第38卷 第18期
Vol.38 No.18

计 算 机 工 程
Computer Engineering

2012年9月
September 2012

·人工智能及识别技术· 文章编号：1000—3428(2012)18—0140—04 文献标识码：A 中图分类号：TP311

基于在线学习和结构约束的目标跟踪算法

王守超，李小霞

(西南科技大学信息工程学院，四川 绵阳 621010)

摘 要：为在复杂环境中对目标进行长期的精确跟踪，提出一种基于在线学习和结构约束的目标检测和跟踪算法。采用改进的光流法对特定目标进行自适应跟踪，实时目标检测采用非层次结构在线学习随机蕨丛分类器。用基于结构约束的非监督学习法精确确定目标位置，以适应目标的形态变化。实验结果表明，该算法能够适应目标的基本形态变化，在目标出现尺寸变化、旋转、部分遮挡或短暂消失时都能稳定精确地跟踪目标。

关键词：随机蕨丛；结构约束；光流法；在线学习；目标跟踪；前向-后向误差

Target Tracking Algorithm Based on Online Learning and Construction Constraint

WANG Shou-chao, LI Xiao-xia

(School of Information Engineering, Southwest University of Science and Technology, Mianyang 621010, China)

【Abstract】 In order to accurately track target during a long term in complex environment, this paper presents an algorithm combining detecting and tracking based on online learning and construction constraint. Improved optical flow method is used to adaptively track specific target. The nonhierarchical structure online learning random ferns classifier is used as real time target detecting method. In order to adapt target shape variation, unsupervised learning method based on construction constraint is used to accurately determine the target position. Experimental results show that this algorithm can adapt to the basic target shape variations and track the target steadily when the targets change in size, rotate, shelter partly or disappear in a short term.

【Key words】 random ferns brake; construction constraint; optical flow method; online learning; target tracking; forward-backward error

DOI: 10.3969/j.issn.1000-3428.2012.18.038

1 概述

视频目标跟踪是计算机视觉的一个重要研究方向，在视频监控、机器人等领域具有广泛的应用前景。针对在复杂环境中对目标进行长期精确跟踪的问题，目前常用的方法是结合基于机器学习的目标检测算法辅助跟踪目标。检测算法分类器的训练需要大量手工标记的样本，由于样本具有不完备性，导致训练出的检测分类器很难达到较高的检测率。基于在线学习的目标检测算法可以有效解决这些问题，其样本的创建和类别标注都是自动进行的，通过对新样本持续的学习，检测分类器可适应检测环境以及目标的变化，逐步提高自身的检测性能。

文献[1]提出了 On-line Boosting 算法，文献[2]在此基础上增加特征选择能力，提出了 On-line Adaboost 算法，从而使在线学习能够用于目标检测与跟踪。此外，比较有代表性的协同跟踪方法(Co-tracking)[3]运用 2 个独立的特征在线训练 SVM 分类器，从而能够对外观不断变化的目标进行跟踪，缺点是目标运动姿态变化较大时会出现跟踪失败。文献[4]通过采用二值特征模型在线训练随机森林分类器进行目标检测，可实现对特定目标的长期跟踪。文献[5]提出了一种基于 Co-training 框架的在线学习跟踪方法，有效避免了模型误差累积和跟踪丢帧等问题。

本文采用改进的光流法对目标进行自适应跟踪，选取可以进行在线学习的非层次结构的随机蕨丛分类器[6]用于目标检测。在跟踪过程中，通过在线学习目标的各种形态变化，建立目标结构约束集，用于对跟踪和检测的结果进行校验，自适应检测和跟踪目标。

2 目标跟踪

首先手工选定所要跟踪目标，设目标矩形为 $B_1=[(x_1,y_1),(x_2,y_2)]$，其中，$(x_1,y_1)$、$(x_2,y_2)$ 分别表示矩形区域的对角坐标。在 B_1 内均匀采样 n 个像素点作为特征点，采用 Lucas-Kanade 光流法对特征点进行跟踪，对应特征点之间的相似程度通常采用归一化互相关(*NCC*)系数度量。*NCC* 度量方法对光照强度不敏感，但当特征点周围灰度变化较大(如刚性变换)时，会出现错误评价，可采用前向-

基金项目：国家自然科学基金资助项目“单目高精度大型物体彩色三维数字化测量原理研究”(60808020)

作者简介：王守超(1987－)，男，硕士研究生，主研方向：多目标检测与跟踪；李小霞，副教授、博士

收稿日期：2011-11-21 **修回日期：**2012-01-12 **E-mail：**wyuting46@163.com

后向误差[7]作为补充。

假设 p_{t-1} 表示前一帧图像中某一特征点的位置，用光流法计算预测其在当前帧的位置 p_t，然后根据当前帧和前一帧计算的数据反向跟踪特征点 p_t，快速计算出其在前一帧图像的位置 $\hat{p}_{t-1}$，则特征点 p_t 的前向-后向误差定义为：

$$FB(p_{t-1},p_t)=\left\|p_{t-1}-\hat{p}_{t-1}\right\| \tag{1}$$

其中，距离计算通常采用欧氏距离，当前向-后向误差大于一定阈值时，特征点跟踪位置不可靠。

运用 Lucas-Kanade 光流法计算前一帧特征点集 F_1 的光流，跟踪得到的特征点集为 F_2，根据 F_2 确定当前帧目标矩形 B_2，具体过程如下：

(1)找出特征点集 F_1 和 F_2 中对应特征点光流被发现的所有特征点，组成新的集合，记为 F_1^* 和 F_2^*。

(2)计算特征点集 F_1^* 和 F_2^* 中对应特征点的 *Ncc* 和 *FB* 误差，排序后分别找出中值 *midNcc* 和 *midFB*。

(3)找出特征点集 F_1^* 和 F_2^* 中 *FB* 误差不大于 *midFB* 且 *Ncc* 不小于 *midNcc* 的特征点，重新记为 F_1^{**} 和 F_2^{**}。

(4)分别计算 F_1^{**} 和 F_2^{**} 中对应特征点的横坐标差值和纵坐标差值，排序后找出差值的中值 d*x* 和 d*y*。

(5)分别计算 F_1^{**} 和 F_2^{**} 中所有特征点两两之间的欧氏距离，然后转换成一维向量，以表征特征点集的疏散程度，再对 2 个向量做点除运算，对得到的结果排序后，找出中值 *e*。

(6)按照下式计算当前帧目标状态：

$$B_2=[(x_1-e_1+\mathrm{d}x,y_1-e_2+\mathrm{d}y),(x_2+e_1+\mathrm{d}x,y_2+e_2+\mathrm{d}y)] \tag{2}$$

其中，$e_1=0.5\times(e-1)\times(x_2-x_1)$；$e_2=0.5\times(e-1)\times(y_2-y_1)$。

3　目标结构约束

根据视频尺寸和选定目标窗口的大小，建立一系列扫描窗口，扫描窗口应尽量包含目标状态可能变化的所有尺寸和位置，同时不能太过细密，需要在跟踪速度和精度之间权衡。由于扫描窗口比较多，因此同一区域可能被多种扫描窗口包含或与很多窗口有重叠的部分，样本创建及目标位置的确定均用到窗口重合率 *overlap* 的计算，其计算公式为：

$$overlap=\frac{W_A\cap W_B}{W_A+W_B-(W_A\cap W_B)} \tag{3}$$

其中，W_A 和 W_B 分别表示 2 个窗口的面积，用矩形区域的像素个数表示。

为了提高目标判别的准确性，目标结构模型采用正结构和负结构一起约束，正结构主要是目标自身的各种形态，负结构选取靠近目标区域的背景图像块[8]，目标结构约束集合的创建和更新过程如图 1 所示。初始正样本只有一个，即与选定目标窗口重合率最大的扫描窗口图像区域，标记类别为 1；负样本选取靠近选定目标区域且方差大于正样本方差 1/4 的扫描窗口图像区域，标记类别为 0。将所有样本变换到 15×15 大小，再归一化为零均值、单位方差，以向量形式存储。

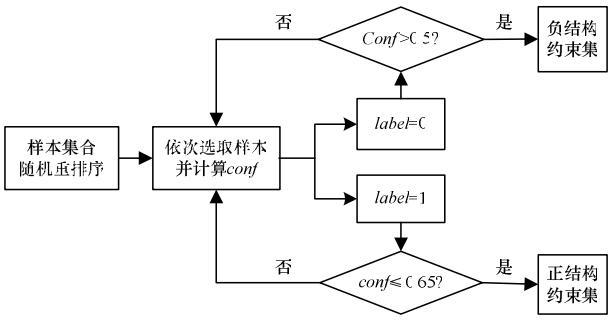


图 1　目标结构约束集合训练过程

为了保证初始训练时正、负结构约束集合不为空，选取唯一的正样本加入正结构约束集，第 1 个负样本加入负结构约束集，分别计算每个样本和正、负结构约束集合中每个元素的 *NCC*，距离用 1 减去 *NCC* 表示，选取最小距离，记为 *dP* 和 *dN*，按照下式计算相对相似度：

$$conf=\frac{dN}{dN+dP} \tag{4}$$

conf 值越大，样本越像目标，反之就越接近背景。

由图 1 可以看出，计算得到的 *conf* 值很容易区分出目标，这是在线学习目标形态的重要过程。当学习完所有目标形态和运动背景，结构约束集合就不再改变。

4　检测算法

设 C_i,1,2,…,*h* 代表类别的集合，*C* 表示代表类别的随机变量，f_i,1,2,…,*n* 表示从图像块中提取的一系列二值特征值，则图像块所属的类别为：

$$\widehat{C}_i=\arg\max_{C_i}P(C=C_i\,|\,f_1,f_2,\cdots,f_n) \tag{5}$$

二值特征值 f_j 的选取方法是比较图像块内随机生成的 2 个像素位置 $d_{j,1}$ 和 $d_{j,2}$ 的灰度值 $I(d_{j,1})$和 $I(d_{j,2})$的大小，取值如下：

$$f_j=\begin{cases}1 & \text{if } I(d_{j,1})<I(d_{j,2})\\ 0 & \text{otherwise}\end{cases} \tag{6}$$

一般情况下需要把所有特征值分成 *m* 组，每组特征值数 *s*=*n*/*m*，每一组就是一棵蕨，然后计算所有蕨的联合概率值，再根据贝叶斯公式，上式简化为：

$$P(f_1,f_2,\cdots,f_n\,|\,C=C_i)=\prod_{k=1}^{m}P(F_k\,|\,C=C_i) \tag{7}$$

其中，$F_k=\left\{f_{\sigma(k,1)},f_{\sigma(k,2)},\cdots,f_{\sigma(k,s)}\right\}$；*k* 表示第 *k* 棵蕨；$\sigma(k,j)$ 表示范围 1,2,…,*n* 的随机排序函数。

为了使分类器对目标形变、图像模糊及噪声等具有较强的稳健性，采用对目标区域及其邻域进行随机仿射变换的方法创建大量正样本。负样本的选取方法与创建目标结构约束集合负样本选取方法一样。每个样本提取 130 个特征值，分成 10 组，每组 13 个，所分类别有 2 类：目标和背景。为了简化计算，提高分类器的决策速度，每棵蕨训练的结果以样本属于目标的概率形式给出，在线检测时最终类别由 10 棵蕨的总投票决定。

在线检测时，在每个扫描窗口图像块内提取二值特征值，送入训练好的随机蕨丛分类器，如果 10 棵蕨蕨值之

和大于 5，则该扫描窗口属于目标。由于二值特征值比较简单，会出现误检情况，因此需要对检测结果进行目标结构约束，设定阈值 α，保留相对相似度较大的检测窗口，记为 $B=\{b_1,b_2,\cdots,b_n\}$，然后进行聚类。根据窗口重合率，给定最小距离阈值 t，聚类过程如下：

(1)将 B 中每个元素归为一类簇，$c_i=\{b_i\}$。

(2)按照下式计算 c 中所有两两元素之间的距离：

$$d(c_i,c_j)=1-o(c_i,c_j) \tag{8}$$

其中，$o(c_i,c_j)$ 表示 c_i 和 c_j 的重合率，找出 $d(c_i,c_j)$ 最小的一对类簇。

(3)如果 $d(c_i,c_j)>t$，则结束；否则，继续下面步骤。

(4) c_i 和 c_j 合并成一个类 c_i，合并方法是对应矩形坐标求平均值，同时对应的相对相似度取平均值。

(5)去除类簇 c_j，返回步骤(2)。

5 目标跟踪与在线学习

首先需要判断跟踪结果是否出现异常，如目标区域超出图像范围、前向-后向误差中值过大，则跟踪失败。跟踪结果需要进行目标结构约束，其相对相似度记为 $conf_t$，检测结果聚类后的相对相似度为 $conf_d$，然后按照图 2 确定目标状态，并判断是否进行在线学习。

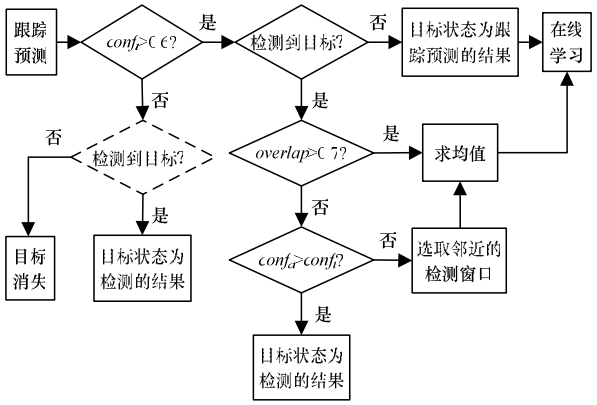


图 2 目标跟踪与在线学习过程

图 2 中虚线框和实线框中的“检测到目标”表示同一过程，均指随机蕨丛算法在当前帧是否检测到目标。如果跟踪并且检测到目标，需要计算跟踪窗口和检测窗口的重合率 $overlap$。窗口求均值的计算方法是检测窗口和跟踪窗口对角坐标求平均。

在线学习包括目标结构约束集合的更新和随机蕨丛分类器的更新 2 个过程。在确定目标区域内选取窗口重合率最大的扫描窗口图像块作为正样本，选取与目标区域重合率小于 0.2 的检测窗口图像块为负样本，然后按照图 1 更新目标结构约束集合。随机蕨丛分类器负样本选取与目标区域重合率小于 0.2 且检测分类器中满足 10 棵蕨蕨值之和大于 1 的扫描窗口图像块，提取特征值后更新蕨值。

6 实验结果与分析

本文算法在 Windows XP 操作系统下，使用 Visual C++ 6.0 开发环境，基于 OpenCV 1.0 开源库实现。实验分为 3 组，第 1 组测试跟踪算法的自适应性，第 2 组测试阈值 α 对检测精度的影响，第 3 组测试算法的整体性能。

图 3 给出了跟踪算法对目标尺寸变化、旋转变化及遮挡过程的自适应性。由图可以看出，跟踪的目标框基本适应目标状态的变化。



(a)尺寸变化



(b)旋转变化



(c)遮挡变化

图 3 跟踪算法对目标尺寸、旋转变化及遮挡的自适应性

选取跟踪算法稳定跟踪目标的 100 帧图像测试检测算法性能，以检测窗口与跟踪窗口重合率大于 0.6 为检测到目标，否则，没有检测到目标或误检，结构约束集合创建后不进行更新，检测算法不进行在线学习，分别选取 α 值为 0、0.5、0.53、0.55、0.6 和 0.65，实验结果如表 1 所示。

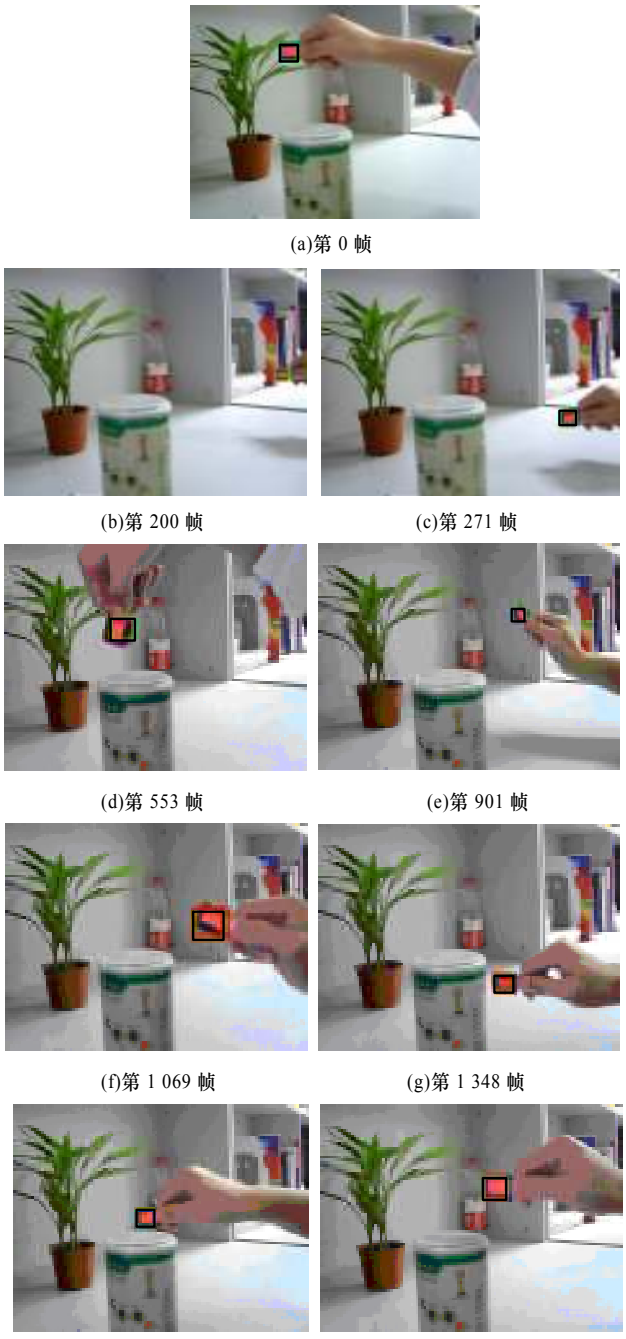
表 1 α 值对检测结果的影响

α 值	检测到的目标帧数	检测到的窗口总数	误检窗口数	目标检测率/(%)	误检率/(%)
0.00	100	476	240	100	50.42
0.50	99	308	133	99	43.18
0.53	99	223	62	99	27.80
0.55	98	148	6	98	4.05
0.60	92	102	0	92	0.00
0.65	63	63	0	63	0.00

从表 1 中可以看出，在没有结构约束(α=0)时，误检率非常高，α 值增大，误检率明显下降，但同时检测率也下降。由于检测到的窗口越多，越容易学习到目标的变化形态，因此选取合适的 α 值，既可以在线更新目标结构约束集合，又可以避免误检，取值一般为[0.55,0.70]。

本文算法整体性能的实验效果如图 4 所示，选取的视频尺寸为 640×480、共 1 546 帧。随机蕨丛分类器训练时的正样本数为 200，旋转角度范围[−45°,45°]，尺寸变化为选定目标尺寸的 0.5 倍~1.5 倍，偏移程度设为目标尺寸的

0.005 倍，添加均值为 0、标准差为 1 的高斯白噪声。聚类过程选取阈值 $t=0.4$，结构约束阈值 $\alpha=0.7$，在线学习过程随机蕨丛正样本选取 10 个，其他条件不变。



(a)第 0 帧　(b)第 200 帧　(c)第 271 帧　(d)第 553 帧　(e)第 901 帧　(f)第 1 069 帧　(g)第 1 348 帧　(h)第 1 447 帧　(i)第 1 501 帧

图 4　跟踪和检测共同确定目标的实验结果

图 4 中黑色框表示最终目标状态。图 4(a)~图 4(c)分别是第 0 帧、第 200 帧、第 271 帧的目标状态，目标从初始位置逐渐远离再回来的过程，主要是目标尺寸发生变化，同时背景环境干扰逐渐变强，从图 4(c)可以看出，算法稳定跟踪到目标；图 4(d)~图 4(f)分别是第 553 帧、第 901 帧、第 1 069 帧的目标状态，在该过程中目标发生很大角度范围内的旋转并加上尺寸变化。在目标状态发生变化、相对相似度减少时，学习过程不断加入新的目标结构，在目标旋转接近 90°出现错切变换时都成功跟踪到目标。图 4(g)~图 4(i)第 1 348 帧、第 1 447 帧、第 1 501 帧的目标状态，错切变换后，由于结构约束集合已经学习到该状态，因此可继续稳定跟踪到目标，第 1 348 帧目标从左侧进入障碍物，逐渐被完全遮挡，第 1 447 帧目标消失后从障碍物上方以一定旋转角度重新出现，由于之前已经学习到目标各种旋转角和尺度下的状态，因此依然稳定跟踪到目标。

本文算法在 2 GB 内存、2.8 GHz CPU 的电脑上处理上述视频的速度为 10 f/s，由于跟踪算法采用的是稀疏光流法跟踪目标区域内均匀采样的 100 个点，因此该部分所消耗的时间很少，大部分时间消耗在目标检测上，视频尺寸越大，扫描窗口越多，所需要的处理时间也就越多。实验时采用 CamShift 算法对上述视频中的目标进行跟踪，处理速度约为 35 f/s，但在目标发生尺寸变化时，不能稳定跟踪目标。目标发生旋转变化后，跟踪失败。

7　结束语

本文算法在自适应跟踪目标的同时结合了随机蕨丛分类器检测目标，应用扫描窗口的方法解决了在线学习算法中样本的创建及类别标注的问题，并通过创建和更新结构约束集合学习目标的各种形态变化，提高了检测算法性能，适合于在复杂环境中对特定目标进行长期精确跟踪。

参考文献

[1] Oza N C, Russell S. Online Bagging and Boosting[C]//Proc. of Artificial Intelligence and Statistics. San Francisco, USA: Morgan Kaufmann, 2001.

[2] Grabner H, Bischof H. On-line Boosting and Vision[C]//Proc. of IEEE Conference on Computer Vision and Pattern Recognition. New York, USA: [s. n.], 2006.

[3] Tang Feng, Brennan S, Zhao Qi, et a1. Co-tracking Using Semi-supervised Support Vector Machines[C]//Proc. of IEEE Conference on Computer Vision. Rio de Janeiro, Brazil: [s. n.], 2007.

[4] Kalal Z, Mikolajczyk K, Matas J. Online Learning of Robust Object Detectors During Unstable Tracking[C]//Proc. of IEEE Conference on Computer Vision. Kyoto, Japan: [s. n.], 2009.

[5] 齐志泉，宋　野，王来生. 基于在线学习的目标跟踪方法研究[J]. 计算机应用研究, 2010, 27(2): 770-774.

[6] Ozuysal M, Calonder M, Lepetit V, et al. Fast Keypoint Recognition Using Random Ferns[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(3): 448-461.

[7] Kalal Z, Mikolajczyk K, Matas J. Forword-Backward Error: Automatic Detection of Tracking Failures[C]//Proc. of International Conference on Pattern Recognition. Istambul, Turkey: [s. n.], 2010.

[8] Kalal Z, Mikolajczyk K, Matas J. P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints[C]//Proc. of IEEE Conference on Computer Vision and Pattern Recognition. San Francisco, USA: IEEE Press, 2010.

编辑　张正兴