复杂情况下自适应特征更新目标跟踪算法

尹宽[1]，李均利[1*]，李丽[1]，储诚曦[2]

[1]四川师范大学计算机科学学院，四川 成都 610101；

[2]宁波大学信息科学与工程学院，浙江 宁波 315211

摘要 为提高复杂情况下目标跟踪的稳健性，提出一种自适应特征更新的目标跟踪算法。对目标提取分级深度特征和手工设计特征，通过不同线性组合方式进行多特征融合，构建多个融合特征器；对不同融合特征器进行可信度判定，选择可信度最高的融合特征作为当前帧的跟踪特征，构建位置相关滤波器，预测出当前帧的目标位置；对跟踪结果进行可靠性检测，可靠性低于阈值则启动融合特征器更新机制，加入时序信息和语义信息进行重跟踪，降低了模型的误差累积。在 OTB-2013 和 OTB-2015 数据库上进行测试，结果表明，与近年来比较流行的 9 种算法相比，提出的算法在快速运动、背景杂波、运动模糊、形变等复杂情况下具有较高的成功率和较好的稳健性。

关键词 机器视觉；目标跟踪；分级深度特征；相关滤波；时序信息

中图分类号 TP391.4 **文献标识码** A **doi**：10.3788/AOS201939.1115002

Adaptive Feature Update Object-Tracking Algorithm in Complex Situations

Yin Kuan[1], Li Junli[1*], Li Li[1], Chu Chengxi[2]

[1] *College of Computer Science, Sichuan Normal University, Chengdu, Sichuan* 610101, *China*;

[2] *Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo, Zhejiang* 315211, *China*

Abstract To improve the robustness of object tracking in complex situations, a new algorithm based on adaptive feature updating is proposed. First, hierarchical deep and hand-crafted features are simultaneously extracted from the object, and multiple fusion feature experts are constructed through multi-feature fusion by using different linear combination methods. Second, the credibility score of each expert is computed and the highest score is selected as the tracking feature of the current frame. A position correlation filter is then constructed to predict the frame′s target position. Finally, the reliability of the tracking result is detected. When this reliability is found to be lower than a certain threshold, the fusion feature updating mechanism is initiated, and the temporal and semantic informations are added to the re-track, which reduces the error accumulation of the model. The proposed algorithm is tested on OTB-2013 and OTB-2015 datasets, and the obtained results are compared with those of 9 recently developed popular algorithms. Our proposed algorithm demonstrates a higher success rate and better robustness in complex situations, such as fast motion, background clutter, motion blur, and deformation, than existing algorithms.

Key words machine vision; object tracking; hierarchical deep feature; correlation filter; temporal information

OCIS codes 150.0155; 100.4999; 100.3008

1 引 言

目标跟踪是计算机视觉领域近年来的研究热点，同时也是各种视频应用中的一个基本任务[1]，广泛应用于智能视频监控、智能交通系统、智能视觉导航、现代化军事、人机交互等领域[2]。现实场景和目标自身存在的复杂性和不确定性[3]，例如场景的光照变化、角度变化、遮挡情况和目标运动过程中出现的姿态变化、尺度变化，给目标跟踪带来了一系列挑战，使得目标跟踪的效果受到一定的影响。因此，如何实现复杂情况下稳健的目标跟踪成为一个值得关注的研究课题。

收稿日期：2019-05-31；**修回日期**：2019-07-04；**录用日期**：2019-07-15

基金项目：国家自然科学基金(61403266,61403196)

* **E-mail**：li.junli@vip.163.com

根据不同的目标建模方式，目标跟踪算法分为基于生成式模型方法和基于判别式模型方法。基于生成式模型方法的思想是对目标进行建模或特征提取，在后续视频序列中进行目标模型与候选目标的相似度计算，相似度最高的候选目标即视为当前帧的跟踪目标。这种方法仅对目标进行建模，没有充分利用背景信息，在跟踪过程中存在一定的局限性，具体表现在当目标外观快速变化或存在遮挡时跟踪效果欠佳。基于判别式模型方法则同时考虑目标信息和背景信息，将跟踪视为一个二分类问题，构建一个稳健的分类器，将目标与背景有效地区分出来，从而实现目标的跟踪。相对于基于生成式模型方法，基于判别式模型方法具有更加稳健的目标跟踪效果，所以这种方法逐渐成为了目标跟踪的主流方法。

基于相关滤波的跟踪算法是一种判别式模型方法，2010 年 Bolme 等[4]提出一种最小输出平方误差滤波器(MOSSE)算法，将相关滤波的思想首次引入目标跟踪领域，把信号域的相关性计算应用在跟踪器中，把跟踪的问题转换为求解两个图像块相似度的问题，相似度最高的响应位置即为跟踪目标的当前位置，提高了滤波器的准确度，同时也将时域计算转换为频域计算，提高了计算的速度，使得算法的速度达到了 669 frame/s。此后，基于相关滤波的跟踪成为目标跟踪的主流方向。针对相关滤波中样本数量不足的问题，Henriques 等[5]在跟踪器中引入循环矩阵思想扩充跟踪样本，提出核函数循环结构跟踪(CSK)算法，提升了跟踪性能；Zhang 等[6]基于跟踪过程中的时空上下文，提出了时空上下文跟踪(STC)算法，对跟踪目标上下文区域的时空关系用贝叶斯框架进行建模，得到目标和邻近区域低级特征统计相关性，通过置信图评估得到目标在新一帧中的位置，提高了跟踪的稳健性；Danelljan 等[7]通过在跟踪算法中加入一个位置滤波器和尺度滤波器，不仅实现了目标的跟踪，还实现了对目标精准的尺度估计。在跟踪算法中，特征的选取在某些程度上对算法的优劣起着关键作用，在 2014 年之前，大部分算法均采用手工特征，如：颜色空间(CN)、梯度直方图(HOG)等。Wang 等[8]提出的深度学习跟踪(DLT)算法是第一个将深度学习思想引入目标跟踪领域的算法，此后深度特征也开始在跟踪领域崭露头角。Danelljan 等[9]利用深度特征进行跟踪，取得了很好的稳健性和准确性；Wang 等[10]提出一种线性组合的多特征融合策略，有效提高了跟踪的效果。

相关滤波虽然以较高的速度和精度在目标跟踪领域得到了广泛应用，但对于在运动过程中的目标易产生快速变形或者目标运动过快等情况时易导致跟踪漂移[11]。单一特征跟踪时，不能对复杂情况下的目标进行稳健的描述，导致跟踪失败；用多特征融合策略时，不明确的特征融合方式不能较好地实现跟踪效果，也会产生较大的计算开销[12]。

针对以上问题，结合多特征融合相关滤波思想，以多线索相关滤波跟踪(MCCT)[10]算法作为基本框架，提出一种复杂情况下的自适应特征更新跟踪方法。针对手工设计特征和深度特征对目标不同侧重表征的特点，将手工设计特征和深度特征进行融合，运用不同的特征组合方式进行线性加权运算，得到不同的融合特征，对融合特征的可信度进行判定，最终选取最可靠特征作为跟踪特征。针对复杂情况，如遮挡、消失后重现、目标形变等，提出一个新的融合特征更新机制，利用目标在连续时间内的相关性，将当前帧特征与时序稳健特征进行线性加权计算，得出更新后的融合特征。

本文的贡献主要在于采用稳健的分级深度特征选取以及融合方式，考虑不同层级深度特征的不同特点，在不同跟踪情况下提取特定的分级深度特征，并采取不同的融合方式，同时增加手工设计特征作为补充，提高定位精度；计算 PSR 评分和融合特征可靠性评分，将两者综合考虑以判别跟踪效果优劣，以此找到跟踪效果欠佳的视频帧，防止持续累积误差造成后续帧的性能下降；考虑时间序列上跟踪目标的稳定性，加入时序特征更新融合机制，稳健应对复杂跟踪情况。不同于直接的特征融合，本文算法充分且稳定地利用了多信息的互补性，并考虑时序上的特征稳定性，通过多特征计算得到滤波响应，用融合响应预测目标的位置。在 OTB-2013 和 OTB-2015 标准数据库上进行大量实验，实验结果验证了本算法切实可行，在复杂情况下跟踪精度有较好的提升。

2　算法概述

提出的算法主要分为 5 个部分：1)将给定视频序列第一帧中标定的目标候选区域位置图像块输入去掉全连接层的预训练深度特征网络 VGG-19 中，提取对应图像的 conv3-4、conv4-4、conv5-4 的深度卷积特征图，同时提取对应图像的 HOG 特征；2)利用线性组合方式，对提取的对应图像的 conv4-4、conv5-4、HOG 三种特征进行加权融合，得到 7 种不

同融合方式的融合特征，以相关性作为融合特征质量评价指标，选出最佳融合特征；3）在进行第 t 帧跟踪时，利用 t-1 帧跟踪结果作为位置中心，确定第 t 帧跟踪的搜索框范围；4）利用最佳的融合特征与搜索范围内提取的融合特征进行相关计算，根据快速傅里叶变换进行滤波器训练和响应图计算，得到响应值最大的点，即当前帧目标的中心点，再利用尺度滤波器计算预测得到最佳跟踪框；5）计算跟踪结果的质量评价指标，判别跟踪是否存在遮挡、快速形变等复杂情况，如遇复杂情况，利用时序稳定性模型和目标语义信息对模型进行更新。算法的整体框图如图 1 所示。

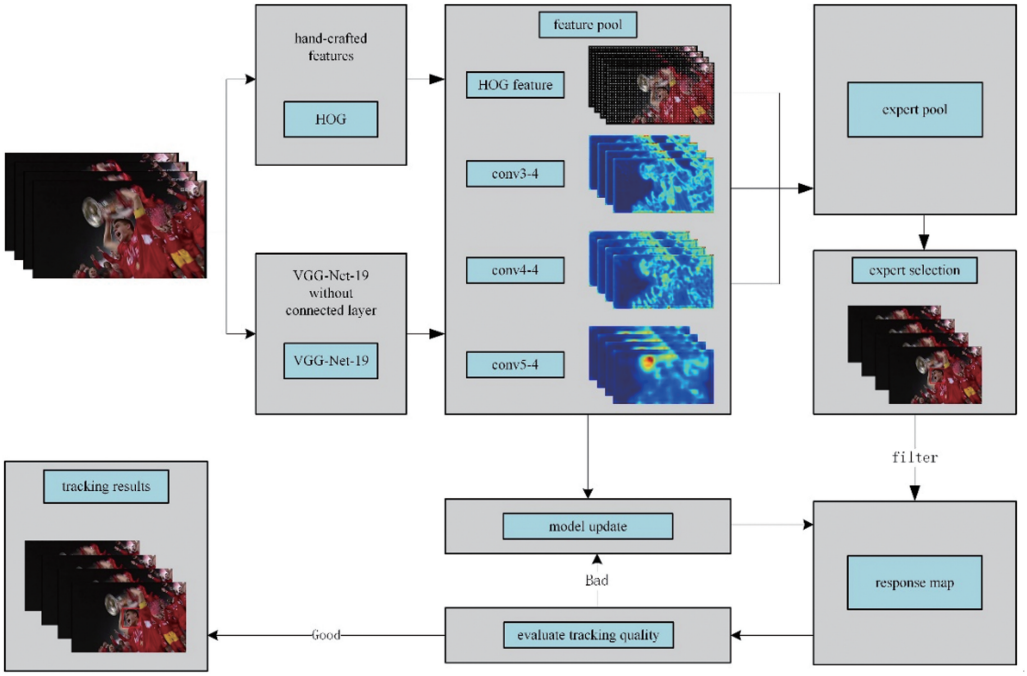

图 1　算法框架图

Fig. 1　Framework of proposed algorithm

3　复杂情况下自适应特征更新目标跟踪算法

3.1　目标特征表征

不同的特征能够从不同角度反映目标的特点，而不同的特征又各有优劣。经典的跟踪算法大都采用手工设计特征，如 HOG[13]、Color Names[14] 等，自深度学习研究得到发展以来，通过深度神经网络提取出来的深度特征在跟踪中也得到了广泛应用。跟踪中常用 VGG（Visual Geometry Group）[15] 作为特征提取网络，深度特征和手工设计特征在跟踪中各有不同侧重点[16]：深度特征含有高层语义信息，但是分辨率低，在跟踪时更注重跟踪的稳健性；而手工设计特征是低层高分辨率的特征，更加强调跟踪的精度，但是在目标外观变化较大时会出现对手工设计特征的跟踪失败。在跟踪中，对当前帧的目标特征进行更新以进行下一帧的跟踪，不管是当前帧结果的稳健性还是精度，都会对特征更新产生一定影响。根据误差传播理论，每一帧的不可靠的跟踪产生的累计误差不断地迭代，均会对之后的跟踪效果产生一定的影响，所以需要在稳健性和精度之间取得一定的平衡，基于利用手工设计特征对深度特征进行补充的特征融合方法，提高了跟踪的效果。

在特征选择方面，HOG 特征能够很好地描述局部信息，以及具有几何、光学不变性，所以本文算法的手工设计特征选择 HOG 特征。通过计算和统计图像归一化的局部区域梯度方向直方图来构成 HOG 特征。随着层数的加深，深度特征的特征分辨率会越来越低，但包含的语义信息则更加丰富。本文算法中的深度特征采用的是去掉全连接层后的 VGG-Net-19 网络的 conv3-4、conv4-4 层和 conv5-4 层提取出来的三种分级深度特征，特征可视化图像如图 2 所示，图 2（a）为原始图像，图 2（b）为 HOG 特征可视化图像，图 2（c）为 conv3-4 特征可视化图，图 2（d）为 conv4-4 特征可视化图，图 2（e）为 conv5-4 特征可视化图。

在特征融合方面，正常情况下进行跟踪时，将 HOG 特征与 conv4-4 层和 conv5-4 层的特征以线

性组合的方式进行多特征融合[10]，通过 $C_3^{(1)}+C_3^{(2)}+C_3^{(3)}=7$ 的组合方式得到 7 个融合特征器，$C_n^{(m)}$ 表示从 n 个对象中不重复地选出 m 个对象的所有组合方式。对 7 个融合特征器进行可靠性评价，选择最可靠的特征进行跟踪，特征融合形式如图 3 所示。

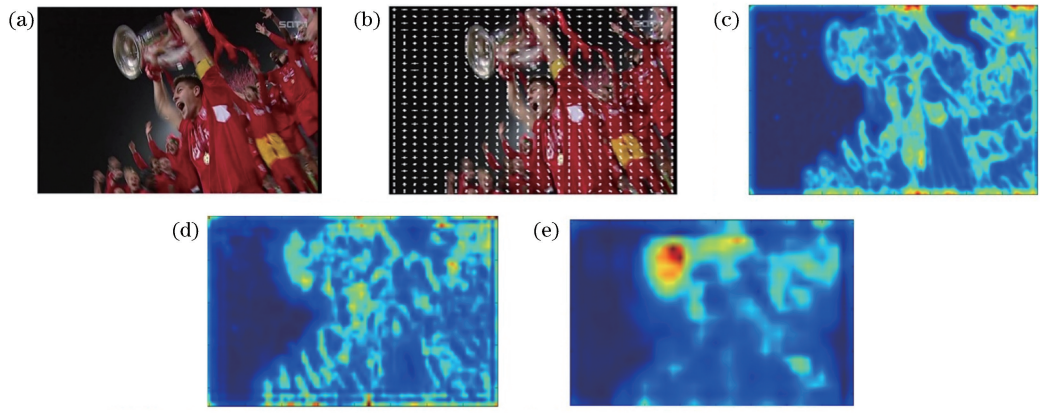

图 2　目标特征可视化。(a)原始图像；(b) HOG 特征；(c) conv3-4 特征；(d) conv4-4 特征；(e) conv5-4 特征

Fig. 2　Target-feature visualization. (a) Original image; (b) HOG feature; (c) conv3-4 feature; (d) conv4-4 feature; (e) conv5-4 feature

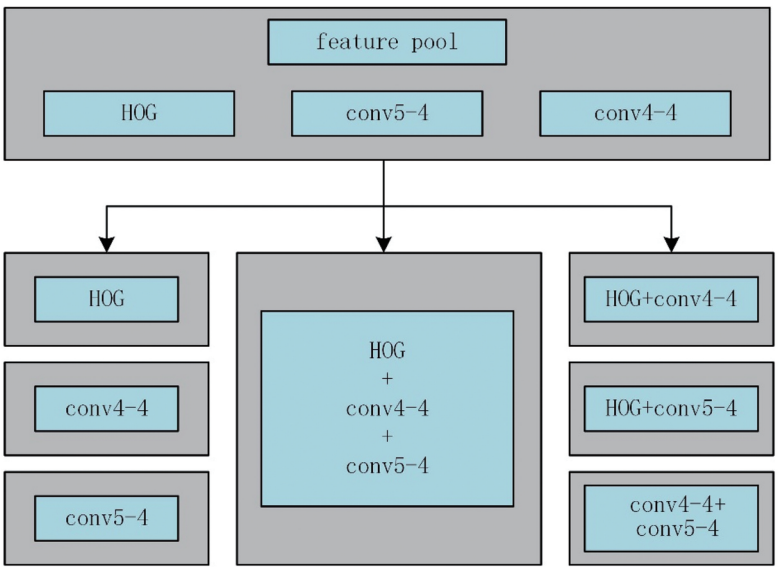

图 3　特征融合示意图

Fig. 3　Schematic of feature fusion

3.2　基于投票的融合特征选择

在跟踪过程中，7 个融合特征器分别单独地对目标进行特征的提取和融合，构成相互独立的特征器以进行后续跟踪。在出现跟踪效果欠佳时，构建一个更新融合特征器进行重跟踪，所以，在正常跟踪过程中，需要从 7 个相互独立的融合特征器中选出 1 个融合特征器进行稳健的跟踪。采用投票的方法选择最优融合特征器进行目标跟踪，从而实现稳健的跟踪效果。

鉴于 7 个融合特征器都是对同一跟踪目标进行的特征提取和融合，所以这 7 个特征融合器之间应具有高度一致性。对 7 个融合特征器两两进行一致性评分计算，将其中与其余 6 个特征器一致性最高的融合特征器作为当前帧跟踪中采用的特征器。融合特征器间一致性的计算表达式为

$$O_{(E_i,E_j)}^{(t)}=\frac{A(B_{E_i}^{(t)}\cap B_{E_j}^{(t)})}{A(B_{E_i}^{(t)}\cup B_{E_j}^{(t)})},\tag{1}$$

式中：$\cap$ 为求交集运算；$\cup$ 为求并集运算；A 为求得的面积；E_i 为第 i 个融合特征器；$B_{E_i}^{(t)}$ 为在跟踪过程的第 t 帧中，第 i 个融合特征器跟踪的结果，即跟踪目标当前结果的预测目标边框；$O_{(E_i,E_j)}^{(t)}$ 为在跟踪过程第 t 帧中，第 i 个融合特征器和第 j 个融合特征器跟踪结果的交并比。(1)式是对不同融合特征器跟踪结果进行交并比(IOU)计算。

为减小 $O^{(t)}_{(E_i,E_j)}$ 值的波动幅度，对 $O^{(t)}_{(E_i,E_j)}$ 采用一种非线性高斯函数计算，将所有交并比的值规范在一个较小的范围内，即

$$O'^{(t)}_{(E_i,E_j)}=\exp[-(1-O^{(t)}_{(E_i,E_j)})^2], \tag{2}$$

式中：$O'^{(t)}_{(E_i,E_j)}$ 为对 $O^{(t)}_{(E_i,E_j)}$ 进行规范化后的值。第 i 个融合特征器的平均一致性计算公式为

$$M^{(t)}_{E_i}=\frac{1}{K}\sum_{j=1}^{K}O'^{(t)}_{(E_i,E_j)}, \tag{3}$$

式中：$M^{(t)}_{E_i}$ 为在第 t 帧跟踪时第 i 个融合特征器与其他融合特征器之间的一致性评分的平均值；K 为融合特征器的数量。在跟踪中，通常来说融合特征器之间的一致性评分在短时间内应具有时间稳定性。因此在一定的短周期 Δt 内融合特征器一致性计算的值的波动程度也反映了 E_i 与其他融合特征器之间的稳定性，可通过计算其标准差 $V^{(t)}_{E_i}$ 进行表征，即

$$V^{(t)}_{E_i}=\sqrt{\frac{1}{K}\sum_{j=1}^{K}[O'^{(t)}_{(E_i,E_j)}-\overline{O}'^{(t-\Delta t+1:t)\,2}_{(E_i,E_j)}]}, \tag{4}$$

式中：$\overline{O}'^{(t-\Delta t+1:t)}_{(E_i,E_j)}=\frac{1}{\Delta t}\sum_{\tau}O'^{(\tau)}_{(E_i,E_j)}$；$\tau$ 为时间索引，且 $\tau\in[t-\Delta t+1,t]$。

为了避免融合特征器性能波动，算法进一步考虑时间稳定性并在计算中引入一个递增的权重序列 $W=\{\rho_0,\rho_1,\cdots,\rho_{\Delta t-1}\}$，对于时序上越近的一致性评分给予更大的权重。此时，平均的加权均值 $M'^{(t)}_{E_i}$ 和标准差 $V'^{(t)}_{E_i}$ 计算公式为

$$M'^{(t)}_{E_i}=\frac{1}{N}\sum_{\tau}W_\tau M^{(t)}_{E_i}, \tag{5}$$

$$V'^{(t)}_{E_i}=\frac{1}{N}\sum_{\tau}W_\tau V^{(\tau)}_{E_i}, \tag{6}$$

式中：W_τ 为权重序列 W 中的第 $\tau-t+\Delta t$ 个权值；N 为归一化因子，$N=\sum_{\tau}W_\tau$。融合特征器最终的一致性评分 $R^{(t)}_{\text{pair}}(E_i)$ 计算公式为

$$R^{(t)}_{\text{pair}}(E_i)=\frac{M'^{(t)}_{E_i}}{V'^{(t)}_{E_i}+\xi}, \tag{7}$$

式中：ξ 为防止分母为 0 的一个极小的约束因子。$R^{(t)}_{\text{pair}}(E_i)$ 越大表示当前融合特征器与其余融合特征器之间的一致性越高，即在跟踪时具有更高的稳定性。

此外，每个融合特征器在进行独立跟踪时，其轨迹具有连续性和平滑性，每个融合特征器的轨迹平滑度在一定程度上表明了其跟踪结果的可靠性。融合特征器的可靠性 $S^{(t)}_{E_i}$ 可以表示为

$$S^{(t)}_{E_i}=\exp\left[-\frac{1}{2\sigma^{(t)}_{E_i}}(D^{(t)}_{E_i})^2\right], \tag{8}$$

式中：$D^{(t)}_{E_i}$ 为第 t 帧跟踪结果 $c(B^{(t)}_{E_i})$ 与第 t-1 帧跟踪结果 $c(B^{(t-1)}_{E_i})$ 之间的欧氏距离；$\sigma^{(t)}_{E_i}$ 为第 i 个融合特征器跟踪结果框的长度 $W(B^{(t)}_{E_i})$ 和宽度 $H(B^{(t)}_{E_i})$ 的平均值。$D^{(t)}_{E_i}$、$\sigma^{(t)}_{E_i}$ 的计算方法分别为

$$D^{(t)}_{E_i}=\|c(B^{(t-1)}_{E_i})-c(B^{(t)}_{E_i})\|, \tag{9}$$

$$\sigma_{E_i}=\frac{1}{2}[W(B^{(t)}_{E_i})+H(B^{(t)}_{E_i})]。 \tag{10}$$

最后采用同特征器间一致性计算的相同方式，并考虑时间稳定性，对跟踪器的最终可靠性进行相同处理，$R^{(t)}_{\text{self}}(E_i)$ 的计算方式为

$$R^{(t)}_{\text{self}}(E_i)=\frac{1}{N}\sum_{\tau}W_\tau S^{(\tau)}_{E_i}。 \tag{11}$$

对当前帧进行跟踪时最终选取的融合特征器的投票分数是将融合特征器间的一致性和融合特征器的可靠性进行综合考虑，通过线性加权的方式计算最终投票分数，计算公式为

$$R^{(t)}(E_i)=\mu R^{(t)}_{\text{pair}}(E_i)+(1-\mu)R^{(t)}_{\text{self}}(E_i), \tag{12}$$

式中：μ 为权重。在跟踪过程中，最终计算出的 $R^{(t)}(E_i)$ 值最大的融合特征器是当前帧中最稳健的融合特征器，用来对当前帧进行跟踪，将其跟踪结果作为最佳跟踪结果。

3.3　复杂情况下更新机制

在跟踪过程中，将当前帧的跟踪结果作为训练数据对目标特征模型进行更新，即以当前帧标注的目标物体作为样本在下一帧跟踪过程中对跟踪目标进行特征模型更新，以进行后续跟踪。在连续的跟踪过程中，每一帧跟踪产生的漂移误差在模型更新中不断累积，这会对后续帧的跟踪效果产生一定的影响。尤其是在目标出现遮挡、短暂消失等复杂情况时，需要选取合适的特征模型进行后续跟踪以保证跟踪效果。

在大多数 DCF 跟踪算法中，通常会采用峰值旁瓣比率(PSR)来衡量目标样本的可靠性[5]，PSR(P)的计算方法为

$$P=\frac{R_{\max}-m}{\sigma}, \tag{13}$$

式中：$R_{\max}$ 为融合特征器评分的最大值；m 和 σ 分别为响应值的均值和标准差。不同特征的平均 PSR ($P^{(t)}_{\text{mean}}$)计算式为

$$P^{(t)}_{\text{mean}}=\frac{1}{3}(P^{(t)}_{\text{H}}+P^{(t)}_{\text{M}}+P^{(t)}_{\text{L}}), \tag{14}$$

式中：$P^{(t)}_{\text{H}}$、$P^{(t)}_{\text{M}}$、$P^{(t)}_{\text{L}}$ 分别表示高级、中级和低级特征响应图的 PSR。PSR 的值能够反映当前样本的

可靠性，值越大说明样本越可靠。

在跟踪过程中出现遮挡或者剧烈形变时，融合特征器的投票评分 $R^{(t)}(E_i)$会出现明显的下降，所以采用 PSR 和 $R^{(t)}(E_i)$均值结合的方式来判定跟踪样本的稳定性以及当前跟踪效果的优劣，即

$$R_{\text{mean}}^{(t)}=\frac{1}{K}\sum_{i=1}^{K}R^{(t)}(E_i),\tag{15}$$

$$S_t=P_{\text{mean}}^{(t)}\cdot R_{\text{mean}}^{(t)},\tag{16}$$

式中：S_t 表示当前帧跟踪结果评分。在跟踪过程中，若当前帧 S_t 急剧减小，可以判定当前帧的跟踪质量不佳，可能是出现了遮挡或者目标形变等问题。针对这种情况，构建了一个引入时间稳定性的目标外观模型更新模块。

考虑到跟踪目标在连续时间序列上会出现一系列的外观变化以及受环境因素影响产生的变化，目标的改变是一个渐变的过程，在时序上存在连续性。而通常跟踪的做法是将当前帧的跟踪结果作为后续跟踪的训练样本进行更新，在出现遮挡、形变、背景复杂等跟踪质量不佳的情况时，误差较大的当前帧跟踪结果作为后续跟踪过程样本进行模型更新会对后续的跟踪产生影响，持续的低质量跟踪过程中不断累积的误差最终可能导致目标跟踪失败。为此，提出考虑时间稳定性的短时记忆特征更新策略，构建一个稳健短时记忆时序特征提取模块，在跟踪过程中，将最终评分较高的帧的特征作为保留时序特征，在跟踪质量不佳时，选取 Δt 时序内的最佳特征作为最终时序特征。提出一个跟踪质量判别机制，当跟踪质量不佳时，对目标表征进行自适应更新：1）利用最邻近稳健短时记忆时序特征以及当前帧的最佳融合特征进行特征提取更新，对于目标形变、光照变化、消失后重现等复杂情况，只有依靠稳健时序特征以及当前帧提取特征信息才能有效地寻找到目标；2）考虑到目标变化的不确定性，融入不包含全连接层的 VGG19 网络 conv3-4 层的包含较高分辨率和较丰富语义信息的特征模型，以提高目标跟踪匹配成功率，在背景复杂、目标旋转等复杂情况时，融合更多的深层高分辨率语义信息能够提升目标跟踪效果。自适应更新计算公式表示为

$$R_{\text{update}}=W_1R_{\text{robust}}+W_2R_{\text{time}}^{(\Delta t)}+W_3F_{\text{conv3-4}},\tag{17}$$

式中：R_{robust} 为当前跟踪器提取的稳健融合特征；$R_{\text{time}}^{(\Delta t)}$ 为时序最邻近稳健特征；$F_{\text{conv3-4}}$ 为包含全连接层的 VGG-Net-19 网络 conv3-4 层深度特征；W_1、W_2、W_3 为权重参数。模型更新机制如图 4 所示，图中 R 表示跟踪质量评分阈值。

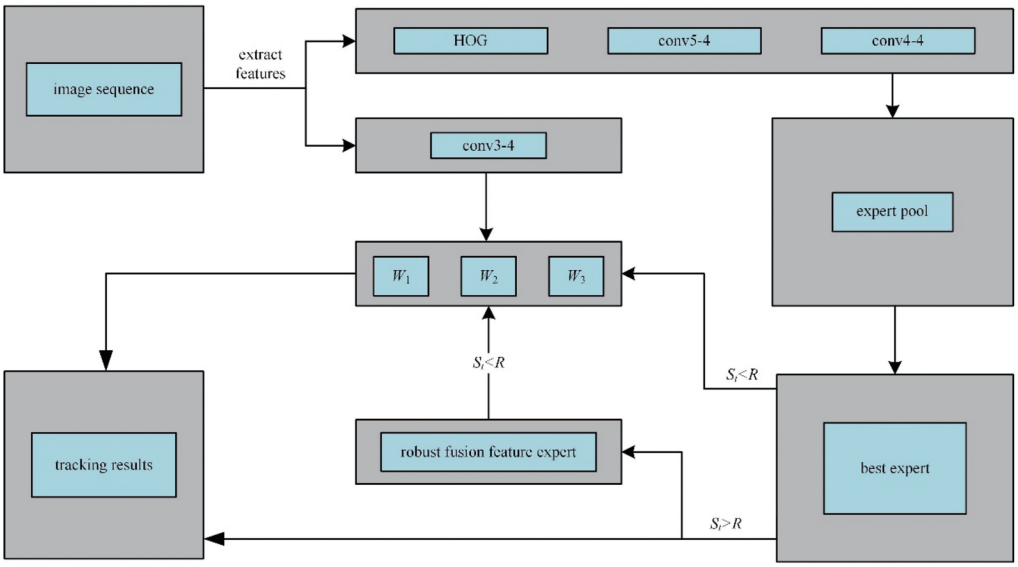

图 4　时序模型更新机制

Fig. 4　Update mechanism of temporal model

3.4　算法总体流程

结合以上对本文算法关键点的描述，算法总体流程如图 5 所示。图中，P_t 表示第 t 帧时目标的位置。x_t，y_t，w_t，h_t 分别代表第 t 帧时，目标位置的 x 坐标、y 坐标、目标框的宽度、目标框的高度。

4　实验与分析

4.1　实验环境

实验平台采用 MATLAB2016a 和 C＋＋在 Matconvnet 上的深度学习库混合编程，硬件环境

Algorithm1：Proposed Tracking Algorithm

Input：Initial target position $P_{t-1}(\mathrm{x}_{t-1},\mathrm{y}_{t-1},\mathrm{w}_{t-1},\mathrm{h}_{t-1})$,the correlation filter.

Output：Estimation object position $P_t(x_t,y_t,w_t,h_t)$

1 Repeat

2 Crop out the searching window in frame t centered at (x_{t-1},y_{t-1}) and extract HOG features and convolutional features;

3 For each sample do computing filter and correlation response map;

4 Estimate the new position (x_t,y_t) on response map;

5 Estimate the new scale of sample and extract feature;

6 Compute confidence score S_t of new patch using (16);

7 **IF** $S_t \le R$

8 Update model of new target using (17);

9 Estimate new position (x_t,y_t) and scale of target;

10 **END IF**

11 Until End of video sequence

图 5　算法流程

Fig. 5　Flow chart of algorithm

为：CPU：Intel(R) Core(TM) i7-8700，3.20 GHz；16 GB 内存；GPU：NVIDIA GeForce 1080Ti。

4.2　评价指标

在 OTB-2013[17]和 OTB-2015[18]数据库上进行大量测试，利用一次性通过评价(OPE)标准来分析算法性能，测试数据库涵盖 11 个属性：背景杂波、快速运动、平面内旋转、平面外旋转、运动模糊、尺度变化、光照变化、低分辨率、遮挡、超出视线范围、形变。每段序列可能有多个属性，背景属性相对复杂，以准确率、成功率作为评价算法性能的指标。为了更好地评价算法性能，将本文算法(ours)同时与近年来比较流行的 MCCT[10]、ECO[19]、CF2[20]、SRDCF[21]、Staple[22]、ADNet[23]、KCF[24]、STRCF[25]、LCT[26]算法进行对比。

4.3　定性分析

图 6 是 10 种跟踪算法在几个视频序列上的部分跟踪结果，视频从上至下、从左至右分别是 Soccer(第 1，132，277，380 帧)、CarScale(第 1，211，214，240 帧)、Bolt2(第 1，49，75，99 帧)、MotorRolling(第 1，67，91，126 帧)、Skiing(第 1，30，49，72 帧)、Ironman(第 1，26，35，108 帧)，通过对 10 个算法在不同视频序列中的跟踪结果进行对比分析，可以发现本文算法的跟踪质量和稳健性比较理想。

4.3.1　遮挡、背景复杂

以“Soccer”视频序列为例，目标在运动过程中多次受到奖杯和周围人群的遮挡，并且目标与背景环境极其相似，多数算法都出现了漂移现象甚至跟踪失败，这是由跟踪器在出现遮挡、背景复杂时学习到背景信息导致的。而本文算法在复杂情况时，判别目标不全依赖当前学习到的特征，而是融入时序稳健特征，可以更好地对目标进行表征。如图 6 第 1 行所示，只有本文算法和 MCCT 算法能够较好地跟踪到目标。

4.3.2　尺度变化

以“CarScale”视频序列为例，目标车辆离镜头由远及近，使得目标在视频序列中尺度发生较大变化，如图 6 第 2 行所示，虽然大多数算法都能跟上目标，但只有本文算法能够更好地进行目标的定位以及尺度估计。

4.3.3　快速运动、形变

以“Bolt2”视频序列为例，视频是短跑比赛场景，目标在视频序列内快速运动，自身也在持续发生形变，如图 6 第 3 行所示。其他算法在跟踪过程中多次出现跟踪漂移和跟踪失败现象，本文算法充分考虑时序信息，对目标的变化实时更新，只有本文算法能够稳定地对目标进行跟踪。

图 6　10 种跟踪算法在部分视频序列上的定性结果显示

Fig. 6　Qualitative results of 10 tracking algorithms for some video sequences

4.3.4　目标平面内/外旋转

以“MotorRolling”视频序列为例，目标摩托车在运动过程中多次出现各种旋转，本文算法采用多特征融合方式，对目标具有更强的表征能力。如图 6 第 4 行所示，除本文算法和 ADNet 算法能够对目标进行稳健跟踪外，其他算法都出现了跟踪失败现象，但本文算法在目标尺度估计上更加精确。

4.3.5　低分辨率

以“Skiing”视频序列为例，该视频序列分辨率较低，本文算法充分考虑特征语义信息，增强了目标的表征能力。如图 6 第 5 行所示，在视频第 21 帧时，除本文算法和 CF2 算法外，其余算法都出现了跟踪失败现象。

4.3.6　光照变化

以“Ironman”视频序列为例，在跟踪过程中，视频序列多次出现强烈的光照变化，本文算法加入实时更新机制，在目标出现较大变化时能够及时对其进行特征更新。如图 6 第 6 行所示，除本文算法和 CF2、MCCT 算法外，其余算法都不能成功跟踪到目标。

为验证本文融合特征器的有效性，以“Soccer”视频序列为例，对每一帧中融合特征器的选取情况进行统计分析，统计结果如表 1 所示。

表 1　特征器频次统计

Table 1　Frequency statistics of feature experts

Expert	1	2	3	4	5	6	7	8
Frequency	18	54	18	80	77	85	25	34

表中 Expert 1～7 对应表示 7 个不同融合特征器，Expert 8 表示自适应更新特征，Frequency 表示在整个跟踪过程中，选取当前融合特征器结果作为最终跟踪结果的视频帧数量。在同一帧跟踪中，不同融合特征器的跟踪结果可视化如图 7 所示。图中由左至右、由上至下分别为“Soccer”视频序列中第 48 帧、第 60 帧、第 67 帧和第 191 帧，不同颜色的虚线矩形框表示不同的特征器结果，最终采用的跟踪结果由对应特征器颜色的实线框表示。由图7可以看

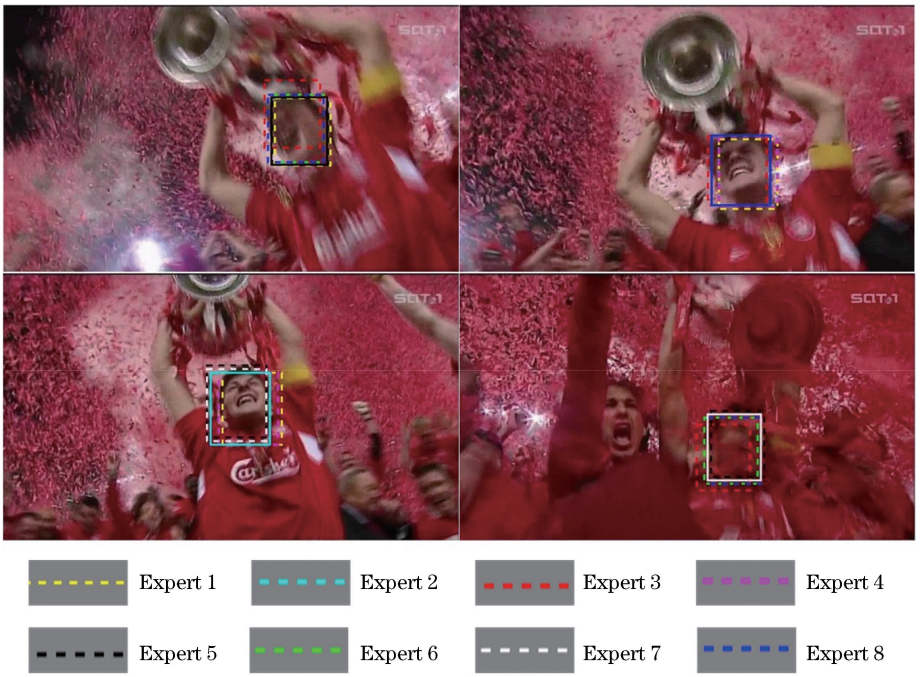

图 7　不同特征器跟踪结果

Fig. 7　Tracking results of different feature experts

出在第 48 帧中，选取了 Expert 5 的跟踪结果作为最终结果；第 60 帧中选取了自适应更新特征的跟踪结果作为最终结果；第 67 帧中选取了 Expert 2 的跟踪结果作为最终结果；第 191 帧中选取了 Expert 7 的跟踪结果作为最终结果。

4.4　定量分析

为进一步全面地评价本文算法，采用跟踪精度和跟踪成功率对算法进行定量分析，图8是本文算

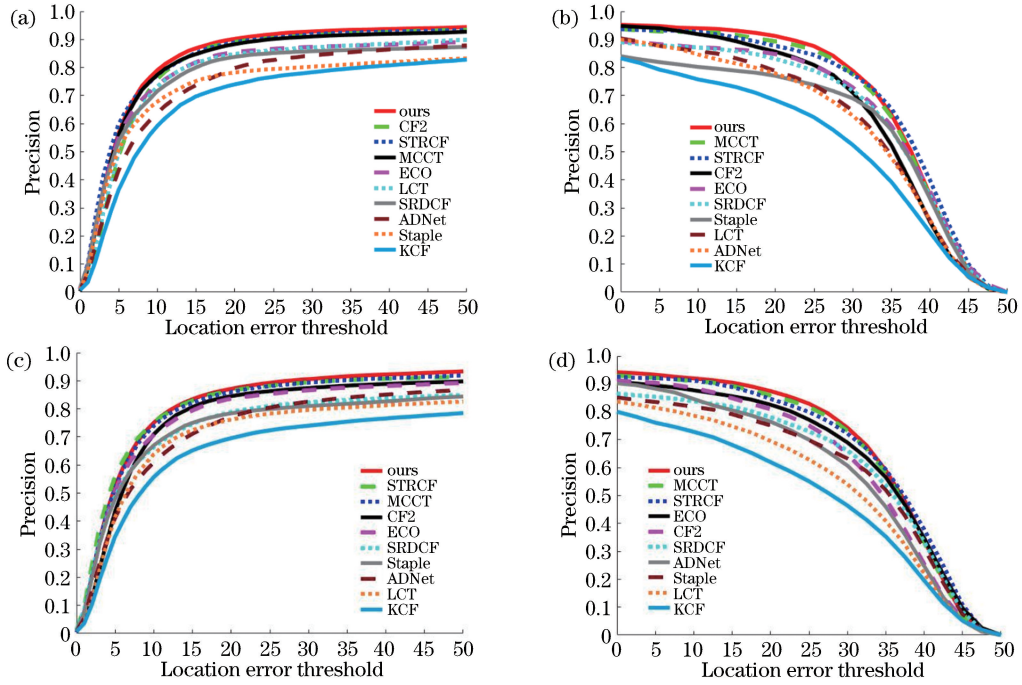

图 8　在 OTB-2013 和 OTB-2015 数据库下，算法跟踪精度和跟踪成功率。(a) OTB-2013 跟踪精度；(b) OTB-2013 跟踪成功率；(c) OTB-2015 跟踪精度；(d) OTB-2015 跟踪成功率

Fig. 8　Tracking accuracy and success rate of algorithm on OTB-2013 and OTB-2015 databases. (a) Tracking accuracy on OTB-2013 database; (b) tracking success rate on OTB-2013 database; (c) tracking accuracy on OTB-2015 database; (d) tracking success rate on OTB-2015 database

法在 OTB-2013 和 OTB-2015 测试数据库上的精度曲线和成功率曲线。

图 8(a)和(b)分别为 8 种跟踪算法在 OTB-2013 数据库上的跟踪精度曲线图和跟踪成功率曲线图，可以看出本文算法不管是跟踪精度还是跟踪成功率都取得了最优效果；图 8(c)和图 8(d)分别为 8 种跟踪算法在 OTB-2015 数据库上的跟踪精度曲线图和跟踪成功率曲线图，可以看出本文算法的跟踪精度和跟踪成功率仍然取得了最优效果。

为更直观地评估本文算法的效果，对算法在 OTB-2013 及 OTB-2015 测试数据库上的跟踪精度和跟踪成功率以及跟踪速率的数据进行分析，如表 2 所示。由数据分析可知，在 OTB-2013 测试数据库下，本文算法的跟踪精度最高，达到了 90.2%，相比排名第二的 CF2 算法精度提高了 1.1%，相比于 STRCF 算法精度提高了 1.3%；在成功率方面，本文算法的跟踪成功率也最高，达到了 87.6%，相比于排名第二的 MCCT 算法成功率提升了 1.3%，相比于 CF2 算法成功率提升了 6.7%。在 OTB-2015 测试数据库下，本文算法的跟踪精度最高，达到了 87.1%，相比于排名第二的 STRDCF 算法精度提高了 0.7%，相比于 MCCT 算法精度提高了 1.1%，相比于 CF2 算法精度提高了 2.6%；在成功率方面，本文算法的跟踪成功率也最高，达到了 82.9%，相比于 STRDCF 成功率提高了 2.9%，相比于 MCCT 算法成功率提升了 1.1%，相比于 CF2 算法成功率提升了 7.8%。在跟踪速率方面，更高的跟踪精度与跟踪成功率是以计算量的增加为代价的，给跟踪速率带来一定的影响。本文算法跟踪速率为 3.9 frame/s，高于 CF2 算法的 1.5 frame/s，仍然满足实时性要求。

表 2　算法在 OTB-2013 和 OTB2015 上精度、成功率值和速率

Table 2　Tracking accuracy, success rate, and speed of algorithm on OTB-2013 and OTB2015 databases

Database	Parameter	Ours	STRCF	MCCT	CF2	ECO	SRDCF	ADNet	Staple	KCF	LCT
OTB-2013	Precision	0.902	0.889	0.883	0.891	0.855	0.838	0.798	0.782	0.740	0.848
	Success rate	0.876	0.845	0.863	0.809	0.806	0.789	0.721	0.738	0.623	0.738
OTB-2015	Precision	0.871	0.864	0.860	0.845	0.836	0.788	0.772	0.784	0.696	0.762
	Success rate	0.829	0.800	0.818	0.751	0.772	0.730	0.700	0.699	0.526	0.629
Average FPS		3.9	28.0	4.2	1.5	54.8	8.0	12.3	97.6	349.3	29.4

为了更进一步地分析本文算法在不同跟踪条件下的跟踪性能，图 9 和图 10 分别给出了算法在 OTB-2013 测试数据库下 11 种不同属性条件下的跟踪精度和成功率。

在 OTB-2013 测试数据库下，测试 11 种不同属性视频序列的结果显示，本文算法的跟踪精度始终处于最优或次优水平，跟踪精度除了在 low resolution、background clutter、deformation 中处于次优，在其他 8 个属性中均得到最优成绩。本文算法的跟踪成功率在 11 种不同属性视频序列上的测试结果始终保持最优。

图 11、图 12 分别给出了算法在 OTB-2015 测试数据库下 11 种不同属性条件下的跟踪精度和成功率。在 OTB-2015 测试数据库下，测试 11 种不同属性视频序列的结果显示，本文算法的跟踪精度始终处于最优或次优水平，跟踪精度除了在 low resolution、deformation、scale variation 中处于次优，在其他 8 个属性中均得到最优成绩。本文算法的跟踪成功率在 11 种不同属性视频序列上的测试结果始终保持最优。

对于长时跟踪情况，本文算法不管是跟踪成功率还是跟踪精度在 8 种对比算法中仍然处于最优。在加入自适应特征更新模块后，复杂情况下的低质量跟踪效果得到提升，减小了当前帧的跟踪误差。由于跟踪时下一帧的跟踪是以当前帧跟踪结果作为初始跟踪状态，所以本文算法在减少当前帧跟踪误差的同时，减小了模型的累积误差，能够提高后续帧跟踪的精度和成功率。本文算法的跟踪精度达到了 0.901，跟踪成功率达到了 0.866。图 13 为本文算法与加入长时跟踪效果较好的算法 LCT[26] 作为对比的算法对 OTB 数据库中较长视频序列中的跟踪精度及跟踪成功率。

OTB-2013、OTB-2015 测试数据库上的跟踪结果显示，本文算法在 11 种不同属性视频序列上的成功率和精度均有较好的提升，平均跟踪速率为 3.9 frame/s，与近年来比较流行的深度学习跟踪算法 MCCT、CF2 等相当。同时，本文算法对于长时跟踪情况同样有较好的提升，能够实现复杂情况下的稳健的跟踪。

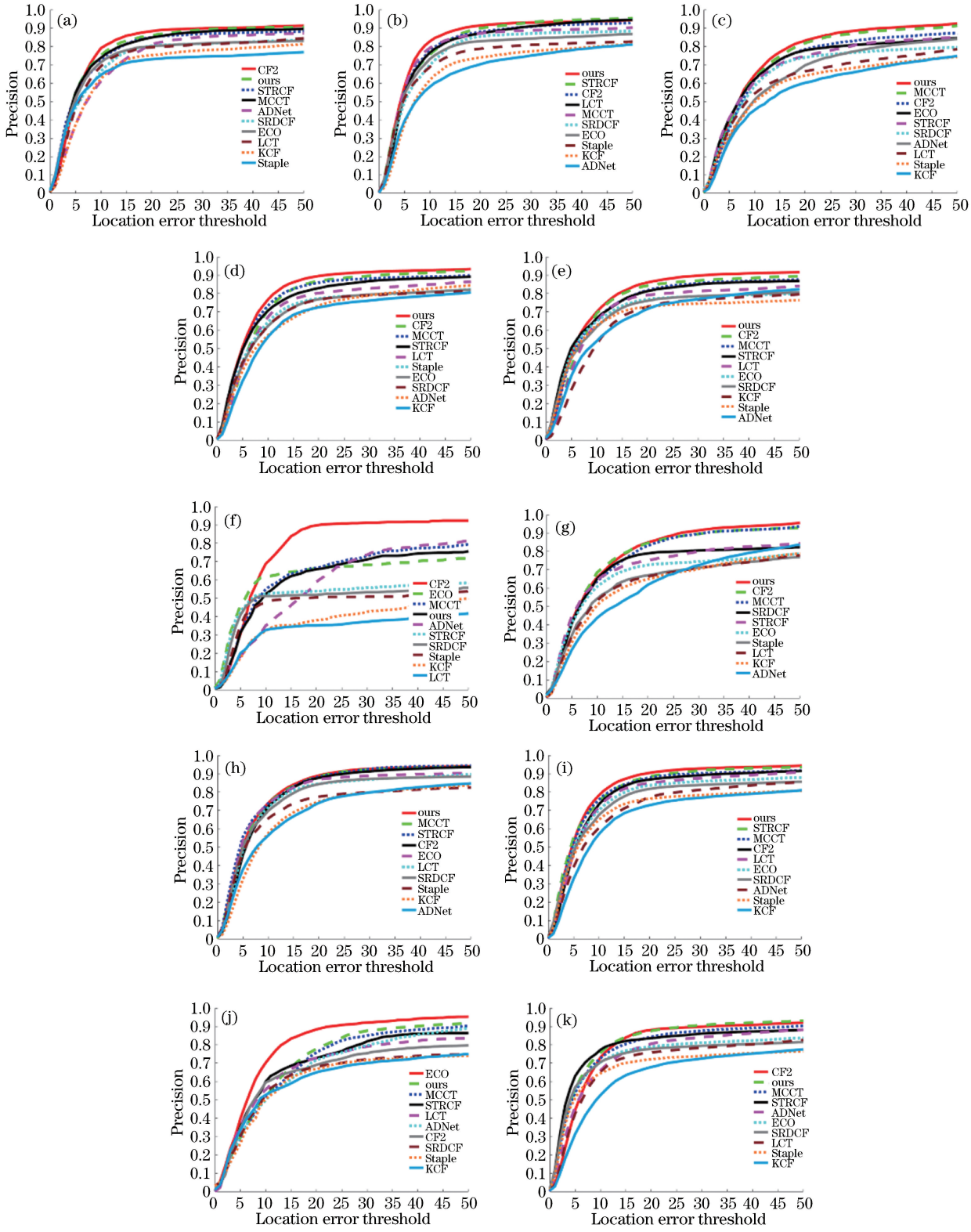

图 9　在 OTB-2013 下 11 种不同属性视频序列跟踪精度。(a)背景杂波;(b)形变;(c)快速运动;(d)平面内旋转;(e)光照变换;(f)低分辨率;(g)运动模糊;(h)遮挡;(i)平面外旋转;(j)超出视线范围;(k)尺度变化

Fig. 9　Tracking precision of 11 different attribute video sequences on OTB-2013 database. (a) Background clutter; (b) deformation; (c) fast motion; (d) in-plane rotation; (e) illumination variation; (f) low resolution; (g) motion blur; (h) occlusion; (i) out-of-plane rotation; (j) out of view; (k) scale variation

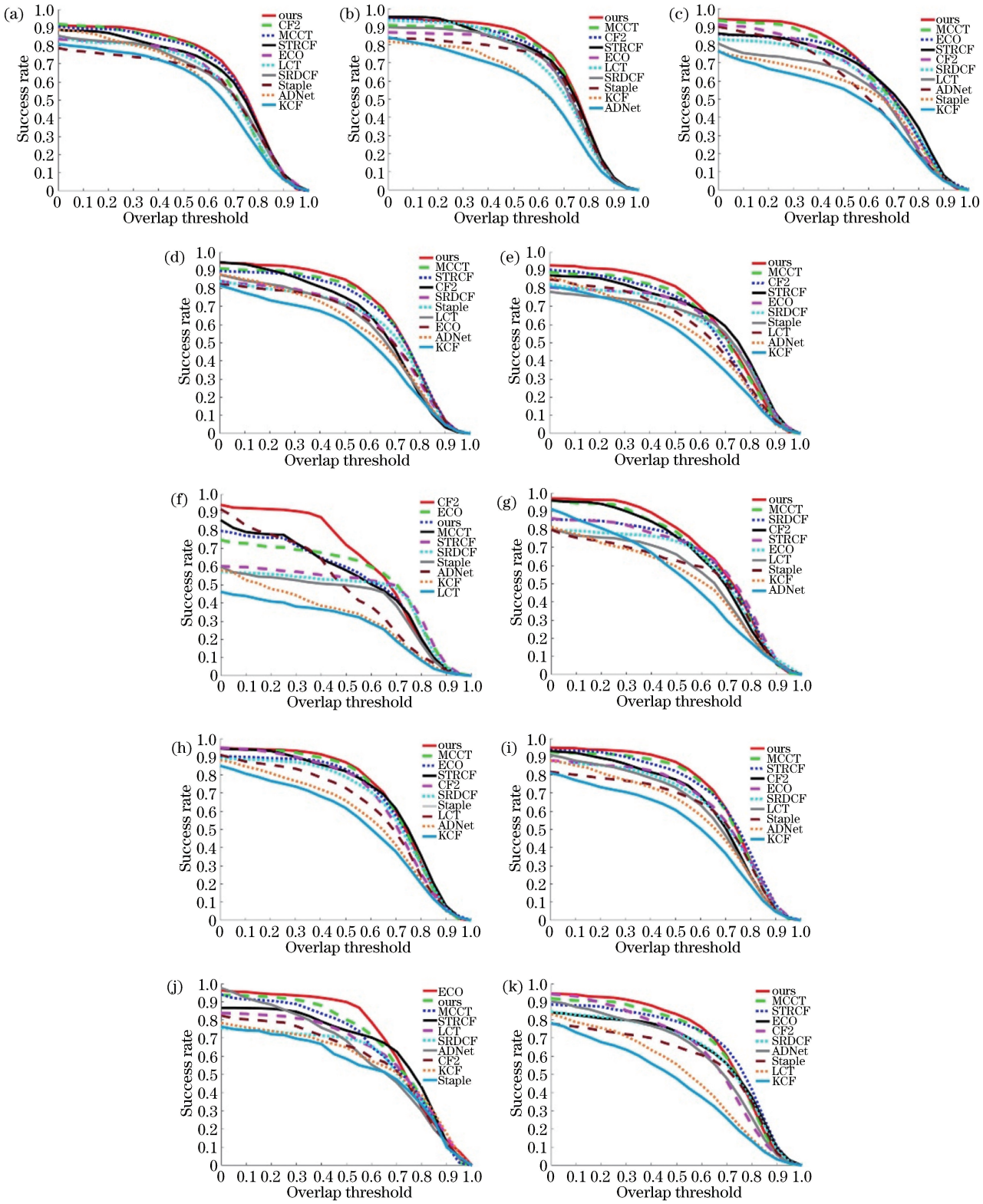

图 10　在 OTB-2013 下 11 种不同属性视频序列跟踪成功率。(a)背景杂波;(b)形变;(c)快速运动;(d)平面内旋转;(e)光照变换;(f)低分辨率;(g)运动模糊;(h)遮挡;(i)平面外旋转;(j)超出视线范围;(k)尺度变化

Fig. 10　Tracking success rates of 11 different attribute video sequences on OTB-2013 database. (a) Background clutter; (b) deformation; (c) fast motion; (d) in-plane rotation; (e) illumination variation; (f) low resolution; (g) motion blur; (h) occlusion; (i) out-of-plane rotation; (j) out of view; (k) scale variation

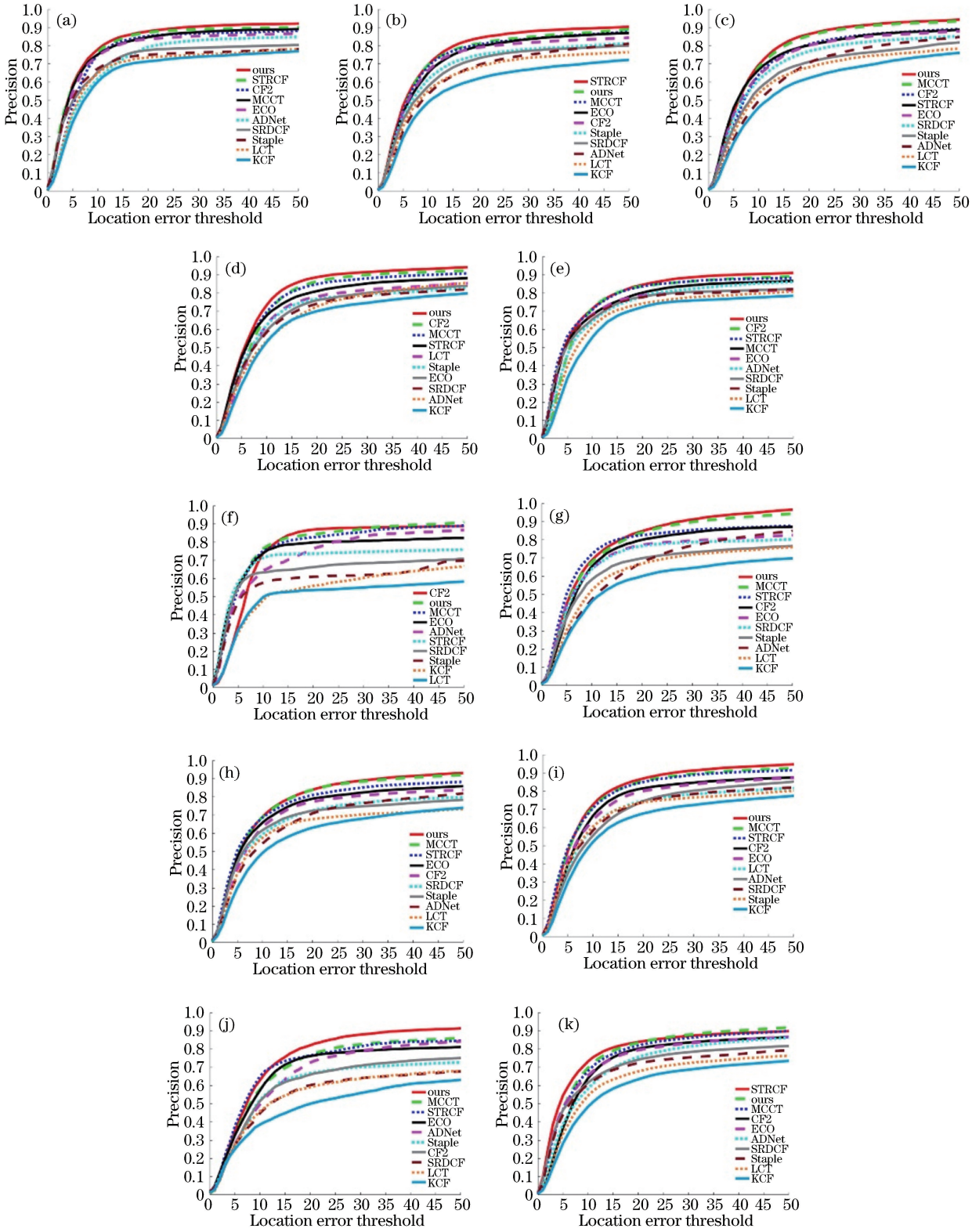

图 11　TB-2015 下 11 种不同属性视频序列跟踪精度。(a)背景杂波;(b)形变;(c)快速运动;(d)平面内旋转;(e)光照变换;(f)低分辨率;(g)运动模糊;(h)遮挡;(i)平面外旋转;(j)超出视线范围;(k)尺度变化

Fig. 11　Tracking precision of 11 different attribute video sequences on OTB-2015 database. (a) Background clutter; (b) deformation; (c) fast motion; (d) in-plane rotation; (e) illumination variation; (f) low resolution; (g) motion blur; (h) occlusion; (i) out-of-plane rotation; (j) out of view; (k) scale variation

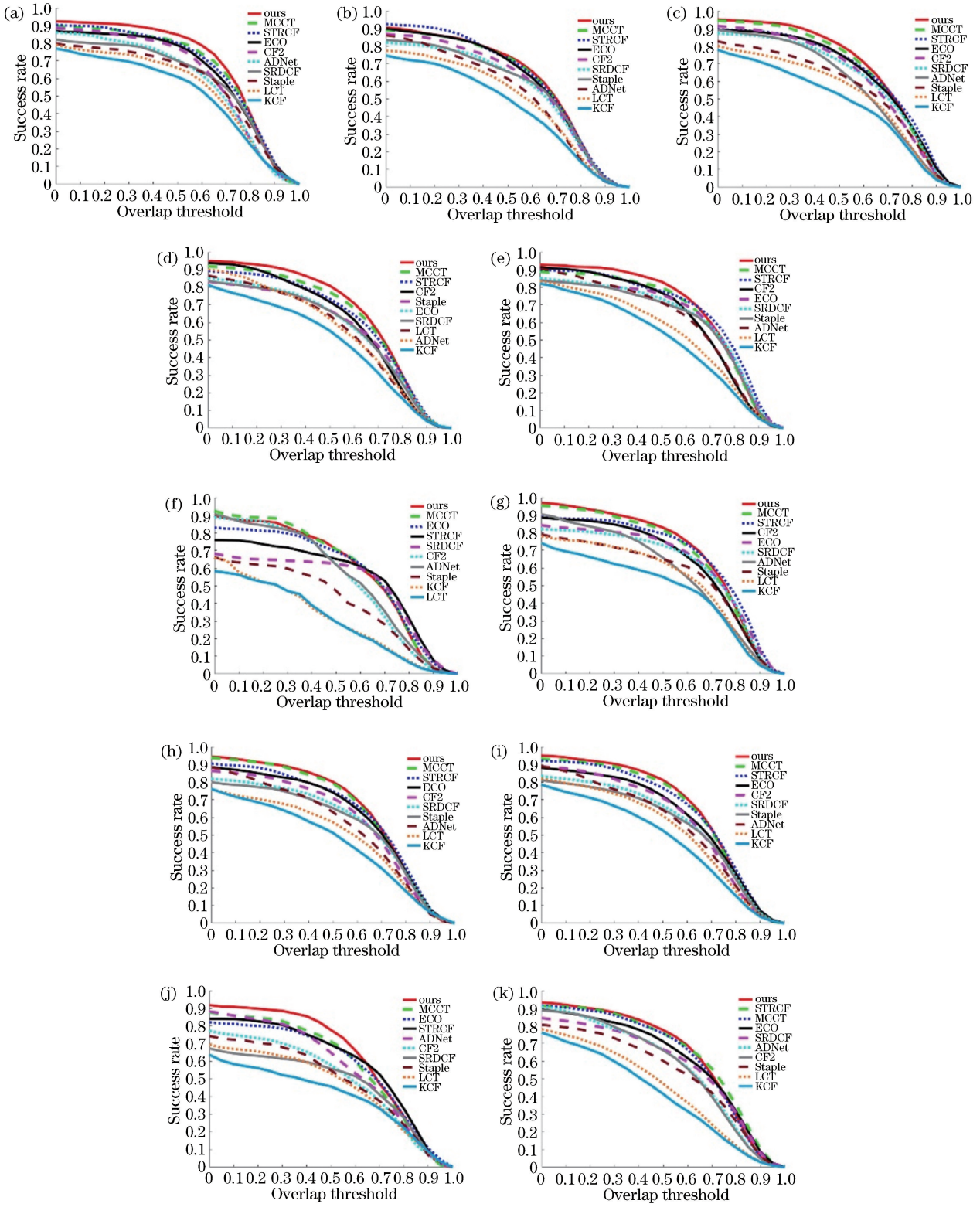

图 12　OTB-2015 下 11 种不同属性视频序列跟踪成功率。(a)背景杂波;(b)形变;(c)快速运动;(d)平面内旋转;(e)光照变换;(f)低分辨率;(g)运动模糊;(h)遮挡;(i)平面外旋转;(j)超出视线范围;(k)尺度变化

Fig. 12　Tracking success rates of 11 different attribute video sequences on OTB-2015 database. (a) Background clutter; (b) deformation; (c) fast motion; (d) in-plane rotation; (e) illumination variation; (f) low resolution; (g) motion blur; (h) occlusion; (i) out-of-plane rotation; (j) out of view; (k) scale variation

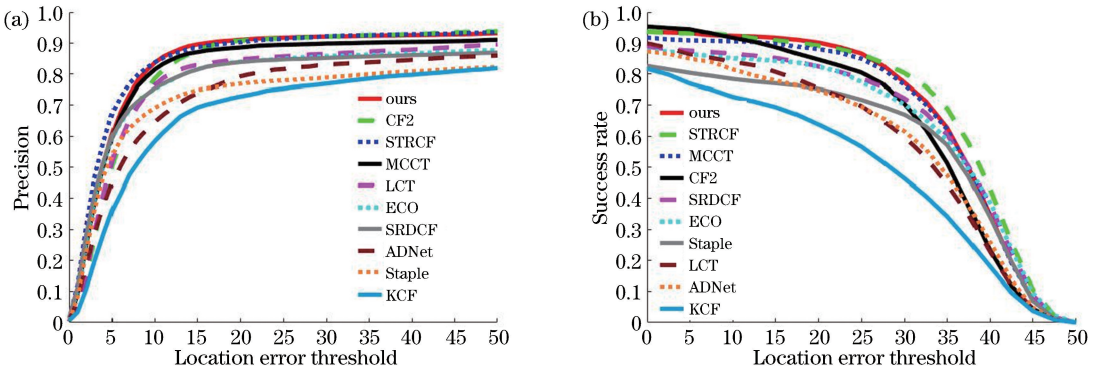

图 13　长时跟踪情况下算法跟踪精度和成功率。(a)跟踪精度;(b)跟踪成功率

Fig. 13　Tracking precision and success rate of algorithm under long-term tracking. (a) Tracking precision; (b) tracking success rate

5　结　　论

提出一种复杂情况下自适应特征更新目标跟踪算法,其主要思想是充分利用不同层级深度特征的丰富语义信息以及手工设计特征定位精度较高的优势,采取线性组合的多特征融合手段得到多个融合特征器,通过置信度评估选择最优融合特征;在目标跟踪阶段,通过可靠性投票计算判别当前跟踪效果,在跟踪效果不佳时采取自适应特征更新方法,从而获得稳健的跟踪效果。上述实验表明,与近年来主流算法相比,本文算法在各种复杂的情况下,有效地提高了目标跟踪的精确性和稳健性,能够较好地适应不同的跟踪环境,获得较高质量的跟踪效果。

参　考　文　献

[1] Wang Q, Zhang L, Bertinetto L, *et al*. Fast online object tracking and segmentation: a unifying approach [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 16-20, 2019, Long Beach, CA, USA. New York: IEEE, 2019: 1328-1338.

[2] Lu H C, Li P X, Wang D. Visual object tracking: a survey [J]. Pattern Recognition and Artificial Intelligence, 2018, 31(1): 61-76.
卢湖川, 李佩霞, 王栋. 目标跟踪算法综述[J]. 模式识别与人工智能, 2018, 31(1): 61-76.

[3] Li J L, Yin K, Chu C X, *et al*. Review of video target tracking technology [J]. Journal of Yanshan University, 2019, 43(3): 251-262.
李均利, 尹宽, 储诚曦, 等. 视频目标跟踪技术综述[J]. 燕山大学学报, 2019, 43(3): 251-262.

[4] 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, 2010: 2544-2550.

[5] 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. Berlin, Heidelberg: Springer, 2012, 7575: 702-715.

[6] Zhang K H, Zhang L, Liu Q S, *et al*. Fast visual tracking via dense spatio-temporal context learning [M] // Fleet D, Pajdla T, Schiele B, *et al*. Computer vision-ECCV 2014. Lecture notes in computer science. Cham: Springer, 2014, 8693: 127-141.

[7] 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 1-5, 2014, University of Nottingham, UK. UK: BMVA Press, 2014.

[8] Wang N Y, Yeung D Y. Learning a deep compact image representation for visual tracking [C] // NIPS'13 Proceedings of the 26th International Conference on Neural Information Processing Systems, December 5-10, 2013, Lake Tahoe, Nevada. New York: ACM, 2013, 1: 809-817.

[9] Danelljan M, Robinson A, Khan F S, *et al*. Beyond correlation filters: learning continuous convolution operators for visual tracking[M] // Leibe B, Matas J, Sebe N, *et al*. Computer vision-ECCV 2016. Lecture notes in computer science. Cham: Springer, 2016, 9909: 472-488.

[10] Wang N, Zhou W G, Tian Q, *et al*. Multi-cue correlation filters for robust visual tracking[C] // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE, 2018: 4844-

4853.

[11] Shen Q, Yan X L, Liu L F, *et al*. Multi-scale correlation filtering tracker based on adaptive feature selection[J]. Acta Optica Sinica, 2017, 37(5): 0515001.
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[J]. 光学学报, 2017, 37(5): 0515001.

[12] 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.

[13] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR′05), June 20-25, 2005, San Diego, CA, USA. New York: IEEE, 2005: 8588935.

[14] Danelljan M, Khan F S, Felsberg M, *et al*. Adaptive color attributes for real-time visual tracking[C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2014, Columbus, OH, USA. New York: IEEE, 2014: 1090-1097.

[15] Simonyan K, Zissermanl A. Very deep convolutional networks for large-scale image recognition[J/OL]. (2015-04-10)[2019-05-30]. https://arxiv.org/abs/1409.1556.

[16] Bhat G, Johnander J, Danelljan M, *et al*. Unveiling the power of deep tracking[M]//Ferrari V, Hebert M, Sminchisescu C, *et al*. Computer vision-ECCV 2018. Lecture notes in computer science. Cham: Springer, 2018, 11206: 493-509.

[17] 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, 2013: 2411-2418.

[18] Wu Y, Lim J, Yang M H. Object tracking benchmark[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1834-1848.

[19] Danelljan M, Bhat G, Khan F S, *et al*. ECO: efficient convolution operators for tracking[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 6931-6939.

[20] Ma C, Huang J B, Yang X K, *et al*. Hierarchical convolutional features for visual tracking[C]//2015 IEEE International Conference on Computer Vision (ICCV), December 7-13, 2015, Santiago, Chile. New York: IEEE, 2015: 3074-3082.

[21] Danelljan M, Hager 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, 2015: 4310-4318.

[22] 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, 2016: 1401-1409.

[23] Yun S, Choi J, Yoo Y, *et al*. Action-decision networks for visual tracking with deep reinforcement learning[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 1349-1358.

[24] 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.

[25] Li F, Tian C, Zuo W M, *et al*. Learning spatial-temporal regularized correlation filters for visual tracking[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE, 2018: 4904-4913.

[26] 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, 2015: 5388-5396.