

自适应学习的时空上下文目标跟踪算法

张 晶,王 旭,范洪博

(昆明理工大学 信息工程与自动化学院,昆明 650500)

摘 要:传统时空上下文目标跟踪(STC)算法在目标发生尺度变化时,跟踪窗口长期不变导致学习的上下文空间模型不具有针对性。为此,提出一种能够进行自适应学习的时空上下文目标跟踪(STC-AL)算法。在前后输出窗口提取尺度不变特征并消除误匹配,对匹配点集进行综合分析后调整输出窗口,并对传统空间模型的学习与更新进行改进。实验结果表明,STC-AL 算法能够适应目标尺度变化,与 STC 算法、CT 算法和 KCF 算法相比,跟踪结果更准确。

关键词:自适应;目标跟踪;时空上下文;尺度不变特征;空间模型

中文引用格式:张 晶,王 旭,范洪博. 自适应学习的时空上下文目标跟踪算法[J]. 计算机工程,2018,44(6):294-299.

英文引用格式:ZHANG Jing, WANG Xu, FAN Hongbo. Spatio temporal context target tracking algorithm of self-adaption learning[J]. Computer Engineering, 2018, 44(6): 294-299.

Spatio Temporal Context Target Tracking Algorithm of Self-adaption Learning

ZHANG Jing, WANG Xu, FAN Hongbo

(School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China)

[Abstract] For the traditional Spatio Temporal Context target tracking (STC) algorithm, no change of tracking's window for a long time leads to learning space context model does not have a targeted problem when the target scale changes. This paper proposes a Spatio Temporal Context target tracking algorithm for Adaptive Learning (STC-AL). The Scale Invariant Feature Transform (SIFT) is extracted from the front and back output windows and used to eliminate false matches. After analyzing the matching point set, the output window is adjusted, and the learning and updating of the traditional spatial model is improved. Experimental results show that STC-AL algorithm can adapt to changes in the target scale, and tracking is more accurate compared with that of STC algorithm, CT algorithm and KCF algorithm.

[Key words] self-adaption; target tracking; Spatio Temporal Context (STC); scale invariant feature; spatio model

DOI:10.19678/j.issn.1000-3428.0047392

0 概述

目标跟踪在信息物理融合系统感知领域里是一个热门话题,通过信息融合技术将不同传感器获取的目标特征信息进行融合来实现场景中的目标跟踪。但为了准确获取目标信息,在进行跟踪时会出现多种问题,其中包括目标物体被遮挡,环境光照变化,目标连续变化而导致的颜色、边缘的相应变化。近年来,研究者针对这些问题提出了很多跟踪算法,如 MIL^[1]、IMIL^[2]、FCT^[3]、IVT^[4]、VTD^[5]、Box-PF^[6]及 TLD^[7]、IA-MCMC^[8]、ANMF-SSAMC^[9]算法等。但是为了能够达到跟踪的鲁棒性,研究者引入了时空上下文的思想,将时间上下文环境与空间上下文信息相结合进行跟踪,即 STC (Spatio Temporal Context)^[10] 算法,达到了较好的鲁棒性。

许多学者对 STC 算法进行了研究并提出了很多改进。文献[11]提出干扰项和有贡献区域的定义,在此基础上进一步细化上下文结构。文献[12]根据上下文不同区域与目标区域运动的相似性赋予信息不同的权值。文献[13]提出利用在线结构化支持向量机建立空间约束信息模型来提高跟踪的准确性。文献[14]提出了抗遮挡机制,将上下文区域分块并基于颜色的相似度来判定目标是否被遮挡,遮挡后采取分块匹配来定位目标的位置。

STC 算法在目标跟踪过程中进行上下文先验模型学习是同等的,离目标中心距离越近灰度贡献值就越大。而因目标运动发生越来越远离镜头的尺度变化,STC 算法窗口的不自适应性会增加不必要区域的模型学习,当运动越来越靠近镜头,就会降低超出窗口区域的目标灰度学习率,因此,其不能充分利

基金项目:国家自然科学基金(61562051);云南省应用基础研究计划重点项目(2014FA029)。

作者简介:张 晶(1974—),男,教授、博士,主研方向为实时嵌入式软件、信息物理融合系统;王 旭,硕士;范洪博,讲师、博士。

收稿日期:2017-05-27 **修回日期:**2017-07-26 **E-mail:**414356764@qq.com

用目标所有的信息进行学习。

为适应目标大小变化并准确进行模型学习从而判断目标位置,本文提出一种自适应学习的 STC 目标跟踪算法。根据目标变化的连续性以及变化导致的目标特征点间距离变化的特性,采取尺度不变特征(Scale Invariant Feature Transform, SIFT)提取方法与 RANSAC 消除误匹配算法^[15-16],匹配前后两帧的尺度不变特征点,综合所有特征点进行尺度变化分析并调整目标窗口,将调整后的窗口作为目标输出,并对目标窗口的上下文区域进行改进。

1 时空上下文视觉跟踪

STC 算法基于贝叶斯框架,在当前帧确定上下文特征集合: $X^c = \{c(z) = (I(z), z) | z \in \Omega_c(x^*)\}$, 其中, x^* 是目标区域中心, $I(z)$ 是目标区域像素 z 的灰度值, $\Omega_c(x^*)$ 是由目标矩形确定的局部上下文区域的图像灰度与位置的统计建模。通过一些基本的概率公式,计算置信图找到似然概率最大的位置,即为跟踪结果。置信图公式为:

$$c(x) = \sum_{c(z) \in X^c} P(x|c(z), o) P(c(z)|o) \quad (1)$$

其中, x 和 z 表示像素坐标, $P(x|c(z), o)$ 是目标 o 与局部上下文区域空间关系的条件概率, $P(c(z)|o)$ 是局部上下文各个点的上下文先验概率。

$P(c(z)|o)$ 上下文先验模型在 STC 算法中进行以下计算:

$$P(c(z)|o) = I(z) \omega_\sigma(z - x^*) \quad (2)$$

$$\omega_\sigma(z - x^*) = a \times \exp\left(-\frac{|z - x^*|^2}{\sigma^2}\right) \quad (3)$$

其中, a 是归一化参数,取值为 $[0, 1]$, σ 是一个尺度参数, σ^2 为高斯函数方差, $\omega_\sigma(z - x^*)$ 是区域中心加权的高斯函数,区域中心像素贡献大,区域边缘像素贡献小。而在 STC 算法中将 $P(x|c(z), o)$ 称为空间上下文模型,定义为:

$$P(x|c(z), o) = h^{SC}(x - z) \quad (4)$$

式(1)中置信函数 $c(x)$ 在 STC 算法中计算如下:

$$c(x) = b \times \exp\left(-\left|\frac{x - x^*}{\alpha}\right|^\beta\right) \quad (5)$$

其中, b 是归一化参数, α 是尺度参数, β 是目标形状参数。

将式(2)~式(5)带入式(1)得:

$$c(x) = h^{SC}(x) \otimes (I(x) \times a \times \exp\left(-\frac{|x - x^*|^2}{\sigma^2}\right)) \quad (6)$$

为了加快 STC 算法的运行速度,将式(6)两边进行快速傅里叶变换,得空间上下文模型为:

$$h^{SC}(x) = F^{-1}\left(\frac{F(b \times \exp\left(-\left|\frac{x - x^*}{\alpha}\right|^\beta\right))}{F(I(z) \omega_\sigma(z - x^*))}\right) \quad (7)$$

综上, STC 算法追踪过程如下:

1) 计算 t 帧人为选定的矩形框置信图:

$$c_t(x) = \sum_{c(z) \in X^c} P(x|c(z), o) P(c(z)|o) = b \times \exp\left(-\left|\frac{x - x^*}{\alpha}\right|^\beta\right)$$

计算上下文先验模型:

$$P(c(z)|o) = I_t(z) \omega_\sigma(z - x^*)$$

$$\omega_\sigma(z - x^*) = a \times \exp\left(-\frac{|z - x^*|^2}{\sigma^2}\right)$$

2) 计算当前帧 $\Omega_c(x^*)$ 上下文区域的空间上下文模型:

$$h_t^{SC}(x) = F^{-1}\left(\frac{F(b \times \exp\left(-\left|\frac{x - x^*}{\alpha}\right|^\beta\right))}{F(I_t(z) \omega_\sigma(z - x^*))}\right)$$

3) 更新空间上下文 $h^{SC}(x)$ 模型:

$$H_2^{SC}(x) = h_1^{SC}(x)$$

$$H_{t+1}^{STC}(x) = (1 - \rho) H_t^{STC}(x) + \rho h_t^{SC}(x) \quad (8)$$

其中, ρ 为模型更新的学习率。

4) 利用 t 帧的 $\Omega_c(x^*)$ 区域在 $t+1$ 帧计算上下文先验模型:

$$P(c(z)|o) = I_{t+1}(x) \omega_\sigma(z - x_t^*)$$

5) 根据式(1)、式(6),已知 $t+1$ 帧的上下文先验模型以及空间上下文模型,计算其置信图为:

$$C_{t+1}(x) = F^{-1}(F(H_{t+1}^{STC}(x)) \odot F(I_{t+1}(x) \omega_\sigma(z - x_t^*))) \quad (9)$$

6) 将 $t+1$ 帧得到的置信图最大点作为目标在 $t+1$ 帧的位置,随后返回第 1) 步继续计算下一帧。将每一帧的置信图的极值点作为目标位置输出。

2 STC-AL 算法详述

2.1 SIFT 特征与匹配

对前 2 帧目标上下文区域进行 SIFT 特征点提取,它在尺度空间中检测极值点,并在特征点处提取其位置、尺度、方向不变量。在以特征点为中心位置处,划分 4×4 共 16 个区域,分别计算 8 个方向直方图,就可提取其 $4 \times 4 \times 8$ 共 128 维的特征向量。图像 SIFT 特征点提取后,从基准图像中查找与该点欧氏距离最近的 2 个特征点,一旦最近距离与次近距离的比值小于给定的阈值,则认为距离最近的点为匹配点,欧式距离公式:

$$d(T, L) = \sqrt{\sum_{i=1}^{128} (t_i - l_i)^2} \quad (10)$$

其中, $T = \{t_i\}_{i=1}^{128}$, $L = \{l_i\}_{i=1}^{128}$ 分别表示 2 张图中的一特征点的特征向量,用 $y = d_i/d_{i-1}$ 表示欧氏距离比值。因为匹配过程中会存在误匹配的情况,本文采取 RANSAC 算法消除误匹配^[16],如图 1 所示。因为 SIFT 匹配算法的耗时长,所以采取最优块优先算法(Best Bin First, BBF)^[17],通过对特征点数据建立 KD 树并优先检索可能性高的最近邻空间,从而加快匹配速度。



图 1 消除误匹配的对比

2.2 自适应窗口调整

本文依据 TLD 算法的窗口调整机制,首先通过对前后两帧的输出窗口进行 SIFT 特征匹配以及消除错误特征点,得到匹配点集合 $U_{t-1} = \{x_{t-1}^1, x_{t-1}^2, \dots, x_{t-1}^n\}$, $U_t = \{x_t^1, x_t^2, \dots, x_t^n\}$ 。其次进行目标窗口尺度变化分析,计算集合 $dx = \{dx_1, dx_2, \dots, dx_n\}$, $dy = \{dy_1, dy_2, \dots, dy_n\}$, dx_i, dy_i 表示匹配特征点 x_{t-1}^i, x_t^i 的横纵偏移量,前后帧的尺度变化参数为:

$$S_i = \frac{dis(x_t^i, x_{t-1}^{i+1})}{dis(x_{t-1}^i, x_{t-1}^{i+1})} \quad (11)$$

其中, $dis(x_t^i, x_{t-1}^{i+1})$ 为第 t 帧第 i 特征点与第 $i+1$ 特征点的欧氏距离, S_i 是前帧目标 2 个匹配点与当前帧匹配的 2 个特征点的距离比例。最后得到集合 $S = \{S_1, S_2, \dots, S_n\}$, 即集合 S 记录了目标所有特征点的欧氏距离比例,最终对这 3 个集合进行排序,用其中位数 m_{Mdx}, m_{Mdy}, m_{MS} 作为当前目标窗口调整参数,调整如下:

$$\begin{aligned} s_1 &= \lambda \times (m_{MS} - \mu) \times w_{t-1} \\ s_2 &= \lambda \times (m_{MS} - \mu) \times h_{t-1} \\ w_t &= w_{t-1} \times m_{MS} \\ h_t &= h_{t-1} \times m_{MS} \\ x_t &= x_{t-1} + m_{Mdx} - s_1 \\ y_t &= y_{t-1} + m_{Mdy} - s_2 \end{aligned} \quad (12)$$

其中, λ, μ 为尺度参数, $w_{t-1}, w_t, h_{t-1}, h_t$ 为第 $t-1$ 帧、第 t 帧目标的矩形框的长宽, $x_{t-1}, y_{t-1}, x_t, y_t$ 为第

$t-1$ 帧、第 t 帧的目标矩形左上角坐标,即经过消除错误特征点后,利用可靠特征点分析目标尺度变化,并输出具体目标矩形 $\{(x_t, y_t), w_t, h_t\}$ 。

2.3 空间上下文模型的学习与更新

在利用式(11)、式(12)自适应窗口调整后,也要对空间模型进行具体窗口确定的上下文区域学习,即目标尺度变大时空间模型要增加对扩展出的区域进行学习,相反,目标尺度变小只要针对缩小后的目标区域进行空间模型学习即可,因此,本文提出目标尺度变化空间模型 h_t^{alSC} , 即自适应上下文空间模型。空间模型的学习改进和更新改进分别为:

$$(1-\ell)h_{t+1}^{SC} \oplus \ell h_{t+1}^{alSC} \quad (13)$$

$$H_{t+1}^{STC} = (1-\rho)H_t^{STC} \oplus \rho((1-\ell)h_{t+1}^{SC} \oplus \ell h_{t+1}^{alSC}) \quad (14)$$

其中, h_{t+1}^{SC} 表示 STC 算法计算得到的空间模型, h_{t+1}^{alSC} 表示利用 SIFT 特征点进行窗口调整后学习得到的空间模型,当目标尺度没有发生大小的变化, $h_{t+1}^{SC} = h_{t+1}^{alSC}$, 式(14)与式(8)计算结果一样,当尺度变大时, h_{t+1}^{SC} 模型小于 h_{t+1}^{alSC} 模型,即计算出来的 2 个空间模型大小不一致,那么进行 $\dots \oplus \dots$ 计算方式,即中心位置对齐后,采取式(13)进行融合学习,具体过程如图 2 所示。这样 h_{t+1}^{alSC} 就包含了扩展出来的目标信息。相反地,当 h_{t+1}^{SC} 模型大于 h_{t+1}^{alSC} 模型,即目标尺度缩小,就要丢弃非目标上下文的信息,只把 h_{t+1}^{alSC} 相对 h_{t+1}^{SC} 的区域进行融合学习即可。目标随着窗口调整的同时空间模型也随之进行自适应学习,然后是空间模型的融合更新。因 h_{t+1}^{SC} 是对 STC 计算出的窗口学习得到的,故 h_{t+1}^{SC} 大小与 H_t^{STC} 大小一致,因此, H_t^{STC} 与利用 $\rho((1-\ell)h_{t+1}^{SC} \oplus \ell h_{t+1}^{alSC})$ 计算出的空间模型同样采取 $\dots \oplus \dots$ 的数据更新。最终式(14)就随着目标尺度变化,空间模型也自适应地进行学习和更新。

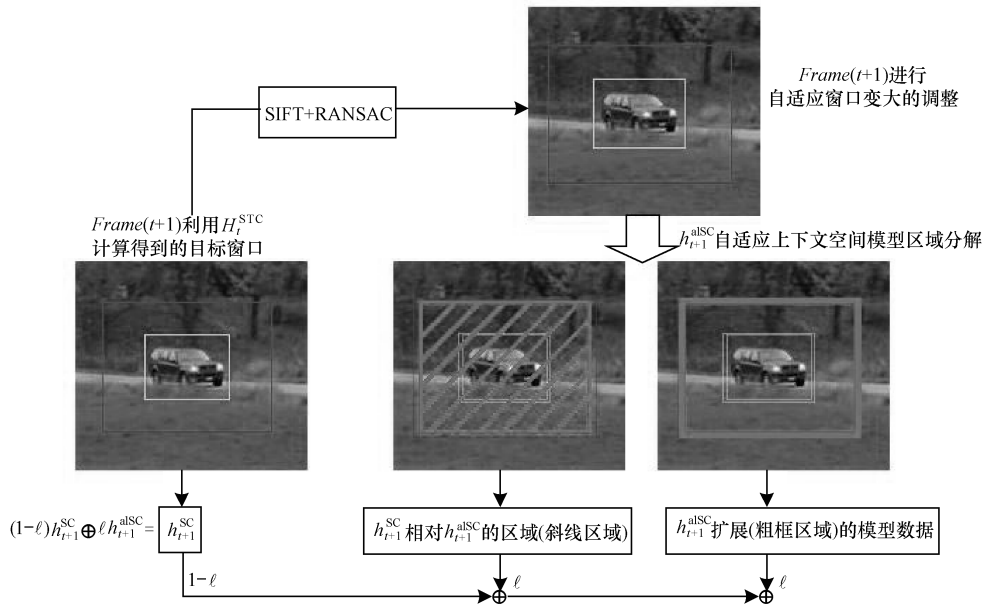


图 2 STC-AL 算法对目标尺度变大的空间上下文模型计算过程

2.4 算法流程

综上所述,自适应窗口的 STC-AL 跟踪流程如图 3 所示。

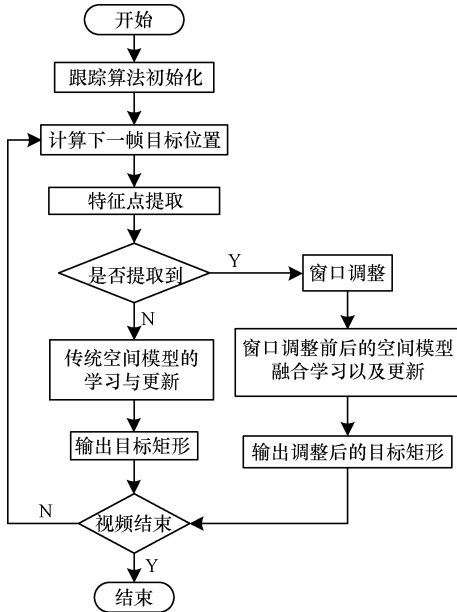


图 3 STC-AL 算法跟踪流程

算法具体步骤如下:

步骤 1 初始化,人为选定目标窗口,并利用 STC 算法输出下一帧目标位置。

步骤 2 加速提取前后目标的 SIFT 特征点,并采用 RANSAC 消除误匹配,若提取不到特征点,即窗口不调整,否则利用匹配点集合进行窗口调整。

步骤 3 对窗口采取第 2.3 节的方法进行空间模型的学习与更新。

步骤 4 计算下一帧目标位置,返回步骤 2。

3 实验结果与分析

为验证 STC-AL 算法的鲁棒性,对 STC 算法、STC-AL 算法进行对比验证,除了 STC 算法外,采用 2 个公开的跟踪算法代码一起参加测试,算法为 CT (Real-Time Compressive Tracking)^[18] 和 KCF (Kernerlized Correlation Filtering)^[19]。实验的视频资源来自文献[20]提到的评测视频库,保证实验的可信度。

本文算法是在 STC 公开的多尺度 C++ 代码上改写的,增加的计算代价主要来自 SIFT 特征点的提取与匹配过程,因此,采取了 k-d 树最优化查找 BBF 方法进行匹配算法的加速,从而保证 STC-AL 算法的实时性。本文的实验环境为:Inter(R) i5-4590 3.30 GHz 处理器、8 GB 内存,软件环境为:Win7 + VS2013 + OpenCV。

3.1 实验参数

在实验过程中,STC-AL 算法的参数设置如下: $\alpha = 2.25, \beta = 1, \rho = 0.075, \lambda = 0.5, \mu = 1, \ell = 0.075$,其他算法均使用了公开源码中的参数。

3.2 实验结果

本文将重点解决目标窗口自适应问题,因此,采取了跟踪重叠率这一评判标准进行评测。跟踪重叠率定义为 $R = \text{Area}(O_s \cap O_g) / \text{Area}(O_s \cup O_g)$,其中, O_s 是跟踪器输出的目标框, O_g 是目标真实大小框, \cap, \cup 表示 2 种框之间的交集与并集, $\text{Area}(\cdot)$ 表示面积大小。 R 的值大于 0.5 表示跟踪的输出是成功的。部分跟踪结果如图 4 ~ 图 6 所示,深灰色框是 STC-AL 算法跟踪结果,浅灰色框是 STC 算法跟踪结果,黑色框是 KCF 算法跟踪结果,白色框是 CT 算法的跟踪结果。

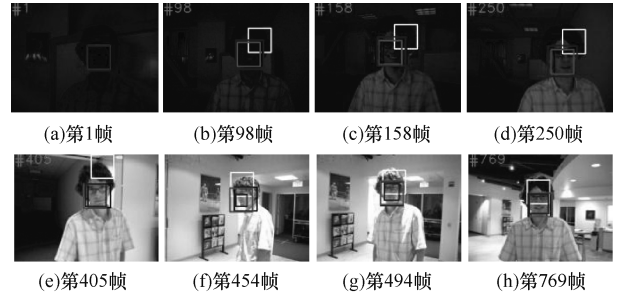


图 4 不同算法对 david 视频序列的跟踪结果

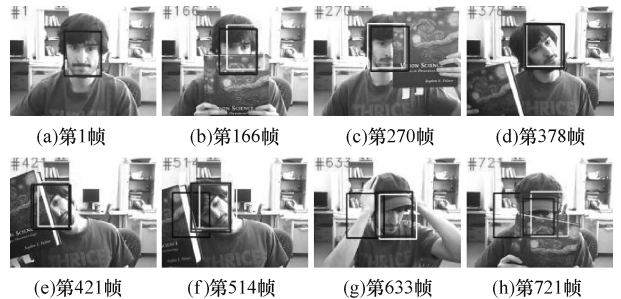


图 5 不同算法对 FaceOcc 视频序列的跟踪结果

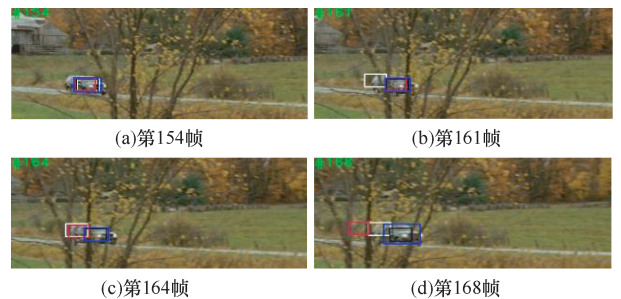


图 6 不同算法对 carScale 视频序列的跟踪结果

图 7 是对 david 视频序列的跟踪重叠率曲线,由图 7 可以看出,在前端时间内,CT 算法以及 KCF 算

法的跟踪重叠率不如 STC 算法和本文的 STC-AL 算法,david 在第 400 帧~第 490 帧这段时间运动特征是远离摄像头的,而在此期间本文算法的输出窗口成功地自适应目标的变化,同时随着目标靠近镜头,也能够调整窗口。在之后的时间内,STC 算法与 STC-AL 算法的重叠率曲线基本一致。可见,本文的自适应学习的跟踪结果有一定优势。

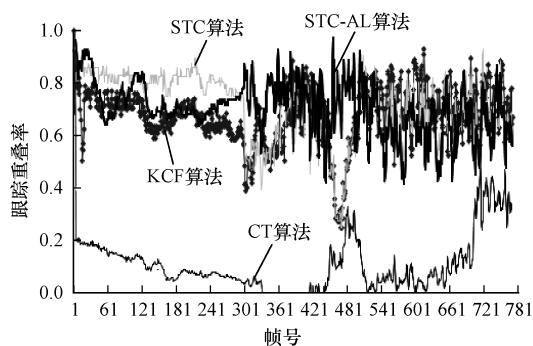


图 7 不同算法对 david 视频序列的跟踪重叠率曲线

图 8 是对 FaceOcc 视频序列的跟踪重叠率曲线图。该视频序列的主要目的是验证遮挡以及旋转是否对算法有影响,从曲线变化可得,KCF 算法在因遮挡导致的跟踪漂移后跟踪率迅速下降,CT 算法的跟踪重叠率就一直不高不低,而 STC 算法与本文算法差距不大,但后期因目标区域被遮挡过大,导致 STC 算法跟踪失败,因此,本文算法的跟踪结果受到影响。故只要 STC 跟踪输出窗口稳定,那么采取本文算法进行窗口调整受遮挡、旋转等干扰影响不大。

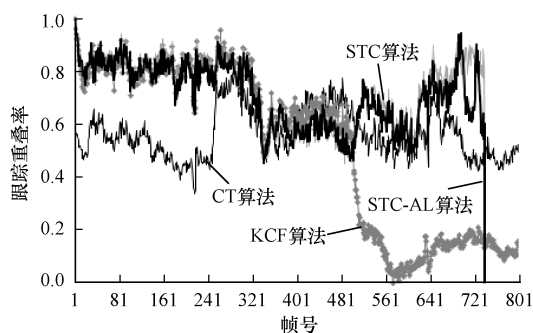


图 8 对 FaceOcc 视频序列的跟踪重叠率曲线

图 9 是 4 个算法对 carScale 视频序列的跟踪重叠率曲线。起初目标的变化不是很明显,KCF 算法、STC 算法以及本文算法差距不大,CT 算法跟踪重叠率其次,但随着目标尺度变化越来越大,STC-AL 算法的自适应能力就得以体现。尤其在第 130 帧~第 180 帧目标被树木遮挡以及接近摄像头的速度也越来越快(图 5 给出了遮挡关键帧的结果图),可以

观察到 STC 跟踪算法发生漂移现象,但是 STC-AL 算法因在前面一直对目标扩展出的信息进行学习,因此在部分遮挡期间仍能准确跟踪。而 KCF 以及 CT 算法虽然还能持续跟踪,但它们的窗口不自适应性导致的跟踪重叠率一直降低。

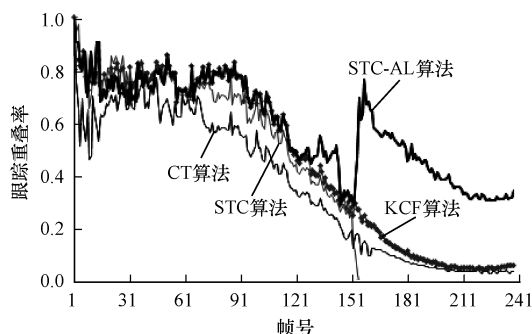


图 9 不同算法对 carScale 视频序列的跟踪重叠率曲线

4 结束语

因目标发生尺度变化时跟踪窗口长期固定化,导致目标信息学习针对性不足。为此,本文提出自适应调整的时空上下文目标跟踪算法,基于 SIFT 尺度不变特征的提取和 RANSAC 算法的消除误匹配,并采取 k-d 树最优化查找 BBF 方法进行匹配算法加速,对匹配点集合进行综合比较分析,判定当前目标的大小变化情况,最后进行具体大小区域的空间模型更新,从而准确学习目标上下文信息。对不同视频序列进行目标跟踪证明了本文算法在一定程度上能够适应目标尺度变化。然而本文算法很依赖传统 STC 算法,一旦发生跟踪漂移甚至跟踪失败,那么本文算法就会受到影响。因此,针对跟踪失败后,目标的再次出现能够重新进行跟踪的设想,是今后需要进一步研究的方向。

参考文献

- [1] BABENKO B, YANG M H, BELONGIE S. Visual tracking with online multiple instance learning [C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2009: 983-990.
- [2] WANG Lijia, JIA Songmin, LI Xiuzhi, et al. Person following for mobile robot using improved multiple instance learning [J]. Acta Automatica Sinica, 2014, 40(12): 2916-2925.
- [3] ZHANG Kaihua, ZHANG Lei, YANG M H. Fast compressive tracking [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(10): 2002-2015.

- [4] ROSS D A, LIM J, LIN R S, et al. Incremental learning for robust visual tracking [J]. International Journal of Computer Vision, 2008, 77(1-3): 125-141.
- [5] KWON J, LEE K M. Visual tracking decomposition [C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2010: 1269-1276.
- [6] LI Zhenxing, LIU Jinmang, LI Song, et al. Group targets tracking algorithm based on box particle filter [J]. Acta Automatica Sinica, 2015, 41(4): 785-798.
- [7] KALAL Z, MIKOLAJCZYK K, MATAS J. Tracking-learning-detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(7): 1409-1422.
- [8] ZHOU Xiuzhuang, LU Yao, LU Jiwen, et al. Abrupt motion tracking via intensively adaptive Markov chain Monte Carlo sampling [J]. IEEE Transactions on Image Processing, 2012, 21(2): 789-801.
- [9] ZHOU Tianfei, LU Yao, DI Huijun. Nearest neighbor field driven stochastic sampling for abrupt motion tracking [C]// Proceedings of International Conference on Multimedia and Expo. Washington D. C., USA: IEEE Press, 2014: 1-6.
- [10] ZHANG Kaihua, ZHANG Lei, LIU Qingshan, et al. Fast visual tracking via dense spatio temporal context learning [C]// Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2014: 127-141.
- [11] DINH T B, VO N, MEDIONI G. Context tracker: exploring supporters and distracters in unconstrained environments [C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2011: 1177-1184.
- [12] XU Jianqiang, LU Yao, LIU Jinwu. Robust visual tracking via weighted spatio-temporal context learning [C]// Proceedings of IEEE International Conference on Image Processing. Washington D. C., USA: IEEE Press, 2015: 413-416.
- [13] XU Yingkun, QIN Lei, LI Guorong, et al. Online discriminative structured output SVM learning for multi-target tracking [J]. IEEE Signal Processing Letters, 2014, 21(2): 190-194.
- [14] 刘万军, 董帅含, 曲海成. 时空上下文抗遮挡视觉跟踪 [J]. 中国图象图形学报, 2016, 21(8): 1057-1067.
- [15] LINDBERGH T. Scale-invariant feature transform [M]. [S. l.]: Betascript Publishing, 2010.
- [16] 常青, 张斌, 邵金玲. 基于 SIFT 和 RANSAC 的特征图像匹配方法 [J]. 华东理工大学学报(自然科学版), 2012, 38(6): 97-101.
- [17] BEIS J S, LOWE D G. Shape indexing using approximate nearest-neighbor search in high-dimensional spaces [C]// Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 1997: 1000-1006.
- [18] ZHANG Kaihua, ZHANG Lei, YANG M H. Real-time compressive tracking [C]// Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2012: 864-877.
- [19] HENRIQUES J F, RUI C, MARTINS P, et al. High-speed tracking with Kernelized correlation filters [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 37(3): 583-596.
- [20] WU Yi, LIM J, YANG M H. Online object tracking: a bench-mark [C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2013: 2411-2418.

编辑 顾逸斐

(上接第293页)

- [9] WANG Q, JIANG J. Comparative examination on architecture and protocol of industrial wireless sensor network standards [J]. IEEE Communications Surveys and Tutorials, 2016, 18(3): 2197-2219.
- [10] GUGLIELMO D D, ANASTASI G, SEGHELLI A. From IEEE 802.15.4 to IEEE802.15.4e: a step towards the internet of things [M]. Cham, Switzerland: Springer International Publishing, 2014.
- [11] DOMENICO D G, BESHRA N, SIMON D, et al. Analysis and experimental evaluation of IEEE802.15.4e TSCH CSMA-CA algorithm [J]. IEEE Transactions on Vehicular Technology, 2017, 66(2): 1573-1588.
- [12] 王剑, 谢闯, 段茂强, 等. 第四十三讲 符合 WIA-PA 标准的工业无线芯片关键技术研究 [J]. 仪器仪表标准化与计量, 2014(1): 18-23.
- [13] 杨志家, 王宏, 周炯. 第二十五章 面向工业测控应用的无线网络协议 SOC 解决方案 [J]. 仪器仪表标准化与计量, 2011(1): 28-30.
- [14] 谢书珊, 吴宁, 陈鑫. 无线传感网节点 SoC 芯片设计研究 [J]. 现代雷达, 2014(10): 74-78, 82.
- [15] WATTEYNE T, VILAJOSANA X, KERKEZ B, et al. OpenWSN: a standards-based low-power wireless development environment [J]. Transactions on Emerging Telecommunications Technologies, 2012, 23(5): 480-493.
- [16] 孙婷婷. 基于 IEEE 802.15.4e 及节点空间分布的无线传感器网络研究 [D]. 西安: 西安电子科技大学, 2014.
- [17] 闵明慧, 杨志家, 李中胜, 等. 工业物联网应用中多时隙帧调度算法研究 [J]. 计算机工程, 2016, 42(11): 15-21, 26.
- [18] CHANG T F, WATTEYNE T, PISTER K, et al. Adaptive synchronization in multi-hop TSCH networks [J]. Computer Networks, 2015, 76(2): 165-176.
- [19] 梁英, 于海斌, 曾鹏. 无线传感器路由协议 [J]. 信息与控制, 2005, 34(6): 325-330.

编辑 金胡考