

视频中快速运动目标的自适应模型跟踪算法

刘宗达, 董立泉*, 赵跃进, 孔令琴, 刘明

北京理工大学光电学院, 北京 100081

摘要 针对现有视频跟踪算法因目标运动机动性强或非对称刚体目标快速变形等情况导致的目标丢失问题, 提出了一种基于相关滤波自适应模型与平均峰值相关能量(APCE)重检机制的视频跟踪算法。自适应模型跟踪算法可根据目标区域的清晰度实时调整模型, 有效保证目标跟踪模型的准确性。实验结果表明, 将自适应模型跟踪算法融入判别式尺度空间跟踪(DSST)模型中, 可增强模型对机动性强或快速变形物体的跟踪效果, 在保证跟踪速度的同时, 使原始 DSST 模型的平均精度提高了 18.3 个百分点, 成功率提高了 15.2 个百分点。此外, 将自适应模型跟踪算法与 APCE 重检机制相结合, 可保证跟踪算法的稳定性。

关键词 机器视觉; 目标跟踪; 模型更新; 自适应模型; 相关滤波; 重检机制

中图分类号 TP391.4

文献标志码 A

doi: 10.3788/AOS202141.1815001

Adaptive Model Tracking Algorithm for Fast-Moving Targets in Video

Liu Zongda, Dong Liqun*, Zhao Yuejin, Kong Lingqin, Liu Ming

School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China

Abstract To address the problem of target loss facing existing video tracking algorithms due to high mobility of targets or rapid deformation of asymmetric rigid targets, this paper proposes a video tracking algorithm based on the correlation filtering adaptive model and the redetection mechanism for average peak-to-correlation energy (APCE). The adaptive model tracking algorithm can adjust the model in real time according to the clarity of the target area, thereby effectively ensuring the accuracy of the target tracking model. Experimental results show that integrating the adaptive model tracking algorithm into the discriminative scale space tracking (DSST) model can enhance the tracking effect of the model on highly mobile or rapidly deforming objects. While guaranteeing tracking speed, the integration also raises the average accuracy of the original DSST model by 18.3 percentage points and the success rate by 15.2 percentage points. In addition, combining the adaptive tracking algorithm with the APCE redetection mechanism can ensure the stability of the algorithm.

Key words machine vision; target tracking; model update; adaptive model; correlation filter; redetection mechanism

OCIS codes 150.0155; 150.1135; 110.1085

1 引言

近年来, 目标跟踪^[1]在计算机视觉中得到了人们的广泛关注, 其在智能安防、智能交通、智能拍摄及自动驾驶等领域都有着重要应用。但目标跟踪算法极易受到目标形变、运动模糊、目标遮挡等因素的

干扰, 从而影响其实际应用。因此, 克服上述困难也成为该领域近年来的研究热点。已有的目标跟踪算法可分为生成式算法和判别式算法, 生成式跟踪算法主要有卡尔曼滤波^[2]、粒子滤波^[3]和均值漂移^[4]等算法。随着深度学习技术的发展, 生成式算法多以深度神经网络提取图像特征, 虽然跟踪精度较高,

收稿日期: 2021-03-08; 修回日期: 2021-04-01; 录用日期: 2021-04-12

基金项目: 国家自然科学基金(61705010, 61935001, 11774031)

通信作者: *kyilind@bit.edu.cn

但速度较慢;且算法运行前需要较长时间的训练,难以进行实际应用。判别式跟踪算法中的相关滤波算法^[5-7]具有计算速度快的优点,且不需要提前训练目标特征,而是先以初始帧中的目标为基准,不断学习其余帧中的目标变化情况,充分利用了不同帧间的差异信息,且很容易做到实时跟踪。现阶段比较流行的相关滤波算法有循环结构核(CSK)算法^[8]、核相关滤波(KCF)算法^[9]、判别式尺度空间跟踪(DSST)算法^[10]、基于循环特征映射的边缘目标跟踪(LMCF)算法^[11]、长时目标跟踪(LCT)算法^[12]等。CSK 算法在误差最小平方和滤波器(MOSSE)^[13]基础上引入了基于核函数的岭回归方法,在保证计算速度的同时提升算法的跟踪精度;KCF 在 CSK 的基础上引入了多通道特征,在特征提取层面提高了算法的跟踪精度;DSST 算法将尺度跟踪融入相关滤波算法中,解决了尺度变化较大的目标对算法跟踪结果的影响;LMCF 算法从模型更新的角度出发,将平均峰值相关能量(APCE)指标和特征最大响应值相结合,以判定跟踪准确度,从而决定是否更新模型;LCT 算法在 DSST 基础上融入了在线邻近算法(KNN)检测,有效提高了算法长期跟踪的稳定性。上述算法对视频中目标机动性较弱、形变较小的情况都有较好的跟踪效果,且能在一定程度上提高目标的跟踪精度。但如果目标的机动性较强、变化较快或图像比较模糊时,容易造成目标丢失且无法再次找回,导致跟踪精度不高、不能实现长期跟踪。

针对上述问题,本文提出了一种基于目标区域清晰度的自适应模型更新算法,可以有效提高算法在目标运动机动性强或非对称刚体目标快速变形等情况下的跟踪精度;同时,结合 LCT 算法与 LMCF 算法建立了合理的目标重检机制,并用现阶段比较流行的YOLOv4(You only look once, v4)目标检测

算法,解决了目标丢失后无法再次找回的问题。

2 相关理论和方法

2.1 相关滤波跟踪理论基础

相关滤波最早应用于信号处理中,是用来描述 2 个信号之间相关性的方法。对于 2 个信号 f 和 g ,其相关性在时间域上的连续和离散形式分别为

$$(f \otimes g)\tau = \int_{-\infty}^{\infty} f^*(t)g(t + \tau)dt, \quad (1)$$

$$(f \otimes g)n = \sum_{-\infty}^{\infty} f^*(m)g(m + n), \quad (2)$$

式中, f^* 为 f 的复共轭, t, τ 分别为连续时间的自变量与变化量, m, n 分别为离散时间的自变量与变化量。对 f 和 g 进行相关计算,得到最大响应值对应的时刻即为 2 个信号最接近的时刻。

根据相关滤波算法的基本思想,可将图像中目标位置的跟踪问题描述为图 1 所示的过程。可以发现,相关滤波算法需要在初始帧中提取一个目标图像滤波模板 h ,并将其与输入图像 f_i 进行相关计算,得到响应矩阵 g_i 。其中,下标 i 为像素序号。越接近目标的点,响应值就越大。良好的跟踪结果计算出的响应矩阵会呈现出较好的单峰性,而跟踪不稳定时,响应矩阵会呈多个波峰的情况,且峰值较低。

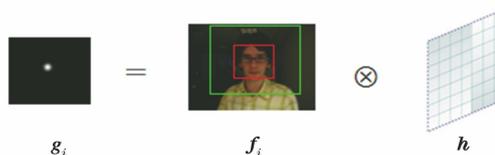


图 1 相关滤波算法在视觉跟踪中的应用

Fig. 1 Application of correlation filtering algorithm in visual tracking

相关滤波跟踪算法的计算过程如图 2 所示,可以发现,整个计算过程主要分为特征提取、在线匹配

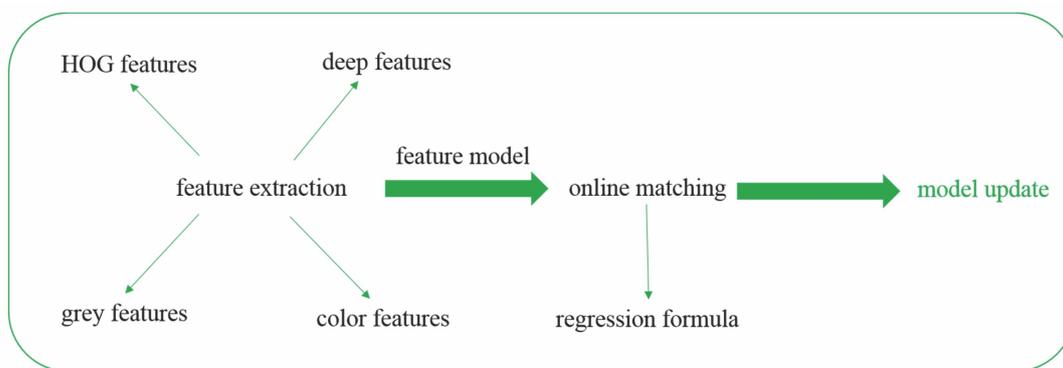


图 2 相关滤波目标跟踪算法的流程图

Fig. 2 Flow chart of the correlation filtering target tracking algorithm

和模型更新三部分。首先,对图像序列的首帧目标进行特征提取,包括图像的灰度特征、颜色特征、方向梯度直方图(HOG)特征以及深度学习算法中的深度特征等;然后,将提取出的特征模型与下一帧图像的待匹配区域进行在线匹配,得到下一帧图像的响应矩阵。在匹配每一帧图像后,都要将得到的特征模型按照一定学习率更新,以学习的新模型对下一帧图像进行匹配,模型的更新方式可表示为

$$\hat{x}^p = (1 - \gamma)\hat{x}^{p-1} + \gamma\hat{x}, \quad (3)$$

式中, \hat{x} 为提取的特征模型, γ 为学习率,取值范围

为 $0.01 \sim 0.1$, p 为帧序列数。

2.2 自适应模型的更新

跟踪过程中目标机动性较强或形变过快会导致运动模糊,跟踪框出现漂移现象,原因是在每一次匹配后的模型更新不准确。当目标的机动性较强或形变较大时,若此时的模型学习率较低(0.01),则仅能学习当前帧中的少部分信息,模型更多存储的是上一帧图像的特征,当处理下一帧图像时,必然会导致目标跟踪不稳定,如图 3 所示。将学习率调至 0.05 时,模型的更新速度跟上了目标的变化速度,因此能够稳定跟踪目标,如图 4 所示。

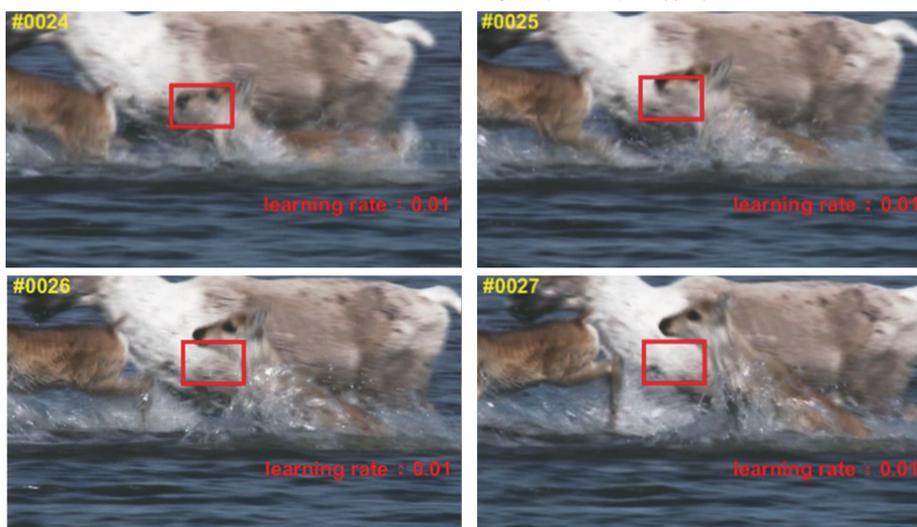


图 3 学习率为 0.01 时的跟踪效果

Fig. 3 Tracking effect with a learning rate of 0.01

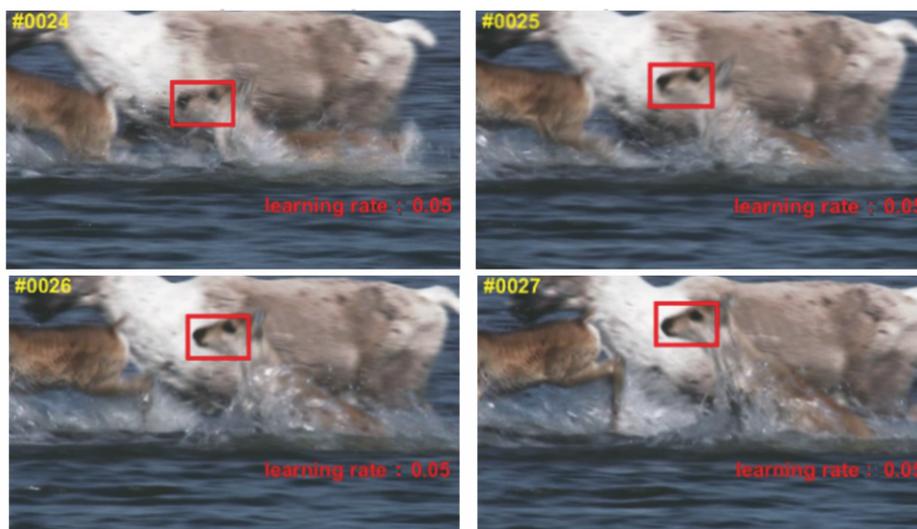


图 4 学习率为 0.05 时的跟踪效果

Fig. 4 Tracking effect with a learning rate of 0.05

针对视频中机动性较强或形变较大的目标,拥有一个稳定的自适应模型更新机制十分重要。因此,本算法基于下一帧候选区域的相对清晰度决定

模型的学习率。为了尽量不影响算法的运算速度,用经典的 Tenengrad 函数^[14]判断候选区域的清晰度。Tenengrad 函数是一种基于图像梯度的函数,

也是一种常用的图像清晰度评价函数。在图像处理中,一般认为比较清晰的图像像素具有更尖锐的边缘,即梯度函数值更大。Tenengrad 函数使用 Sobel 算子提取水平和垂直方向的梯度值,具体步骤如下。

1) 设 Sobel 卷积核为 G_x, G_y , 则图像 I 在点 (x, y) 处的梯度可表示为

$$S(x, y) = \sqrt{G_x * I(x, y) + G_y * I(x, y)} \quad (4)$$

2) 定义 Tenengrad 函数的值为

$$X_{\text{Tenengrad}} = \frac{1}{n} * \sum_x \sum_y S(x, y)^2, \quad (5)$$

式中, n 为评价区域的像素总数。Tenengrad 函数值越大,表明评价区域的清晰度越高;反之,则表明评价区域的清晰度越低。首先,固定一个基本学习率 L_{base} 和一个基本清晰度 C_{base} , 其中,基本清晰度 C_{base} 由首帧帧的清晰度评价值得到。然后,在其余帧中计算基本清晰度与当前帧清晰度的比值,并将该比值与基本学习率的乘积定义为学习率。最后,充分考虑了运动目标过于模糊时由特征提取不准确导致的目标丢失情况,设置了清晰度阈值 T_c , 若清晰度低于阈值 T_c , 则立即将学习率调至为 0, 可表示为

$$\begin{cases} l = L_{\text{base}} \times \frac{C_{\text{base}}}{c}, & c > T_c, \\ l = 0, & c \leq T_c \end{cases} \quad (6)$$

式中, l 为更新的学习率, c 为当前时刻的清晰度值。

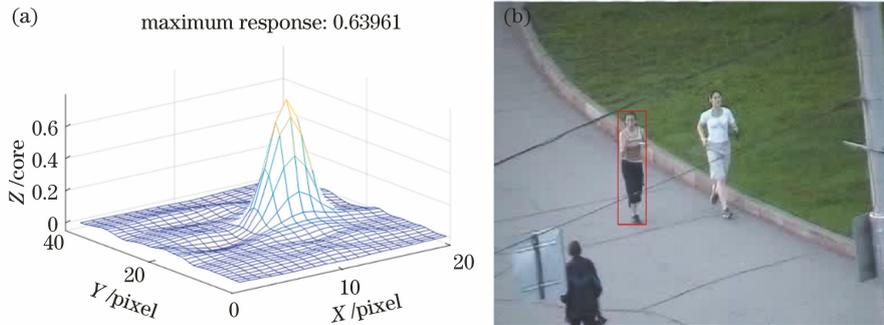


图 5 跟踪良好时的响应图及最大响应值。(a) 响应图; (b) 跟踪效果图

Fig. 5 Response graph and maximum response value when tracking is good. (a) Response graph; (b) tracking effect graph

APCE 值能反映响应图的波动情况,可表示为

$$X_{\text{APCE}} = \frac{|F_{\text{max}} - F_{\text{min}}|^2}{\text{mean} \left[\sum_{w,z} (F_{w,z} - F_{\text{min}})^2 \right]}, \quad (7)$$

式中, F_{max} 为响应值的最大值, F_{min} 为响应值的最小值, $F_{w,z}$ 为响应图对应位置的响应值, w 和 z 分别为水平和垂直方向的像素坐标。响应图的波动越强烈, APCE 值就越小;反之, APCE 值就越高。跟踪情况良好和出现遮挡时的响应图对应的 APCE

2.3 APCE 重检机制

若目标在连续运动过程中出现严重遮挡或加速度过大使目标变得模糊时,也会导致特征提取不准确,无法进行长时间跟踪。目标跟踪的特点是一旦模型被污染,将无法再次追回,需要手动重新抓取目标,大大降低了算法的整体成功率,使其难以在实际工程中应用。因此,建立合适的目标重检机制具有重要意义。为了建立正确的重检机制,必须准确判断目标何时丢失,以节约计算资源,提高运算速率。为了实现这一目标,本算法将文献[8]提出的 APCE 与最大响应作为跟踪稳定性的判断条件。

用滤波模板进行相关计算后,可得到目标区域与模型的特征响应图。该响应图能描述模型与目标区域的相似程度,若响应图呈单峰分布,表明跟踪情况良好,且峰值点位置为探测区域的目标中心点,其对应数值也被称为最大响应值,如图 5 所示,此时,最大响应值处于一个较高水平,为 0.63961,跟踪情况相对稳定。其中,core 为响应值得分。当特征响应图的最大值较小时,表明跟踪已经遇到了困难,如图 6 所示,此时,最大响应值为 0.21811。但在遇到相似物邻近或遮挡导致的跟踪框漂移情况下,最大响应值不会一直保持很低的水平,此时,算法可能无法判断跟踪出现困难。因此,最大响应值仅是判定跟踪稳定的必要条件之一,仅使用单一方法作为判定条件是不完全准确的。

值如图 7、图 8 所示,可以发现,出现遮挡等跟踪困难的情况时,响应图的波动明显较大,且 APCE 值呈较低水平,此时跟踪已经不稳定。综上所述,虽然无法得到准确判断跟踪稳定的充要条件,但结合两个必要条件也能在一定程度上判断跟踪的稳定程度,因此,结合最大响应值与 APCE 值可作为目标检测的判断条件。

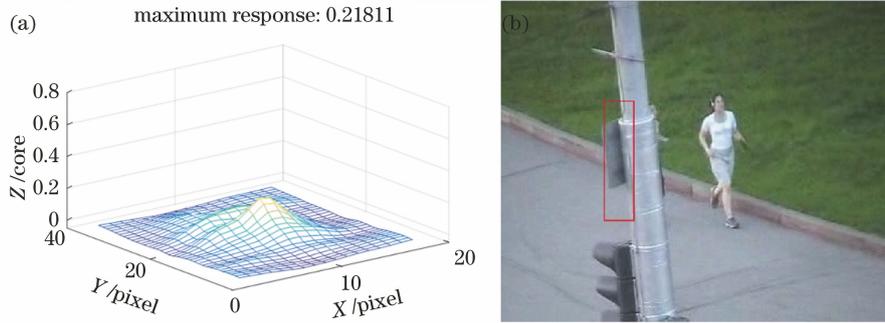


图 6 跟踪困难时的响应图及最大响应值。(a)响应图;(b)跟踪效果图

Fig. 6 Response graph and maximum response value when tracking is difficult. (a) Response graph; (b) tracking effect graph

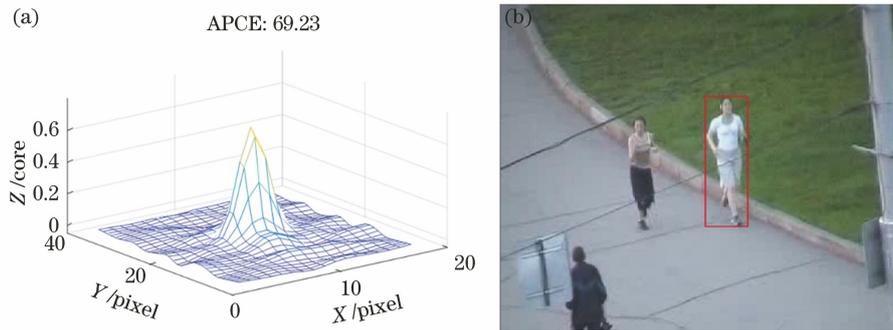


图 7 跟踪良好时的响应图及 APCE 值。(a)响应图;(b)跟踪效果图

Fig. 7 Response graph and APCE value when tracking is good. (a) Response graph; (b) tracking effect graph

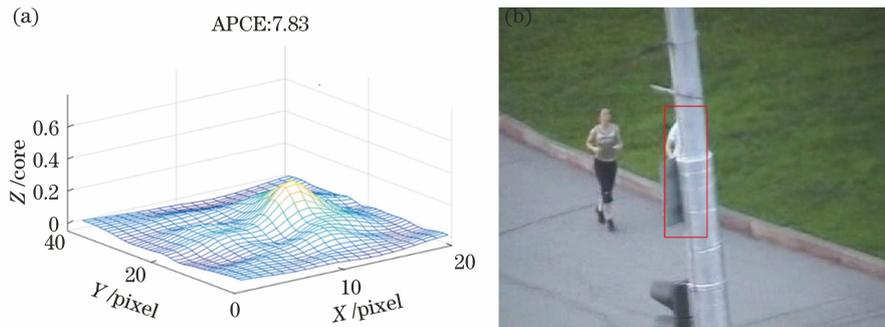


图 8 跟踪困难时的响应图及 APCE 值。(a)响应图;(b)跟踪效果图

Fig. 8 Response graph and APCE value when tracking is difficult. (a) Response graph; (b) tracking effect graph

本算法使用现阶段速度与精度兼顾且效果较好的 YOLOv4^[15] 检测算法,由于 YOLOv4 为基于深度学习的目标检测算法,需要在算法应用之前对跟踪目标的类别进行训练,即该算法无法对未知种类的目标进行检测和跟踪。用 YOLOv4 算法检测出的目标边界框会有一个置信度得分,当该得分大于设定阈值 T_{detect} 时,程序自动判断检测成功,算法转入跟踪阶段,且检测的边界框会作为跟踪的初始框。当连续几帧图像的最大响应值与 APCE 值均小于预先设定的阈值 $T_{\text{max_response}}$ 和 T_{APCE} 时,可认为此时的跟踪已经发生漂移,则开启重检模式,再次进入检测算法。

3 实验结果与性能评估

3.1 自适应模型更新机制

为了验证自适应模型更新算法的有效性,选用 OTB100 测试集作为本算法的测试数据,OTB50 和 OTB100 数据集是 Wu 等^[16-17] 先后在 2013 年和 2015 年公布的数据集,共包含 100 种目标跟踪测试集,且融入了目标遮挡、目标形变、光照变化、尺度变化、运动模糊、快速运动等复杂情况。OTB100 数据集还提出了具体的跟踪精度判别指标,即一次通过评估(OPE)。OPE 包括两种判别方式,第一种是距离精度,即平均精度(Precision)曲线,可以描述预测

框中心与真实框中心距离小于一定像素数的视频帧数占视频总帧数的百分比,该距离精度也被称为位置误差阈值。第二种是面积精度,即成功率曲线。可以描述预测框与真实框交集面积与二者并集面积之比大于一定比率的视频帧数占视频总帧数的百分比,该比率阈值也称为面积比率阈值。平均精度和成功率可表示为

$$X_{\text{precision}} = \frac{\hat{F}_{|c_1 - c_2| < L}}{F}, \quad (8)$$

$$X_{\text{success}} = \frac{\hat{F}_{\frac{s_1 \cap s_2}{s_1 \cup s_2} > T}}{F}, \quad (9)$$

式中, $X_{\text{precision}}$ 为平均精度, X_{success} 为成功率, F 为测

试视频总帧数, \hat{F} 为跟踪成功的帧数, C_1 和 C_2 分别为测试框中心和真实框中心的像素值, L 为位置误差阈值,一般取为 20, S_1 与 S_2 分别为测试框与真实框的面积, T 为面积比率阈值,一般取为 0.5。

实验使用的硬件平台:处理器为 i5-4430、主频为 3.0 GHz 的 CPU。将本算法融入经典的 DSST 算法中并将基本学习率 L_{base} 设为 0.02,最低清晰度阈值 T_c 设为 3.0。用 OTB100 数据集中包含运动机动性较强物体的测试帧进行实验,加入自适应模型更新机制的 DSST 算法与原始 DSST 算法的跟踪结果如图 9 所示。可以发现,未加入自适应模型更新机制的 DSST 算法目标跟踪失败,而加入自适应模型更新机制的算法能成功跟踪目标。

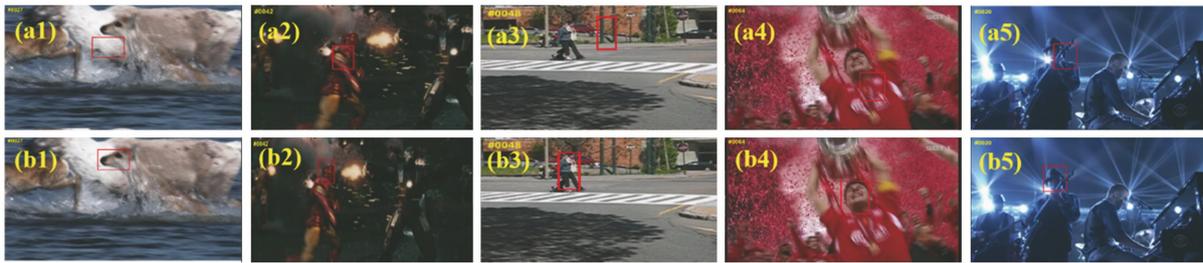


图 9 不同算法对快速移动物体的跟踪结果。(a)固定模型;(b)自适应模型

Fig. 9 Tracking results of fast motion objects by different algorithms. (a) Fixed model; (b) adaptive model

对 OTB100 数据集中的所有数据进行测试,OTB100 数据集中的 100 组测试数据已由其团队全部标注好,并发布了计算 OPE 的代码框架。不同算法经过 OTB100 数据集的测试代码得到的两个 OPE 评价曲线如图 10 所示,对比算法包括 CSK 算法、单目标长时间跟踪(TLD)算法、KCF 算法、LCT 算法、DSST 算法以及加入自适应模型机制的 DSST 算法(本算法)。从图 10(a)可以发现,DSST 算法的平均精度为 69.3%,而本算法的平均精度为 87.6%,比 DSST 算法提高了 18.3 个百分点,而 LCT、KCF、TLD、CSK 算法的平均精度分别为

85.3%,71.9%,60.3%,53.2%,均低于本算法。从图 10(b)可以发现,DSST 算法的成功率为 61.5%,而本算法的成功率为 76.7%,比 DSST 算法提高了 15.2 个百分点,而 LCT、KCF、TLD、CSK 算法的成功率分别为 76.9%,63.7%,49.6%,43.3%。虽然本算法的整体成功率略低于 LCT 算法,但本算法中的机制可移植性较强。

图 11~图 13 为 OTB100 数据集中所有快速移动(fast motion)、移动模糊(motion blur)及快速形变(rapid deformation)目标的 OPE 测试曲线。可以发现,在快速移动目标帧序列上,DSST 算法的平

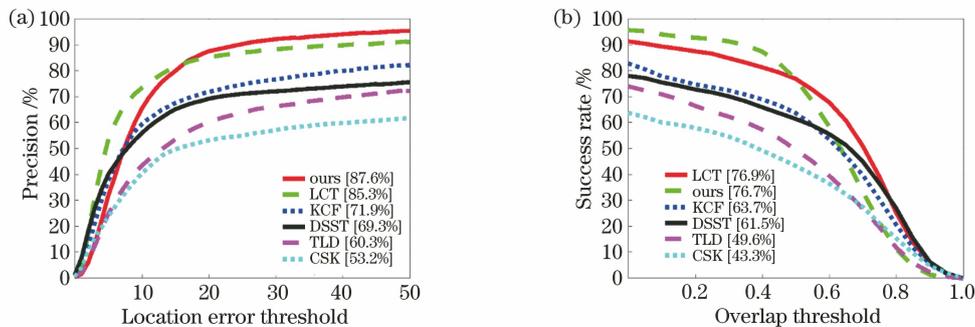


图 10 不同算法的 OPE 曲线。(a)平均精度;(b)成功率

Fig. 10 OPE curves of different algorithms. (a) Average precision; (b) success rate

均精度和成功率分别为 53.2% 和 54.3%，而本算法的平均精度和成功率分别为 80.8% 和 73.7%；在所有模糊运动目标帧序列上，DSST 算法的平均精度和成功率分别为 48.7% 和 50.3%，而本算法的平均精度和成功率分别为 81.1% 和 71.0%；对于快速变

形的目标序列，DSST 算法的平均精度和成功率分别为 56.8% 和 56.5%，而本算法的平均精度和成功率分别为 87.7% 和 74.7%。这表明本算法中使用的机制能在一定程度上改善目标运动机动性强或非对称刚体目标快速变形等情况导致的目标丢失情况。

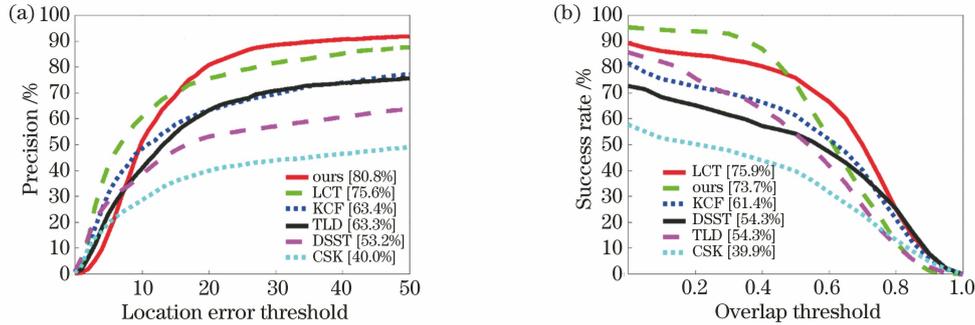


图 11 不同算法在快速移动目标序列中的 OPE 曲线。(a)平均精度;(b)成功率

Fig. 11 OPE curves of different algorithms in fast motion target sequences. (a) Average precision; (b) success rate

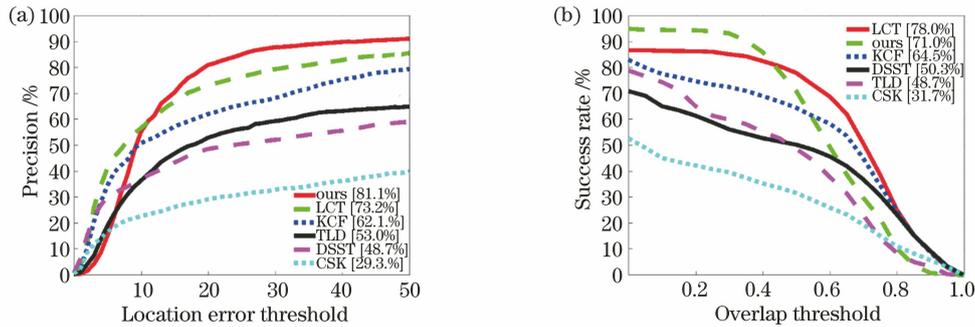


图 12 不同算法在移动模糊目标序列中的 OPE 曲线。(a)平均精度;(b)成功率

Fig. 12 OPE curves of different algorithms in the motion blur target sequence. (a) Average precision; (b) success rate

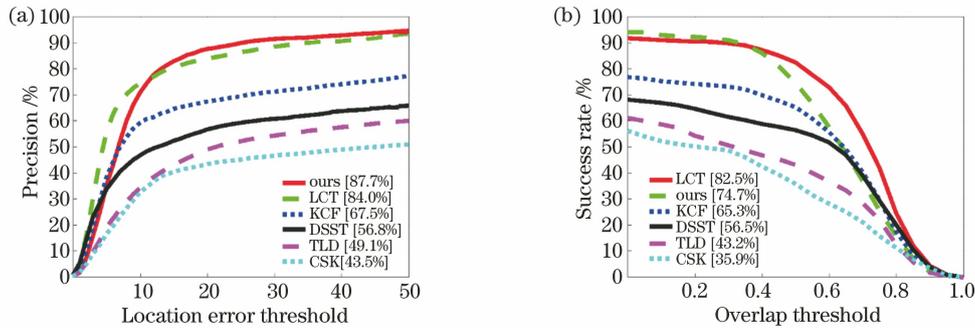


图 13 不同算法在快速变形目标序列上的 OPE 曲线。(a)平均精度;(b)成功率

Fig. 13 OPE curves of different algorithms in the rapid deformation target sequences. (a) Average precision; (b) success rate

将图 10~图 13 中的数据整理到表 1 中,同时给出了不同算法在实验平台中的运算速率,可以发现,相比 DSST 算法,本算法的计算速率略有降低,但仍能满足实时性的要求,其中,FPS 表示算法在一秒内能处理的帧数。

3.2 目标重检机制及其实用性分析

为了验证本算法中自适应模型与重检机制的可

行性及实用性,选取包含滑雪运动员的视频帧序列和手持手机的快速移动视频帧序列(以滑雪运动员和手机为目标)各 5 个,共包含约 10500 帧数据。测试显卡的 GPU 为 NVIDIA Titan 2080Ti,CPU 为 i7-6700。将最大响应值稳定性的判定阈值 $T_{\max_response}$ 设为 0.3,APCE 稳定性判定阈值 T_{APCE} 设为 30,目标重检判定阈值 T_{detect} 设为

表 1 不同算法的性能分析

Table 1 Performance analysis of different algorithms

Algorithm		Ours	LCT	DSST	KCF	CSK	TLD
Fast motion	precision /%	80.8	75.6	53.2	63.4	40.0	63.3
	success rate /%	73.3	75.9	54.3	61.4	39.9	54.3
Motion blue	precision /%	81.1	73.2	48.7	62.1	29.3	53.0
	success rate /%	71.0	78.0	50.3	64.5	31.7	48.7
Rapid deformation	precision /%	87.7	84.0	56.8	67.5	43.5	49.1
	success rate /%	74.7	82.5	65.3	56.5	35.9	43.2
Precision /%		87.6	85.3	69.3	71.9	53.2	60.3
Success rate /%		76.7	76.9	61.5	63.7	43.3	49.6
Speed /FPS		37.5	30.4	40.5	48.1	26.9	21.7

0.85, 部分帧的测试结果如图 14 所示。可以发现, 前 3 帧算法并没有检测到运动员, 因此, 第 2 帧以前未发现边界框, 在第 3 帧图像中出现 YOLOv4 算法边界框, 证明检测成功。在此之后, 跟踪算法对该边界框内的 skater 进行特征提取, 并开始制作模型。

第 4 帧以后跟踪算法开始运行, 并出现了代表跟踪算法的边界框。在第 65 帧时, 由于运动速度变化过快, 导致了模型污染, 触发了重检判断机制。在第 66 帧图像中算法重新检测到了 skater, 从而完成了检测→跟踪→检测的闭环过程。



图 14 本算法对滑雪运动员数据集的测试结果

Fig. 14 Test results of our algorithm on the skier data set

图 15 为本算法对测试集中跟踪困难情况的跟踪效果图(包含高山滑雪运动员和快速移动的手机)。图 15(a)中包含运动员和手机运动机动性强的情况, 从第 116 和 662 帧数据可以发现, 当运动员出现急转弯及手机快速左右摇晃时, 目标会变得模糊, 导致跟踪出现不稳定, 触发了重检机制, 从而在下一帧中立即检测到新的目标, 重新画出边界框; 图 15(b)中第 249 和 230 帧数据中目标已经出现了明显的运动变形, 触发重检机制后也可以重新捕获

目标; 图 15(c)中第 672 和 100 帧数据分别出现了无边界框情况, 原因是在触发检测机制后没有检测到目标, 第 100 帧图像由于 YOLOv4 算法本身出现了漏检情况, 而第 672 帧图像中有障碍物(旗杆)完全遮挡了目标, 使检测算法无法识别。但在下一帧后, 本算法也立即检测到了该目标, 验证了本算法具有一定的连续性跟踪和抗遮挡能力; 图 15(d)中本算法出现了错跟、误跟情况, 此时, 也能立即触发重检机制。

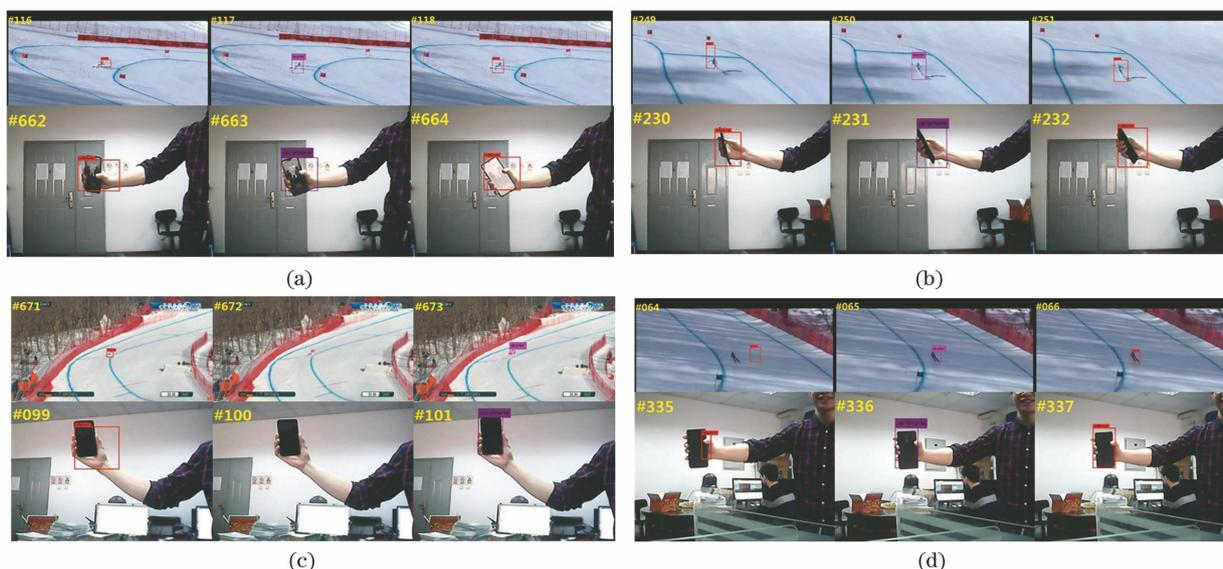


图 15 重检机制的验证结果。(a)目标模糊;(b)目标变形;(c)连续性差;(d)错位跟踪

Fig. 15 Verification result of the recheck mechanism. (a) Target is blurred; (b) target is deformed; (c) poor continuity; (d) dislocation tracking

对上述采集的测试集标注目标真实框,并效仿 OPE 的评价方法,得到了关于自制数据集的评价结果(包含 skater 和 phone 的所有测试数据),其 OPE 曲线如图 16 所示。可以发现,仅有自适应模型的 DSST 算法(adapt)面对目标机动性更强、变形更快的视频帧的平均精度和成功率均处于较低水平,在位置误差阈值小于 20 的精度仅有

45.5%,面积比率阈值大于 0.5 处仅有 45.7%。原因是面对这类跟踪困难的情况,模型极易跟丢目标,且很难再次重新找回;而加入重检机制(adapt+redetect)后,即便是跟踪十分困难的情况,依旧能够重新找回目标,从而进行稳定跟踪,在自制数据集测试集中本算法的平均精度和成功率分别为 86.7%和 82.1%。

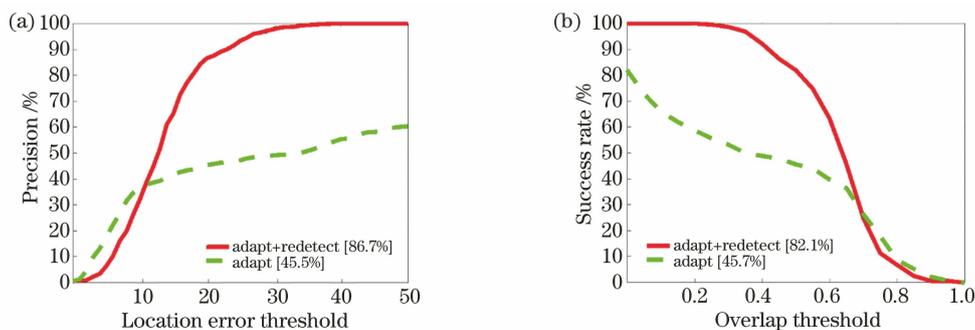


图 16 本算法在自制数据集上的 OPE 曲线。(a)平均精度;(b)成功率

Fig. 16 OPE curve of our algorithm on the self-made data set. (a) Average precision; (b) success rate

4 结 论

为了解决视频中目标运动机动性强或非对称刚体目标快速变形等情况导致的目标丢失问题,提出了一种基于相关滤波的自适应模型与 APCE 重检机制的视频跟踪算法。为了验证自适应模型机制的有效性,将自适应模型跟踪机制融入 DSST 跟踪算法,并将融入该机制后的 DSST 算法在整个 OTB100 数据集上进行测试,得到跟踪结果的平均

精度为 87.6%,成功率为 76.7%。将融入 DSST 前后的算法与当前比较流行的目标跟踪算法进行对比,验证了本算法的有效性及准确性。最后,将本算法应用到实际场景中进行验证,实现了检测→跟踪→检测的闭环过程。虽然加入目标检测算法后仅能对特定类别目标进行跟踪,牺牲了算法应用的广泛性,但该机制在工程应用上可实现长时目标跟踪,能够在特定场景下解决实际问题。

参 考 文 献

- [1] Meng L, Yang X. A survey of object tracking algorithms [J]. *Acta Automatica Sinica*, 2019, 45(7): 1244-1260.
孟球, 杨旭. 目标跟踪算法综述 [J]. *自动化学报*, 2019, 45(7): 1244-1260.
- [2] Shi L C, An Y L, Su B H, et al. An improved background subtraction algorithm based on Kalman filtering [J]. *Laser & Optoelectronics Progress*, 2018, 55(8): 081003.
施龙超, 安玉磊, 苏秉华, 等. 一种改进的基于卡尔曼滤波的背景差分算法 [J]. *激光与光电子学进展*, 2018, 55(8): 081003.
- [3] Yang G W, Yan S M, Wang Y Z. V-shaped seam tracking based on particle filter with histogram of oriented gradient [J]. *Chinese Journal of Lasers*, 2020, 47(7): 0702002.
杨国威, 闫树明, 王以忠. 基于方向梯度直方图粒子滤波的 V 型焊缝跟踪 [J]. *中国激光*, 2020, 47(7): 0702002.
- [4] Vojir T, Noskova J, Matas J. Robust scale-adaptive mean-shift for tracking [J]. *Pattern Recognition Letters*, 2014, 49: 250-258.
- [5] Zhang Z, Sun J, Yang L T. Tracking algorithm based on correlation filter fusing with keypoint matching [J]. *Acta Optica Sinica*, 2019, 39(2): 0215001.
张哲, 孙瑾, 杨刘涛. 融合相关滤波与关键点匹配的跟踪算法 [J]. *光学学报*, 2019, 39(2): 0215001.
- [6] Ge B Y, Zuo X Z, Hu Y J. Long-term object tracking based on feature fusion [J]. *Acta Optica Sinica*, 2018, 38(11): 1115002.
葛宝义, 左宪章, 胡永江. 基于特征融合的长时目标跟踪算法 [J]. *光学学报*, 2018, 38(11): 1115002.
- [7] Wang H Y, Wang L, Yin W R, et al. Multi-scale correlation filtering visual tracking algorithm combined with target detection [J]. *Acta Optica Sinica*, 2019, 39(1): 0115004.
王红雨, 汪梁, 尹午荣, 等. 结合目标检测的多尺度相关滤波视觉跟踪算法 [J]. *光学学报*, 2019, 39(1): 0115004.
- [8] Henriques J F, Caseiro R, Martins P, et al. Exploiting the circulant structure of tracking-by-detection with kernels [M]//Fitzgibbon A, Lazebnik S, Perona P, et al. *Computer vision - ECCV 2012. Lecture notes in computer science*. Heidelberg: Springer, 2012, 7575: 702-715.
- [9] 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.
- [10] 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, September, 2014, Nottingham*. British: BMVA Press, 2014: 1-11
- [11] 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.
- [12] Ma C, Yang X K, Zhang C Y, et al. Long-term correlation tracking [C]//*2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 7-12, 2015, Boston, MA, USA*. New York: IEEE Press, 2015: 5388-5396.
- [13] 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.
- [14] Fang L P, Chen Z X, Chen P, et al. Focus evaluation for automatic recognition in corneal curvature measurement [J]. *Optical Technique*, 2019, 45(2): 218-223.
方丽萍, 陈兆学, 陈鹏, 等. 角膜曲率测量中自动识别的对焦评价 [J]. *光学技术*, 2019, 45(2): 218-223.
- [15] Bochkovskiy A, Wang C Y, Liao H Y M. YOLOv4: optimal speed and accuracy of object detection [EB/OL]. (2020-04-23)[2021-03-01]. <https://arxiv.org/abs/2004.10934>.
- [16] Wu Y, Lim J, Yang M H. Online object tracking: a benchmark [C]//*2013 IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2013, Portland, OR, USA*. New York: IEEE Press, 2013: 2411-2418.
- [17] Wu Y, Lim J, Yang M H. Object tracking benchmark [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(9): 1834-1848.