

基于自适应多层卷积特征决策融合的目标跟踪

陈法领^{1,2,3,4,5}, 丁庆海^{1,6}, 罗海波^{1,2,4,5*}, 惠斌^{1,2,4,5}, 常铮^{1,2,4,5}, 刘云鹏^{1,2,4,5}

¹中国科学院沈阳自动化研究所, 辽宁 沈阳 110016;

²中国科学院机器人与智能制造创新研究院, 辽宁 沈阳 110169;

³中国科学院大学, 北京 100049;

⁴中国科学院光电信息处理重点实验室, 辽宁 沈阳 110016;

⁵辽宁省图像处理与视觉计算重点实验室, 辽宁 沈阳 110016;

⁶航天恒星科技有限公司, 北京 100086

摘要 针对在复杂环境中目标尺度变化、形状变化以及场景光照变化、背景干扰等因素导致的目标跟踪稳定性下降问题, 提出一种基于自适应多层卷积特征决策融合的目标跟踪算法。首先, 通过卷积神经网络 VGG-Net-19 提取目标候选区域的多层卷积特征; 其次, 在相关滤波模型框架下, 利用这些卷积特征构建多个弱跟踪器; 接着, 根据每个弱跟踪器的决策损失变化自适应地调节它们的决策权重, 完成基于多层卷积特征的目标位置估计; 然后, 根据尺度相关滤波模型在目标中心区域进行多尺度采样, 并利用相邻帧的尺度变化先验分布完成对目标尺度的预测。选取 51 组具有多种挑战因素的视频序列对所提算法的跟踪性能进行测试。实验结果表明, 与当前主流的目标跟踪算法相比, 所提算法取得了更高的跟踪精度和成功率, 同时可以较好地适应目标的尺度变化, 并且在目标发生形变、场景出现光照变化及背景干扰等复杂条件下仍具有较好的跟踪鲁棒性。

关键词 机器视觉; 目标跟踪; 决策融合; 卷积神经网络; 卷积特征; 尺度自适应

中图分类号 TP391.4

文献标志码 A

doi: 10.3788/AOS202040.2315002

Target Tracking Based on Adaptive Multilayer Convolutional Feature Decision Fusion

Chen Faling^{1,2,3,4,5}, Ding Qinghai^{1,6}, Luo Haibo^{1,2,4,5*}, Hui Bin^{1,2,4,5},

Chang Zheng^{1,2,4,5}, Liu Yunpeng^{1,2,4,5}

¹Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, Liaoning 110016, China;

²Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang, Liaoning 110169, China;

³University of Chinese Academy of Sciences, Beijing 100049, China;

⁴Key Laboratory of Opto-Electronic Information Processing, Chinese Academy of Sciences, Shenyang, Liaoning 110016, China;

⁵Liaoning Key Laboratory of Image Understanding and Computer Vision, Shenyang, Liaoning 110016, China;

⁶Space Star Technology Co., Ltd., Beijing 100086, China

Abstract To address the tracking stability degradation caused by target scale variation, deformation, illumination variation, and background clutter in complex scenes, a target tracking algorithm based on adaptive multilayer convolutional feature decision fusion is proposed. Initially, multilayer convolutional features are extracted from a target candidate region using the VGG-Net-19 convolutional neural network. Then, under a correlation filter model framework, the extracted convolutional features are employed to construct several weak trackers. Decision weights are adjusted adaptively based on the fluctuation of the decision losses of these weak trackers, and the target position is estimated based on the multilayer convolutional features. Next, according to a scale correlation filter model, multiple scale image patches are sampled at the target center position. Taking advantage of the prior distribution of

收稿日期: 2020-08-07; 修回日期: 2020-08-23; 录用日期: 2020-08-31

基金项目: 国家重点研发计划(2018YFC080192)、中国科学院国防科技创新重点基金(Y8K4160401)

* E-mail: luohb@sia.cn

scale variation between adjacent frames, its scale is predicted. Fifty-one video sequences with multiple challenging attributes are selected to evaluate the tracking performance of the proposed algorithm. The experimental results demonstrate that the proposed algorithm has higher tracking accuracy and success rate compared with state-of-the-art target tracking algorithms. The proposed algorithm adapts well to target scale variation. In addition, it improves the target tracking robustness under target deformation, illumination variation, and background clutter conditions.

Key words machine vision; target tracking; decision fusion; convolutional neural network; convolutional feature; scale adaption

OCIS codes 150.0155; 150.1135; 100.4999

1 引 言

目标跟踪主要依据视频序列起始帧给出的目标先验信息,在后续的视频序列中估计出目标的状态信息,属于机器视觉领域的一个重要研究方向,在智能监控、智能交通、无人驾驶及机器人自主导航等领域具有非常广泛的应用^[1-2]。虽然目标跟踪算法已经取得了很大进展,但在复杂环境中,由于容易受目标尺度变化、形状变化或场景中光照变化、背景干扰等不利因素影响,实现稳定可靠的目标跟踪仍是一项具有挑战性的任务^[1]。

基于相关滤波的目标跟踪由于具有较强的跟踪鲁棒性和较高的跟踪速度,受到了国内外学者的广泛关注。在 Bolme 等^[3]首次将相关滤波理论应用于目标跟踪后, Henriques 等^[4]采用梯度方向直方图(HOG)特征将相关滤波目标跟踪框架扩展为多通道模式,提高了算法对目标的跟踪能力。Bertinetto 等^[5]通过融合 HOG 特征和颜色直方图特征,进一步提升了算法的目标跟踪性能。文献[6]则在 HOG 特征基础上,利用颜色名(CN)特征对跟踪结果进行修正。文献[7]通过融合 HOG、CN、局部二值模式(LBP)3种特征,提高了算法对目标的跟踪精度。文献[8-9]则分别采用基于深度学习的目标检测模型和关键点匹配算法,解决了跟踪失败后的目标重检测问题。

上述方法由于采用的都是传统手工特征,在复杂环境下容易受外界因素干扰和目标自身变化影响,目标跟踪的稳定性还有待提升。近年来,随着卷积神经网络(CNN)在图像分类、目标识别等领域不断取得突破性进展,卷积特征表现出了强大的目标表征能力,并被应用到了目标跟踪领域^[2]。Wang 等^[10]在粒子滤波框架下通过栈式降噪自编码网络提取了图像特征,并对目标进行了跟踪。文献[11]利用不同层级的卷积特征分别建立通用网络和专用网络来改善目标跟踪的性能。Ma 等^[12]采用一种由粗到精的分层目标定位策略,该策略提高了目标跟

踪的精度。Qi 等^[13-14]利用不同的深层卷积特征构建多个弱分类器,并采用在线决策方法完成了目标跟踪,后续为提高跟踪性能又改进了决策方法。熊昌镇等^[15]利用两层不同的卷积特征进行切换跟踪。文献[16]则通过残差网络提取的卷积特征对目标进行跟踪。

上述基于卷积特征的目标跟踪方法尽管取得了比传统手工特征更好的跟踪效果,但仍存在一些不足。由于它们大多采用的是 CNN 提取的最后几层卷积特征,这些深层的卷积特征包含了丰富的目标语义信息,但空间分辨率较低,使目标的轮廓模糊并丢失了部分纹理信息,在进行目标跟踪时不利于实现对目标的精确定位。而浅层的卷积特征虽具有较高的空间分辨率,能够精确定位目标,但容易受外界干扰,目标跟踪的鲁棒性不佳。为取得更好的跟踪效果,需要综合利用不同层级的卷积特征,因此本文提出一种自适应多层卷积特征决策融合的目标跟踪算法。首先通过卷积神经网络 VGG-Net-19 提取目标候选区域的 5 层(Conv1-2、Conv2-2、Conv3-4、Conv4-4、Conv5-4)卷积特征;然后利用相关滤波模型分别为每一层卷积特征训练一个弱跟踪器,并采用自适应决策方法对多个弱跟踪器的跟踪结果进行融合,进而完成对目标的跟踪;在此基础上对候选目标中心区域进行多尺度采样,并通过尺度相关滤波模型和相邻帧的尺度变化先验分布完成对目标尺度的估计;选取 51 组视频序列来测试所提算法的跟踪性能,并对其结果和近年来主流的目标跟踪算法的结果进行对比分析。

2 相关滤波目标跟踪模型

基于相关滤波的目标跟踪方法利用目标特征与滤波器模板的相关操作来构造目标函数^[3-4],以目标的实际响应与期望响应的误差平方和最小化为准则,训练得到相关滤波器,即

$$\operatorname{argmin}_{h_p} \|h_p \star f_p - g_p\|_2^2 + \lambda_p \|h_p\|_2^2, \quad (1)$$

式中: h_p 为相关滤波器; f_p 为目标的特征表示; g_p

为目标的期望响应; λ_p 为防止过拟合的正则化参数; \star 为循环相关操作。利用循环相关的离散傅里叶变换(DFT)特性, (1)式在频域上的解为

$$\mathbf{H}_p = \frac{\mathbf{G}_p^* \otimes \mathbf{F}_p}{\mathbf{F}_p^* \otimes \mathbf{F}_p + \lambda_p \mathbf{I}}, \quad (2)$$

式中: \mathbf{H}_p 、 \mathbf{F}_p 及 \mathbf{G}_p 分别为 \mathbf{h}_p 、 \mathbf{f}_p 及 \mathbf{g}_p 的 DFT; \otimes 表示叉乘; 上标 $*$ 表示复共轭; \mathbf{I} 为单位矩阵。

当目标的特征表示具有多通道模式时, (1)式可以转换为

$$\operatorname{argmin}_{\mathbf{h}_{p,d}} \left\| \sum_{d=1}^D \mathbf{h}_{p,d} \star \mathbf{f}_{p,d} - \mathbf{g}_p \right\|_2^2 + \lambda_p \sum_{d=1}^D \|\mathbf{h}_{p,d}\|_2^2, \quad (3)$$

式中: D 为目标特征的通道数。同样, 相关滤波器在多通道模式下的各通道频域表示为

$$\mathbf{H}_{p,d} = \frac{\mathbf{G}_p^* \otimes \mathbf{F}_{p,d}}{\sum_{d=1}^D \mathbf{F}_{p,d}^* \otimes \mathbf{F}_{p,d} + \lambda_p \mathbf{I}}. \quad (4)$$

在进行目标跟踪时, 提取目标候选区域的 D 通

道特征图 \mathbf{z} , 其 DFT 为 \mathbf{Z} , 则它的相关滤波响应为

$$\mathbf{f} = \mathcal{F}^{-1} \left(\sum_{d=1}^D \mathbf{H}_{p,d}^* \otimes \mathbf{Z}_d \right). \quad (5)$$

(5)式中, 最大响应值的坐标即为目标位置。

3 多层卷积特征决策融合

3.1 卷积特征选择

随着高性能计算设备的快速发展和大规模训练数据集的不断更新, 涌现了众多性能优异的 CNN 模型, 其中 VGG-Net^[17] 由牛津大学和 Google 公司共同开发, 它通过 ImageNet 数据集中的 130 万幅图片训练得到, 具有出色的目标特征提取能力和迁移学习能力。CNN 作为一种多层级的信息感知器, 它的每个卷积层都能够获得输入图像的不同特征表示。图 1 为采用 VGG-Net 子集中的 VGG-Net-19 网络提取的不同层卷积特征图的可视化表示, 为了便于比较, 将各层卷积特征图插值为相同分辨率。

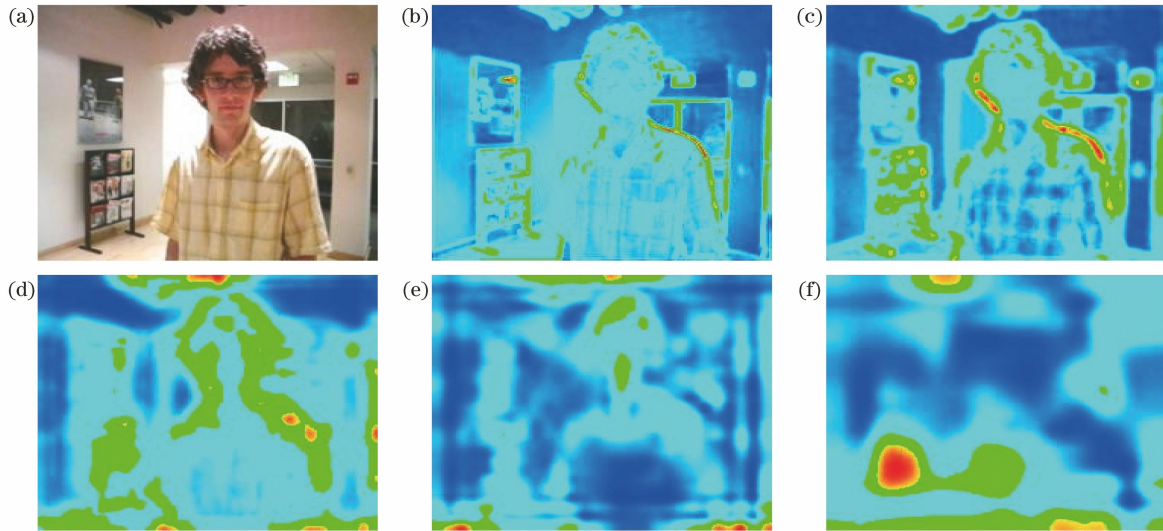


图 1 VGG-Net-19 网络中不同层卷积特征的可视化结果。(a)输入图像;(b) Conv1-2 层特征;(c) Conv2-2 层特征;(d) Conv3-4 层特征;(e) Conv4-4 层特征;(f) Conv5-4 层特征

Fig. 1 Visualization results of convolutional features in different layers on VGG-Net-19. (a) Input image; (b) feature in Conv1-2 layer; (c) feature in Conv2-2 layer; (d) feature in Conv3-4 layer; (e) feature in Conv4-4 layer; (f) feature in Conv5-4 layer

从图 1 可知: 与输入图像相比, 浅层(Conv1-2、Conv2-2)的卷积特征具有较高的空间分辨率, 同时保留了较清晰的目标轮廓、纹理等细节信息, 在目标跟踪时有利于完成精确的定位; 而深层(Conv4-4、Conv5-4)的卷积特征的空间分辨率较低, 包含了更多的目标语义信息, 在目标跟踪时有利于处理目标的外观变化和背景干扰。为了更充分地利用卷积特征进行目标跟踪, 选取 VGG-Net-19 网络中提取的

Conv1-2、Conv2-2、Conv3-4、Conv4-4 及 Conv5-4 共 5 层卷积特征进行目标跟踪。

3.2 自适应决策融合

为解决多个专家的在线决策问题, 文献[18]通过计算各个专家的累积代价函数来确定他们的决策权重, 该方法在计算累积代价函数时并未考虑当前决策损失相对历史决策损失的变化。但在复杂环境下的目标跟踪中, 目标自身和外界环境的变化是不

可避免的,各个专家在不同时刻的决策损失是有差异的,因此在目标跟踪时需要考虑各个专家决策损失的动态变化,并据此调节它们的累积代价函数,进而调整它们的决策权重以适应目标跟踪的变化。受文献[14,18]启发,本文提出一种自适应在线决策方法以实现多个跟踪器跟踪结果的融合。首先为所选取的5层卷积特征各训练一个相关滤波器,得到5个弱跟踪器;然后根据这些弱跟踪器的决策损失变化动态调节它们的累积代价函数,进而得到自适应的决策权重,实现多个弱跟踪器的融合跟踪。

由于CNN中的池化操作,随着卷积层级的加深,卷积特征图的尺寸逐渐减小。为了更好地融合不同层级的卷积特征,采用双线性插值方式将各层卷积特征图调整为相同尺寸。对于尺寸调整后的第 t 帧的第 l 层卷积特征图 $\mathbf{z}_{t,l} \in \mathbf{R}^{M \times N \times D_l}$, M 、 N 及 D_l 分别表示该卷积特征图的宽、高及通道数,其相关滤波响应为

$$\mathbf{f}_{t,l} = \mathcal{F}^{-1} \sum_{d=1}^{D_l} (\mathbf{H}_{t,l,d}^* \otimes \mathbf{Z}_{t,l,d})。 \quad (6)$$

(6)式中,最大响应值对应的坐标即为利用该层卷积特征得到的目标位置,即

$$\mathbf{p}_{t,l} = (m_{t,l}, n_{t,l}) = \underset{M,N}{\operatorname{argmax}}(\mathbf{f}_{t,l})。 \quad (7)$$

式中: $m_{t,l}$ 和 $n_{t,l}$ 为利用第 l 层特征得到的目标坐标。在利用每一层卷积特征得到它们对应的目标位置后,按照在线决策方法确定目标的最终位置,表达式为

$$\mathbf{p}_t = \sum_{l=1}^L \alpha_{t,l} \times \mathbf{p}_{t,l}, \quad (8)$$

式中: L 为弱跟踪器的个数,它与卷积特征图的层数相对应; $\alpha_{t,l}$ 为每一层卷积特征的决策权重, $\sum_{l=1}^L \alpha_{t,l} = 1$ 。在融合各个弱跟踪器的跟踪结果得到最终的目标位置后,每一层的弱跟踪器都将有一个对应的决策损失。为准确评估各个弱跟踪器的融合效果,定义第 t 帧的第 l 层弱跟踪器的决策损失为

$$\zeta_{t,l} = 1 - O_{S_{t,l}}, \quad (9)$$

式中: $O_{S_{t,l}} = \frac{\operatorname{Area}(B_{\operatorname{Box}_{t,l}} \cap B_{\operatorname{Box}_t})}{\operatorname{Area}(B_{\operatorname{Box}_{t,l}} \cup B_{\operatorname{Box}_t})}$,表示第 l 层弱跟踪器得到的目标区域 $B_{\operatorname{Box}_{t,l}}$ 与最终融合得到的目标区域 B_{Box_t} 的重叠度, $O_{S_{t,l}}$ 值越大,表示两者的重叠度越高,则该弱跟踪器的决策损失越小。在得到每一层的决策损失后,第 t 帧总的决策损失为

$$\bar{\zeta}_t = \sum_{l=1}^L \alpha_{t,l} \times \zeta_{t,l}。 \quad (10)$$

这里,定义第 t 帧的第 l 层弱跟踪器在当前帧的代价函数 $R_{t,l}$ 和连续 T 帧的累积代价函数 $R_{C_{t,l}}$ 分别为

$$R_{t,l} = \bar{\zeta}_t - \beta_{t,l} \zeta_{t,l}, \quad (11)$$

$$R_{C_{t,l}} = R_{C_{t-1,l}} + R_{t,l}, \quad (12)$$

式中: $R_{C_{t-1,l}}$ 为前 $T-1$ 帧的累积代价函数。 $\beta_{t,l}$ 由双曲正切函数定义,表达式为

$$\beta_{t,l} = \tanh(\gamma s_{t,l}), \quad (13)$$

式中:形状调节系数 $\gamma > 0$ 。(13)式用于反映第 l 层弱跟踪器在当前帧相对于历史帧的决策损失变化。 $s_{t,l}$ 的表达式为

$$s_{t,l} = \frac{\zeta_{t,l} - \mu_{t,l}}{\sigma_{t,l}}, \quad (14)$$

式中:该弱跟踪器连续 T 帧的决策损失均值 $\mu_{t,l} = \frac{1}{T} \sum_{\tau=t-T+1}^t \zeta_{\tau,l}$;该弱跟踪器连续 T 帧的决策损失方

差 $\sigma_{t,l} = \sqrt{\frac{1}{T-1} \sum_{\tau=t-T+1}^t (\zeta_{\tau,l} - \mu_{t,l})^2}$ 。

当 $s_{t,l}$ 为负时,表示第 l 层弱跟踪器在当前帧的决策损失低于其历史帧的平均决策损失,此时(13)式中的 $\beta_{t,l}$ 也将为负,那么(11)式中的当前帧代价函数 $R_{t,l}$ 相对总的决策损失将增加。同样的,(12)式中的累积代价函数 $R_{C_{t,l}}$ 也将增加,这说明当该弱跟踪器在当前帧的跟踪性能优于历史帧时,应增大其当前帧的决策权重。反之,该弱跟踪器在当前帧的代价函数和累积代价函数都在减小,则降低其在当前帧的决策权重。这样,可以根据各个弱跟踪器的决策损失变化来自适应地调节它们的代价函数和累积代价函数,然后动态调整它们的决策权重,进而更好地适应目标跟踪的变化。

通过求解(12)式的累积代价函数^[18],得到在第 $t+1$ 帧的第 l 层弱跟踪器的决策权重为

$$\alpha_{t+1,l} = \frac{[R_{C_{t,l}}]_+}{c_t} \exp \frac{([R_{C_{t,l}}]_+)^2}{2c_t}, \quad (15)$$

式中: $[R_{C_{t,l}}]_+ = \max\{0, R_{C_{t,l}}\}$ 为非负值; c_t 为尺度因子。图2为所提自适应决策融合跟踪算法的过程示意图。

3.3 尺度自适应方法

在跟踪过程中,目标发生尺度变化现象是非常普遍的,传统的相关滤波模型^[3-4]仅对目标位置进行预测,并未考虑目标的尺度变化。为了进一步提升算法的跟踪性能,通过最小化(16)式得到的尺度相关滤波器 $\mathbf{h}_{s,j}$ 来估计目标的尺度变化^[19]。

$$\underset{\mathbf{h}_{s,j}}{\operatorname{argmin}} \left\| \sum_{j=1}^J \mathbf{h}_{s,j} \star \mathbf{f}_{s,j} - \mathbf{g}_s \right\|_2^2 + \lambda_s \sum_{j=1}^J \|\mathbf{h}_{s,j}\|_2^2, \quad (16)$$

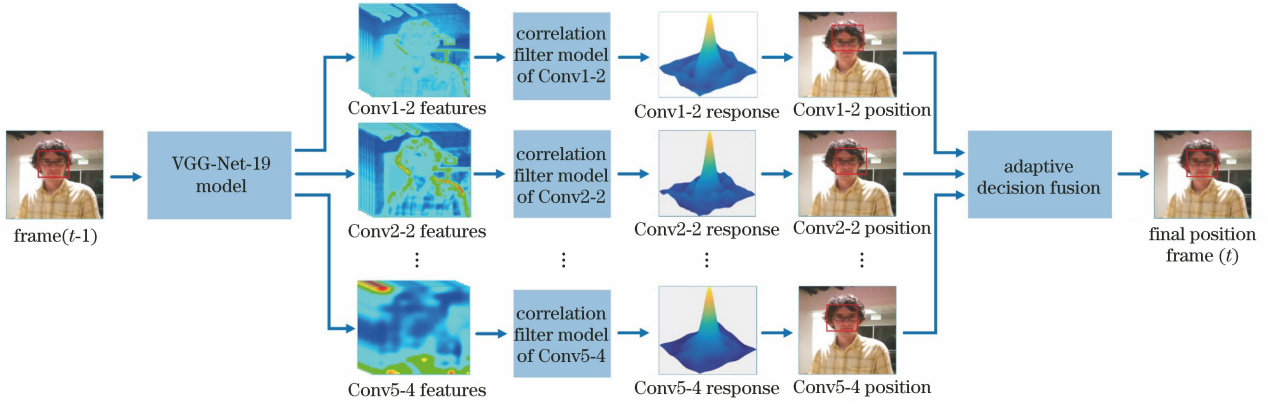


图 2 所提算法的过程示意图

Fig. 2 Procedure diagram of proposed algorithm

式中: $f_{s,j}$ 为尺度样本的多通道特征表示; g_s 为期望的尺度响应; λ_s 为防止过拟合的正则化系数; J 为尺度样本的特征通道数。与(4)式类似, 尺度相关滤波器每个通道在频域上表示为

$$H_{s,j} = \frac{G_s^* \otimes F_{s,j}}{\sum_{j=1}^J F_{s,j}^* \otimes F_{s,j} + \lambda_s I}, \quad (17)$$

式中: $H_{s,j}$ 、 $F_{s,j}$ 、 G_s 分别为 $h_{s,j}$ 、 $f_{s,j}$ 、 g_s 的 DFT。

在得到当前帧的目标位置 p_t 后, 利用目标在前一帧的尺度 s_{t-1} , 按照(18)式, 在候选目标的中心区域进行多尺度采样。

$$s_{t,n} = \{s_n = s_{t-1} + k \Delta s, k \in [-K/2, K/2]\}, \quad (18)$$

式中: Δs 、 k 和 K 分别为尺度样本的采样间隔、步长和样本数; n 为尺度样本编号。与文献[20]不同, 为了更精确地估计目标的尺度变化, 本文对尺度样本的采样间隔进行了更细致的划分, 同时增加尺度样本数, 可以适应更大范围的目标尺度变化。在得到尺度样本后, 通过双线性插值方式将它们调整为相同大小, 并分别提取它们的 HOG 特征进行尺度估计。之所以选择 HOG 特征, 主要是因为尺度估计与正常跟踪时的目标位置估计不同, 它需要采集不同尺度下的样本, 如果每个尺度样本都利用 CNN 进行特征提取, 则需要很大的计算开销; 另外, HOG 特征中, 通过计算尺度样本中各点的梯度幅值和角度, 并进行梯度的投影映射和归一化, 可以得到 18 通道的有向梯度特征、9 通道的无向梯度特征及 4 通道的纹理特征。这些特征能够有效表征目标的轮廓信息和形状信息, 并具有旋转和光照不变性。在提取这些尺度样本的 HOG 特征后, 对它们进行 DFT, 得到尺度样本的频域集合 $F_{t,s_n} = \{F_{t,s_1}, F_{t,s_2}, \dots, F_{t,s_K}\}$, 利用目标在第 $t-1$ 帧的尺度相关

滤波器 $H_{t-1,s,j}$, 得到当前帧各尺度样本的相关滤波响应为

$$f_{t,s_n} = \mathcal{F}^{-1} \left(\sum_{j=1}^J H_{t-1,s,j}^* \otimes F_{t,s_n,j} \right). \quad (19)$$

通过寻找(19)式的最大值, 获得当前帧的目标尺度估计, 即

$$s_t = \underset{s_n}{\operatorname{argmax}} (f_{t,s_n}). \quad (20)$$

考虑到在目标跟踪时相邻帧的时间间隔很短, 相邻帧的目标尺度变化能够近似为一种高斯分布^[21], 则目标尺度变化的最大后验概率估计的表达式为

$$p(s_t | f_{t,s_n}) = p(f_{t,s_n} | s_t) p(s_t), \quad (21)$$

式中: $p(f_{t,s_n} | s_t)$ 为尺度相关滤波器得到的尺度变化似然估计; $p(s_t)$ 为尺度变化的先验分布。

3.4 模型更新机制

由于目标在跟踪过程中会不可避免地出现尺度、旋转和形状等变化, 同时还会受到场景光照变化、背景干扰等因素影响, 为避免跟踪漂移, 需要对跟踪模型进行必要的更新。对各卷积层的相关滤波器和尺度相关滤波器进行更新的表达式分别为

$$\begin{cases} A_{t,l,d_l} = (1 - \eta_p) A_{t-1,l,d_l} + \eta_p G_{t,l}^* \otimes F_{t,l,d_l} \\ B_{t,l,d_l} = (1 - \eta_p) B_{t-1,l,d_l} + \eta_p \sum_{d_l=1}^{D_l} F_{t,l,d_l}^* \otimes F_{t,l,d_l} \\ H_{t,l,d_l} = \frac{A_{t,l,d_l}}{B_{t,l,d_l} + \lambda_p I} \end{cases}, \quad (22)$$

$$\begin{cases} A_{t,s,j} = (1 - \eta_s) A_{t-1,s,j} + \eta_s G_{t,s}^* \otimes F_{t,s,j} \\ B_{t,s,j} = (1 - \eta_s) B_{t-1,s,j} + \eta_s \sum_{j=1}^J F_{t,s,j}^* \otimes F_{t,s,j} \\ H_{t,s,j} = \frac{A_{t,s,j}}{B_{t,s,j} + \lambda_s I} \end{cases}, \quad (23)$$

式中： \mathbf{A}_{t-1,l,d_l} 、 \mathbf{B}_{t-1,l,d_l} 分别为第 l 层的相关滤波器在 $t-1$ 帧的分子项和分母项； η_p 为更新率； $\mathbf{A}_{t-1,s,j}$ 、 $\mathbf{B}_{t-1,s,j}$ 分别为尺度相关滤波器在 $t-1$ 帧的分子项

和分母项； η_s 为更新率。

3.5 算法流程

所提算法的具体工作流程如图 3 所示。

-
- Input: target initial position $p_1=(m_1, n_1)$ and scale $s_1=(w_1, h_1)$ in the first frame
 Output: estimated target position $p_t=(m_t, n_t)$ and scale $s_t=(w_t, h_t)$ in each subsequent frame
- 1) train the correlation filter model \mathbf{H}_{1,l,d_l} with convolutional features for each weak tracker and the scale correlation filter model $\mathbf{H}_{1,s,j}$ with HOG features, initialize the parameters $\alpha_{1,l}$, $\zeta_{1,l}$ and $R_{1,l}$ of each weak tracker
 - 2) for $t=2,3,\dots,T$ do
 - 3) for $l=1,2,\dots,L$ do
 - 4) calculate the estimated target position $p_{t,l}$ of each weak tracker according to Eq. (6) and (7)
 - 5) end for
 - 6) calculate the final target position p_t using Eq. (8)
 - 7) sample multi-scale image patches at the estimated target center and accomplish its scale estimation s_t using Eq. (18), (19), (20), and (21)
 - 8) for $l=1,2,\dots,L$ do
 - 9) calculate the decision loss $\zeta_{t,l}$ using Eq. (9) and update the cumulative regret $R_{\zeta,t,l}$ of each weak tracker according to Eq. (12)
 - 10) calculate the next frame decision weight $\alpha_{t+1,l}$ of each weak tracker according to Eq. (15)
 - 11) update the correlation filter model \mathbf{H}_{1,l,d_l} of each weak tracker using Eq. (22)
 - 12) end for
 - 13) update the scale correlation filter model $\mathbf{H}_{1,s,j}$ using Eq. (23)
 - 14) end for
-

图 3 所提算法的工作流程

Fig. 3 Flow chart of proposed algorithm

4 实验结果与分析

4.1 实验环境与测试条件

完成算法的计算机硬件配置:CPU 为 Intel(R) Core(TM) i5-2500,主频 3.30GHz,内存 8 GB,用于加速计算的 GPU 为 Nvidia GeForce GTX1050。算法软件配置为 Matlab R2016b 和 Matconvnet1.0。实验选取了 OTB50 数据集^[22]中的 51 组视频序列进行目标跟踪的性能测试,这些视频序列中包含了目标跟踪面临的多种挑战因素。在实验中,所提算法的具体参数设置:(8)式中各个弱跟踪器的初始决策权重统一设置为 0.2;(13)式中 $\gamma=0.2$,决策损失的统计帧数 $T=6$;(18)式中尺度样本采样参数分别为 $\Delta s=0.015, k=1, K=33$;(22)、(23)式中更新率分别为 $\eta_p=0.01, \eta_s=0.025$ 。

采用文献[22]给出的一次性通过评估(OPE)方法,距离精度(DP)及重叠精度(OP)两种评价指标对目标跟踪算法的性能进行评估,在进行 DP 和 OP 指标评价时,中心位置误差(CLE)和重叠率(OS)的阈值分别设置为 20 和 0.5。另外,为验证所提算法的跟踪性能,在 OTB50 数据集中对 39 种算法进行对比,这些算法包括所提算法、基于深度特征的

DLT 算法^[10]、FCNT 算法^[11]、HDT 算法^[13]、基于相关滤波的 KCF 算法^[4]、DSST 算法^[19]、全卷积孪生网络 SiamFC 算法^[23]、基于结构化支持向量机的 DLSSVM 算法^[24]、多专家融合跟踪的 MEEM 算法^[25]、采用迁移学习的 TGPR 算法^[26],另外还包括了在 OTB50 数据集中表现优异且具有代表性的其他 29 种算法,如 Struck 算法、SCM 算法、TLD 算法及 ASLA 算法等。

4.2 多层卷积特征融合性能分析

为了验证所提算法的目标跟踪效果,在 OTB50 数据集中对所提算法与采用单层卷积特征的各个弱跟踪器进行对比,各个弱跟踪器除了采用的卷积特征与所提算法存在差异外,其余的参数设置均与所提算法一致,对比实验结果如图 4 所示。

图 4 中 Conv1-2、Conv2-2、Conv3-4、Conv4-4 及 Conv5-4 分别为各个单层卷积特征所对应的 5 个弱跟踪器。可以看出:多层卷积特征融合后的跟踪性能均优于各个单层特征;相对于次优的弱跟踪器 Conv4-4,融合后的 DP 评价指标和 OP 评价指标分别提高了 7.5% 和 8.3%。这说明相对于单层的卷积特征,所提算法可以有效提升目标跟踪的性能。

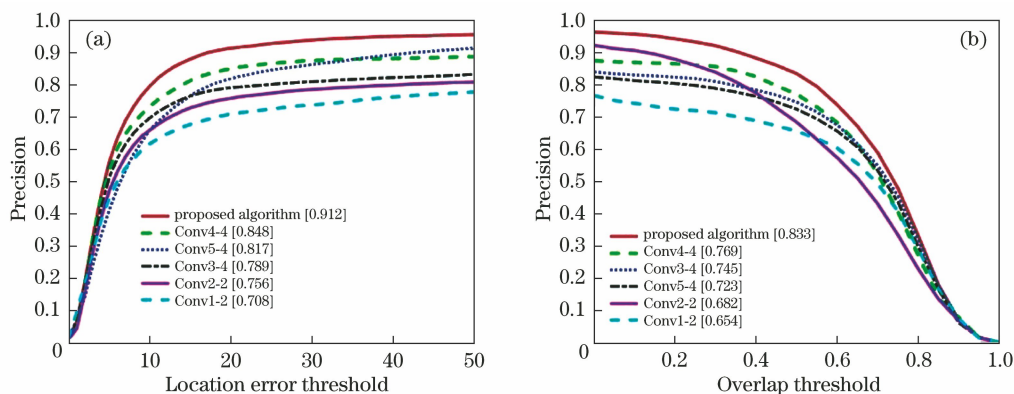


图 4 所提算法与各个弱跟踪器的距离精度和成功率。(a)距离精度;(b)成功率

Fig. 4 Distance precision and success rate of proposed algorithm and other weak trackers. (a) Distance precision; (b) success rate

4.3 尺度自适应性能分析

选取 OTB50 数据集中存在尺度变化的 28 组视频序列对不同算法进行性能评估。图 5 为 39 种

算法在 28 组尺度变化视频序列中的 DP 和 OP 两种评价指标的对比结果,为了方便对比,仅列出指标排名前 12 的算法。

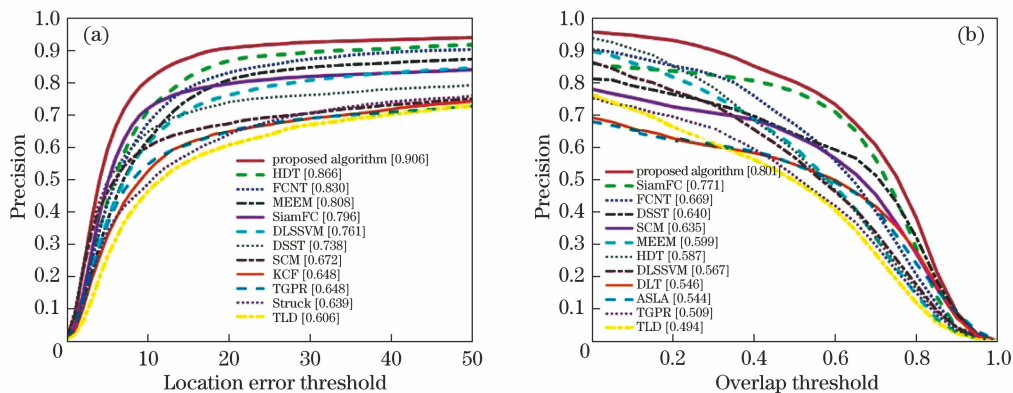


图 5 不同算法在尺度变化视频序列中的距离精度和成功率。(a)距离精度;(b)成功率

Fig. 5 Distance precision and success rate of different algorithms under scale variation sequences.

(a) Distance precision; (b) success rate

在图 5(a)中,所提算法的 DP 评价指标为 0.906,相对于排名第 2 的 HDT 算法,提高了 4.6%;在图 5(b)中,所提算法的 OP 评价指标为 0.801,相对于排名第 2 的 SiamFC 算法,提高了 3.9%。但需要指出的是,在这些存在尺度变化的视频序列中,DP 指标排名第 2 的 HDT 算法的 OP 指标相对于所提算法降低了 26.7%,这主要是由于 HDT 算法缺乏尺度估计策略,它虽然也能较准确地定位目标,但给出的跟踪框不能精准并完整地覆盖目标。另外,OP 指标排名第 2 的 SiamFC 算法的 DP 评价指标和所提算法相比降低了 12.1%。由此可以看出,所提算法可以较好地适应目标的尺度变化,并且其尺度预测性能也优于对比算法。

4.4 整体跟踪性能分析

4.4.1 定量分析

为了更加充分地验证所提算法的跟踪性能,在

OTB50 数据集的全部 51 组视频序列中对所提算法和其他 38 种算法进行对比测试。图 6 为 39 种算法在全部 51 组视频序列中的整体 DP 和 OP 评价指标的对比结果,同样,这里只列出每种指标排名前 12 的算法。

在图 6(a)中,所提算法的 DP 评价指标相对排名第 2 的 HDT 算法提高了 2.6%,其 OP 指标相对于 HDT 算法则提升了 13.0%。在图 6(b)中,所提算法的 OP 评价指标相对于排名第 2 的 SiamFC 算法提高了 6.4%,其 DP 评价指标相对于 SiamFC 算法则提升了 12.7%。综上所述可以看出,所提算法的整体跟踪性能更佳。

为进一步评估所提算法的性能,在 OTB50 数据集包含相关挑战因素的视频序列上分别对图 6 中 DP 和 OP 评价指标排名前 12 的算法进行对比分析。这些挑战因素共有 11 种,具体包括:光照

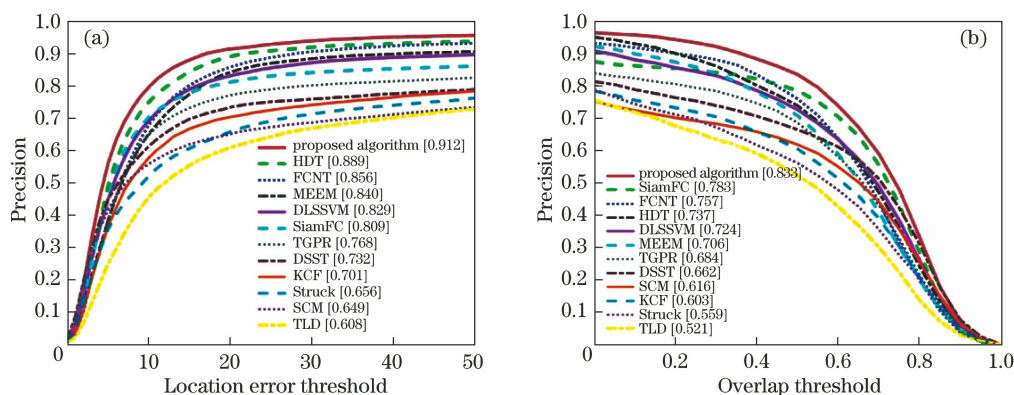


图 6 不同算法在全部视频序列中的距离精度和成功率。(a)距离精度;(b)成功率
Fig. 6 Distance precision and success rate of different algorithms under all sequences.

(a) Distance precision; (b) success rate

变化(IV)、形状变化(DEF)、尺度变化(SV)、遮挡(OCC)、运动模糊(MB)、快速运动(FM)、平面旋转(IPR)、非平面旋转(OPR)、超出视野(OV)、背景干

扰(BC)、低分辨率(LR)。算法的 DP 和 OP 评价指标的对比结果如表 1、2 所示,可以看出:所提算法在大部分挑战因素中取得了最优(显示加粗)和次优的

表 1 排名前 12 的算法在 11 种不同挑战因素下的 DP 评价指标

Table 1 DP scores of the top twelve algorithms under eleven challenging attributes

Algorithm	IV	DEF	SV	OCC	MB	FM	IPR	OPR	OV	BC	LR
Proposed algorithm	0.868	0.876	0.906	0.908	<u>0.828</u>	0.837	0.881	0.897	<u>0.766</u>	0.886	0.847
HDT	<u>0.845</u>	0.884	<u>0.866</u>	<u>0.874</u>	0.840	<u>0.782</u>	<u>0.869</u>	<u>0.871</u>	<u>0.679</u>	<u>0.871</u>	<u>0.846</u>
FCNT	0.830	0.917	0.830	0.797	0.789	0.767	0.811	0.831	0.741	0.799	0.765
MEEM	0.778	0.859	0.808	0.814	0.740	0.757	0.809	0.853	0.730	0.808	0.494
DLSSVM	0.754	<u>0.892</u>	0.761	0.811	0.749	0.731	0.807	0.840	0.697	0.818	0.553
SiamFC	0.709	0.744	0.796	0.802	0.700	0.721	0.744	0.788	0.777	0.732	0.659
TGPR	0.709	0.812	0.648	0.738	0.662	0.611	0.724	0.750	0.509	0.730	0.399
DSST	0.730	0.636	0.738	0.692	0.544	0.513	0.768	0.725	0.511	0.694	0.497
KCF	0.657	0.698	0.648	0.695	0.571	0.534	0.691	0.678	0.590	0.676	0.387
Struck	0.558	0.521	0.639	0.564	0.551	0.604	0.617	0.597	0.539	0.585	0.545
SCM	0.594	0.586	0.672	0.640	0.339	0.333	0.597	0.618	0.429	0.578	0.305
TLD	0.537	0.512	0.606	0.563	0.518	0.551	0.584	0.596	0.576	0.428	0.349

表 2 排名前 12 的算法在 11 种不同挑战因素下的 OP 评价指标

Table 2 OP scores of the top twelve algorithms under eleven challenging attributes

Algorithm	IV	DEF	SV	OCC	MB	FM	IPR	OPR	OV	BC	LR
Proposed algorithm	0.758	<u>0.786</u>	0.801	0.839	0.748	0.768	0.769	0.793	0.742	0.774	0.738
HDT	0.662	0.765	0.587	0.739	0.748	0.705	0.715	0.716	0.685	<u>0.769</u>	0.636
FCNT	<u>0.755</u>	0.813	0.669	0.706	0.724	0.718	0.688	0.724	<u>0.748</u>	0.721	0.620
MEEM	0.667	0.719	0.599	0.694	0.721	<u>0.726</u>	0.643	0.695	0.741	0.741	0.472
DLSSVM	0.648	0.778	0.567	0.723	<u>0.730</u>	0.693	0.690	0.714	0.712	0.745	0.503
SiamFC	0.679	0.705	<u>0.771</u>	<u>0.769</u>	0.666	0.699	<u>0.725</u>	<u>0.756</u>	0.794	0.705	<u>0.651</u>
TGPR	0.629	0.770	0.509	<u>0.665</u>	0.655	0.590	<u>0.645</u>	<u>0.661</u>	0.527	0.688	<u>0.410</u>
DSST	0.681	0.610	0.640	0.632	0.528	0.503	0.679	0.632	0.512	0.627	0.497
KCF	0.543	0.628	0.474	0.580	0.561	0.523	0.613	0.579	0.610	0.630	0.355
Struck	0.491	0.473	0.471	0.493	0.518	0.567	0.528	0.506	0.550	0.545	0.410
SCM	0.568	0.565	0.635	0.599	0.339	0.335	0.560	0.575	0.449	0.550	0.308
TLD	0.460	0.456	0.494	0.468	0.482	0.473	0.476	0.497	0.516	0.388	0.327

(显示加下划线)跟踪结果。综合上述实验结果,所提算法除了具有较好的尺度适应能力外,在其他10种常见的目标跟踪挑战因素中也均具有较好的跟踪性能。

4.4.2 定性分析

为了更加直观地评估所提算法的跟踪性能,对所提算法和其他5种算法(包括HDT算法、FCNT算法、DSST算法、MEEM算、TGPR算法)在6组具有不同挑战因素的视频序列Doll、Carscale、

Freeman3、Skating1、Matrix、Liquor 中进行对比验证,结果分别如图7~12所示。

图7是Doll序列的跟踪结果,展示了算法对目标尺度变化的适应能力。如图7(a)~(c)所示,当目标尺度逐渐增大时,HDT算法、FCNT算法、MEEM算法及TGPR算法都只能跟踪到目标局部。在图7(d)中,当目标尺度缩小时,FCNT算法的跟踪框中包含了较多的背景信息,只有所提算法和DSST算法可以较好地适应目标的尺度变化。

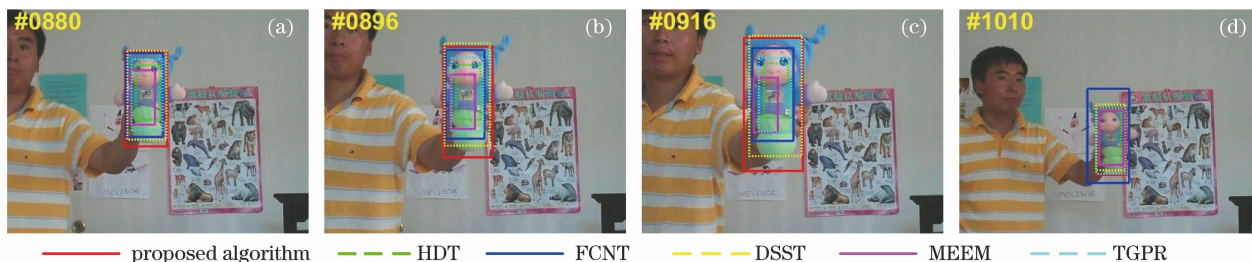


图7 不同算法在Doll序列中的跟踪结果

Fig. 7 Tracking results of different algorithms on Doll sequence

图8是Carscale序列的跟踪结果,相对于图7,该序列中的目标尺度变化更为剧烈。当目标尺度变化不大时,各个算法均能较好地跟踪目标,如图8(a)所示;当目标尺度逐渐增大时,只有所提算

法和DSST算法可以较完整地覆盖目标,如图8(b)所示;当目标的尺度变化进一步加剧时,如图8(c)、(d)所示,DSST算法也只跟踪到了目标局部,但所提算法仍可较好地适应目标尺度的剧烈变化。

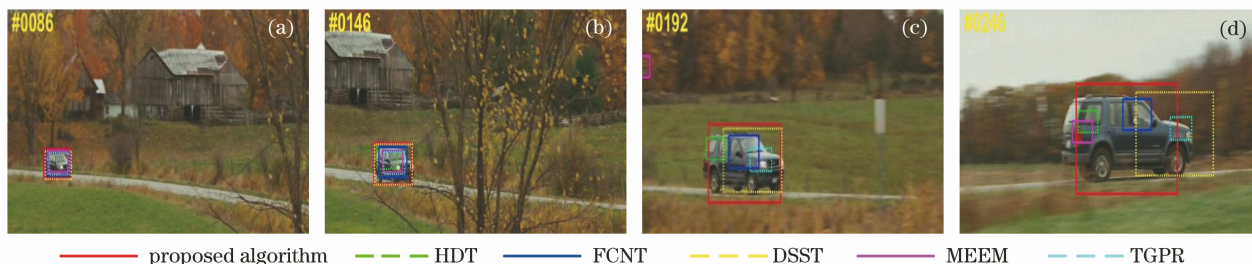


图8 不同算法在Carscale序列中的跟踪结果

Fig. 8 Tracking results of different algorithms on Carscale sequence

图9是Freeman3序列的跟踪结果,目标在该序列中发生了多次平面内和非平面内的旋转变化。在图9(a)中,TGPR算法跟踪到了错误目标,而HDT算法、FCNT算法、DSST算法及MEEM算法

也只跟踪到了目标局部;在后续的第424帧、432帧及456帧中,HDT算法、FCNT算法及DSST算法先后丢失了目标,所提算法在此过程中则一直较准确地跟踪了目标。

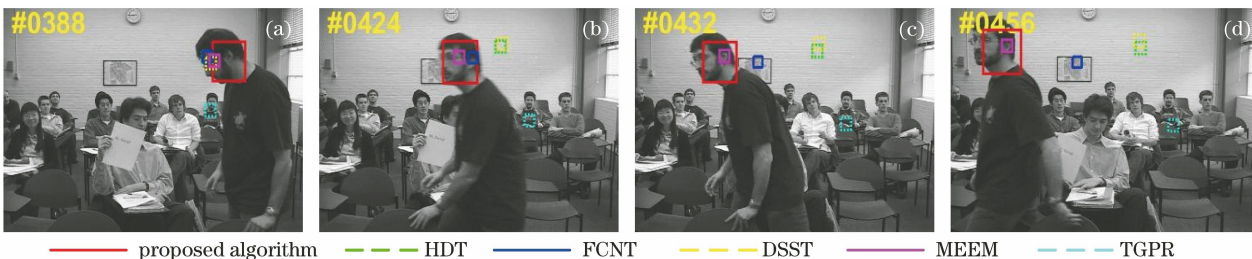


图9 不同算法在Freeman3序列中的跟踪结果

Fig. 9 Tracking results of different algorithms on Freeman3 sequence

图 10 是 Skating1 序列的跟踪结果,目标在该序列中发生了较明显的形状变化。如图 10(a)所示,MEEM 算法和 TGPR 算法出现了明显的跟踪漂移,并在第 288 帧时出现了跟踪失败;FCNT 算

法和 DSST 算法在第 310 帧时出现了不同程度的跟踪漂移,这种现象在第 314 帧时进一步加剧;但所提算法和 HDT 算法则能一直较稳定地跟踪目标。



图 10 不同算法在 Skating1 序列中的跟踪结果

Fig. 10 Tracking results of different algorithms on Skating1 sequence

图 11 是 Matrix 序列的跟踪结果,该序列出现了多次明显的光照变化。FCNT 算法和 TGPR 算法在第 4 帧时出现了明显的跟踪漂移,并在第 7 帧时丢失了目标;同样的,DSST 算法在第 30 帧时跟

踪到了背景上;HDT 算法和 MEEM 算法的跟踪框在第 50 帧时包含了较多的背景信息,而所提算法表现出了较强的光照变化适应能力。

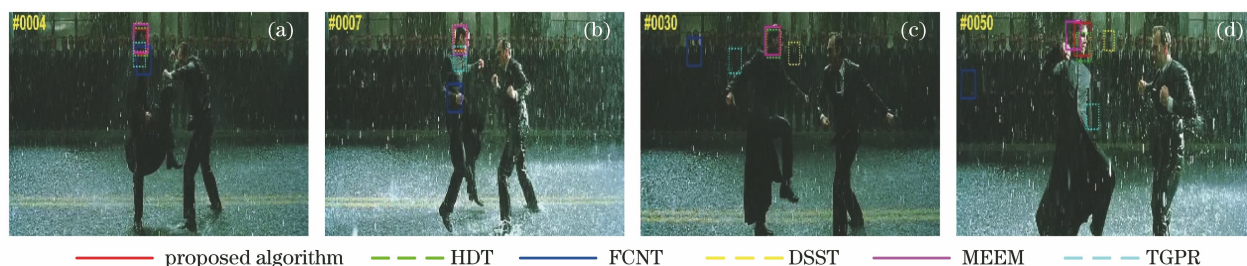


图 11 不同算法在 Matrix 序列中的跟踪结果

Fig. 11 Tracking results of different algorithms on Matrix sequence

图 12 是 Liquor 序列的跟踪结果,该序列的背景比较复杂并且存在较多的相似目标干扰。在图 12(a)中,DSST 算法和 TGPR 算法已经跟踪到了错误目标;在图 12(b)中,当目标开始运动时,FCNT 算法也出现了跟踪失败现象,HDT 算法的

跟踪框中则包含了较多的背景信息,并在图 12(c)中跟踪到了干扰目标上;在图 12(d)中,MEEM 算法的跟踪框也不能完整地覆盖目标,所提算法则出色地完成了跟踪任务。

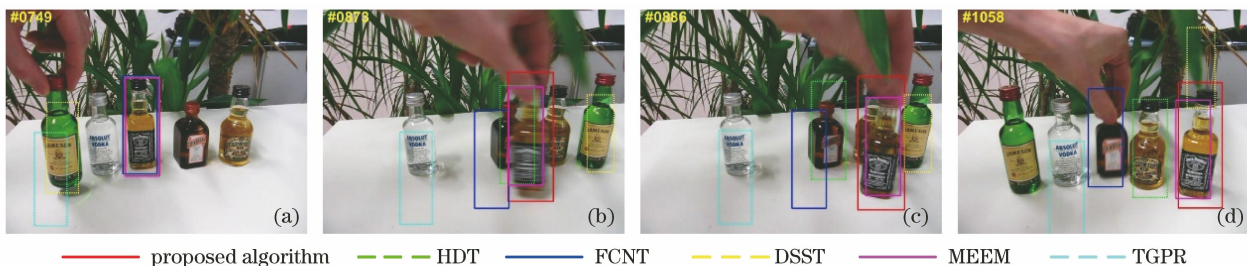


图 12 不同算法在 Liquor 序列中的跟踪结果

Fig. 12 Tracking results of different algorithms on Liquor sequence

综上,所提算法不仅能够适应目标尺度的快速变化,而且可以适应目标的旋转和形状变化,同时能够应对外界环境的光照变化和背景干扰等多种挑战因素。

4.5 算法跟踪的实时性分析

所提算法主要是利用 CNN 提取的卷积特征在相关滤波模型框架下完成目标跟踪,它的时间复杂度可以表示为

$$T_c = O\left(\sum_{l=1}^L M_l^2 K_l^2 D_{l-1} D_l\right) + O[MND \log(MND)], \quad (24)$$

式中： $O\left(\sum_{l=1}^L M_l^2 K_l^2 D_{l-1} D_l\right)$ 为卷积特征提取的时间复杂度； M_l 和 K_l 分别为卷积特征图和卷积核的尺寸； $O[MND \log(MND)]$ 为相关滤波计算的时间复杂度。

在实际应用中，可以利用目标跟踪算法的执行效率反映时间复杂度，目标跟踪算法的执行效率通常由每秒完成目标跟踪的图像帧数(FPS)来表征，这个值越大，说明算法的实时性越好，执行效率也越高。在采用 GPU 进行加速计算的情况下，所提算法在 OTB50 数据集集中的 51 组视频序列的 FPS 平均值为 11.6。表 3 为不同算法的实时性比较，并给出了算法对应的代码实现方式，其中 M 表示 Matlab, C 表示 C 和 C++。

表 3 不同算法在所有测试视频序列中的平均跟踪速度

Table 3 Average tracking speed of different algorithms on all video sequences

Parameter	Proposed algorithm	HDT	FCNT	DSST	MEEM	TGPR
Code format	M+C	M+C	M	M+C	M+C	C
Average FPS	11.6	10	3	32.2	10	3

表 3 中，所提算法、HDT 算法和 FCNT 算法都是在相关滤波模型下利用相同 CNN 提取的卷积特征完成目标跟踪的，它们的时间复杂度属于同一量级。但由于所提算法和 HDT 算法在运行时调用了部分 C 和 C++ 计算模块，因此它们的实时性高于 FCNT 算法。另外，相对于所提算法，HDT 算法使用了更多的深层卷积特征，它的卷积特征通道数多于所提算法，因此实时性比所提算法稍差。所提算法与 DSST 算法相比，所提算法在目标跟踪时使用了目标表征能力更强的卷积特征，因此它的时间复杂度有所提高，但是所提算法的 DP 和 OP 评价指标与 DSST 算法相比分别提高了 24.6% 和 25.8%，跟踪性能得到明显提升。最后，与 MEEM 算法和 TGPR 算法相比，所提算法采用了基于相关滤波的跟踪模型，可以通过快速傅里叶变换降低时间复杂度，并利用目标表征能力更强的卷积特征使 DP 评价指标分别提高了 8.6% 和 18.8%，OP 评价指标分别提高了 18.0% 和 21.8%。

5 结 论

根据相关滤波目标跟踪模型，提出了一种基于

自适应多层卷积特征决策融合的目标跟踪算法。为了充分发挥卷积特征的目标表征能力，选取了卷积神经网络 VGG-Net-19 提取的不同层级的卷积特征，利用这些卷积特征在相关滤波模型框架下训练得到了多个弱跟踪器。同时为了适应目标跟踪过程中的变化，根据各个弱跟踪器的决策损失，自适应地调整这些弱跟踪器的决策权重，融合得到一个强跟踪器完成对目标的位置估计。接着通过尺度相关滤波模型和相邻帧之间的目标尺度变化先验分布实现对目标的尺度变化估计。最后在 51 组具有多种挑战因素的视频序列上对所提算法进行了跟踪性能测试，并与当前其他性能优异的主流目标跟踪算法进行对比分析。实验结果表明，所提算法不仅能较好地适应目标的尺度变化，并且在目标发生旋转变、形状变化及场景中出现光照变化、背景干扰等多种挑战因素下仍具有较高的跟踪精度，整体性能优于对比算法。另外，在实验中也发现，所提算法在目标被严重遮挡时也会出现跟踪失败现象，后续将增加目标重检测机制以进一步提高算法的目标跟踪性能。

参 考 文 献

- [1] Lu H C, Wang D. Online visual tracking[M]. Singapore: Springer, 2019.
- [2] Li X, Zha Y F, Zhang T Z, et al. Survey of visual object tracking algorithms based on deep learning[J]. Journal of Image and Graphics, 2019, 24(12): 2057-2080.
李玺, 查宇飞, 张天柱, 等. 深度学习的目标跟踪算法综述[J]. 中国图象图形学报, 2019, 24(12): 2057-2080.
- [3] 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.
- [4] 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.
- [5] 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.

- [6] Zhang G S, Hao J X. Tracking algorithm based on location correction mechanism and model update strategy[J]. *Information and Control*, 2020, 49(2): 177-187.
张国山, 郝婧漩. 基于位置修正机制和模型更新策略的跟踪算法[J]. *信息与控制*, 2020, 49(2): 177-187.
- [7] Chang M, Shen K, Zhang X D, et al. Correlation filter tracking based on adaptive feature fusion and model updating[J]. *Acta Optica Sinica*, 2019, 39(9): 0915001.
常敏, 沈凯, 张学典, 等. 自适应特征融合和模型更新的相关滤波跟踪[J]. *光学学报*, 2019, 39(9): 0915001.
- [8] 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.
- [9] 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.
- [10] Wang N Y, Yeung D Y. Learning a deep compact image representation for visual tracking[C]//Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS), December 5-10, 2013, Lake Tahoe, NV, USA. New York: Curran Associates, 2013: 809-817.
- [11] Wang L J, Ouyang W L, Wang X G, et al. Visual tracking with fully convolutional networks[C]//2015 IEEE International Conference on Computer Vision (ICCV), December 7-13, 2015, Santiago, Chile. New York: IEEE Press, 2015: 3119-3127.
- [12] 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 Press, 2015: 3074-3082.
- [13] Qi Y K, Zhang S P, Qin L, et al. Hedged deep 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: 4303-4311.
- [14] Qi Y K, Zhang S P, Qin L, et al. Hedging deep features for visual tracking[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019, 41(5): 1116-1130.
- [15] Xiong C Z, Che M Q, Wang R L, et al. Robust real-time visual tracking via dual model adaptive switching [J]. *Acta Optica Sinica*, 2018, 38(10): 1015002.
熊昌镇, 车满强, 王润玲, 等. 稳健的双模型自适应切换实时跟踪算法[J]. *光学学报*, 2018, 38(10): 1015002.
- [16] Jin H, Li X Y. Image saliency detection based residual network feature fusion tracking algorithm[J]. *Laser and Optoelectronics Progress*, 2020, 57(18): 181025.
金惠, 李新阳. 基于图像显著性检测的残差网络特征融合跟踪算法[J]. *激光与光电子学进展*, 2020, 57(18): 181025.
- [17] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[EB/OL]. (2015-04-10)[2020-08-07]. <https://arxiv.org/abs/1409.1556>.
- [18] Chaudhuri K, Freund Y, Hsu D. A parameter-free hedging algorithm[C]//Proceeding of the 23rd Annual Conference on Neural Information Processing Systems (NIPS), December 7-10, 2009, Vancouver, BC, Canada. New York: Curran Associates, 2009: 297-305.
- [19] Danelljan M, Häger G, Shahbaz Khan F, et al. Accurate scale estimation for robust visual tracking [C] // Proceedings of the British Machine Vision Conference 2014, September 1-5, 2014, Nottingham, UK. Guildford: BMVA Press, 2014.
- [20] Chen F L, Ding Q H, Chang Z, et al. Multi-scale kernel correlation filter algorithm for visual tracking based on the fusion of adaptive features [J]. *Acta Optica Sinica*, 2020, 40(3): 0315001.
陈法领, 丁庆海, 常铮, 等. 自适应特征融合的多尺度核相关滤波目标跟踪[J]. *光学学报*, 2020, 40(3): 0315001.
- [21] Bibi A, Ghanem B. Multi-templatescale-adaptive kernelized correlation filters[C]//2015 IEEE International Conference on Computer Vision Workshop (ICCVW), December 7-13, 2015, Santiago, Chile. New York: IEEE Press, 2015: 613-620.
- [22] 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.
- [23] Bertinetto L, Valmadre J, Henriques J F, et al. Fully-convolutional Siamese networks for object tracking[M]//Hua G, Jégou H. *Computer vision-ECCV 2016. Lecture notes in computer science*. Cham: Springer, 2016, 9914: 850-865.
- [24] Ning J F, Yang J M, Jiang S J, et al. Object tracking via dual linear structured SVM and explicit

- feature map[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 4266-4274.
- [25] Zhang J M, Ma S G, Sclaroff S. MEEM: robust tracking via multiple experts using entropy minimization [M]//Fleet D, Pajdla T, Schiele B, et al. Computer vision-ECCV 2014. Lecture notes in computer science. Cham: Springer, 2014, 8694: 188-203.
- [26] Gao J, Ling H B, Hu W M, et al. Transfer learning based visual tracking with Gaussian processes regression [M]//Fleet D, Pajdla T, Schiele B, et al. Computer vision-ECCV 2014. Lecture notes in computer science. Cham: Springer, 2014, 8691: 188-203.