFT). ISIFT 的主要目标是在不改变每个特征点描述符的同时, 减少特征点数目, 从而减少特征提取和匹配的时间. ISIFT 预先定义特征点的数目, 并不搜索整个尺度空间, 而是进行如下迭代过程: 首先, 随机搜查某个尺度空间, 获得该尺度的局部极值点; 其次, 若搜索到更有可能的局部极值点, 则进行更新.

假设局部极值点位于 blob 区域^[3-5] 内, 提出搜索局部极值点的机制: 当检测到 blob 区域时, 对该区域的局部极值点进行检测; 当没有检测到 blob 区域时, 忽略该区域搜索的结果, 在其他区域开始新的搜索. 判定某一点 P 是否位于一个 blob 区, 可采用如下公式:

$$\text{isBlob} = \begin{cases} \text{True}, & | \text{DoG}(p) | > T_{\text{Scale}}; \\ \text{False}, & \text{otherwise}; \end{cases} \quad (6)$$

其中 T_{Scale} 是一个与尺度相关的阈值.

ISIFT 算法描述如下:

Definitions

N_{Keys} // 要求的特征点数目

N_{Samples} // 样本数且 $N_{\text{Samples}} > N_{\text{Keys}}$

N_{trials} // 迭代次数

N_{scales} // 尺度图像数目

Initialization: Keypoints $\leftarrow \{ \}$

For (Scale = 1, Scale $\leq N_{\text{scales}}$, Scale++) {

$(x_i, y_i) = \text{RandomCoord}()$

isBlob ($p(x_i, y_i)$) is True then $\forall i = 1, 2, \dots, N_{\text{Samples}}$;

$i = 0$

while ($i < N_{\text{Samples}}$ and $| \text{Keypoints} | < N_{\text{Keys}}$) {

$i = i + 1$, trial = 0

ExtremaFound = False

While (trial $< N_{\text{trials}}$ and $\neg \text{ExtremaFound}$) {

if isExtremum (x_i, y_i) then

Key _{i} = KeypointDescriptor (x_i, y_i)

Keypoints = Keypoints $\cup \{ \text{Key}_i \}$

ExtremaFound = True

else

$(x_i, y_i) = \text{findNewCandidate}(x_i, y_i)$

end if

trial \leftarrow trial + 1 } }

其中: 函数 KeypointDescriptor() 确定坐标为 (x_i, y_i) 的特征点描述符, 随机函数 RandomCoord() 返回一系列现尺度下的 (x, y) 坐标值. 函数 isExtremum() 检测 (x_i, y_i) 是否是局部极值. 函数 findNewCandidate() 搜索同一尺度下点 (x_i, y_i) 周围的 8 个相邻点, 如果比较值 $p(x_i, y_i)$ 为正, 则返回

的是极大值的坐标；如果比较值 $p(x_i, y_i)$ 为负，则返回的是极小值的坐标。

4 Monte Carlo 定位算法

Monte Carlo 定位^[6,7]作为一种概率定位方法，原理基于贝叶斯算法，通过从传感信息递归估计位姿状态空间的概率分布来实现定位。Monte Carlo 定位的关键是应用 N 个带权值的随机样本来表示机器人位姿的可信度 $\text{Bel}(l)$ ，样本空间 $S = \{s_i | i = 1, 2, \dots, N\}$ 。样本集通过离散化状态近似表示机器人实际位姿的可信度。每个采样 s_i 由机器人位姿 $l_i = \langle x_i, y_i, \theta_i \rangle$ 和权值 p_i 组成，即 $s_i = \langle l_i, p_i \rangle \in S$ 。 p_i 表示机器人位于该点的概率，且 $\sum_{i=1}^N p_i = 1$ 。

与 Markov 定位相似，Monte Carlo 定位方法也包括基于运动模型预测和感知模型更新两个阶段。

4.1 运动模型

对于两轮差分驱动移动机器人，其运动模型^[8] $P(l_i | l_{i-1}, u_{i-1})$ 可利用机器人运动学来建立。在 i 时刻，设两轮差分驱动机器人相对于全局坐标系的位姿状态为 $l_i = (x_i, y_i, \theta_i)^T$ ， $u_i = (\delta_i, \alpha_i)^T$ 表示里程计模型的输入控制变量。其中： u_i 表示 i 时刻的输入矢量， δ_i 表示 i 时刻车体中心走过的距离， α_i 表示 i 时刻车体转过的角度。则由里程计测量所得到的机器人新位姿为

$$l_{i+1} = \begin{bmatrix} x_{i+1} \\ y_{i+1} \\ \theta_{i+1} \end{bmatrix} = \begin{bmatrix} x_i + \delta_i \cos \theta_i \\ y_i + \delta_i \sin \theta_i \\ \theta_i + \alpha_i \end{bmatrix} + \vartheta_i, \quad (7)$$

其中 ϑ_i 为随机噪声。在每个运动周期中，移动机器人根据运动模型 $P(l_i | l_{i-1}, u_{i-1})$ ，利用式(7) 采样得到下一时刻粒子集的概率分布，也就是利用运动模型来实现粒子集的预测过程。

4.2 感知模型

在 Monte Carlo 定位方法中，采样集通过感知模型 $P(z_i | l_i)$ ^[9,10] 重新设置权值。在基于视觉的定位中，感知数据是由位姿场景图像提取的特征组成的特征向量。

对于每个采样 s_i ，其权值 p_i 表示机器人处于位姿 $l_i = \langle x_i, y_i, \theta_i \rangle$ 的概率。当机器人获取传感器信息时，根据感知模型更新所有采用的权值，即 $p_i = \eta p(z_i | l_i)$ 。其中 η 为规一化因子，以确保所有采样的概率之和为 1。当两幅图像的 SIFT 特征向量生成后，采用特征点特征向量的欧氏距离作为两幅图像中特征点的相似性判定度量。通过计算这些邻近样本图像与实际观察到的图像的相似程度计算采样 s_i 的权值 p_i 。

设机器人在新的位姿实际观测到的环境模型为

图像 I_k ，则采用 s_i 的权值按下式更新：

$$p_i = \frac{1}{N} \sum_{j=1}^N d(I_k, I_j) (D - \text{dist}(I_i, I_j)) (\Phi - \delta(\theta_i, \theta_j)). \quad (8)$$

其中： N 为采用最邻近法确定的距离采样 s_i 较近的样本图像的数目， D 为设定的最大直线距离， Φ 为设定的最大角度距离， $\text{dist}(I_i, I_j)$ 为采样 s_i 所处的位置与预先采集样本图像的位置之间的直线距离， $\delta(\theta_i, \theta_j)$ 为采样 s_i 所处的角度与预先采集样本图像的角度之间的距离， $d(I_k, I_j)$ 为实际观测到的场景图像 I_k 与样本图像 I_j 之间的相似程度。最后对 p_i 进行归一化处理。

5 仿真实验

5.1 ISIFT 算法实验

为了研究本文算法中采样 NSamples 与检测到的特征点的关系，设计了如下实验：应用上述算法对 100 幅不同的图像进行实验，并且设定 NKey 为 1000。实验结果如图 2 所示，因为 SIFT 对整个图像进行运算，所以特征点数目不变，平均特征点数目为 262。迭代 SIFT 通常检测到更少的特征点，如果样本取得足够大，则能检测到与 SIFT 相同数目的特征点。

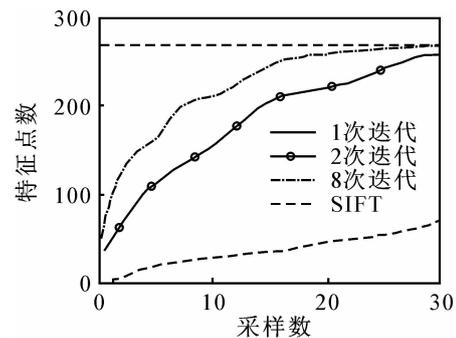


图 2 采样数、迭代次数和特征数的关系

5.2 全局定位实验

基于 ISIFT 图像检索的 Monte Carlo 定位系统，在 GRANDAR AS-R 移动机器人上作了仿真实验，实验环境为 $50\text{m} \times 30\text{m}$ 的办公室和走廊。ISIFT 通过随机过程获得特征点，使得在训练阶段和定位阶段获得图像的特征点可能不同。考虑到训练阶段对计算时间的要求较低，所以训练阶段采用经典的 SIFT 算法，而定位阶段使用 ISIFT 算法。

在训练阶段，人工控制机器人沿着规划好的路径行走，同时按距离间隔 50cm ，角度间隔 45° 采集样本图像，在走廊采集样本图像 1000 幅，在各个办公室共采集图像 720 幅。机器人自定位环境模型构成了数据库特征，每个位置数据库都存储一个特征集，这些特征从一系列位置已知的图像中抽取。然后

让机器人沿着同样的路径行走,并在各个位置采集图像,同时记录机器人的实际位姿,以便计算定位误差.通过实际采集到的数据对该系统进行离线测试.

机器人不知道自己的初始位姿,所以开始产生的采样集由400个位于整个位姿状态空间均匀分布的采样组成.在运动过程中,当机器人采集到的当前图像与数据库图像匹配时,粒子会分布在该数据库图像坐标周围.随着机器人不断接近,并搜集到足够证据以确信机器人实际位置的样本分布,在几次感知更新后,样本几乎集中在机器人真实位姿上.图3显示了实验室中全局定位采样的收敛过程及位置跟踪过程.

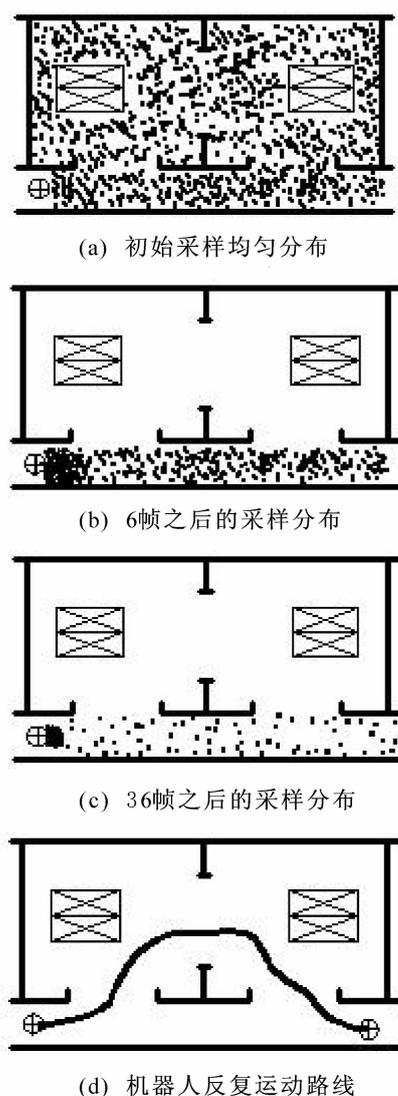


图3 定位过程描述

6 结 论

本文采用 ISIFT 算法进行特征点的抽取,减少了特征抽取和匹配过程中的计算量.在此基础上,提

出一种基于单目视觉的机器人 Monte Carlo 自定位方法.仿真实验结果表明,本文所述定位方法能有效地解决机器人的全局定位问题,并具有良好的计算性能.

参考文献(References)

- [1] David G Lowe. Distinctive image features from scale-invariant keypoints[J]. *Int J of Computer Vision*, 2004, 60(2): 91-110.
- [2] Lindeberg T. Scale-space theory: A basic tool for analysing structures at different scales[J]. *J of Applied Statistics*, 1994, 21(2): 224-270.
- [3] Tammimi H, Andreasson H, Treptow A, et al. Localization of mobile robots with omnidirectional vision using particle filter and iterative SIFT[J]. *Robotics and Autonomous Systems*, 2006, 54(9): 758-765.
- [4] Lindeberg T. Scale-space theory in computer vision [M]. Dordrecht: Kluwer Academic Publishing, 1994: 423-435.
- [5] 杨鹏, 高晶, 刘作军, 等. 基于全景与前向视觉的足球机器人定位方法研究[J]. *控制与决策*, 2008, 23(1): 75-78.
(Yang P, Gao J, Liu Z J, et al. Localization for robot soccer based on omni-vision and front-vision[J]. *Control and Decision*, 2008, 23(1): 75-78.)
- [6] Dellaert F, Burgard W, Fox D. Condensation algorithm for robust and vision-based mobile robot localization [C]. *Proc of the IEEE Computer Society Conf on CVPR*. Colorado, 1999: 2588-2594.
- [7] Fox D, Burgard W, Dellaert F. Monte Carlo localization: Efficient position estimation for mobile robots[C]. *Proc of the AAAI-99*. Orlando, 1999: 343-349.
- [8] 陈伟, 吴涛, 李政, 等. 基于粒子滤波的单目视觉 SLAM 算法[J]. *机器人*, 2008, 30(3): 242-247.
(Chen W, Wu T, Li Z, et al. A monocular vision SLAM algorithm based on particle filter[J]. *Robot*, 2008, 30(3): 242-247.)
- [9] 李桂芝. 自主移动机器人导航定位技术研究[D]. 北京: 北京科技大学, 2005.
(Li G Z. Research on navigation and localization for autonomous[D]. Beijing: Beijing University of Science and Technology, 2005.)
- [10] Jana Koseck'a, Li F Y, Yang X L. Global localization and relative positioning based on scale-invariant keypoints [J]. *Robotics and Autonomous Systems*, 2005, 52(1): 27-38.