

文章编号: 2095-0411 (2012) 02-0064-05

基于改进 SIFT 特征和粒子滤波的目标识别仿真研究*

尹 康, 段锁林, 邹 凌

(常州大学 机器人研究所, 江苏 常州 213164)

摘要: 针对在目标识别中原始 SIFT (尺度不变特征转换) 特征算法计算量大, 特征点匹配耗时长等缺陷, 采用一种改进的 SIFT 特征算法。在原始的 SIFT 算法基础上简化了特征描述符, 以及对匹配算法进行了改进, 考虑到识别过程中目标物体的特征点会发生变化, 因此结合粒子滤波来实现对目标物体的识别。仿真结果表明: 该方法继承了原始 SIFT 算法的优点, 有效地避免了一些干扰, 减小了计算量, 在结合粒子滤波算法后能够有效地更新特征点的匹配, 最终实现了对目标物体准确的识别。

关键词: SIFT 算法; 特征提取; 关键点匹配; 粒子滤波; 目标识别

中图分类号: TP 242.6

文献标识码: A

Research on the Object Recognition Based on Improved SIFT Feature and Particle Filter

YIN Kang, DUAN Suo-lin, ZOU Ling

(Institute of Robotics, Changzhou University, Changzhou 213164, China)

Abstract: For the problems of much calculation and consuming more time in the original SIFT feature algorithm, a kind of improved SIFT feature algorithm is used. Based on the original SIFT feature algorithm, the feature descriptor is simplified and the matching algorithm is improved. Considering the changing of feature descriptors in object recognition, the particle filter algorithm is combined with the target object recognition. Simulation result shows that this algorithm retains the advantages of original SIFT features algorithm, some disturbances are avoided and calculated amount is decreased. In combination with the particle filter algorithm it can effectively update feature point matching. The objects can be recognized reliably by using the combination algorithm.

Key words: SIFT algorithm; feature extraction; key point matching; particle filter; object recognition

目标识别技术是机器视觉领域中十分重要的课题之一, 图像特征点的配准度量由于对位置变化、灰度变化、图像形变以及光照等图像的复杂空间变化具有较好的适应能力。因此, 基于图像特征点配准的方法成为了图像配准技术界的主流方向和重要发展趋势^[1], 多年前就有学者在此方面做了重要研

究。图像灰度自相关函数的特征点检测算子^[2], 利用了图像灰度自相关函数的特征点匹配思想对目标物体进行识别, 此两种典型算法对于图像在出现旋转、光照变化时具有良好的稳定性。

近年来, 基于局部不变量描述子的方法在目标识别和匹配方面取得了显著的进展, 最具代表性的

* 收稿日期: 2012-03-20

基金项目: 常州市科技计划项目资助 (CJ20110023)

作者简介: 尹康 (1988—), 男, 江苏苏州人, 硕士生; 通讯联系人: 段锁林。

是由 David G Lowe 在 2004 年提出的尺度不变特征变换^[3] (Scale Invariant Feature Transform, SIFT), 并且通过实验证明了 SIFT 算法对于光照变化、几何形变、图像旋转等变化具有不变性和稳定性。但是随着识别技术的发展和需要, 特别是对识别要求精度较高的场景中, SIFT 算法的计算复杂性和耗时的缺点就凸显出来。利用 PCA 对 SIFT 特征描述符进行降维^[4], 降低了局部信息的误匹配率, 但计算量依然较大, 而且全局向量描述范围大小固定, 不具备尺度不变性。使用改进的 Canny 滤波器对 SIFT 描述符进行改进^[5], 使其适用于高精度识别场景, 但算法的复杂度大大提高。本文将 SIFT 特征描述符进行简化, 并对匹配算法进行改进。

在大多数目标识别场合, 目标物体往往是在一个较复杂的环境下, 非目标物体会对识别过程产生干扰。因此, 候选目标的选择也是非常重要的, 主要解决的是目标物体在“哪里”的问题, 具有代表性的有均值漂移 (mean shift) 方法^[6], 该方法将识别问题转化为最优化问题, 求解代价函数最小值的问题, 但其只适用于线性高斯情况, 对于非线性高斯情形, 效果不佳。对于此情况, 可以结合粒子滤波获得近似最优解, 以达到准确识别的目的。

通过对 SIFT 特征描述符进行简化, 并对匹配算法进行改进, 结合仿真验证了算法法的有效性。

1 SIFT 算法

1.1 图像的尺度空间

Koenderink 和 Lindeberg 的研究证明高斯核是实现尺度变换的唯一线性变换核, 假设云台 CCD 摄像机所获取的图像为 $I(x, y)$ 则这幅图像在不同尺度下的尺度空间 $L(x, y, \sigma)$ 可通过图像与高斯核 $G(x, y, \sigma)$ 的卷积来实现, 即:

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

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

其中 (x, y) 是空间坐标, σ 是尺度空间因子, 该值越小表示图像被平滑得越少, 相应的尺度也越小。大尺度对应图像的概貌特征, 小尺度对应图像的细节特征。

1.2 SIFT 特征匹配

常规 SIFT 特征匹配可以简述为如下 4 个主要

步骤:

(1) 尺度空间极值点检测。将摄像机获取的图像通过高斯核函数的连续滤波, 形成高斯金字塔图像, 进而再对相邻尺度的 2 个高斯图像进行相减得到 2 高斯图像的差值 DoG (Difference of Gaussians) 金字塔多尺度空间, 在 DoG 空间检测极值点。对该空间中的每个点与相邻尺度的点逐个进行比较就可以得到关键点所处的位置和相应的尺度。极值点的检测就是看该点是否均大于或小于它的 26 个邻域点 (包括该点相同尺度的 8 个邻域点, 以及大于该尺度的 9 个邻域点和小于该尺度的 9 个邻域点), 极值检测所采用的尺度空间函数定义为存在常数乘性尺度因子 k 的相邻尺度高斯函数的差分与原始图像的卷积, 取高斯差分图像的局部值, $D(x, y, \sigma)$ 可以表示为:

$$D(x, y, \sigma) = [G(x, y, k\sigma) - G(x, y, \sigma)] * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (3)$$

其中: $D(x, y, \sigma)$ 为尺度空间函数, k 为乘性尺度因子, 其取值可以根据高斯金字塔来予以获取。图 1 和图 2 分别表示传统高斯金字塔与多层多级高斯金字塔的示意图。

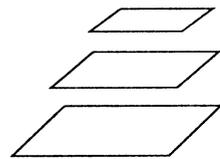


图 1 传统高斯金字塔

Fig. 1 Traditional gaussian pyramid

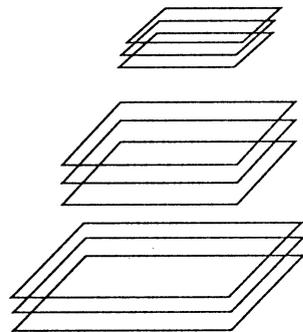


图 2 多级多层高斯金字塔

Fig. 2 Multi-stage multilayer gaussian pyramid

在获取极值点以后, 还要去除掉低对比度和不稳定的边缘响应点, 以此来获取高匹配能力和高抗噪声能力的特征点。

(2) 确定特征点方向。利用关键点邻域像素的梯度方向分布特性, 可以指定关键点参数方向。通过求每个极值点的梯度来为极值点赋予方向。像素点的梯度、梯度幅值和梯度方向可分别描述为:

$$\text{grad } I(x, y) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right) \quad (4)$$

$$m(x, y) =$$

$$\sqrt{[L(x+1, y) - L(x-1, y)]^2 + [L(x, y+1) - L(x, y-1)]^2} \frac{f_i}{\sqrt{\sum_{j=1}^8 f_j}}, \quad i=1, 2, \dots, 8 \quad (8)$$

$$\theta(x, y) = \tan^{-1} \left[\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)} \right] \quad (6)$$

其中 $I(x, y)$ 为摄像机所获取的图像函数, $L(x, y)$ 为所获取图像在不同尺度下的尺度空间函数。

对于每个特征点, 在特征点为中心的邻域窗口内采样, 用直方图统计邻域像素的梯度方向。00—360 是梯度直方图的取值范围, 100 为一柱, 共 36 柱, 梯度直方图的峰值对应角度就为特征点的主方向。

(3) 生成特征描述符。以特征点为中心取 8×8 的窗口, 在 4×4 的图像小块上计算 8 个方向的梯度方向直方图, 绘制每个梯度方向累加值, 形成一个种子点。一个特征点用 4×4 共 16 个种子点描述, 而每个种子点有 8 个方向向量信息, 所以, 每个特征点产生 $4 \times 4 \times 8$ 共 128 个数据, 即形成了 128 维的 SIFT 特征向量或特征描述符。

(4) 特征匹配。当目标物体图像的特征描述符生成以后, 就可以确定待测图像中关键点特征向量与目标物体图像中关键点特征向量的欧式距离, 欧式距离的计算公式为:

$$d = \text{sqrt}(\sum x_{i1} - x_{i2})^2 \quad i=1, 2, \dots, n \quad (7)$$

其中 x_{i1} 表示第一个点的第 i 维坐标, x_{i2} 表示第二个点的第 i 维坐标。则关键点相似性的度量可以根据欧式距离做判定。

2 改进的 SIFT 算法

仿真研究表明, 由于传统 SIFT 算法在建立尺度空间和生成特征描述符时花费了大量的时间, 而且匹配时间会随着图像的尺寸变大而增加。因此, 改进算法的主体思想是减少生成特征描述符的计算量和提高特征匹配速度^[7-8]。

2.1 特征描述符的简化

当特征点的主方向与坐标轴的主方向一致时, 简化的步骤如下所述:

(1) 以特征点为圆心, 以 8 pixel 为半径作圆形窗口, 确定特征点的邻域范围后, 从 1 pixel 至 7 pixel 为半径, 作以特征点为圆心的圆, 将邻域分

为 8 个子区域。

(2) 将各子区域内像素的灰度累加值 f_i 进行统计, 灰度累加值归一化后, 可以得到:

$$f_i = \frac{f_i}{\sqrt{\sum_{j=1}^8 f_j}}, \quad i=1, 2, \dots, 8 \quad (8)$$

(3) 计算各子区域内像素的灰度累加差分 $d f_i$, 通过一阶差分可得:

$$Dd f_i = \begin{cases} |f_i - f_{i+1}|, & i=1 \\ |2f_i - f_{i-1} - f_{i+1}|, & 1 < i < 8 \\ |f_i - f_{i-1}|, & i=8 \end{cases} \quad (9)$$

(4) 特征点的描述符由归一化灰度累加值和灰度累加差分构成, 就可以得到一个 16 维的描述符:

$$F = (\overline{f_1}, \overline{f_2}, \dots, \overline{f_8}, d \overline{f_1}, d \overline{f_2}, \dots, d \overline{f_8}) \quad (10)$$

通过上述过程建立的特征描述符考虑到了邻域内多个子区域的灰度信息与关联, 对平移变换、尺度变换、旋转变换等具有不变性。

改进 SIFT 算法以同心圆邻域内灰度累加值以及差分作为特征描述符, 将原先的 128 维的特征描述符简化为了 16 维的特征描述符, 大量的减少了计算量, 从而增加了算法的快速性。

2.2 匹配算法的改进

对目标图像与摄像机获取的场景图像进行基于特征点的匹配, 关键点在于要确定正确匹配点的数目。假设图像间的变换全为仿射变换

$$\begin{bmatrix} x_{21} & x_{22} & \dots & x_{2q} \\ y_{21} & y_{22} & \dots & y_{2q} \\ 1 & 1 & \dots & 1 \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & t_x \\ a_{10} & a_{11} & t_y \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1q} \\ y_{11} & y_{12} & \dots & y_{1q} \\ 1 & 1 & \dots & 1 \end{bmatrix} \quad (11)$$

其中 $(x_{11}, y_{11}), (x_{12}, y_{12}), \dots, (x_{1q}, y_{1q})$ ($q \geq 3$) 为目标物体图像部分特征点的坐标, 由 (11) 式可知, 该变换有 6 个自由度, 因此若检测到 3 组不共线的匹配对, 就可以认为是目标物体并进行跟踪。若是在实际实验中, 为提高跟踪的准确性和实时性, 可以适当增加匹配对的个数。

综上分析, 再根据尺度空间结构分析, 在目标识别中可考虑匹配过程从金字塔顶级开始, 自底而上进行匹配, 当目标一旦被认定, 即可停止匹配过程, 具体步骤如下:

(1) 用传统高斯金字塔结构表达待检测图像;

(2) 对金字塔顶级的图像, 进行尺度空间细化表达, 并在该尺度内检测 DOG 图像的极点, 准确定位特征点, 获取特征点的主方向并且对其进行简化;

(3) 将获取的特征点与参考图像的所有特征点进行匹配, 去除错误点后, 若正确匹配对达到给定数目后, 随即完成识别。若正确匹配对数目不够, 则进入下一级图像匹配。

(4) 进入下一级匹配后, 重复 (2) (3) 两步, 直至达到给定的匹配对数, 完成识别过程。

特征匹配结果如下图所示:

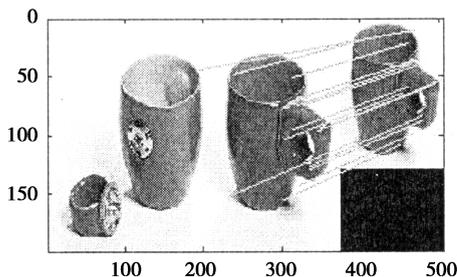


图 3 特征匹配结果

Fig. 3 Feature matching result

如若将该算法运用到机器人对目标物体的识别中, 目标物体与机器人会发生相对位移, 则特征点信息将会出现偏差。而粒子滤波算法可以根据更新粒子权重来估计当前目标物体特征点的信息, 可以有效增强目标识别的实时性。

3 粒子滤波算法

粒子滤波的主要思想是用一组具有权值的粒子来完全地描述后验概率分布。根据蒙特卡罗理论, 当粒子的数目足够多时, 这组具有权值的粒子就能完全地描述后验概率分布, 此时, 粒子滤波就是最优的贝叶斯估计^[9]。于 20 世纪 90 代被提出, 并由 Isard 引入计算机视觉领域。对于简单的识别问题, 目标状态就是目标的位置信息, 对于较为复杂的识别场合, 目标的状态还包括目标的旋转角度、尺度等信息。特别是在对非刚性目标识别, 其复杂性在于目标物体轮廓复杂, 背景复杂, 目标旋转以及摄像机发生相对位移等。如果定义为 t 时刻的目标状态和观测值, 则可以将目标识别问题转化为后验概率 $p(x_t | Z_{1:t})$ 的求解问题, 其中 $Z_{1:t} = (Z_1, \dots, Z_t)$ 是 t 时刻为止获得的所有观测值。则利用一组带有权重的粒子 $\{X_{t-1}^i, \omega_{t-1}^i\}_{i=1}^N$ 来近似逼近后验概率 $p(x_{t-1} | Z_{1:(t-1)})$ 。其中 X_{t-1}^i 表示

粒子, 也叫随机状态, 表示可能的目标状态, ω_{t-1}^i 表示粒子的权重。后验概率计算方法如下:

$$p(x_{t-1} | Z_{1:(t-1)}) \approx \sum_{i=1}^N \omega_{t-1}^i \delta((X_{0:t-1} - X_{0,t-1}^i)) \quad (12)$$

新粒子由适当的重要采样函数产生, 此函数依赖于目标状态和观测数据, 即

$$X_t^i \sim q(X_t | X_{t-1}^i, Z_t) \quad (13)$$

新粒子权重用下面的式子进行更新:

$$\omega_t^i \propto \omega_{t-1}^i \frac{p(Z_t | X_t^i) p(X_t | X_{t-1}^i)}{q(X_t | X_{t-1}^i, Z_t)} \quad (14)$$

新的粒子由适当的状态变换函数产生, 此函数定义为:

$$X_t = F(X_{t-1}, U_t) \quad (15)$$

其中, U_t 表示系统噪声, $F(\cdot)$ 描述了目标的运动状况。

如果取 $q(X_t^i | X_{t-1}^i, Z_t) = p(X_t | X_{t-1}^i)$, 则新的粒子权重通过计算观测似然函数来更新:

$$\omega_t^i \propto \omega_{t-1}^i p(Z_t | X_t^i) \quad (16)$$

粒子滤波算法流程如下:

已知 $t-1$ 时刻的粒子及权重 $\{X_{t-1}^i, \omega_{t-1}^i\}_{i=1}^N$ 和 $t-1$ 时刻的状态估计值 \tilde{X}_{t-1} , 则在 t 时刻:

(1) 当 $i=1, \dots, N$,

产生新粒子: $X_t^i = F_t(\tilde{X}_{t-1}, U_t^i)$,

更新粒子权重: $\omega_t^i \propto \omega_{t-1}^i p(Z_t | X_t^i)$;

(2) 归一化粒子权重: $\omega_t^i = \omega_t^i / \sum_{i=1}^N \omega_t^i$;

(3) 计算 t 时刻的状态估计值 \tilde{X}_t 。

4 结合算法描述

改进的 SIFT 特征和粒子滤波相结合的识别算法描述如下:

(1) 计算图像的每个粒子的权重。对粒子集合进行重采样, 得到新的粒子集合;

(2) 利用多级多层高斯金字塔计算图像的 SIFT 特征点;

(3) 采用改进的 SIFT 特征算法进行匹配;

(4) 计算所有粒子权重并进行归一化;

(5) 计算加权平均状态;

(6) 对目标物体进行定位并识别。

5 仿真研究

为了验证改进的 SIFT 特征和粒子滤波相结合

的算法在目标识别应用中的有效性和优越性，在 Matlab 7.0 平台下对其进行验证性实验。利用结合算法对实验可能遇到的特殊情况进行识别仿真，图像均受不同程度的旋转，拉伸，和光照变化等影响，识别效果较好。以下就选取日常场景图像的识别结果。理想情况识别效果如图 4。

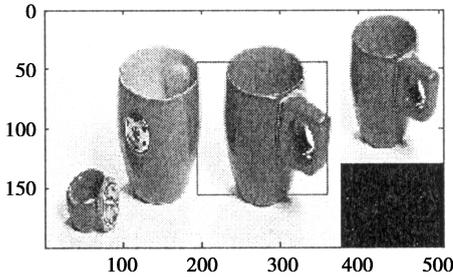


图 4 理想情况识别效果

Fig. 4 The ideal situation recognition result

结果表明：在理想情况下结合算法能够准确的识别出目标物体。

5.1 目标旋转拉伸对识别的影响

由于 CCD 摄像机在获取图像时可能会使用不同的焦距或视角，图像会产生一定程度的旋转或者拉伸，图 5 表示目标物体产生了 90° 的旋转并且有较大拉伸。

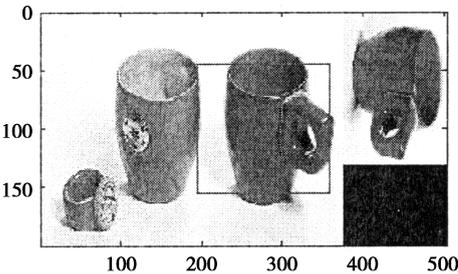


图 5 目标旋转拉伸后识别效果

Fig. 5 Target rotation after stretching recognition result

结果表明目标物体图像发生旋转拉伸变化时结合算法能够有效匹配目标物体特征点并进行准确识别。不管目标物体如何的拉伸旋转，其特征点总是存在的，故不会影响识别结果。

5.2 目标受光照变化对识别的影响

由于实验环境的光照会受到时间和地点的影响，因此识别的对光照变化保持不变性非常重要。图 6 表示亮度减小的识别仿真效果。

结果表明：当目标物体所在环境发生光照变化时结合算法也能准确识别。结合算法是采用匹配并更新特征点来对目标物体进行识别，只要特征点存

在并且可见，则与光照的变化因素无关。

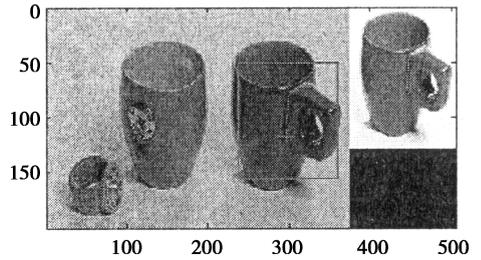


图 6 光照变化后识别效果

Fig. 6 Recognition result after illumination change

6 结 语

本文将改进的 SIFT 特征与粒子滤波算法应用目标识别研究中，基本满足了对目标物体进行准确识别这一要求。对目标物体的各种变化和環境变化，该算法也有较好的识别效果。需要指出的是，对特征点提取的过程中，需要在各个尺度上进行计算，没有考虑目标物体被遮挡的情况，此种情况下目标物体的特征点在配准过程中是否会遇到问题，还有待下一步研究。另外，本研究所要识别的目标物体背景相对而言较单一，如果在较复杂的背景中识别目标物体，该算法可能会出现错误匹配的点。所以，在今后的研究中还要对该融合算法进行改进与优化。

参考文献：

[1] 赵芹, 周涛, 舒勤. 基于特征点的图像配准技术探讨 [J]. 红外技术, 2006, 28 (6): 327-330.
 [2] 李冬梅, 王延杰. 一种基于特征点匹配的图像拼接技术 [J]. 微计算机信息, 2008, 24 (5): 296-298.
 [3] D G Lowe. Distinctive image features from scale-invariant keypoints [J]. International Journal of Computer Vision, 2004, 60 (2): 90-110.
 [4] 秦雪, 侯进. 一种改进的 SIFT-PCA 算法在图像检索中的应用 [J]. 西南科技大学学报, 2011, 26 (4): 66-70.
 [5] 杨秋菊, 肖雪梅. 基于改进 Canny 特征点的 SIFT 算法 [J]. 计算机工程与设计, 2011, 32 (7): 2428-2431.
 [6] 王雷光, 郑晨, 代沁伶. 基于均值漂移聚类的油罐识别 [J]. 电脑知识与技术, 2011, 7 (17): 4155-4157.
 [7] 孙林, 吴相林, 周莉, 等. 对扭转鲁棒的 SIFT 图像匹配在商标识别中的应用 [J]. 计算机工程与设计, 2011, 32 (4): 1406-1410.
 [8] 张书真, 宋海龙, 向晓燕, 等. 采用快速 SIFT 算法实现目标识别 [J]. 计算机系统应用, 2010, 19 (6): 83-85.
 [9] 王法胜, 赵清杰. 一种用于解决非线性滤波问题的新型粒子滤波算法 [J]. 计算机学报, 2008, 31 (2): 346-352.