

融合 GMS 检测和置信度判别的 TLD 目标跟踪

郭伟, 杨琛*, 曲海成, 邢宇哲

辽宁工程技术大学软件学院, 辽宁 葫芦岛 125105

摘要 针对目标在遮挡、尺度变化等复杂场景下易产生模型漂移问题, 基于跟踪学习检测(TLD)框架提出一种结合基于网格的运动统计(GMS)检测和置信度判别的长时目标跟踪算法。首先在跟踪模块中采用快速判别尺度空间的相关滤波器(fDSST)作为跟踪器, 利用位置滤波器和尺度滤波器对上一帧目标进行位置与尺度的判别, 并依据 TLD 算法中跟踪模块与检测模块的独立性, 将跟踪模块结果输入检测模块中, 采用平均峰值相关能量(APCE)对模板更新进行置信度判别。在检测模块中先引入 GMS 网格运动统计作为检测器, 使具有快速旋转不变性特征的 ORB(Oriented FAST and Rotated BRIEF)算法对上一帧目标进行特征匹配, 再利用网格运动统计对匹配结果进行过滤, 实现目标位置的粗定位, 依据预测位置对目标检测区域进行适当的动态缩减, 最后使用级联分类器对目标进行精准定位。结果表明, 本文提出的跟踪方法在有效防止模型漂移的情况下, 大大提高了算法的跟踪速度, 同时对目标遮挡、尺度变化及旋转等挑战环境也具有较好的准确性和鲁棒性。

关键词 图像处理; 目标跟踪; 模型漂移; 运动统计; 动态缩减

中图分类号 TP399

文献标志码 A

doi: 10.3788/LOP202158.0410009

TLD Target Tracking Fused with GMS Detection and Confidence Discrimination

Guo Wei, Yang Chen*, Qu Haicheng, Xing Yuzhe

School of Software, Liaoning Technical University, Huludao, Liaoning 125105, China

Abstract In order to solve the problem of model drift in complex scenes such as occlusion and scale variation, this paper proposes a long-term target tracking algorithm based on TLD framework, which integrates GMS detection and confidence discrimination. First, in tracking module, the fast discriminating scale space correlation filter (fDSST) is used as the tracker, and the position filter and scale filter are used to distinguish the position and scale of the target in the previous frame. According to the independence of the tracking module and the detection module in the TLD algorithm, the results of the tracking module are input into the detection module, and the average peak-to-correlation energy (APCE) is used to determine the template update to judge the confidence. In the detection module, GMS grid motion statistics is used as the detector to make the ORB algorithm with fast rotation invariance feature to match the target in the previous frame, and then the grid motion statistics is used to filter the matching results to achieve the rough positioning of the target position, and the target detection area is reduced dynamically according to the prediction position. Finally, the cascaded classifier is used to locate the target accurately. The results show that the tracking method proposed in this paper can greatly improve the tracking speed of the algorithm while effectively preventing model drift, and has better accuracy and robustness to challenging environments such as target occlusion, scale variation and rotation.

Key words image processing; target tracking; model drift; motion statistics; dynamic cutting

OCIS codes 100.3008; 100.4999

收稿日期: 2020-06-16; 修回日期: 2020-07-12; 录用日期: 2020-08-07

基金项目: 国家自然科学基金(41701479)、辽宁省自然科学基金(20180550529)

* E-mail: 312553405@qq.com

1 引 言

随着计算机视觉技术的兴起,视觉目标跟踪技术已经成为该领域的研究热点^[1],在医学图像处理、无人机、自动驾驶、国防安全等方面得到广泛应用。但由于跟踪过程中存在目标形变、遮挡、尺度变化等多种复杂因素,目标跟踪仍然面临着巨大的挑战。

视觉目标跟踪算法主要分为生成式模型^[2-4]和判别式模型^[5-10]。生成式模型的主要思想是计算当前候选目标与目标模型的相似程度,相似度最高的候选样本作为目标进行跟踪^[11]。经典的生成式算法如基于粒子分布统计的粒子滤波算法^[2]、基于概率密度分布的 Mean-Shift 算法^[3]、基于特征点匹配的光流跟踪算法^[4]等。但生成式模型没有考虑到目标的背景信息,在目标遮挡、光照变化或背景杂乱等环境下通常会发生模型漂移。

判别式方法在一定程度上规避了生成式模型的缺点,目标和背景都作为训练模型的基本信息,同时将目标从背景中抽离出来,得到当前帧的目标位置。Bolme 等^[5]提出最小化误差平方和滤波跟踪器(MOSSE),该算法利用傅里叶变换实现了从时域到频域的转换计算方式,在降低运算量的同时大大提高了跟踪速度;Henriques 等^[6-7]提出核化跟踪检测循环结构算法(CSK)^[6],通过循环矩阵稠密采样,加入高斯核函数,有效避免了逆矩阵的复杂运算;随后又利用核函数对 MOSSE 进行改进,提出了核相关滤波跟踪(KCF)^[7],该算法摒弃单通道灰度特征,引入了多通道梯度方向直方图(HOG)特征,将其映射到非线性空间,进一步提高了算法性能;Danelljan 等^[8-10]提出了空间正则化的相关滤波器(SRDCF)^[8],在训练过程中加入空间正则化,有效解决了边界效应产生的模型漂移问题,同时使用高斯赛德尔函数求解滤波器,但是过多的迭代次数降低了滤波器的跟踪速度。2017 年, Danelljan 等^[10]在判别式尺度空间跟踪(DSST)^[9]的基础上提出快速判别尺度空间的相关滤波器(fDSST),通过对位置滤波器主成分分析(PCA)特征降维和尺度滤波器正交三角(QR)分解降维进行加速,使得算法更加鲁棒。

上述基于相关滤波的跟踪算法在跟踪性能、速度上都各有优劣,但跟踪过程中极易受到遮挡、快速运动、尺度变化等因素的影响,导致跟踪失败,因而需要一种跟踪与检测并行的算法对目标进行长时跟踪,以达到更加鲁棒的跟踪效果。

2012 年, Kalal 等^[12]提出跟踪学习检测(TLD)算法,在该算法中,目标跟踪与检测并行,两个模块的输出结果都参与学习过程,通过学习机制再反作用于跟踪器和检测器,以此达到长时间跟踪的目的。

文献[13]在光流法的基础上引入 Kalman 滤波解决目标严重遮挡的问题,并采用帧差法确定目标运动区域的范围,但该算法仍然受到光流法应用场景的限制,对光线极为敏感,易导致模型漂移,帧差法的引入虽然提高了目标检测效率,但跟踪精度却大幅下降。文献[14-15]均在检测模块中使用 Kalman 滤波器对检测区域进行预测,以达到缩小检测范围的效果,但由于运动目标的不规则性, Kalman 滤波的应用受到了极大的限制。文献[16]利用 Mean-Shift 算法通过在线更新模型解决运算复杂度高的问题,并通过动态调整分类器参数,提升算法的精确度,但当目标发生尺度变化时,跟踪框保持宽度不变,极易跟踪失败。以上改进算法在一定程度上提高了目标跟踪的准确性和实效性,但由于 TLD 算法在整体性能上依赖于跟踪模块与检测模块之间相互的配合,所以仅仅对跟踪器或检测器进行改进,并不能使得算法总体跟踪性能得到明显的提高。

因此,针对上述问题,本文做出以下改进:1)考虑到 fDSST 算法在尺度空间判别和跟踪速度上的明显优势,本文在跟踪模块中将其作为跟踪器,并依据 TLD 框架下跟踪模块与检测模块的独立性,在模板更新上利用检测模块对跟踪模块结果的准确性进行判别——使用平均峰值相关能量(APCE)对模板进行有选择地更新,在防止复杂场景下模型漂移的同时,大大提高了 fDSST 的跟踪性能;2)在检测模块中引入网格运动统计(GMS)对目标位置进行粗定位,并利用其构造一种检测区域动态缩减的检测方法,目标区域子窗口的减少不但节省了大量的跟踪时间,保证了跟踪实时性,更使得级联分类器对目标的定位更加精确。

2 TLD 算法

2.1 TLD 算法

TLD 算法是一种单目标长时间跟踪算法,该算法主要由跟踪模块、检测模块、学习模块和综合模块组成^[17]。

跟踪模块采用中值流跟踪算法,利用目标框来表示(被跟踪)目标,并在连续的相邻帧之间估计目标的运动轨迹,其原理是先采用金字塔 Lucas-

Kanade 光流法在当前帧目标框的特征点中跟踪到下一帧特征点的对应位置,并引入对图像噪声干扰和亮度变化稳健性较好的归一化互相关(NCC)^[18]和差值平方和(SSD)^[19]作为其评判准则^[20],将NCC与正反向跟踪检测机制结合起来产生预测位置,根据相邻两帧之间的位置与尺度变化计算位移中值,并将此作为跟踪模块的输出。

检测模块可以看作一个级联分类器结构,由方差分类器、集合分类器和最近邻分类器组成,利用滑动窗口全局扫描的方法对每帧图片进行样本筛选。其中,方差分类器对灰度值方差小于初始跟踪框50%的子窗口进行淘汰,而另一半子窗口输入至集合分类器;集成分类器含有若干个基本分类器,每个基本分类器对子窗口图像块进行像素比较,产生对应的二进制编码,并根据编码计算出相应的后验概率分布,若所有后验概率分布平均值大于50%,则集成分类器将其认定为目标,然后进入最近邻分类器,对新样本进行相关相似度计算,若大于设定阈值则认为为正样本(即目标),否则将被分类为负样本。

学习模块是根据跟踪与检测结果对当前帧的正负样本进行采样,采用P-N学习策略^[21]对目标模型进行学习更新,其中P专家负责检出被误分为负样本的正样本,并将其添加到正样本集中,N专家负责修正被误分为正样本的负样本,修正之后将其添加到正样本集中。

综合模块则是按照一定的融合策略对跟踪和检测模块的跟踪框进行评测,即先对检测结果进行聚类操作,然后根据聚类结果和跟踪结果的置信度,对二者进行加权作为跟踪的最终输出结果;若跟踪与检测没有得到目标框,则跟踪失败。

2.2 TLD 算法的不足

TLD 算法虽然利用跟踪、检测和学习实现了长时间目标跟踪,但是该算法还是存在以下不足:

1)以 Lucas-Kanade 光流法为核心的跟踪模块有三个性质假设前提,即亮度恒定性、时间连续性以及运动空间一致性^[22]。当在跟踪过程中发生目标遮挡、光照变化或离开视野时,便会破坏其假设,模型易产生漂移导致跟踪失败。

2)在检测过程中采用滑动窗口全局扫描方法,对每一帧包含的全部子窗口进行检索,由于检测区域过大,子窗口繁多(一幅 640 pixel×480 pixel 的 JPEG 图像内就包含多达七万多个子窗口),而其中绝大部分并不包含目标,大大降低了检测模块的运行速度,严重影响了目标跟踪的实时性。

3 本文算法

在 TLD 框架的基础上,本文首先采用加入了 APCE 阈值增强加权策略的 fDSST 算法对跟踪模块进行重新设计,改善了 TLD 算法在目标遮挡、丢失或光照变化等情况下模型漂移的现象;同时在检测模块中引入 GMS 检测器对目标进行粗定位,并对预测位置进行动态缩减,减少大量子窗口扫描,提高了目标检测速度,保证算法的实时性。本文算法的整体框架如图 1 所示。

3.1 基于 fDSST 算法的跟踪模块

fDSST 算法^[11]采用位置滤波和尺度滤波对目标位置和尺度进行预测。首先,该算法以目标为中心,图像块 f^l 表示目标区域,并提取该区域 d 维 HOG 特征,利用最小化代价函数构造 d 维相关滤波器 h^l ,表示为

$$\epsilon = \left\| g - \sum_{l=1}^d h^l * f^l \right\|^2 + \lambda \sum_{l=1}^d \| h^l \|^2, \quad (1)$$

式中: f^l 为训练样本, $l \in \{1, 2, \dots, d\}$; g 表示 f^l 的响应输出,中心响应峰值即为当前目标位置;* 表示循环相关运算; λ 为正则项系数,避免过滤器的过拟合。

为了简化运算步骤,降低计算复杂度,通过快速傅里叶变换(FFT)将(1)式进行频域建模,求解得到

$$H^l = \frac{\overline{GF^l}}{\sum_{k=1}^d \overline{F^k F^k} + \lambda} = \frac{A_t^l}{B_t}, \quad (2)$$

式中: G, H^l, F^l 为(1)式中 g, h^l, f^l 经过离散傅里叶变换(DFT)得到的对应变量; $\overline{\cdot}$ 表示 \cdot 的共轭转置; t 为图像帧数。

将(2)式进行模型更新,可快速得到更加鲁棒的近似结果,即

$$\begin{cases} A_t^l = (1 - \eta) A_{t-1}^l + \eta \overline{GF_t^l}, & l = 1, 2, \dots, d \\ B_t = (1 - \eta) B_{t-1} + \eta \sum_{k=1}^d \overline{F_t^k F_t^k} \end{cases}, \quad (3)$$

式中: η 为学习率,视为样本模型的更新参数。

在第 $t-1$ 帧处,提取不同维度的候选样本 Z , Z_t^l 为 Z 经 DFT 得到的正样本区域。将频域滤波输出转换为时域滤波结果,求解相关滤波响应值 Y_t ,即

$$Y_t = \mathcal{F}^{-1} \left\{ \frac{\sum_{l=1}^d \overline{A_{t-1}^l} Z_t^l}{B_{t-1} + \lambda} \right\}, \quad (4)$$

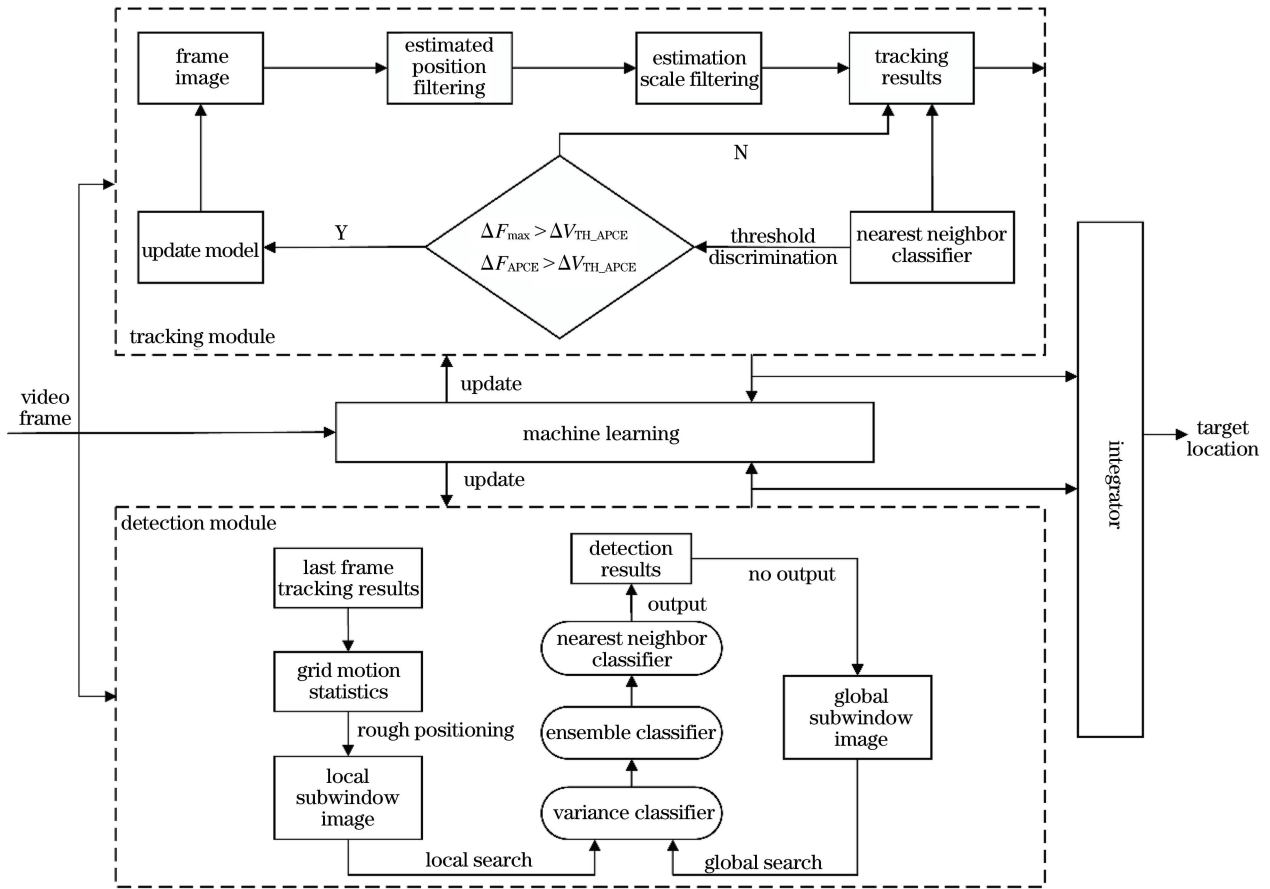


图 1 整体框架示意图

Fig. 1 Schematic of overall frame

式中： \mathcal{F}^{-1} 表示离散傅里叶逆变换 (IFFT)； Y_t 的最大响应值位置即为当前帧目标估计位置 L_t 。

针对跟踪过程中目标尺度变化问题，在位置滤波器的基础上加入尺度滤波器。其思想是在当前帧中，先利用二维位置滤波器定位当前帧目标的位置（得到的目标尺度仍为前一帧的尺度），并加入一维尺度滤波器，以当前区域的中心位置为中心，从获取到的候选样本中找到最优匹配尺度。目标样本尺度的定义为

$$a^c w \times a^c h, c \in \left\{ \left[-\frac{S-1}{2}, \dots, \frac{S-1}{2} \right] \right\}, \quad (5)$$

式中： a 为尺度因子； w 和 h 分别为目标框的宽度和高度； S 为尺度滤波器的大小，依此得到多个尺度下不同大小的候选样本，再利用 (4) 式计算尺度滤波器的最大响应值，即为当前目标尺度，将其定义为 D 。

3.2 模型更新策略

根据 (3) 式可知，fDSST 算法在跟踪过程中采用的是对每一帧图像进行更新的策略，但对跟踪结

果并没有进行可靠性的判定，当目标出现遮挡、消失或光照变化时，更新模型只会增大跟踪器对目标的识别误差，最终导致模型漂移。针对此状况，本文在大量的实验基础上发现相关滤波跟踪响应图的峰值及振荡程度能够对目标当前状态进行判别，依此更新模型从而达到长时间稳定跟踪的效果。

本文算法利用 TLD 跟踪模块与检测模块相互独立的性质，首先将第 t 帧跟踪模块结果 P_t 输入至检测模块中的最近邻分类器，设定阈值，利用响应图的峰值状态及其振荡程度判断模型是否更新。文献 [23] 提出一种 APCE 计算跟踪置信度的方法，能够通过响应图振荡程度有效避免低于均值的响应峰值带来的模型更新，定义为

$$E_{APCE} = \frac{|F_{\max} - F_{\min}|^2}{\text{mean} \left[\sum_{w,h} (F_{w,h} - F_{\min})^2 \right]}, \quad (6)$$

式中： F_{\max} 、 F_{\min} 和 $F_{w,h}$ 分别表示最大响应值、最小响应值以及 (w, h) 位置上的响应值。APCE 的值越大，响应图中目标位置的峰值越明显，周围检测区域


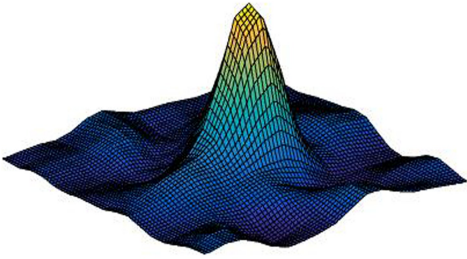

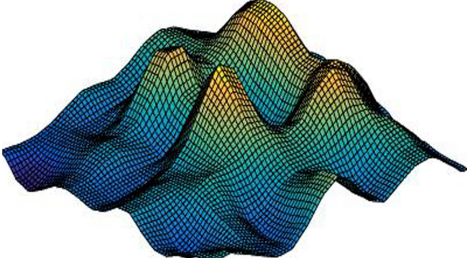
无明显波动(尖锐的单峰为最佳理想状态),目标的可信度也随之增加,反之则越低。

经过多次实验发现,在跟踪过程中当目标受到光照变化、旋转、偏离视野等因素的干扰时,响应图会受到不同程度的振荡,影响模型更新。故本文采取对 APCE 值设定阈值方法进行更新判别,当目标跟踪性能受到不良影响时,增强阈值的判断能力能够使模型更新更加准确,若直接在阈值相应均值上正向加权,则不满足上述条件,所以本文采取对阈值增强加权的策略,阈值 ΔV_{TH_APCE} 的计算公式为

$$\Delta V_{TH_APCE} = \frac{1}{\Delta T^\alpha} \times 10^\beta \times \gamma, \quad (7)$$

表 1 DragonBaby 序列在不同帧数的响应图及滤波器更新情况

Table 1 Response maps and filter updates of DragonBaby sequence in different frames

Frame	Response map	Filter update
<p># 73</p> 		Yes
<p># 101</p> 		No

跟踪模块算法具体步骤如下。

1) 训练滤波器模型。根据矩形框标定的第一帧目标区域,对位置滤波器和尺度滤波器进行训练。

2) for $t = 2, 3, \dots, n$

输入:第 t 帧图像 I_t ,上一帧目标位置 L_{t-1} 、目标尺度 D_{t-1} 。

输出:当前帧目标跟踪结果 P_t 。

①目标位置估计。根据 Y_{t-1} 从当前帧中提取目标区域,以其位置为中心,按照第 $t-1$ 帧矩形框的 2 倍大小进行候选样本 Z_t 的采集,代入(4)式经 IFFT 得到响应值 Y_t ,再通过计算 $\max(Y_t)$ 最大响应值得到目标当前的位置 L_t 。

②目标尺度估计。根据位置滤波器获得当前目标位置 L_t ,利用一维尺度滤波器提取该目标位置不

同尺度的候选样本 Z_t ,并依据(5)式计算尺度响应值,最大响应尺度值即为当前尺度 D_t 。

③根据 L_t 和 D_t 从图像 I_t 中提取跟踪结果 P_t 。

④模板更新。将 P_t 输入至检测模块最近邻分类器中,当 ΔF_{max} 和 ΔF_{APCE} 同时大于阈值 ΔV_{TH_APCE} 时,对位置滤波器和尺度滤波器进行更新;反之,两个滤波器均不进行更新。

Table 1 Response maps and filter updates of DragonBaby sequence in different frames

同尺度的候选样本 Z_t ,并依据(5)式计算尺度响应值,最大响应尺度值即为当前尺度 D_t 。

③根据 L_t 和 D_t 从图像 I_t 中提取跟踪结果 P_t 。

④模板更新。将 P_t 输入至检测模块最近邻分类器中,当 ΔF_{max} 和 ΔF_{APCE} 同时大于阈值 ΔV_{TH_APCE} 时,对位置滤波器和尺度滤波器进行更新;反之,两个滤波器均不进行更新。

3.3 基于 GMS 的检测区域动态缩减的检测模块

由于 TLD 检测模块中直接采用滑动窗口全局扫描策略使得过大的检测区域内包含上万个子窗口,大大降低了检测速度,同时子窗口包含的冗余信息会被作为正样本学习到学习模块的样本库中,从而增大检测模块的误差,严重影响检测器的检测性

能。故本文在检测模块中首先引入 GMS 检测器^[24],其原理是先依据 ORB(Oriented FAST and Rotated BRIEF)算法进行特征点匹配,再利用网格运动统计对特征匹配结果进行筛选,过滤匹配错误的特征,对目标位置实现粗定位。

I_a 为匹配图像, I_b 为待匹配图像。当计算两个区域中特征匹配的概率时,首先假设区域 a 与区域 b 表示两幅图像的另一物体区域,其中 f_a^T 为正确匹配, f_a^F 为错误匹配, f_a^b 表示 f_a 在区域 b 中的正确匹配,则其发生的概率为 $P(f_a^T) = r$, 设 P_T 和 P_F 分别表示在区域 a 中一个特征点匹配到区域 b 中正确的概率和错误的概率,

$$P_T = P(f_a^T) + P(f_a^F)P(f_a^b | f_a^F) = r + (1-r)\theta q/Q, \quad (9)$$

$$P_F = P(f_a^F)P(f_a^b | f_a^F) = (1-r)\theta q/Q, \quad (10)$$

式中: q 和 Q 分别为区域 b 和整个图像 I_b 中的特征点个数; θ 为权重值。

正确匹配的区域中会出现定量的错误匹配特征点,为了增大特征匹配结果的差异对其进行快速区分,文献[25]构造了基本平滑运动核,将图像划分为 3×3 个网格(网格运动划分如图 2 所示),则每个网格特征点估计量 S_i 呈二项分布,公式为

$$S_i \sim \begin{cases} B(q, P_T), & \text{if } x_i \text{ is true} \\ B(q, P_F), & \text{if } x_i \text{ is false} \end{cases} \quad (11)$$

且同时每个核特征点估计量表达式为

$$S_{a,b} = \sum_{k=1}^K |x_{a^k b^k}|, \quad (12)$$

式中: K 为不相交的网格个数 ($K=9$); $|x_{a^k b^k}|$ 为区域 a 与区域 b 成功匹配的点对数量。此时,相邻网格区域特征点匹配的估计量 S_i 的二项分布如下:

$$S_i \sim \begin{cases} B(Kq, P_T), & \text{if } x_i \text{ is true} \\ B(Kq, P_F), & \text{if } x_i \text{ is false} \end{cases} \quad (13)$$

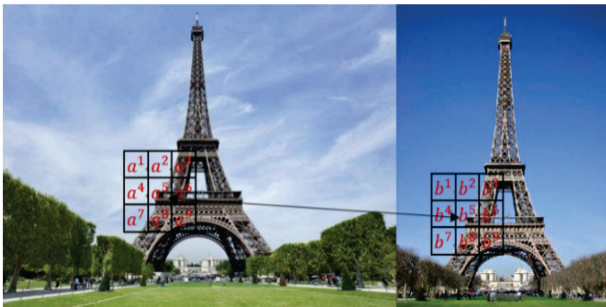


图 2 网格运动划分效果图

Fig. 2 Grid motion partition effect diagram

网格的划分令区域特征点估计量 S_i 的值逐渐增大,故在通常情况下,设置判定阈值 $\tau = e_F + \sigma s_F \approx \sigma s_F \approx \sigma \sqrt{q}$, 其中 e_F 表示均值, s_F 为方差,可得到每个核区域匹配点对数量的二值化表达式为

$$N_{\text{cell}} - N_{\text{pair}}\{a,b\} \in \begin{cases} T, & S_{a,b} > \tau_a = \sigma \sqrt{q_a} \\ F, & \text{othersize} \end{cases} \quad (14)$$

式中: σ 为权重值 ($\sigma=6$); T 表示正确匹配; F 表示错误匹配。

此时,剔除特征点估计量 S_i 小于阈值 τ 的匹配对,剩余网格中则包含较多符合匹配条件的特征点,为样本提供了可靠的特征匹配对,从而实现目标的粗定位。然后根据检测情况调整目标检测区域,首先对其进行局部子窗口扫描,再使用级联分类器实现对目标的精确定位。GMS 检测器检测效果如图 3 所示。

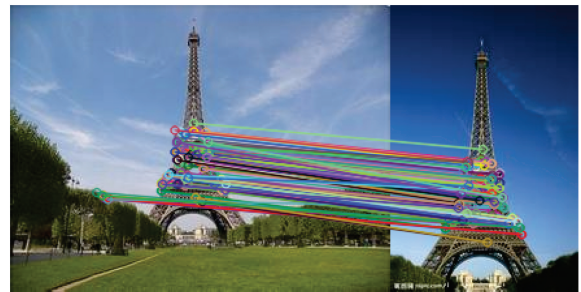


图 3 GMS 检测器检测效果图

Fig. 3 GMS detector detection effect diagram

本文检测模块算法具体步骤如下。

for $t=2,3,\dots,n$

输入: 第 t 帧图像 I_t , 上一帧目标位置 L_{t-1} 、目标尺度 D_{t-1} 。

输出: 当前帧检测结果 P_t 。

1) 根据 L_{t-1} 和 D_{t-1} 在第 $t-1$ 帧中选取图像, 根据(9)式、(10)式计算正确匹配点邻域内匹配正确和错误的概率;

2) 使用网格运动统计方法利用(14)式计算出区域匹配的点对数,保留大于阈值 τ 的网格区域匹配点对数量,得到预测区域的中心位置;

3) 以该位置为中心,在当前帧中取 2 倍 D_{t-1} 大小的图像块为预测区域,对预测区域仍采用 TLD 算法中滑动窗口的策略来生成检测子窗口;

4) 将生成的检测子窗口输入级联分类器,最终通过的窗口即为检测模块的输出结果;

5) 若无窗口输出,则对输入图像进行全局搜索以生成检测子窗口,然后转步骤 4。

4 实验

4.1 实验环境与参数设置

本文算法(标记为 Ours)的实验均在 Matlab R2016b 上进行,实验的硬件环境配置为 Intel(R) Core(TM) i5-4210U CPU @ 2.40 GHz, 8 GB RAM, windows 10, 64 位的操作系统。为了验证本文算法的鲁棒性和有效性,实验采用公开数据集 OTB-2015^[25] 中 100 个视频序列进行测试,同时选取 KCF^[7]、SRDCF^[9]、fDSST^[11]、TLD^[12]、Staple^[26] 5 种当下主流算法进行定量分析和定性分析。

实验参数如下:在位置滤波器中,特征维度 $d=32$,正则项系数 $\lambda=0.002$,学习率 $\eta=0.0025$ 。在尺度滤波器中,尺度个数 $S=17$,学习率 $\eta=0.02$ 。在(7)式中,权值参数 $\alpha=30, \beta=32, \gamma=1$,根据实验测试结果可以设置(9)式和(10)式中 $\theta=3$ 。

4.2 定性分析

从上述的 100 个视频序列中选取 8 个同时包含多种挑战因素的视频序列(表 2)进行实验,视频中包含 11 种挑战性环境,分别为光照变化(IV)、尺度变化(SV)、遮挡(OCC)、形变(DEF)、运动模糊(MB)、快速运动(FM)、平面外旋转(OPR)、平面内旋转(IPR)、背景杂乱(BC)、离开视野(OV)、低分辨率(LR)。并列出了本文算法与其他 5 种主流算法的跟踪效果图,如图 4 所示。

表 2 实验视频序列信息

Table 2 Information of the test video sequences

Video sequence	Frame	Challenge factor
Basketball	725	OCC, DEF, IV
BlurOwl	631	MB, FM, IPR, SV
Football	362	OCC, BC, IPR, OPR
Freeman4	283	SV, OCC, BC, IPR, OPR
Human4	667	IV, OCC, SV, DEF
Liquor	1741	IV, SV, OCC, MB, FM, OPR, OV, BC
Skating2-2	473	SV, OCC, DEF, FM, OPR
Soccer	392	SV, IV, BC, FM, OCC, MB, IPR, OPR

1) Basketball 序列中目标频繁发生遮挡和光照变化现象。在第 23 帧,当目标出现暂时遮挡时,SRDCF、KCF、Staple 算法出现漂移现象;在第 303 帧,目标被完全遮挡后, fDSST 算法跟踪框漂移至相似人物,跟踪失败,其余算法跟踪框轻微漂移;第 650 帧,在光照变化的影响下,以 Lucas-Kanade 光流法为核心的 TLD 算法跟踪失败;除此之外,其余算法定位成功,但是从跟踪框可以看出,本文算法对目标的尺度大小和定位更加准确,具有较好的跟踪性能。

2) BlurOwl 序列存在着因目标快速运动或摄像机抖动而产生的运动模糊现象。从第 77 帧开始,摄像机的抖动使得 Staple、SRDCF 和 TLD 算法的跟踪框逐渐漂移,直至跟踪框脱离目标无法重新跟踪,跟踪失败;在第 441 帧, fDSST 算法因目标出现平面内旋转和尺度变化而产生轻微漂移现象;最终在整个跟踪过程中,本文算法和 KCF 算法始终可以有效防止摄像机抖动带来的漂移现象,表现稳定。

3) Football 序列背景杂乱,目标被严重遮挡且伴有大量相似干扰物。在第 283 帧,相似物的干扰及局部遮挡现象会导致 KCF、fDSST 算法跟踪框出现漂移;在第 305 帧时, KCF 算法跟踪器回归,重新开始跟踪;目标的快速移动致使 SRDCF、TLD、Staple 算法最终跟踪失败;在第 362 帧,只有本文算法和 KCF 算法能够完成跟踪,但相比之下本文算法跟踪框标定的目标大小更加准确,效果最佳。

4) Freeman4 序列中,初始跟踪的目标尺寸非常小,且分辨率低,被跟踪目标由小变大的过程中还时而遇到遮挡现象。从第 51 帧开始,由于报纸的挥动遮挡,各跟踪器漂移不定,当目标出现由小到大的尺度变化及周围相似物的干扰时,只有本文算法可以对跟踪结果进行判定,同时利用检测器快速检测目标,具有良好的跟踪性能。

5) Human4 序列与 Liquor 序列中均出现严重的尺度变化与光照变化现象。以 Human4 序列为例进行分析,从第 131 帧开始, fDSST 和 Staple 算法因目标被遮挡物覆盖之后,跟踪框相继偏离目标,跟踪失败;目标的光照变化和尺度变化使得 TLD、SRDCF 跟踪器无法判定原目标,本文算法和 KCF 算法表现较为稳定,但因为目标的尺度由大变小, KCF 算法不能很好地适应目标尺度变化,只有本文算法可以更加精准地框定目标,跟踪效果最佳,由此可见,本文算法对目标的尺度变化、光照变化与遮挡更具鲁棒性。

6) Skating2-2 序列中目标始终呈现快速运动、形变和旋转,且伴有尺度变化和遮挡现象。从第 118 帧开始, fDSST 跟踪器学习到更多与目标相近颜色的背景信息,导致跟踪框漂移到背景;在目标经过多次旋转和形变后, SRDCF 算法不能精确地定位目标的尺度,最终丢失目标,跟踪失败;当男士被女士多次遮挡再现时, TLD 与 Staple 算法跟踪框相继漂移到女士身上,跟踪失败;由于本文算法加入了网格运动统计检测机制,能够很好地对目标外观进行检测匹配,因此达到良好的跟踪效果。

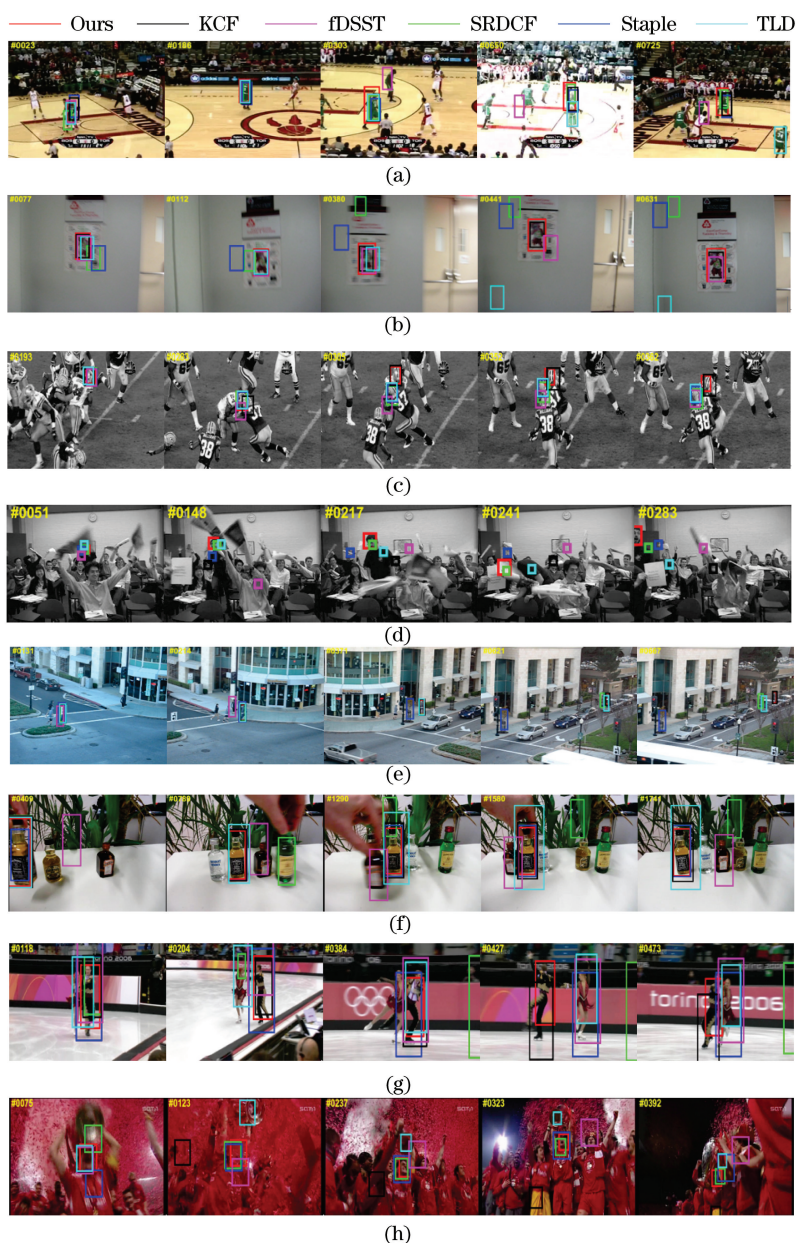


图 4 不同算法在部分序列上的跟踪效果。(a)Basketball; (b)BlurOwl; (c)Football; (d)Freeman4; (e)Human4; (f)Liquor; (g)Skating2-2; (h)Soccer

Fig. 4 Tracking effects of different algorithms on partial sequences. (a)Basketball; (b)BlurOwl; (c)Football; (d)Freeman4; (e)Human4; (f)Liquor; (g)Skating2-2; (h)Soccer

7) Soccer 序列中目标存在严重的遮挡和相似背景干扰,且伴有尺度变化、快速运动和运动模糊问题。在第 123 帧,目标尺度发生变化,KCF 算法无法适应而导致逐渐偏离目标,跟踪失败;TLD 与 fDSST 跟踪器由于相似颜色的背景干扰在 123 帧后逐渐产生漂移,无法准确跟踪目标;SRDCF 与 Staple 算法虽然在 323 帧能够跟踪目标,但由于目标快速运动,跟踪精度明显下降;从第 392 帧可以看出,本文算法始终能够以较高精度跟踪目标,对于遮挡、相似背景干扰与运动模糊的场景也表现出较好

的鲁棒性。

4.3 定量分析

本文采用中心位置误差(CLE)、跟踪精确度(DP)和成功率(OP)作为衡量跟踪性能的评价标准,采用跟踪速度(FPS)来评价算法实时性。其中,CLE 表示跟踪算法目标框中心点与真实目标框中心点的欧氏距离,其值越小则表明跟踪精度越高;DP 表示中心位置误差 CLE 小于给定阈值的帧数与全部视频序列总帧数的比值,通常阈值默认为 20 pixel;OP 表示算法目标框重合率大于某个设定

阈值的帧数占总体视频帧数的比值,通常情况下阈值默认为 0.5。

图 5 为不同跟踪算法在 OTB-2015 数据集上的跟踪精度与成功率的曲线对比图。本文算法跟踪精

度与成功率分别为 0.836 和 0.774,皆优于其他 5 种对比算法,且相比 fDSST 算法分别提升了 15.4% 和 16%,较 TLD 基础算法也分别提升了 24.7% 和 27.9%。

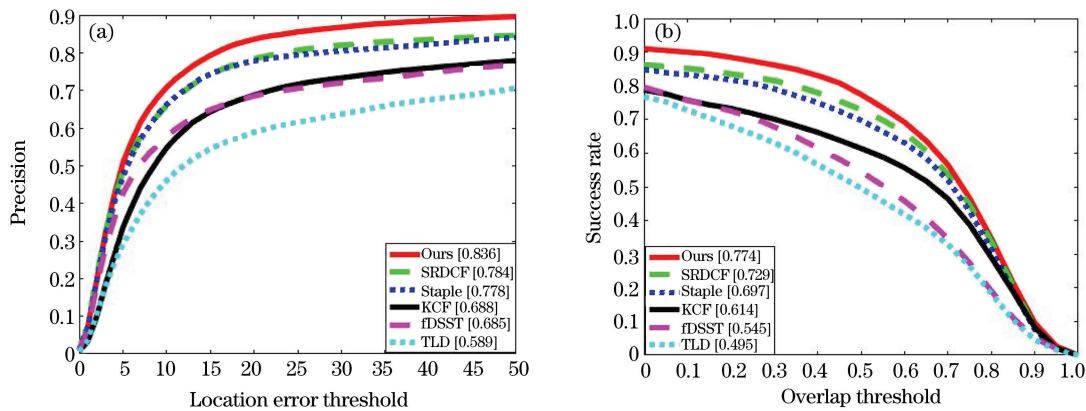


图 5 不同跟踪算法的精度与成功率曲线图。(a)精度;(b)成功率

Fig. 5 Precision and success rate curves of different tracking algorithms. (a) Precision; (b) Success rate

为了更好地分析本文算法的综合性能,对 11 种挑战性环境下算法的跟踪精度和成功率进行了对比

分析。图 6 和图 7 分别为部分挑战性环境下跟踪精度和成功率曲线对比图。

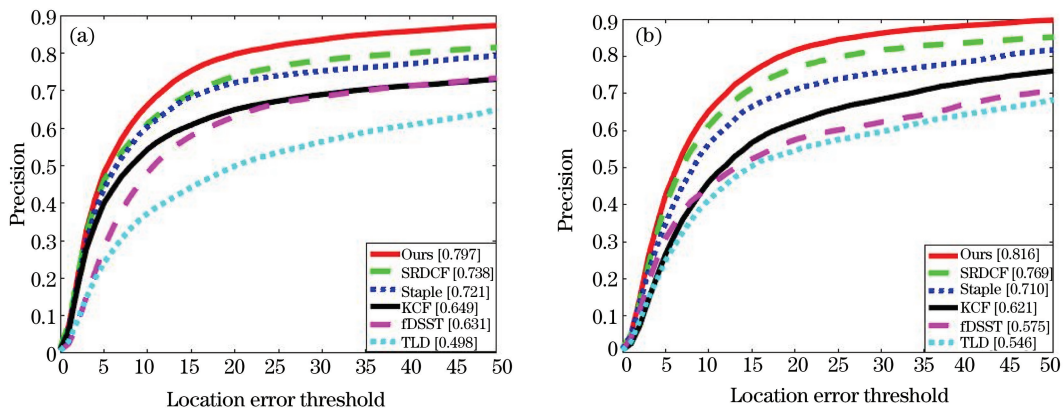


图 6 部分挑战环境下跟踪精度曲线对比图。(a)尺度变化;(b)快速运动

Fig. 6 Comparison of tracking precision under partial challenge environment. (a)Scale variation; (b)fast motion

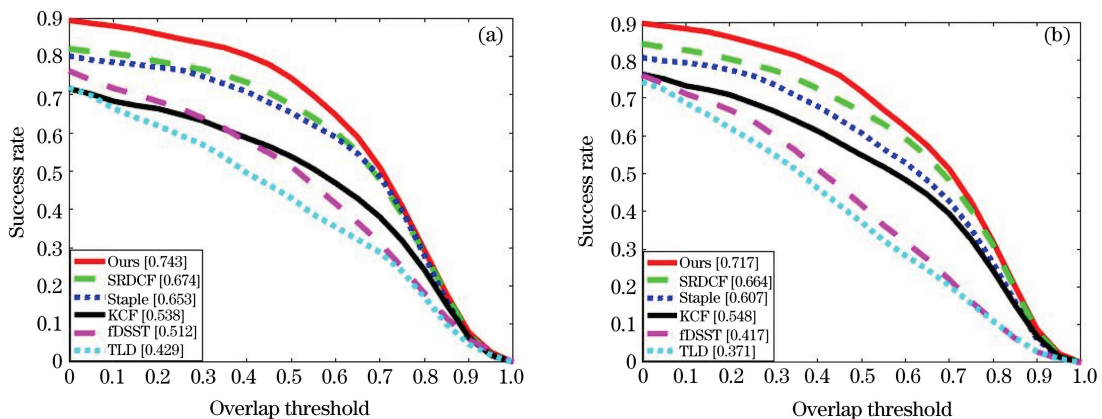


图 7 部分挑战环境下成功率曲线对比图。(a)遮挡;(b)尺度变化

Fig. 7 Comparison of success rate under partial challenge environment. (a)Occlusion; (b) scale variation

表 3 为不同算法在 11 种挑战环境下跟踪精度对比数据,其中,加粗字体和下划线分别表示最佳和次佳结果。从表 3 可以看出,本文算法在 IV、SV、OCC、MB、FM 等 7 种挑战环境下的跟踪精度均排名第一,在 OV、DEF 和 IPR 这三个挑战环境下分别以 0.006、0.001 和 0.009 的微弱劣势位居第二。

表 3 不同算法在 11 种挑战环境下跟踪精度对比

Table 3 Track precision comparison of different algorithms under 11 challenging environments

Challenge	Ours	KCF	fDSST	SRDCF	Staple	TLD
IV	0.788	0.724	0.720	<u>0.781</u>	0.778	0.515
OV	<u>0.726</u>	0.501	0.480	0.732	0.668	0.349
SV	0.797	0.631	0.649	<u>0.738</u>	0.721	0.498
OCC	0.800	0.632	0.602	0.724	<u>0.727</u>	0.515
BC	0.739	<u>0.744</u>	0.695	0.745	0.741	0.544
MB	0.793	0.601	0.570	<u>0.767</u>	0.699	0.547
FM	0.816	0.621	0.575	<u>0.769</u>	0.710	0.546
DEF	<u>0.746</u>	0.619	0.560	0.730	0.747	0.490
LR	0.849	0.560	0.602	<u>0.663</u>	0.610	0.538
IPR	<u>0.746</u>	0.685	0.696	0.730	0.755	0.550
OPR	0.799	0.663	0.651	<u>0.731</u>	0.726	0.507

表 4 为本文算法与 5 种对比算法在 11 种挑战环境下成功率对比数据,其中,加粗字体和下划线分别表示最佳和次佳结果。从表 4 可以看出,本文算法在 SV、IV、OCC、FM、BC 等 7 种挑战环境下的成功率均优于其他算法,在 MB 和 OPR 这两个挑战环境下分别以 0.03 和 0.026 的劣势居于第二位。

表 4 不同算法在 11 种挑战环境下成功率对比

Table 4 Success rate comparison of different algorithms under 11 challenge environments

Challenge	Ours	KCF	fDSST	SRDCF	Staple	TLD
IV	0.752	0.550	0.651	<u>0.735</u>	0.715	0.435
OV	0.514	0.457	0.443	0.558	<u>0.548</u>	0.311
SV	0.717	0.417	0.548	<u>0.664</u>	0.607	0.371
OCC	0.743	0.512	0.538	<u>0.674</u>	0.653	0.429
BC	0.794	0.597	0.619	<u>0.711</u>	0.679	0.465
MB	<u>0.729</u>	0.550	0.553	0.759	0.650	0.530
FM	0.759	0.526	0.538	<u>0.717</u>	0.645	0.506
DEF	0.714	0.503	0.505	<u>0.659</u>	0.656	0.415
LR	0.623	0.295	0.644	<u>0.625</u>	0.472	0.319
IPR	0.664	0.541	0.659	0.659	<u>0.663</u>	0.471
OPR	<u>0.636</u>	0.516	0.569	0.619	0.662	0.417

由上述实验数据可以看出,本文算法在基于

TLD 框架下设计 APCE 置信度判别和 GMS 检测,使得目标在遮挡、光照变化等复杂环境下可以有效防止模型漂移,具有较高的跟踪精度,同时对快速运动、目标旋转、背景复杂等场景也表现出了一定的鲁棒性。

4.4 实时性分析

为了验证本文算法的实时性,对上述 8 个视频序列进行测试,不同算法的跟踪速度数据如表 5 所示,其中,加粗字体和下划线分别表示最佳和次佳结果。本文算法在 8 个视频集上的平均帧率达到了 25.89 frame/s,具有较好的实时跟踪效果。

表 5 不同算法在视频序列上的平均跟踪速度

Video sequence	Average tracking speed of different algorithms on video sequences unit:frame/s					
	Ours	KCF	fDSST	SRDCF	Staple	TLD
Basketball	25.706	18.562	12.019	<u>24.563</u>	19.563	14.145
BlurOwl	19.598	22.440	17.212	25.309	<u>23.764</u>	11.908
Football	24.248	19.957	15.665	<u>22.862</u>	21.771	12.557
Freeman4	41.393	29.849	30.702	37.423	<u>39.878</u>	16.312
Human4	<u>17.469</u>	13.259	10.589	19.134	16.180	9.102
Liquor	30.196	21.573	22.620	<u>26.987</u>	25.109	17.445
Skating2-2	21.281	20.056	17.225	<u>21.565</u>	22.323	13.608
Soccer	26.640	15.422	11.874	17.211	<u>19.797</u>	10.794

本文算法通过置信度判定对模型进行有选择地更新,改变了传统跟踪器每帧更新的方式,同时在检测模块实现目标粗定位及检测区域动态缩减,减少了目标区域子窗口的数量,节约了跟踪时间,保证了跟踪实时性。在测试的视频序列中,平均中心位置误差较低,跟踪精确度较高,具有良好的跟踪性能。

5 结 论

本文基于 TLD 框架在跟踪模块引入 fDSST 作为跟踪器,依据跟踪模块与检测模块相互独立的特性,在模型更新上利用检测模块对跟踪模块结果的准确性进行判别,加入 APCE 阈值增强加权的策略进行模型更新,一定程度上抑制了 TLD 算法在目标遮挡、丢失或光照变化等环境下模型漂移的现象;在检测模块中首先采用 GMS 对目标位置进行粗定位,并对预测位置进行动态缩减,构造一种检测区域动态缩减的检测方法,减少了大量目标区域子窗口的扫描,不但节省了大量的跟踪时间,保证了跟踪实时性,更使得级联分类器对目标的定位更加精确。在公开数据集 OTB-2015 上,本文算法与当前 5 种

主流算法进行实验对比,结果表明,本文算法可以有效防止由目标遮挡、光照变化、尺度变化等因素产生的模型漂移现象,且同时对上述复杂环境的挑战具有较强的稳定性,跟踪速度基本满足实时性的需求。

参 考 文 献

- [1] Ge B Y, Zuo X Z, Hu Y J. Review of visual object tracking technology [J]. *Journal of Image and Graphics*, 2018, 23(8): 1091-1107.
葛宝义, 左宪章, 胡永江. 视觉目标跟踪方法研究综述 [J]. *中国图象图形学报*, 2018, 23(8): 1091-1107.
- [2] 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): 125-141.
- [3] Vojir T, Neskova J, Matas J. Robust scale-adaptive mean-shift for tracking [J]. *Pattern Recognition Letters*, 2014, 49:250-258.
- [4] Xiao J, Zhu S P, Huang H, et al. Object detecting and tracking algorithm based on optical flow [J]. *Journal of Northeastern University (Natural Science)*, 2016, 37(6): 770-774.
肖军, 朱世鹏, 黄杭, 等. 基于光流法的运动目标检测与跟踪算法 [J]. *东北大学学报(自然科学版)*, 2016, 37(6): 770-774.
- [5] Bolme D S, Beveridge J R, Draper B A, et al. Visual object tracking using adaptive correlation filters[C]//2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 13-18, 2010, San Francisco, CA, USA. New York: IEEE Press, 2010: 2544-2550.
- [6] Henriques J F, Caseiro R, Martins P, et al. Exploiting the circulant structure of tracking-by-detection with kernels [M] // *Computer Vision - ECCV 2012*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012: 702-715.
- [7] Henriques J F, Caseiro R, Martins P, et al. High-speed tracking with kernelized correlation filters[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(3): 583-596.
- [8] Danelljan M, Häger G, Khan F S, et al. Learning spatially regularized correlation filters for visual tracking[C]//2015 IEEE International Conference on Computer Vision (ICCV), December 7-13, 2015, Santiago, Chile. New York: IEEE Press, 2015: 4310-4318.
- [9] Danelljan M, Häger G, Khan F S, et al. Accurate scale estimation for robust visual tracking [C] // *Proceedings of the British Machine Vision Conference 2014*, Nottingham. British Machine Vision Association, 2014: 65.1-65.11.
- [10] Danelljan M, Häger G, Khan F S, et al. Discriminative scale space tracking [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 39(8): 1561-1575.
- [11] Liu W J, Sun H, Jiang W T. Correlation filter tracking algorithm for adaptive feature selection[J]. *Acta Optica Sinica*, 2019, 39(6): 0615004.
刘万军, 孙虎, 姜文涛. 自适应特征选择的相关滤波跟踪算法 [J]. *光学学报*, 2019, 39(6): 0615004.
- [12] Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(7): 1409-1422.
- [13] Lü R P, Cai X Y, Dong L, et al. Context object tracking algorithm based on TLD framework [J]. *Video Engineering*, 2015, 39(9): 6-9, 43.
吕柄蓬, 蔡肖芋, 董亮, 等. 基于 TLD 框架的上下文目标跟踪算法 [J]. *电视技术*, 2015, 39(9): 6-9, 43.
- [14] Zhou X, Qian Q M, Ye Y Q, et al. Improved TLD visual target tracking algorithm[J]. *Journal of Image and Graphics*, 2013, 18(9): 1115-1123.
周鑫, 钱秋朦, 叶永强, 等. 改进后的 TLD 视频目标跟踪方法 [J]. *中国图象图形学报*, 2013, 18(9): 1115-1123.
- [15] Sun C J, Zhu S H, Liu J W. Fusing Kalman filter with TLD algorithm for target tracking [C] // 2015 34th Chinese Control Conference (CCC), July 28-30, 2015, Hangzhou, China. New York: IEEE Press, 2015: 3736-3741.
- [16] Sun Y J, Zhang L Y, Yun X. Visual tracking algorithm based on region estimation and adaptive classification[J]. *Laser & Optoelectronics Progress*, 2019, 56(18): 181001.
孙彦景, 张丽颖, 云霄. 基于区域预估与自适应分类的视觉跟踪算法 [J]. *激光与光电子学进展*, 2019, 56(18): 181001.
- [17] Fu M, Xing C J. Improved TLD target tracking method based on frame difference [J]. *Electronic Design Engineering*, 2017, 25(7): 183-186.
付苗, 邢藏菊. 帧间差法对 TLD 跟踪算法的改进 [J]. *电子设计工程*, 2017, 25(7): 183-186.
- [18] Briechle K, Hanebeck U D. Template matching using fast normalized cross correlation [J]. *Proceedings of SPIE*, 2001, 4387: 95-102.
- [19] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector[C]//*European Conference on*

- Computer Vision. Springer International Publishing, 2016.
- [20] Lü B K, Wu C M, Tian X P. Fast image matching algorithm based on best-buddies similarity[J]. *Laser & Optoelectronics Progress*, 2020, 57(10): 101018. 吕波凯, 吴成茂, 田小平. 一种基于最佳伙伴相似性的快速图像匹配算法[J]. *激光与光电子学进展*, 2020, 57(10): 101018.
- [21] Kalal Z, Matas J, Mikolajczyk K. P-N learning: bootstrapping binary classifiers by structural constraints [C] // 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 13-18, 2010, San Francisco, CA, USA. New York: IEEE Press, 2010: 49-56.
- [22] Liu H J. Design and implementation of moving target detection system based on FPGA + DSP [D]. Nanjing: Nanjing University of Science and Technology, 2015. 刘恒建. 基于 FPGA+DSP 的运动目标检测系统的设计与实现[D]. 南京: 南京理工大学, 2015.
- [23] Wang M M, Liu Y, Huang Z Y. Large margin object tracking with circulant feature maps[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 4800-4808.
- [24] Bian J W, Lin W Y, Matsushita Y, et al. GMS: grid-based motion statistics for fast, ultra-robust feature correspondence[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 2828-2837.
- [25] Wu Y, Lim J, Yang M H. Object tracking benchmark [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(9): 1834-1848.
- [26] Bertinetto L, Valmadre J, Golodetz S, et al. Staple: complementary learners for real-time tracking[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 1401-1409.