

结合孪生网络重检的长期目标跟踪算法^①



王 林, 郑有玲

(西安理工大学 自动化与信息工程学院, 西安 710048)

通信作者: 郑有玲, E-mail: 979045393@qq.com

摘 要: 针对传统长期相关滤波器使用特征单一、跟踪失败后无法再次捕捉到目标的缺点, 提出一种结合深度学习的多特征融合长期目标跟踪算法. 本算法在长期相关跟踪算法 (long-term correlation tracking, LCT) 的基础上, 采用多特征融合的方式, 将局部二值模式特征、改进的方向梯度直方图特征以及颜色特征相融合, 来提高跟踪算法的鲁棒性. 由于 LCT 算法选择随机蕨分类器进行目标重检, 对检测范围有局限性且重检精度较低, 故采用基于深度学习的孪生网络实例搜索 (SINT) 方法对全局图像进行目标重检. 本文的实验在 OTB100 数据集上进行, 结果表明: 本文算法与 LCT 算法相比, 距离精度和成功率分别提升了 13% 和 10.3%.

关键词: 长期目标跟踪; 相关滤波; 特征融合; 深度学习; 目标重检; 特征融合

引用格式: 王林, 郑有玲. 结合孪生网络重检的长期目标跟踪算法. 计算机系统应用, 2022, 31(4): 188-195. <http://www.c-s-a.org.cn/1003-3254/8425.html>

Long-term Target Tracking Algorithm Based on Twin Network Rechecking

WANG Lin, ZHENG You-Ling

(Faculty of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China)

Abstract: The traditional long-term correlation filter uses a single feature and cannot capture the target again after tracking failure. Considering this, the study proposes a multi-feature fusion long-term target tracking algorithm combined with deep learning. On the basis of the long-term correlation tracking (LTC) algorithm, the proposed algorithm uses multi-feature fusion to join together the local binary pattern feature, the improved directional gradient histogram feature, and the color feature to promote the robustness of the tracking algorithm. Since the LCT algorithm adopts a random fern classifier to recheck the target, which has a limited detection range and low rechecking accuracy, the deep learning-based twin network instance search (SINT) method is employed to recheck the global image. The experiment in this study is carried out on the OTB100 dataset, and the results show that compared with the LCT algorithm, the proposed algorithm has improved the range accuracy and the success rate by 13% and 10.3% respectively.

Key words: long-term object tracking; correlation filter; feature fusion; deep learning; target reinspection; feature fusion

目标跟踪是计算机视觉领域的一个主流方向^[1], 而长期目标跟踪因更接近实际应用而受到越来越多的关注^[2]. 但对于长期目标跟踪算法中存在的形变、运动模糊、遮挡及目标短暂出视野等问题, 所以, 建立一个同时具有鲁棒性和良好运行速度的长期目标跟踪系统是

有必要的.

相关性, 通常是指两种事物中存在的联系. 对于信号处理中的函数来说, 相关即判断两个函数在某时刻的相似程度. 在目标跟踪中, 相关滤波方法的本质就是在运动模型预测的下一帧目标位置中找到与给定目标

^① 基金项目: 陕西省科技计划重点项目 (2017ZDCXL-GY-05-03)

收稿时间: 2021-06-30; 修改时间: 2021-07-30; 采用时间: 2021-08-12; csa 在线出版时间: 2022-03-22

产生最大响应的区域来作为下一帧中的目标位置. 相关滤波^[3]由于其性能优越, 研究者一直对其进行研究. 2010年, 文献[4]首次将相关滤波用于目标跟踪, MOSSE算法也是“相关”思想在目标跟踪中应用的开端, 该算法的目的就是寻找一个能在目标位置响应最大的滤波器, 且使用傅里叶变换将相关运算转换到频域, 很大程度上提高了算法的运行速度. 文献[5]提出了CSK算法, 利用循环矩阵的特性达到了密集采样的目的, 增加了样本的可分性. 随后, 文献[6]提出KCF (kernelized correlation filter) 算法, 它拓宽了已有算法所使用的特征范围, 选择了对方向敏感的方向梯度直方图特征, 提高了算法的精度. 同时文献[7]提出一种利用颜色特征的目标跟踪算法, 也获得不错的效果. 随着深度学习的发展, 文献[8]提出一种利用高级语义特征的ECO算法, 该算法同时压缩了训练样本空间, 在提升算法性能的同时加快了算法运行速度. 文献[9]提出的算法根据不同的应用场景采用不同的特征选择, 使算法可以应对更多干扰因素.

在目标跟踪的过程中, 目标会因严重遮挡和出视野造成目标丢失, 为解决这一问题, 2011年, 文献[10]提出TLD算法, 是一种将目标检测和目标跟踪相结合的跟踪器, 实现在目标丢失后重新出现在视野时, 检测器可以再次捕捉到目标. 由于其使用特征点跟踪的模式, 跟踪的进度较低. 为提高精度, 文献[11]提出了LCT算法, 一种基于相关滤波和随机森林分类器结合的长时目标跟踪器, 有效提高了算法的精度, 但检测范围存在局限性.

综上, 针对长期目标跟踪算法LCT出现的特征使用单一和重检分类器的局限性问题, 本文在将目标跟踪任务分解为目标的平移和尺度估计的基础上, 融合了改进的方向梯度直方图特征、颜色特征和局部二值模式特征, 提高跟踪器的鲁棒性; 用孪生网络实例搜索替代了LCT中用于目标重检的森林分类器, 解决了森林分类器存在的检测范围局限性的问题.

1 长期相关滤波算法

长期相关滤波算法 (long-term correlation tracking, LCT) 是基于相关滤波器的跟踪算法, 它从单一框架中学习了两个回归模型. 一个是时间上下文回归器, 利用目标和周围时间相关性, 积极适应平移估计来解决重大变形和遮挡问题; 另一个是目标回归器, 仅模拟目标

外观, 并建立外观金字塔以进行尺度估计. 目标回归器同时也作为判别回归模型, 用最大响应值评估跟踪结果置信度, 并使用阈值 τ_t 来控制检测器是否被激活. 选择随机森林分类器作为在目标跟踪失败时的重检测器.

1.1 核相关滤波

LCT跟踪器将跟踪任务分解成位置估计和尺度估算两个子任务. 分别由两个相关滤波器 R_c 和 R_t 来实现.

相关滤波器的获得方式, 是利用循环矩阵的性质对大小为 $M \times N$ 个像素的图像进行密集采样得到更多的循环样本即 $x_{m,n}(m,n)$, 训练岭回归分类器如式(1):

$$\min \sum_{m,n} \|\varphi(x_{m,n}) \cdot \omega - y(m,n)\|^2 + \lambda \|\omega\|^2 \quad (1)$$

其中, $\varphi(x_{m,n})$ 是核空间映射函数; ω 为所求滤波器; λ 为正则化项参数.

最终计算求得 ω 的具体表达式为 $\omega = \sum_{m,n} \alpha(m,n) \varphi(x_{m,n})$, 对系数 α 做傅里叶变换如式(2):

$$A = F(a) = \frac{F(y)}{F(\varphi(x) \cdot \varphi(x))} \quad (2)$$

在新的图像帧中, 需要对每一个预估的目标框计算相应的输出值, 计算方式为:

$$\hat{y} = F^{-1}(A \odot F(\varphi(z) \cdot \varphi(\hat{x}))) \quad (3)$$

所得响应的最大值被认作新一帧中感兴趣区域的位置.

1.2 更新机制与目标重检

在更新机制和目标重检测方面, LCT算法均使用设定阈值的方法来控制是否进行模型更新和目标重检.

当滤波器的响应值 y_t 大于阈值 τ_a 时, 更新滤波器 R_t , 模型更新方式如式(4)、式(5):

$$\hat{x}^t = (1 - \alpha)\hat{x}^{t-1} + \alpha x^t \quad (4)$$

$$\hat{A}^t = (1 - \alpha)\hat{A}^{t-1} + \alpha \quad (5)$$

其中, t 代表当前帧, α 是学习速率.

当目标发生遮挡或出视野时, 算法会出现跟丢目标的情况. 这时为了保持跟踪器模型的准确性, 要停止目标模型更新. 为此, LCT算法通过判定跟踪框响应值与规定阈值大小来判定是否更新模型.

跟踪器在新一帧所得目标框的响应值记为该帧的置信度, 研究者经多次实验选定置信度阈值, 低于该值, 证明所得目标错误率较高, 需要再次检测. 此时暂停模型更新, 开始进行目标重检测. 选择将高置信度的每个

跟踪结果调整为 15×15 大小, 以形成灰度特征向量来训练随机森林分类器. 同时, 将重检器获得的检测框位置作为这一帧图像确定的目标位置, 并继续进行跟踪.

2 优化的 LCT 算法

2.1 更新机制与目标重检

选择合适的特征有助于跟踪器更好的学习目标的外观模型. HOG 特征, 对几何方向敏感而对颜色变化不敏感. 为了得到一个性能优良的跟踪器, 本文将多种特征融合在一起, 即使用局部二值模式 (LBP) 特征、

改进的方向梯度直方图 (FHOG) 特征以及颜色 (CN) 特征进行融合.

2.1.1 FHOG 特征

FHOG 特征由 Felzenszwalb 等人^[12] 提出, 它在 HOG (histogram of oriented gradient) 特征的基础上利用每个 cell 的 4 个邻域进行归一化和截断处理, 由于方向敏感的划分方式和不敏感划分方式在不同目标类别上表现效果各异, 故选择对方向敏感的 (18 维) 特征和对方向不敏感的 (9 维) 特征, 但特征维度太大对特征在行列两方向上累加, 进行特征降维提取方式如图 1 所示.

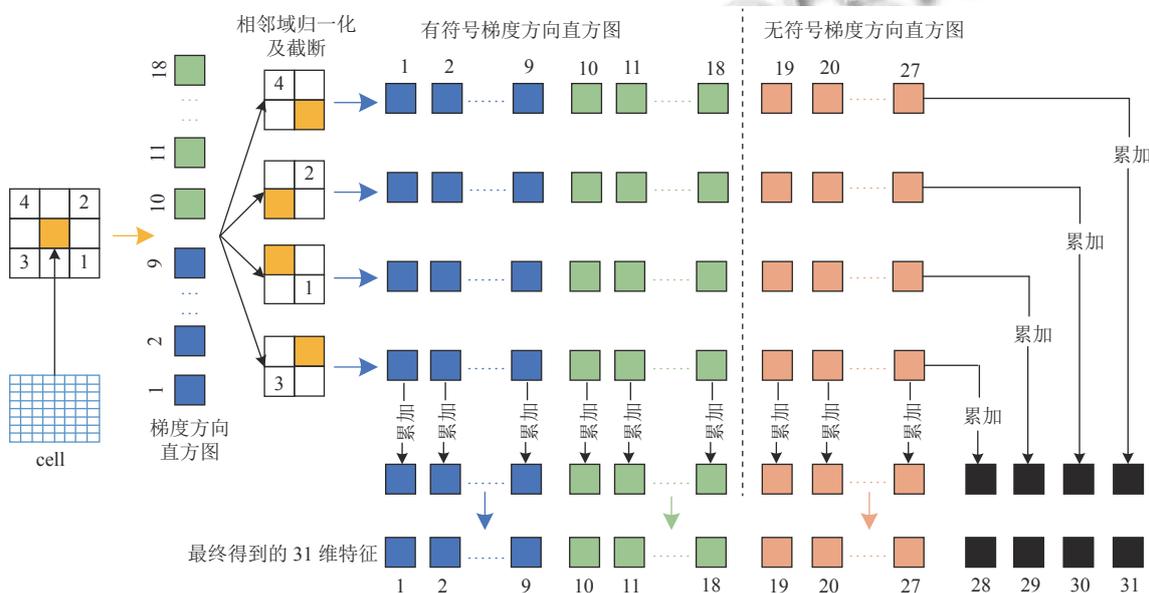


图 1 FHOG 特征提取原理图

2.1.2 LBP 特征

LBP (local binary pattern) 是局部二值模式, 体现图像的局部纹理特征, 可以有效对抗旋转和灰度的变化. 提取方式为: 选中图像中的某一像素并将其作为阈值, 以其为中心建立 3×3 的窗口, 让每一小格像素值分别与阈值比较. 若此值大于或等于阈值, 则在 LBP 编码图中, 对应位置记为 1, 反之记为 0. 对产生 8 位二进制数十进制化, 得到 LBP 编码值. LBP 特征是对目标外观模型的纹理特征的体现, 反映物体表面的变化. 它的提取示意图如图 2 所示.

不涉及目标空间位置的情景. 相比其他特征, 颜色特征以低复杂度获得研究者的青睐. 该特征将传统的 RGB 空间的特征映射为 11 维度的 CN 特征, 在面对光照变化的干扰因素时可以较好地应对和处理, 从而保证跟踪效果的准确性.

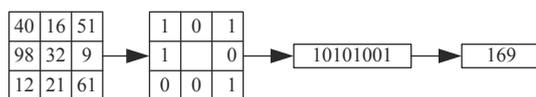


图 2 LBP 特征提取原理图

2.1.3 CN 特征

CN (color names) 特征是图像重要的特征之一, 但颜色特征对一些形变不敏感, 在一些判别性场景中应用广泛. CN 颜色特征应对目标形变效果不佳, 适用于

2.1.4 特征融合策略

对当前图像序列采用 FHOG 特征、LBP 特征、CN 特征 3 种特征根据各自权重实现融合. 首先用公式

$$\omega = \sum_{m,n} \alpha(m,n)\varphi(x_{m,n})$$

它们为 $\hat{\omega}_{hog}$ 、 $\hat{\omega}_{lbp}$ 、 $\hat{\omega}_{cn}$ 。在跟踪的过程中,分别由式(6)–式(8)计算3个表现模型与新一帧图像的相关响应图。

$$f_{hog}(z) = F^{-1}(\hat{\omega}_{hog} \odot \hat{z}) \quad (6)$$

$$f_{lbp}(z) = F^{-1}(\hat{\omega}_{lbp} \odot \hat{z}) \quad (7)$$

$$f_{cn}(z) = F^{-1}(\hat{\omega}_{cn} \odot \hat{z}) \quad (8)$$

其中, $\hat{\omega}$ 为特征的表现模型, \hat{z} 为当前图像的特征向量的傅里叶变换, $f(z)$ 为得到的响应图。由式(9)–式(11)计算各相关响应图的最大值,并将其作为特征融和的权值依据。

$$\alpha = \frac{\max(f_{hog}(z))}{\max(f_{hog}(z)) + \max(f_{lbp}(z)) + \max(f_{cn}(z))} \quad (9)$$

$$\beta = \frac{\max(f_{lbp}(z))}{\max(f_{hog}(z)) + \max(f_{lbp}(z)) + \max(f_{cn}(z))} \quad (10)$$

$$\gamma = \frac{\max(f_{cn}(z))}{\max(f_{hog}(z)) + \max(f_{lbp}(z)) + \max(f_{cn}(z))} \quad (11)$$

α 、 β 、 γ 分别为FHOG特征、LBP特征、CN特征的融合权重。最后,根据所得的融合权重得到最终的响应如式(12):

$$f(z) = \alpha f_{hog}(z) + \beta f_{lbp}(z) + \gamma f_{cn}(z) \quad (12)$$

2.2 孪生网络重检机制

本文提出使用基于深度学习的重检测器。用全局孪生网络实例搜索 SINT 来改善随机森林分类器对目标重检测范围的局限性,从而提高重检测器的跟踪性能。

SINT 网络模型如图 3 所示,左右两边网络结构相同。左边网络存储目标模板信息,右边网络用于目标区域特征提取,孪生网络的工作核心是学习一个匹配函数,能够将每幅图像对应特征空间中的一个特征向量。训练时,度量距离即向量差异,最终,通过拟合此距离来表示与输入图像的相似度差异。

当 SINT 在外部视频中学习一个用于跟踪的通用匹配函数,可以稳健地处理一个对象在视频序列中可能经历的常见外观变化。学习到的函数可以不需要任何调整,直接应用到新的跟踪视频中。该视频中的目标对象是之前没有看到过的对象。故本文利用 ALOV 数据集中训练好的 SINT 网络模型参数。当重检模块被激活时, SINT 简单的将第一帧中初始目标区域与新帧中的候选目标区域进行匹配,并通过匹配函数,返回最相似的区域。候选框的选择采用半径采样策略,即围绕前一帧的预测位置,在不同半径的圆上均匀采样位置,搜

索半径设置为第一帧中初始框的较长轴。将重检响应值与阈值 τ_r 比较,若大于阈值则确定目标位置;若小于阈值则在更大的搜索区域内采用滑动窗口生成目标候选区域 $\{c_i\}_{i=1}^N$,最终检测结果如式(13):

$$\hat{c} = \arg \max_{c_i} v(x_{obj}, c_i), \quad i = 1, 2, \dots, N \quad (13)$$

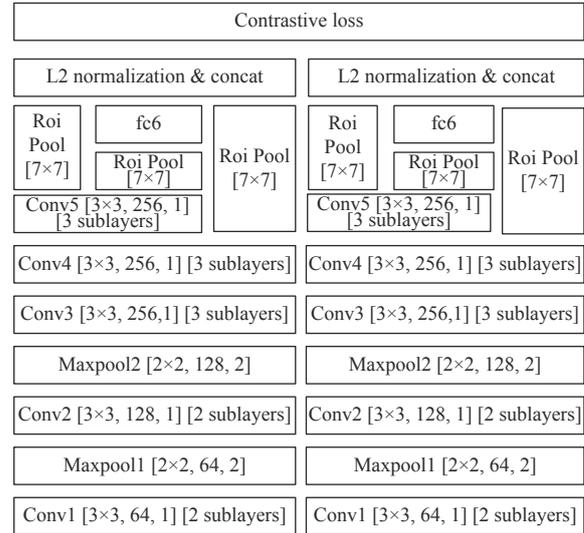


图3 SINT模型的整体结构

2.3 改进的算法流程

改进的算法流程图如图 4 所示,具体步骤描述如算法 1。

算法 1. 改进算法

输入: 初始目标中心点位置 pos_1 与尺度 s_1 。

输出: 输出目标估计状态中心点位置 pos_t 与尺度 s_t , 两个滤波器模型 R_c 和 R_r 。

- Step 1. 根据 pos_1 和 s_1 , 从第 1 帧中裁剪搜索窗口并提取 3 种特征。
- Step 2. 用 R_c 和相关滤波分数计算相关响应图 y_t , 估计新的目标中心点位置 pos_t 。
- Step 3. 在新的目标中心点位置 pos_t 周围构建目标金字塔, 然后用滤波器 R_r 和式 $s_t = \arg \max(\max(y_1), \max(y_2), \dots, \max(y_s))$ 估计最优尺度 s_t 。
- Step 4. 使用最优尺度下的响应值来确定跟踪结果的置信度, 若 $\max(y_s) > \tau_r$ 执行 Step 6 否则激活重检模块。
- Step 5. 使用孪生网络实例搜索 SINT 搜索目标的新位置, 若新位置 pos_r 的响应值大于 τ_r , 则 $pos_t = pos_r$ 。
- Step 6. 若 $\max(y_t) > \tau_a$, 则使用式(4)、式(5)更新模型。
- Step 7. 若序列结束, 则停止算法, 否则重复 Step 3~Step 6。

3 实验结果与分析

3.1 实验环境及配置

本文所提算法的核相关滤波器的参数设置和 LCT

算法设置一致. 对于重检器孪生网络, 训练时, 总共从 ALOV 数据集中采样了 60 000 对帧, 使用 ImageNet 预先训练好的网络参数 Siamese 网络进行微调. 初

始设定学习率为 0.001、权重衰减为 0.001. 学习率每隔两个周期下降 10 倍. 当验证损失不再减少时, 停止调优.

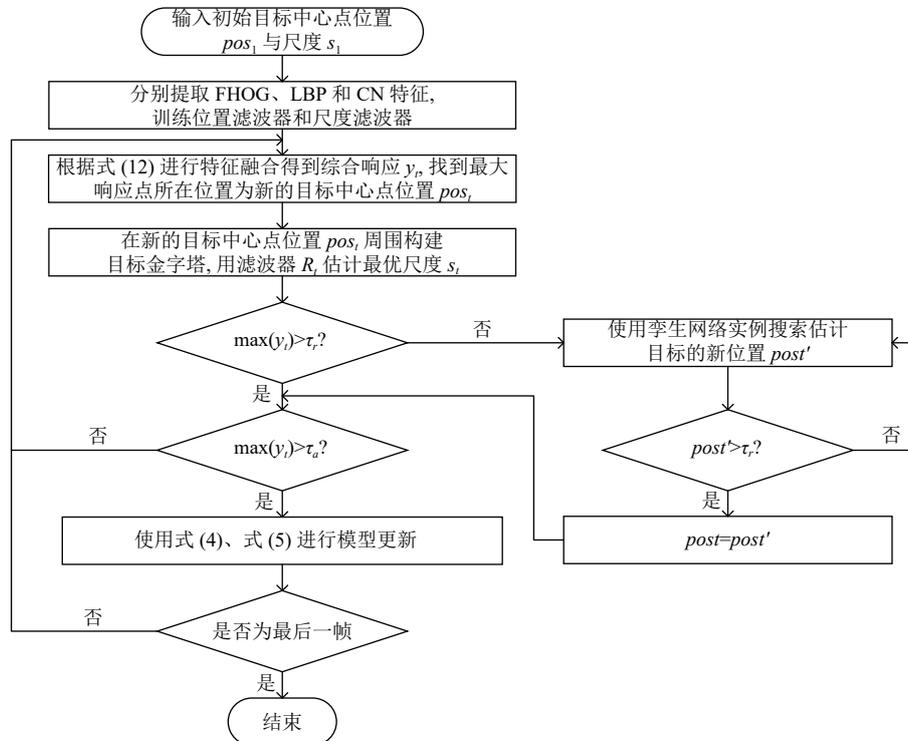


图 4 改进的算法流程图

算法的实验代码在 LCT 的开源的 Matlab 代码上进行改进. 将重检测模块用训练好的孪生网络实例搜索代替原有算法中的随机蕨分类器, 使用的软件平台为 Matlab 2020a. 硬件配置为: NVIDIA GTX 1070 GPU, 8 GB 内存, 使用 Caffe 深度学习框架.

3.2 评估指标

本文选择重叠成功率 (overlap success rate, OSR), 距离精度 (distance precision, DP) 和选择帧率 (frame per second, FPS) 评测算法且本文都选择采用一次性评估方式, 即只在视频序列第一帧进行初始化.

重叠成功率是指给定的标注框与算法得到的实际框之间的重叠率大于阈值的帧数与总帧数的比. 本文阈值选择 0.5. 重叠得分定义如式 (14):

$$S = \frac{|B_t \cap B_{gt}|}{|B_t \cup B_{gt}|} \quad (14)$$

其中, B_t 表示实际跟踪得到的框, B_{gt} 表示标注框, $| \cdot |$ 表示区域内的像素数.

中心位置误差 CLE 指的是实际框中心 (x_t, y_t) 与

标注框中心 (x_{gt}, y_{gt}) 的欧氏距离.

$$CLE = \sqrt{(x_t - x_{gt})^2 + (y_t - y_{gt})^2} \quad (15)$$

距离精度是指中心位置误差小于设定阈值的视频帧数占总帧数的比例. 本文阈值为 20 pixels.

使用帧率来评估算法的运行速度.

3.3 定性分析

由于 OTB100 公开数据集涵盖了大量的具有挑战性情况的视频集, 包含遮挡、形变、运动模糊、光照变化、目标出视野等. 为了与其他算法进行对比, 将本文与 4 种跟踪算法进行对比, 它们分别是 LCT、LCT+、KCF、ECO. 表 1 为实验所得数据, 图 5 为 5 种算法在 OTB100 数据集上得到的精度曲线图和成功率曲线图. 由表 1 可知, 本文算法精确度为 80.7%, 重叠成功率为 71.6%, 运行速率为 16.5 fps, 与原始算法对比, 本文算法的距离精度和重叠成功率分别提高了 13% 和 10.3%. 相较于深度学习的目标跟踪算法, 本文算法在

与其性能相当的基础上,运行速度为 16.5 fps 可以实现实时跟踪.图 6、图 7、图 8 分别为遮挡、尺度变化和运动模糊情况下的结果对比图.由对比得出,与主流算法相比,本文提出算法的抗干扰能力更强,也更具鲁棒性.

3.4 定量分析

从实验视频中选取 5 个跟踪成功的视频对本文算法与主流算法进行定性分析.所选视频序列存在遮挡(OCC)、尺度变化(SV)、形变(DEF)、运动模糊(MB)等不同的挑战.跟踪结果如图 9 所示.

表 1 在 OTB100 数据集上 5 种跟踪算法对比

数据集	指标	本文	LCT	LCT+	KCF	ECO
OTB100遮挡数据集 (48)	DPR (%)	78.6	69.7	72.3	63.2	70.4
	OSR (%)	69.3	52.8	60.5	48.5	63.2
	Speed (fps)	17.2	29.5	26.4	221.2	10.5
OTB100尺度变化数据集 (63)	DPR (%)	80.6	81.3	72.6	63.3	73.5
	OSR (%)	72.3	71.7	70.2	43.6	69.7
	Speed (fps)	15.4	28.9	17.7	247.1	7.5
OTB100运动模糊数据集 (29)	DPR (%)	82.9	52.1	65.4	63.3	70.6
	OSR (%)	73.2	59.4	64.8	43.5	57.9
	Speed (fps)	16.9	23.5	17.4	240	10.2
平均值	DPR (%)	80.7	67.7	70.1	69.6	71.5
	OSR (%)	71.6	61.3	65.1	45.2	63.6
	Speed (fps)	16.5	27.3	20.5	236.1	9.4

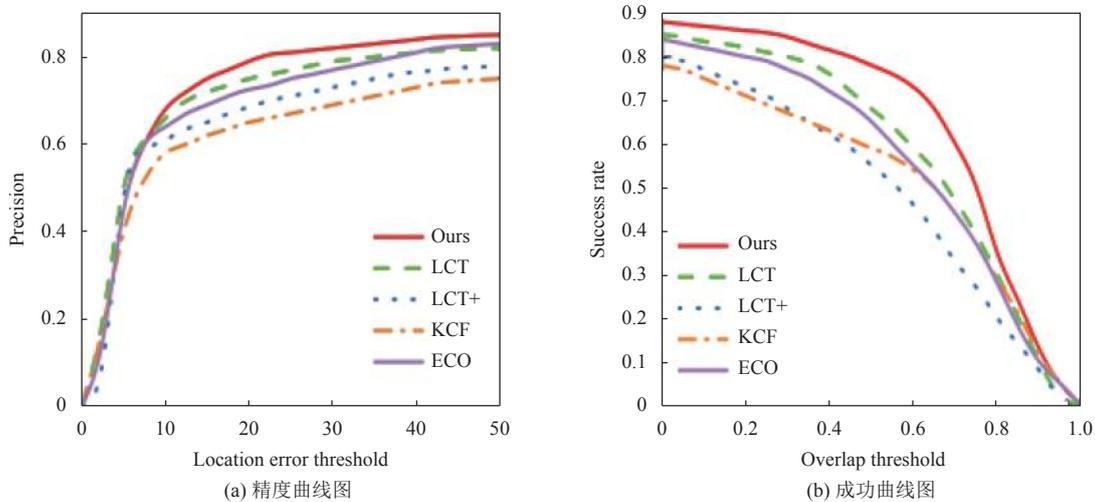


图 5 OTB100 数据集的结果对比

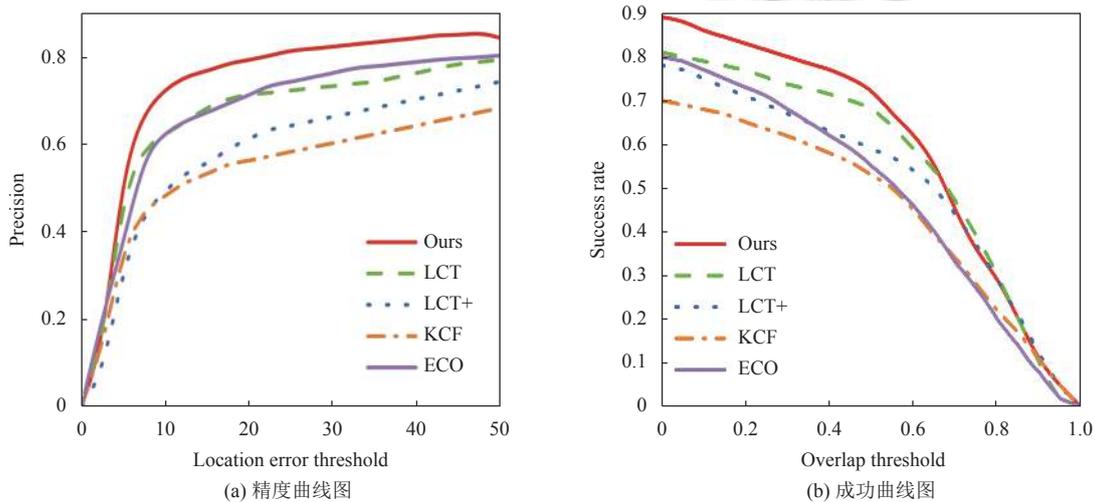


图 6 OTB100 的遮挡数据集结果对比

如图 9 所示,在图 9(a) Jogging 和图 9(e) Learning 视频序列中,所有算法在目标没有被遮挡时,都能稳定的

跟踪,图 9(a) 序列在 53 帧及图 9(e) 序列 262 帧、334 帧在目标发生严重遮挡时,本文算法使用孪生网络重检

器, 目标重现时, 可再次捕捉目标并继续跟踪. 而 KCF 算法和 ECO 算法在目标被严重遮挡后, 无法重新跟踪目标. 在图 9(b) Bird2 中, 目标多次发生平面内旋转、平面外旋转、形变、运动模糊等情况, 51 帧目标发生平面内旋转之后, 发生了形变, 我们的算法可以继续跟踪目标而 LCT 算法和 KCF 出现了跟踪失败的情况, LCT+ 算法和 ECO 算法的跟踪框也发生了偏离. 第 91 帧到

99 帧, 目标由于平面内旋转再次发生形变, 而此时多个算法出现跟丢的情况, 只有本文算法可以稳定跟踪目标. 在图 9(c) BlurCar1 中, 第 786 帧出现严重的运动模糊状况, 跟踪算法都能继续跟踪目标, 而本文算法能更准确地跟踪目标. 本文算法在目标发生严重遮挡时, 由于目标周围的上下文仍可能保持不变, 所以时间上下文回归器 R_c 可以很好地应对目标严重遮挡, 依然稳定地跟踪目标.

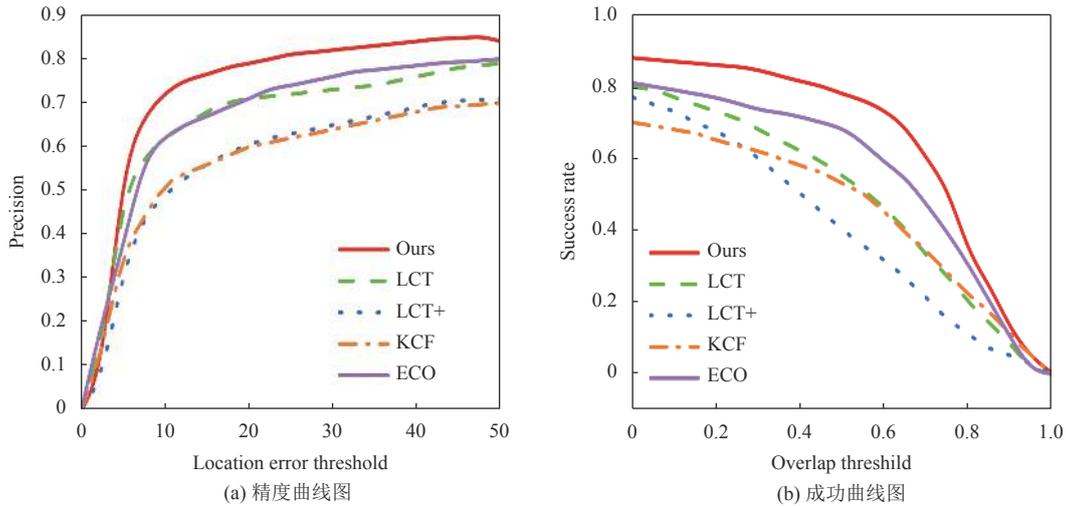


图 7 OTB100 的尺度变化数据集结果对比

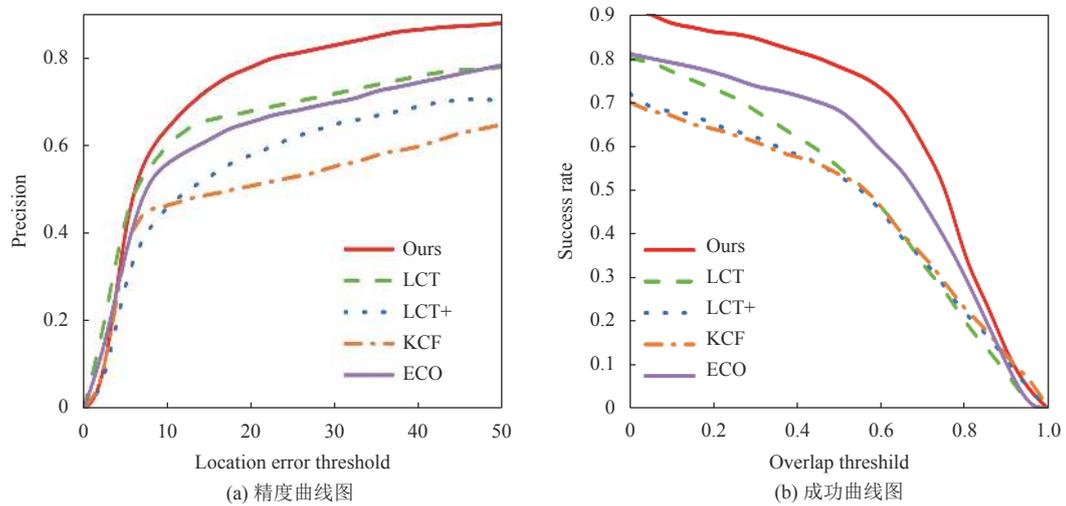


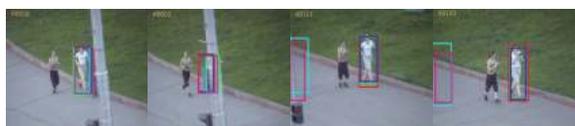
图 8 OTB100 的运动模糊数据集结果对比

4 结论

针对长期相关滤波跟踪算法特征单一、重检精度低, 提出了一种结合深度学习的多特征融合的长期目标跟踪算法. 在 LCT 算法的基础上, 使用 FHOG 特征、LBP 特征、CN 特征来训练位置滤波器 R_c 和尺度

滤波器 R_r , 当目标发生遮挡或出视野时, 采用选定阈值判断是否激活孪生网络重检器. 在目标位置构建目标金字塔及尺度滤波器 R_r 判断目标尺度并与相应阈值来判断是否更新目标模型. 本文的算法在 OTB100 数据集上进行了实验, 得到的结果表明, 与 LCT 算法相比,

本文算法在跟踪精度和覆盖成功率上都有所提高;和一些相关算法对比,本文算法的跟踪性能有所提升.但在运行速率上有待提高.



(a) Jogging



(b) Bird2



(c) BlurCar1



(d) Learning



(e) Coke

— Ours — LCT — LCT+ — KCF — ECO

图9 不同算法在5个视频序列上的跟踪效果

参考文献

- 1 黄凯奇, 陈晓棠, 康运锋, 等. 智能视频监控技术综述. 计算机学报, 2015, 38(6): 1093–1118. [doi: [10.11897/SP.J.1016.2015.01093](https://doi.org/10.11897/SP.J.1016.2015.01093)]
- 2 尹宏鹏, 陈波, 柴毅, 等. 基于视觉的目标检测与跟踪综述.

自动化学报, 2016, 42(10): 1466–1489.

- 3 Chen Z, Hong ZB, Tao DC. An experimental survey on correlation filter-based tracking. *Computer Science*, 2015, 53(6025): 68–83.
- 4 Bolme DS, Beveridge JR, Draper BA, *et al.* Visual object tracking using adaptive correlation filters. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Francisco: IEEE, 2010. 2544–2550.
- 5 Henriques JF, Caseiro R, Martins P, *et al.* Exploiting the circulant structure of tracking-by-detection with kernels. *Proceedings of the 12th European Conference on Computer Vision*. Florence: Springer, 2012. 702–715.
- 6 Henriques JF, Caseiro R, Martins P, *et al.* High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(3): 583–596. [doi: [10.1109/TPAMI.2014.2345390](https://doi.org/10.1109/TPAMI.2014.2345390)]
- 7 Danelljan M, Khan FS, Felsberg M, *et al.* Adaptive color attributes for real-time visual tracking. 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014. 1090–1097.
- 8 Danelljan M, Bhat G, Khan FS, *et al.* ECO: Efficient convolution operators for tracking. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu: IEEE, 2017. 6931–6939.
- 9 沈秋, 严小乐, 刘霖枫, 等. 基于自适应特征选择的多尺度相关滤波跟踪. 光学学报, 2017, 37(5): 0515001.
- 10 Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(7): 1409–1422. [doi: [10.1109/TPAMI.2011.239](https://doi.org/10.1109/TPAMI.2011.239)]
- 11 Ma C, Yang XK, Zhang CY, *et al.* Long-term correlation tracking. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston: IEEE, 2015. 5388–5396.
- 12 Felzenszwalb PF, Girshick RB, McAllester D, *et al.* Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, 32(9): 1627–1645. [doi: [10.1109/TPAMI.2009.167](https://doi.org/10.1109/TPAMI.2009.167)]