

DOI:10.11918/202105112

# 一种无人机航拍目标的长期跟踪算法

黄 鹤<sup>1,2</sup>, 张 科<sup>1</sup>, 陈永安<sup>1</sup>, 王会峰<sup>1</sup>, 茹 锋<sup>1,2</sup>, 王 珺<sup>2</sup>

(1. 长安大学 电子与控制工程学院, 西安 710064; 2. 西安市智慧高速公路信息融合与控制重点实验室(长安大学), 西安 710064)

**摘 要:** 针对无人机在航拍目标跟踪的复杂场景过程中, 运动目标可能会被遮挡或不确定跟踪, 导致视觉模型出现逐渐损坏、漂移和不可逆转失败等问题, 提出了一种无人机航拍目标的长期跟踪算法。首先, 进行互补分类器多特征自适应融合设计, 在贝叶斯分类器中采用颜色直方图特征, 在相关滤波器中采用方向梯度直方图、灰度以及颜色名特征; 结合多种特征的优点, 构建目标鲁棒性的外观以适应复杂场景。然后, 在相关滤波器中加入自适应时空正则化项。在空间正则化参数中引入局部变化, 实现学习时限制像素可信度较低的滤波器; 而在时间正则化中, 根据全局响应自适应地调整滤波器的学习, 并用初始滤波器约束更新范围, 这样在缓解边界效应的同时, 有效防止滤波器退化。最后, 在上述基础上, 加入重检测模块, 使得跟踪过程更加准确。实验结果表明, 本文算法可以适应无人机航拍的复杂场景, 缓解边界效应和防止滤波器退化。与同类主流算法相比, 在目标经历严重遮挡、移出视野等情况后, 仍然满足实时性需要, 获得了较好的跟踪效果。

**关键词:** 长期跟踪; 相关滤波; 边界效应; 滤波器退化; 目标重检测

中图分类号: TP391.4

文献标志码: A

文章编号: 0367-6234(2022)05-0104-13

## A long-term tracking algorithm for UAV aerial photography

HUANG He<sup>1,2</sup>, ZHANG Ke<sup>1</sup>, CHEN Yongan<sup>1</sup>, WANG Huifeng<sup>1</sup>, RU Feng<sup>1,2</sup>, WANG Jun<sup>2</sup>

(1. School of Electronics and Control Engineering, Chang'an University, Xi'an 710064, China;

2. Xi'an Key Laboratory of Intelligent Expressway Information Fusion and Control (Chang'an University), Xi'an 710064, China)

**Abstract:** In view of the problem that the moving target may be occluded or tracked under uncertain conditions during the aerial target tracking of UAV in complex scenes, resulting in the gradual damage, drift, and irreversible failure of the visual model, a long-term tracking algorithm for UAV was proposed. Firstly, the complementary classifier was designed for multi-feature adaptive fusion. The color histogram feature was used in Bayesian classifier, and the directional gradient histogram, grayscale, and color name features were used in the correlation filter. In combination with the advantages of multiple features, the robust appearance of the target was built to adapt to the complex scenes. Secondly, an adaptive spatial-temporal regularization term was added to the correlation filter. Local changes were introduced into the spatial regularization parameters for the implementation of the filter with low pixel credibility during learning. In temporal regularization, the learning of the filter was adaptively adjusted according to the global response, and the initial filter was used to constrain the update range, which effectively prevented filter degradation while mitigating boundary effects. Finally, a re-detection module was added to make sure the accuracy of the tracking process. Experimental results show that the proposed algorithm could adapt to the complex scenes of UAV aerial photography, alleviate boundary effects, and prevent filter degradation. In comparison with similar mainstream algorithms, the proposed algorithm could still meet the real-time requirements and achieve better tracking effect even when the target experienced serious occlusion or moved out of view.

**Keywords:** long-term tracking; correlation filtering; boundary effects; filter degradation; target re-detection

随着无人机的迅速发展, 基于相关滤波器的航拍跟踪方法成为目前的研究热点<sup>[1]</sup>。文献[2]首次  
在目标跟踪中应用相关滤波, 并结合灰度特征提出

了MOSSE滤波器; 文献[3]通过引入循环矩阵, 提出了CSK算法, 解决了样本冗余问题; 在此基础上, 文献[4]结合多通道方向梯度直方图(histogram of oriented gradient, HOG)特征<sup>[5]</sup>提出了KCF算法; 文献[6]利用颜色名(color name, CN)属性提出了一种自适应颜色属性相关滤波器ACNCF。上述MOSSE、CSK、KCF和ACNCF算法都采用固定的跟踪窗口, 无法适应目标尺度的变化。为了解决该问题, 文献[7]提出了DSST算法, 添加了尺度估计的相关滤波器, 但额外训练增加了计算量; 文献[8]引入空间正则

收稿日期: 2021-05-23

基金项目: 国家自然科学基金面上项目(52172324); 国家重点研发计划(2018YFB1600600); 陕西省重点研发计划(2021GY-285, 2021SF483); 陕西省自然科学基金基础研究计划(2021JM-184); 西安市智慧高速公路信息融合与控制重点实验室(长安大学)开放基金(300102321502)

作者简介: 黄 鹤(1979—), 男, 教授, 博士生导师

通信作者: 黄 鹤, huanghe@chd.edu.cn

化提出了 SRDCF 算法,但采用的迭代方式求解速度很慢;文献[9]结合 HOG 与颜色直方图两种特征,提出了 Staple 互补分类器,但其跟踪结果易受复杂环境影响;文献[10]提出了 BACF 算法,对背景辨识能力较强,但目标严重遮挡时其跟踪精度低;文献[11]提出了 ECO 算法,跟踪耗时较多;为缓解滤波器退化的问题,文献[12]提出了 STRCF 算法,在目标状态突变时限制滤波器的更新速度,但目标遮挡时其跟踪表现不好;文献[13]提出了 LCT 跟踪算法,将在线随机森林分类器用于目标重检测,但其寻优精度和收敛速度仍需进一步提升。因此,本文提出了一种航拍目标的长期跟踪算法,从建立鲁棒性外观模型、缓解边界效应、防止滤波器退化、目标重检测 4 个角度改进传统算法,提升了计算精度和速度,解决了目标遮挡、移出视野等问题。

## 1 改进算法

### 1.1 多特征互补分类器自适应融合设计

在相关滤波器中,将 FHOG 特征、CN 特征和灰度特征进行特征融合,在贝叶斯分类器中则使用颜色直方图特征。二者组成互补分类器,利用多特征的优势构建鲁棒性的外观模型。改进算法流程见图 1。

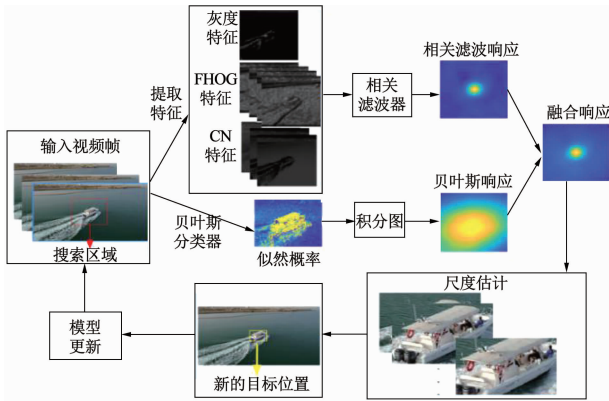


图 1 互补分类器多特征融合算法框图

Fig. 1 Block diagram of multi-feature fusion algorithm of complementary classifier

#### 1.1.1 贝叶斯分类器的设计

统计目标前景和背景颜色直方图,分别记为  $H_0$  和  $H_B$ ,根据贝叶斯后验概率预测图像块中每个像素点  $x$  的目标概率,计算方式如式(1)所示:

$$P(x \in \mathbf{O} | \mathbf{O}, \mathbf{B}, b_x) \approx \frac{P(b_x | x \in \mathbf{O})P(x \in \mathbf{O})}{\sum_{\Omega \in \{\mathbf{O}, \mathbf{B}\}} P(b_x | x \in \Omega)P(x \in \Omega)} \quad (1)$$

式中:  $\mathbf{O}$  和  $\mathbf{B}$  分别表示目标前景和背景区域,  $b_x$  代表某个像素点  $x$  的值位于直方图第  $b$  个区间中。

式(1)化简为

$$P(x \in \mathbf{O} | \mathbf{O}, \mathbf{B}, b_x) \approx \frac{H_0(b_x)}{H_0(b_x) + H_B(b_x)} \quad (2)$$

根据式(2)得到似然概率图,利用积分图计算似然概率图中与目标尺寸大小相关的矩形区域内所有像素之和,获得贝叶斯分类器的响应见图 2。

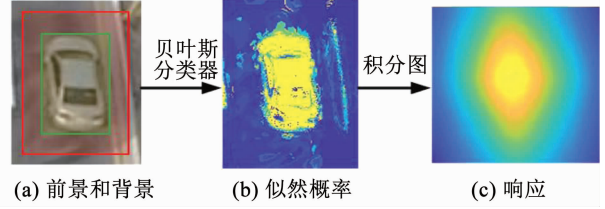


图 2 目标前景和背景、似然概率以及贝叶斯分类器响应

Fig. 2 Diagram of target foreground and background, likelihood probability, and Bayesian classifier response

积分图上  $(x, y)$  处的值  $S(x, y)$  为

$$S(x, y) = \sum_{x_1 \leq x, y_1 \leq y} L(x_1, x_2) \quad (3)$$

式中:  $L(x_1, x_2)$  为似然概率图位置  $(x_1, x_2)$  的概率值,  $S(x, y)$  为  $(x, y)$  左上角区域中概率值之和。通过式(4),可以快速计算任意矩形区域中所有概率值的和,也即响应图。

$$\mathbf{R}_{\text{hist}}(m, n) = S(x, y) + S(u, v) - S(x, v) - S(u, y) \quad (4)$$

式中:  $\mathbf{R}_{\text{hist}}$  为贝叶斯分类器响应,  $(x, y)$  为矩形右下角坐标,  $(u, v)$  为矩形左上角坐标,  $m$  和  $n$  分别为矩形长和宽。

颜色直方图在线更新为

$$\begin{cases} \mathbf{H}_0^t = (1 - \eta_{\text{hist}}) \mathbf{H}_0^{t-1} + \eta_{\text{hist}} \hat{\mathbf{H}}_0^t \\ \mathbf{H}_B^t = (1 - \eta_{\text{hist}}) \mathbf{H}_B^{t-1} + \eta_{\text{hist}} \hat{\mathbf{H}}_B^t \end{cases} \quad (5)$$

式中:  $\eta_{\text{hist}}$  为学习率,  $\mathbf{H}_0^t$ 、 $\mathbf{H}_B^t$  和  $\hat{\mathbf{H}}_0^t$ 、 $\hat{\mathbf{H}}_B^t$  分别表示第  $t$  帧更新后的和计算得到的目标前景、背景直方图。

#### 1.1.2 相关滤波器中的多特征融合设计

将一幅 RGB 彩色图像变换为单通道的灰度图像有多种形式,本文使用的形式为

$$G = R \times 0.299 + G \times 0.587 + B \times 0.114 \quad (6)$$

FHOG 特征提取的过程为:1) 建立像素级特征映射;2) 空间聚合,将梯度方向划分为 18 个方向敏感通道和 9 个方向不敏感通道;3) 归一化和截断。

根据 4 个相对邻域定义归一化因子:

$$F_{\delta, \gamma}(i, j) = (\| \mathbf{C}(i, j) \|^2 + \| \mathbf{C}(i + \delta, j + \gamma) \|^2 + \| \mathbf{C}(i, j + \gamma) \|^2 + \| \mathbf{C}(i + \delta, j) \|^2)^{1/2} \quad (7)$$

式中:  $\delta, \gamma \in \{i, j\}$ ;  $\mathbf{C}(i, j)$  为单元特征向量;  $E_{\beta}(\mathbf{v})$  表示向量  $\mathbf{v}$  用  $\beta$  截断,将向量  $\mathbf{v}$  中大于  $\beta$  的值设为  $\beta$ 。对 cell 特征归一化和截断后,串联起来得到新的特征:

$$\mathbf{H}(i, j) = \begin{pmatrix} E_{\beta}(\mathbf{C}(i, j)/F_{-1, -1}(i, j)) \\ E_{\beta}(\mathbf{C}(i, j)/F_{+1, -1}(i, j)) \\ E_{\beta}(\mathbf{C}(i, j)/F_{+1, +1}(i, j)) \\ E_{\beta}(\mathbf{C}(i, j)/F_{-1, +1}(i, j)) \end{pmatrix} \quad (8)$$

根据方向敏感和不敏通道,最终得到 31 维 FHO 特征<sup>[19]</sup>。利用 RGB 值和颜色名之间建立的字典映射表(32 768 × 10),可以快速提取 10 维 CN 特征<sup>[20]</sup>。将 FHO、CN 和 1 维灰度特征融合,得到一个 42 维的新特征,构建更加鲁棒的外观模型。在相关滤波框架下目标位置周围提取图像块,并融合 3 种特征,维数记为  $k \in \{1, \dots, K\}$ 。岭回归训练得到滤波器  $\mathbf{h}$  如下:

$$\arg\min_{\mathbf{h}} \left\{ \left\| \sum_{k=1}^K \mathbf{h}^k \otimes \boldsymbol{\phi}^k - \mathbf{y} \right\| + \lambda \sum_{k=1}^K \mathbf{h}^k \right\} \quad (9)$$

式中:  $\mathbf{h}^k$  和  $\boldsymbol{\phi}^k$  为  $k$  通道的滤波器和特征,  $\otimes$  为循环相关操作,  $\mathbf{y}$  为高斯期望输出, 参数  $\lambda$  为正则化项。对式(9)求解可得到滤波器  $\mathbf{h}$ 。直接求解无法解决边界效应和滤波器退化问题,故后续还需加入自适应时空正则化设计。

### 1.1.3 互补分类器自适应融合设计

贝叶斯分类器可以较好区分目标和背景,但对光照变化、目标相似物比较敏感。而多特征相关滤波对目标形状变化比较敏感。如果某个分类器受复杂环境影响导致跟踪结果较差,直接将响应图融合会影响最终跟踪效果,故互补分类器采用自适应融合。在输入视频序列中提取图像块作为搜索区域,对贝叶斯和相关滤波分类器独立训练;利用前者得到每个像素的得分,同时结合积分图计算贝叶斯分类器的响应;然后将后者作用于提取的 42 维特征,得到相关滤波响应;将两种响应按照图 1 融合。定义两种响应中心偏离距离  $M_d$ ,表示如下:

$$M_d = \sqrt{(u_{cf} - u_{hist})^2 + (v_{cf} - v_{hist})^2} \quad (10)$$

式中( $u_{cf}, v_{cf}$ )和( $u_{hist}, v_{hist}$ )分别为相关滤波器和贝叶斯分类器的响应图峰值位置。根据  $M_d$  可得到响应偏移,但无法估计跟踪可靠性。另外,贝叶斯鲁棒性不如相关滤波器,故根据贝叶斯响应去判断跟踪可靠性  $M_h$ ,定义为

$$M_h = \frac{N(\mathbf{R}_{hist} > \eta_h \max(\mathbf{R}_{hist}))}{S_m \times S_n} \quad (11)$$

式中: $N(\mathbf{R}_{hist} > \eta_h \max(\mathbf{R}_{hist}))$ 表示区域响应图中响应值大于  $\eta_h \max(\mathbf{R}_{hist})$  的像素数; $S_m \times S_n$ 为响应图尺寸;参数  $\eta_h$ 用于控制响应值的变化范围。定义第  $t$  帧的自适应融合状态  $S_A(t)$  如下所示( $\tau_d$ 和  $\tau_h$ 为阈值参数):

$$S_A(t) = \begin{cases} 1, & \text{if } M_d > \tau_d \text{ and } M_h > \tau_h \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

1)  $S_A(t)$  为 1 时,表示贝叶斯响应是可靠的,将其与相关滤波的响应图线性融合,得到融合响应图:

$$\mathbf{R} = (1 - \alpha)\mathbf{R}_{cf} + \alpha\mathbf{R}_{hist} \quad (13)$$

式中: $\mathbf{R}_{cf}$ 为相关滤波分类器响应, $\mathbf{R}_{hist}$ 为贝叶斯分类器响应, $\alpha$ 为常数融合系数。

2) 当  $S_A(t)$  为 0 时,表示响应不确定,采用自适应融合策略,调整式(8)为

$$\mathbf{R} = (1 - \alpha')\mathbf{R}_{cf} + \alpha'\mathbf{R}_{hist} \quad (14)$$

式中  $\alpha'$  表示贝叶斯分类器响应图在融合响应图中的占比。

## 1.2 自适应时空正则化

### 1.2.1 拟解决的两个问题

1) 边界效应问题。边界效应严重限制了跟踪性能,讨论如下:a) 训练阶段:循环位移产生大量样本,但只有一个是真实的,滤波器训练前样本集需要叠加余弦窗,消除样本边界的不连续性,减少不合理样本。但余弦窗将边缘区域像素全变成零,会滤掉需要学习的有限背景信息。b) 检测阶段:余弦窗限制了检测区域,简单扩充检测图像块会降低算法速度。如果目标移动到检测区域边缘,部分目标像素会被余弦窗过滤掉,造成跟踪结果变差;一旦部分目标移出检测区域,此时余弦窗会过滤掉仅存的目标像素,造成跟踪失败。

2) 滤波器退化问题。相关滤波跟踪器通常在每一帧中更新模型,当遇到目标遮挡、移出视野等复杂场景时,会造成滤波器的污染和退化,导致跟踪失败。为解决边界效应问题,当前多采用扩大搜索区域的方式,会引入大量背景信息供滤波器学习,遮挡、光照变化等也会加重滤波器的退化。

### 1.2.2 响应变化

定义局部响应变化向量  $\boldsymbol{\Pi} = [\|\boldsymbol{\Pi}^1\|, \|\boldsymbol{\Pi}^2\|, \dots, \|\boldsymbol{\Pi}^T\|]$ ,在目标边界框中的二维可视化,为空间正则化做准备,它的第  $i$  个元素  $\|\boldsymbol{\Pi}^i\|$  定义为

$$\|\boldsymbol{\Pi}^i\| = \frac{\mathbf{R}_i[\psi_{\Delta}]^i - \mathbf{R}_{i-1}^i}{\mathbf{R}_{i-1}^i} \quad (15)$$

式中: $\mathbf{R}_i[\psi_{\Delta}]$ 为平移算子,可以使两个响应图  $\mathbf{R}_i$  和  $\mathbf{R}_{i-1}$  中的两个峰重合,消除运动影响; $\mathbf{R}^i$  为响应图  $\mathbf{R}$  中第  $i$  个元素。定义全局响应变化变量  $\Pi_2$ ,表示两个响应图的变化,这里使用欧几里得范数定义,公式如下:

$$\Pi_2 = \|\mathbf{R}_i[\psi_{\Delta}] - \mathbf{R}_{i-1}\|_2^2 \quad (16)$$

$\Pi$  还表示跟踪框的局部可信度,光照变化或者部分遮挡的情况下都能使外观的可信度降低,滤波器在学习这些外观时应受到惩罚,避免受到干扰。过大的  $\Pi_2$  可能表示跟踪结果错误,应停止滤波器学习。相对较大的值则加速滤波器学习,提高目标对

环境变化的适应性。

### 1.2.3 模型自适应时空正则化

#### 1) 自适应空间正则化

像素可信度较低的滤波器在学习时应加以限制,这里通过空间正则化参数  $\tilde{\mathbf{u}}$  中引入局部变化  $\Pi$  实现:

$$\tilde{\mathbf{u}} = \mathbf{P}^T \delta \log(\Pi + 1) + \mathbf{u} \quad (17)$$

式中:  $\mathbf{P}^T \in \mathbb{R}^{T \times T}$  为用于裁剪对象所在滤波器的中心部分;  $\delta$  是一个常数,用于调整局部响应变化的权重;而  $\mathbf{u}$  用于缓解边界效应。由于空间惩罚在局部响应变化显著的像素位置,滤波器将在一定程度上避免学习新的外观。

#### 2) 自适应时间正则化

引入时间正则化,滤波器的变化率在损失函数中由固定参数  $\theta$  惩罚,但这并不通用。故尝试通过联合优化惩罚项和滤波器确定超参数值。因此,定义了一个参考值  $\tilde{\theta}$ ,根据全局响应优化目标函数:

$$\tilde{\theta} = \frac{\zeta}{1 + \log(\nu \Pi_2 + 1)}, \Pi_2 \leq \phi \quad (18)$$

式中  $\zeta$  和  $\nu$  表示超参数。当  $\Pi_2$  高于阈值  $\phi$  时,即响应图中存在畸变,滤波器停止学习;如果低于  $\phi$ ,响应图变化越剧烈,  $\tilde{\theta}$  越小,从而可以解除对滤波器时间变化的限制,且在外观变化较大时,滤波器可以更快地学习。

根据全局响应自适应调整滤波器的学习,可以减缓滤波器退化,但随着模型更新中的累积误差,还可能产生跟踪漂移。更新过程中唯一确认没有受污染的是初始滤波器。如果滤波器与初始滤波器变化过大,模型很可能完全退化了,需要改进。在学习过程中,应该考虑用初始滤波器去约束更新范围。根据最近  $N_r$  帧滤波器的平均值与初始滤波器的差值变化来惩罚滤波器的学习,为了优化求解方便,前  $N_r$  帧不根据初始滤波器对滤波器的学习进行惩罚,公式如下:

$$\bar{\mathbf{h}}_{N_r}^k = \begin{cases} \mathbf{h}_1^k, t < N_r \\ \frac{1}{N_r} \sum_{t=t-N_r+1}^t \mathbf{h}_t^k, t \geq N_r \end{cases} \quad (19)$$

式中:  $\bar{\mathbf{h}}_{N_r}^k$  为最近  $N_r$  帧滤波器的均值,  $t$  为当前帧数。联合空间正则化和时间正则化,提出一种新的目标函数为

$$\begin{aligned} \varepsilon(\mathbf{H}_t, \theta_t) = & \frac{1}{2} \|\mathbf{y} - \sum_{k=1}^K \mathbf{h}_t^k \otimes \boldsymbol{\phi}_t^k\|_2^2 + \\ & \frac{1}{2} \sum_{k=1}^K \|\tilde{\mathbf{u}} \odot \mathbf{h}_t^k\|_2^2 + \frac{\theta_t}{2} \sum_{k=1}^K \|\mathbf{h}_t^k - \mathbf{h}_{t-1}^k\|_2^2 + \\ & \frac{1}{2} \|\theta_t - \tilde{\theta}\|_2^2 + \frac{\theta_1}{2} \sum_{k=1}^K \|\bar{\mathbf{h}}_{N_r}^k - \mathbf{h}_1^k\|_2^2 \end{aligned} \quad (20)$$

式中:  $\tilde{\theta}$  为参考值,  $\theta_t$  为优化的时间正则化参数,  $\theta_1$  为固定的时间正则化参数,  $\tilde{\mathbf{u}}$  为式(17)计算得到的自适应空间正则化。时间正则化中,使用  $\|\mathbf{h}_t^k - \mathbf{h}_{t-1}^k\|_2$  约束滤波器的学习速度,使用  $\|\bar{\mathbf{h}}_{N_r}^k - \mathbf{h}_1^k\|_2$  限制滤波器的变化范围。

### 1.2.4 ADMM 优化求解

采用 ADMM 优化求解式(20),构建辅助变量  $\hat{\mathbf{g}}_t = \sqrt{T} \mathbf{F} \mathbf{h}_t$  ( $\hat{\mathbf{G}} = [\hat{\mathbf{g}}_t^1, \hat{\mathbf{g}}_t^2, \hat{\mathbf{g}}_t^3, \dots, \hat{\mathbf{g}}_t^K]$ )。  $\mathbf{F} \in \mathbb{C}^{T \times T}$  为正交矩阵,  $T$  为提取特征的长度,  $\hat{\cdot}$  表示离散傅里叶变换。频域中式(20)的增广拉格朗日形式表示为

$$\begin{aligned} L_t(\mathbf{H}_t, \theta_t, \hat{\mathbf{G}}_t, \hat{\mathbf{M}}_t) = & \varepsilon(\mathbf{H}_t, \theta_t, \hat{\mathbf{G}}_t) + \\ & \frac{\gamma}{2} \sum_{k=1}^K \|\hat{\mathbf{g}}_t^k - \sqrt{T} \mathbf{F} \mathbf{h}_t^k\|_2^2 + \\ & \sum_{k=1}^K (\hat{\mathbf{g}}_t^k - \sqrt{T} \mathbf{F} \mathbf{h}_t^k)^T \hat{\mathbf{m}}_t^k \end{aligned} \quad (21)$$

式中:  $\hat{\mathbf{M}}_t = [\hat{\mathbf{m}}_t^1, \hat{\mathbf{m}}_t^2, \dots, \hat{\mathbf{m}}_t^K] \in \mathbb{R}^{T \times K}$  为拉格朗日乘子傅里叶变换,  $\gamma$  为步长正则化参数。通过分配:

$$\mathbf{V}_t^k = \frac{\mathbf{m}_t^k}{\gamma} (\mathbf{V}_t^k = [V_t^1, V_t^2, \dots, V_t^K]) \quad (22)$$

式(21)可以变换为

$$\begin{aligned} L_t(\mathbf{H}_t, \theta_t, \hat{\mathbf{G}}_t, \hat{\mathbf{V}}_t) = & \varepsilon(\mathbf{H}_t, \theta_t, \hat{\mathbf{G}}_t) + \\ & \frac{\gamma}{2} \sum_{k=1}^K \|\hat{\mathbf{g}}_t^k - \sqrt{T} \mathbf{F} \mathbf{h}_t^k + \hat{\mathbf{v}}_t^k\|_2^2 \end{aligned} \quad (23)$$

利用 ADMM 求解以下子问题:

#### 1) 子问题 $\hat{\mathbf{G}}$ :

$$\begin{aligned} \hat{\mathbf{G}}^* = \operatorname{argmin}_{\hat{\mathbf{G}}} \{ & \frac{1}{2} \|\hat{\mathbf{y}} - \sum_{k=1}^K \hat{\mathbf{g}}_t^k \otimes \hat{\boldsymbol{\phi}}_t^k\|_2^2 + \\ & \frac{\theta_t}{2} \sum_{k=1}^K \|\hat{\mathbf{g}}_t^k - \hat{\mathbf{g}}_{t-1}^k\|_2^2 + \\ & \frac{\gamma}{2} \sum_{k=1}^K \|\hat{\mathbf{g}}_t^k - \sqrt{T} \mathbf{F} \mathbf{h}_t^k + \hat{\mathbf{v}}_t^k\|_2^2 \} \end{aligned} \quad (24)$$

式中\*为子问题的最优解。式(24)直接求解非常困难,因此,在每个像素的  $K$  个通道对  $\hat{\boldsymbol{\phi}}_t^k$  进行采样,结合谢尔曼-莫里森公式推导,得到其解为

$$\mathbf{I}_j^*(\hat{\mathbf{G}}_t) = \frac{1}{\gamma + \theta_t} (\mathbf{I} - \frac{\mathbf{I}_j(\hat{\boldsymbol{\phi}}_t) \mathbf{I}_j(\hat{\boldsymbol{\phi}}_t)^T}{\theta_t + \gamma + \mathbf{I}_j(\hat{\boldsymbol{\phi}}_t)^T \mathbf{v}_j(\hat{\boldsymbol{\phi}}_t)}) \boldsymbol{\rho} \quad (25)$$

式中  $\mathbf{I}_j(\hat{\boldsymbol{\phi}}) \in \mathbb{C}^{K \times 1}$  表示包含像素  $j$  ( $j = 1, 2, \dots, T$ ) 上  $\hat{\boldsymbol{\phi}}$  的所有  $K$  个通道的值的向量。

向量  $\boldsymbol{\rho}$  采用式(26)表示如下:

$$\boldsymbol{\rho} = \mathbf{I}_j(\hat{\boldsymbol{\phi}}) \hat{\mathbf{y}}_j + \theta_t \mathbf{I}_j(\hat{\mathbf{G}}_{t-1}) - \gamma \mathbf{I}_j(\hat{\mathbf{V}}_t) + \gamma \mathbf{I}_j(\sqrt{T} \mathbf{F} \mathbf{H}_t) \quad (26)$$

#### 2) 子问题 $\mathbf{h}$ :

$$\mathbf{h}^{k*} = \underset{\mathbf{h}^k}{\operatorname{argmin}} \left\{ \frac{1}{2} \|\tilde{\mathbf{u}} \odot \mathbf{h}_t^k\|_2^2 + \frac{\theta_1}{2} \sum_{k=1}^K \|\bar{\mathbf{h}}_{N_r}^k - \mathbf{h}_1^k\|_2^2 + \frac{\gamma}{2} \sum_{k=1}^K \|\hat{\mathbf{g}}_t^k - \sqrt{T} \mathbf{F} \mathbf{h}_t^k + \hat{\mathbf{v}}_t^k\|_2^2 \right\} \quad (27)$$

得到  $\mathbf{h}^k$  的解为

$$\mathbf{h}^{k*} = \begin{cases} [\tilde{\mathbf{U}}^T \tilde{\mathbf{U}} + \gamma T \mathbf{I}]^{-1} \gamma T (\mathbf{v}_t^k + \mathbf{g}_t^k) = \frac{\gamma T (\mathbf{v}_t^k + \mathbf{g}_t^k)}{(\tilde{\mathbf{u}} \odot \tilde{\mathbf{u}}) + \gamma T}, t \leq N_r \\ \frac{\gamma T (\mathbf{v}_t^k + \mathbf{g}_t^k) - \theta_1 \mathbf{h}_1^k / N_r + 1 / N_r^2 (\sum_{t=N_r}^{t-1} \mathbf{h}_t^k)}{(\tilde{\mathbf{u}} \odot \tilde{\mathbf{u}}) + \gamma T + \theta_1 / N_r^2}, t > N_r \end{cases} \quad (28)$$

式中  $\tilde{\mathbf{U}} = \operatorname{diag}(\tilde{\mathbf{u}}) \in \mathbb{R}^{T \times T}$  为对角矩阵。

3) 子问题  $\theta_t$ :

$$\theta_t^* = \underset{\theta_t}{\operatorname{argmin}} \left\{ \frac{\theta}{2} \sum_{k=1}^K \|\hat{\mathbf{g}}_t^k - \hat{\mathbf{g}}_{t-1}^k\|_2^2 + \frac{1}{2} \|\theta_t - \tilde{\theta}\|_2^2 \right\} = \tilde{\theta} - \frac{\sum_{k=1}^K \|\hat{\mathbf{g}}_t^k - \hat{\mathbf{g}}_{t-1}^k\|_2^2}{2} \quad (29)$$

4) 拉格朗日乘子更新: 在解决了上述 3 个子问题后, 可以将拉格朗日乘子更新为

$$\hat{\mathbf{V}}^{i+1} = \hat{\mathbf{V}}^i + \gamma^i (\hat{\mathbf{G}}^{i+1} - \hat{\mathbf{H}}^{i+1}) \quad (30)$$

式中  $i$  和  $i+1$  为迭代索引。步长正则化常数  $\gamma$  (初始等于 1) 的形式为

$$\gamma^{(i+1)} = \min(\gamma_{\max}, \beta \gamma^i), \beta = 10, \gamma_{\max} = 10\,000 \quad (31)$$

通过迭代求解上述 4 个子问题, 可以有效优化目标函数, 在  $t$  帧中得到最优滤波器  $\hat{\mathbf{G}}_t$  和时间正则化参数  $\theta_t$ , 然后在  $t+1$  帧中使用  $\hat{\mathbf{G}}_t$  进行检测。

$$\mathbf{R}_t = I_{\text{FT}} \left[ \sum_{k=1}^K (\hat{\mathbf{z}}_t^k \odot \hat{\mathbf{g}}_{t-1}^k) \right] \quad (32)$$

式中:  $\mathbf{R}_t$  为第  $t$  帧响应图,  $I_{\text{FT}}$  为傅里叶逆变换操作,  $\hat{\mathbf{z}}_t^k$  为在第  $t$  帧提取特征的傅里叶形式。

### 1.3 尺度估计

为解决目标尺寸变化问题<sup>[14]</sup>, 引入尺度相关滤波器。提取多个不同尺度样本, 建立尺度金字塔。设目标为  $P \times R$ , 提取  $s$  个大小为  $a^n P \times a^n R$  的训练样本,  $a$  为尺度系数,  $n \in \{[(s-1)/2], \dots, [(s-1)/2]\}$ , 所有样本调整到相同的一维尺寸。通过计算最大响应获得目标尺度。

### 1.4 模型自适应更新

传统相关滤波跟踪算法每一帧更新模型比较耗时, 需要采用稀疏更新来优化, 即目标变化足够大时再更新滤波器, 这样会带来不必要的计算。因此, 本文每  $N_s$  帧更新一次滤波器, 尺度模型仍每帧更新一次, 不受  $N_s$  影响。大量实验表明, 因为稀疏更新减少了对最近训练样本的过拟合, 模型适度不频繁

更新反而会提升跟踪效果。此外, 还有利于样本受到遮挡影响时, 模型学习到较少的干扰噪声。贝叶斯分类器中颜色直方图模型每帧都更新, 但缺少评估当前跟踪状态, 易学习到较多噪声。故引入一种自适应方式进行更新。当  $\Pi_2 > \phi$  时, 说明出现了光照变化、遮挡及移除视野等情况, 自适应融合状态  $S_A(t)$  认为颜色响应可能不可靠。此时仍快速更新, 会造成模型漂移, 因此需对学习率  $\eta_{\text{hist}}$  进行调整, 公式为

$$\eta_{\text{hist}} = \begin{cases} \varepsilon \times \eta'_{\text{hist}}, & \text{if } S_A(t) = 1 \text{ or } \Pi_2 > \phi \\ \eta'_{\text{hist}}, & \text{otherwise} \end{cases} \quad (33)$$

式中  $\eta'_{\text{hist}}$  为颜色直方图模型的初始学习率, 跟踪可靠时用来更新; 不可靠时, 用参数  $\varepsilon$  ( $0 < \varepsilon < 1$ ) 修正学习率。

### 1.5 算法流程

自适应时空正则化跟踪算法流程见图 3。

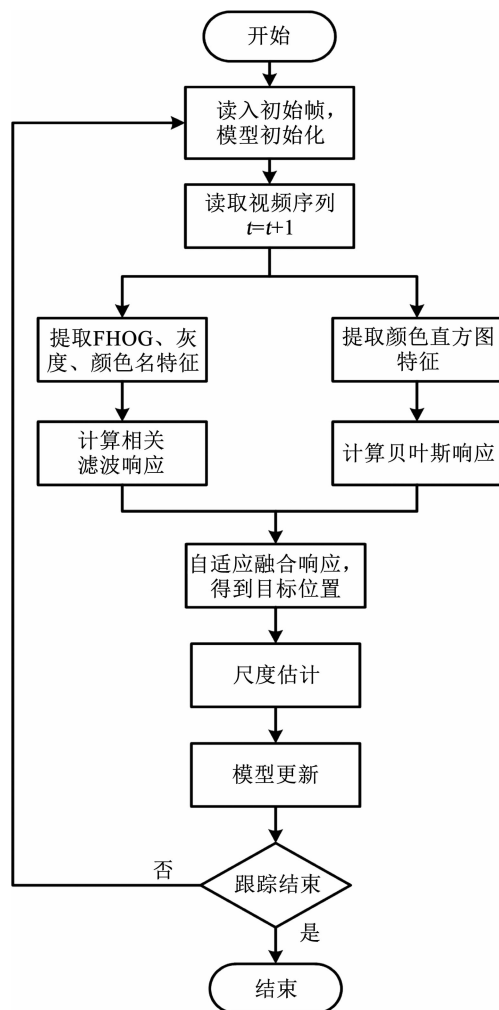


图 3 本文算法流程图

Fig. 3 Flow chart of proposed algorithm

## 2 重检测模块设计

加入重检测模块可以检测目标跟踪质量。当检测到跟踪不可靠时, 恢复丢失目标, 从而提高处理异

常情况的鲁棒性。但频繁调用检测器,会影响算法的速度。

## 2.1 跟踪不确定性检测

相关滤波器根据相关响应进行目标定位,为了防止滤波器退化,设计一种新的不确定性检测机制 $q_t$ ,通过峰值旁瓣比 $P_{SR}$ 和峰值绝对值定义,如式(34)所示:

$$q_t = P_{SR}(\mathbf{R}_t) \cdot \max(\mathbf{R}_t) \quad (34)$$

式中 $\mathbf{R}_t$ 为第 $t$ 帧的相关响应图。在跟踪不确定性检测中, $q_t$ 的相对值取决于目标外观的变化,是正常跟踪过程中不确定度的一个微弱指标。当在相对较短的时间尺度上出现遮挡时,定位质量将会显著降低。设 $\bar{q}_t$ 为最近 $N_q$ 连续帧上计算得到的平均定位质量,如果 $\bar{q}_t$ 和 $q_t$ 之间的比值超过预定阈值 $\tau_q$ ,则认为跟踪不确定,即

$$\bar{q}_t / q_t > \tau_q \quad (35)$$

出现遮挡时,比值显著提高,表明遮挡过程中跟踪高度不确定,见图4。

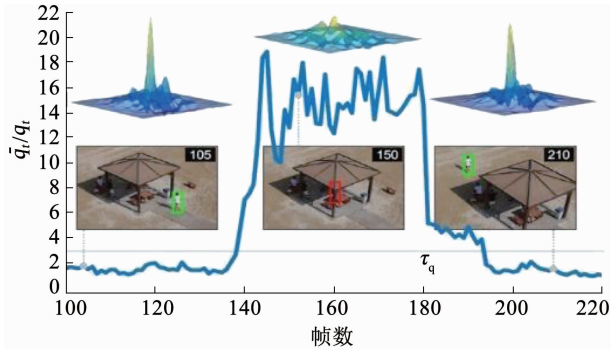


图4 平均跟踪质量与当前跟踪质量的比值变化

Fig. 4 Variation of ratio of average tracking quality to current tracking quality

另外,结合全局响应变量 $\Pi_2$ ,设计一种新的目标当前跟踪状态判断 $C_t$ ,如式(36)所示:

$$C_t = \begin{cases} 1, & \text{otherwise} \\ 0, & \text{if } \bar{q}_t / q_t > \tau_q \text{ or } \Pi_2 > \phi \end{cases} \quad (36)$$

式中: $\tau_q$ 为比值阈值, $\phi$ 为全局响应阈值。 $C_t$ 为1表示跟踪状态是可信的,将短期跟踪结果直接输出; $C_t$ 为0表示跟踪不确定,需进行目标恢复。

## 2.2 目标恢复

为实现目标恢复,需使用不被错误训练样本污染的视觉模型,即初始化时学习的滤波器。对于短期遮挡而言,可能产生更好检测效果。虽然长期跟踪可以减少短期模型污染,但却无法终止污染。因此,需要维护多个重检测滤波器 $\mathbf{H}^{\text{DE}} = \{\mathbf{h}_i^{\text{DE}}\}_{i=1, \dots, N^{\text{DE}}}$ ,

利用更新频率处理污染。上标DE表示变量属于本节的重检测滤波器。滤波器频繁更新可以学习最近的外观变化,能适应目标变化,但也增加了学习噪声的概率。而更新频率较低,遇到干扰较多时,可以增加鲁棒性学习,但也可能因为更新过慢,不能适应目标的快速变化,导致跟踪失败。因此,需维护多个重检测滤波器,其中一个从不更新(初始滤波器),保证出现类似于初始训练视图时,模型可以从污染中恢复。第 $i$ 个滤波器每 $n_i^{\text{DE}}$ 帧更新一次,类似于短期滤波器:

$$\mathbf{h}_i^{\text{DE}} = (1 - \eta) \mathbf{h}_{i-1}^{\text{DE}} + \eta \mathbf{h}_i^{\text{ST}} \quad (37)$$

式中: $\mathbf{h}_i^{\text{DE}}$ 为保存的第 $i$ 个检测滤波器, $\mathbf{h}_i^{\text{ST}}$ 为在第 $t$ 帧求出的短期跟踪滤波器,ST表示变量属于短期跟踪,即1.2节算法, $\eta$ 为学习率。

目标遮挡消失后重新出现的位置,较大可能就在消失点附近,因此添加了一个随机行走运动模型作为目标搜索区域大小增长方法,也即目标的运动先验,表示为

$$\pi(\mathbf{X}_t) = N(\mathbf{X}_t; \mathbf{X}_c, \Sigma_t) =$$

$$\exp[-(\mathbf{X}_t - \mathbf{X}_c)^T \Sigma_t^{-1} (\mathbf{X}_t - \mathbf{X}_c) / 2] \quad (38)$$

式中: $\pi(\mathbf{X}_t)$ 是一个高斯函数,对角协方差 $\Sigma_t = \text{diag}(\sigma_{xt}^2, \sigma_{yt}^2)$ , $\mathbf{X}_c$ 为最后一个可靠估计的位置中心。 $\Delta_t$ 为上次确认的跟踪评估与当前帧的差值绝对值。模型方差随 $\Delta_t$ 增加而逐渐增大,在遮挡时间内逐渐增加尺度搜索,例如 $[\sigma_{xt}^2, \sigma_{yt}^2] = [x_w, x_h] a_s^{\Delta_t}$ ,其中, $a_s$ 为尺度增长参数, $x_w$ 和 $x_h$ 分别为目标的长和宽。目标重检测时,从 $\mathbf{H}^{\text{DE}}$ 中挑选一个滤波器,对周围零填充,然后与整张图片提取的特征进行相关操作得到响应 $\mathbf{R}^{\text{DE}}$ ,与运动先验 $\pi(\mathbf{X}_t)$ 相乘得到最大响应位置作为待检测的估计位置 $\mathbf{X}_t^{\text{DE}}$ ,即

$$\mathbf{X}_t^{\text{DE}} = \max(\pi(\mathbf{X}_t) \odot \mathbf{R}^{\text{DE}}) \quad (39)$$

在 $\mathbf{X}_t^{\text{DE}}$ 周围提取特征后与之前存储的所有滤波器分别进行相关操作。为提升效率,每幅图像只评估单个滤波器,循环使用 $\mathbf{H}^{\text{DE}}$ 中所有滤波器和目标大小尺度 $\mathbf{S}^{\text{DE}}$ 在随后的图像序列中检测,直到找回目标。通过 $\mathbf{R}^{\text{DE}}$ 计算得到跟踪质量 $q_t^{\text{DE}}$ ,与短期组件的跟踪质量 $q_t^{\text{ST}}$ 比较,质量分数较高的位置为最终目标位置估计,公式如下:

$$(\mathbf{X}_t, q_t) = \begin{cases} (\mathbf{X}_t^{\text{ST}}, q_t^{\text{ST}}), & q_t^{\text{ST}} \geq q_t^{\text{DE}} \\ (\mathbf{X}_t^{\text{DE}}, q_t^{\text{DE}}), & \text{otherwise} \end{cases} \quad (40)$$

如果估计的位置是可靠的,短期组件和检测器更新,否则模型不更新。

## 2.3 算法流程

基于重检测的长期跟踪算法,具体流程如下。

### 步骤1:初始化

长期跟踪器在第1帧时初始化,训练得到初始模型  $\mathbf{h}_1^{\text{ST}}$  并保存。在剩余帧中,在不同的时间尺度上保存  $N^{\text{DE}}$  个视觉模型,用于目标定位和检测  $\mathbf{H}^{\text{DE}} = \{\mathbf{h}_i^{\text{DE}}\}_{i=1, \dots, N^{\text{DE}}}$ ,没有更新的模型等于初始化模型,如  $\mathbf{h}_1^{\text{DE}} = \mathbf{h}_1^{\text{ST}}$ 。

### 步骤2:计算相关滤波响应

在第  $t$  帧提取前一帧目标定位周围的一个区域,使用短期跟踪滤波器  $\mathbf{h}_{t-1}^{\text{ST}}$  计算相关响应  $\mathbf{R}_t^{\text{ST}}$ 。

### 步骤3:跟踪可靠性判断

根据相关响应  $\mathbf{R}_t^{\text{ST}}$  得到目标位置  $\mathbf{X}_t^{\text{ST}}$ ,并计算定位质量  $q_t^{\text{ST}}$  和全局响应变量,然后得到目标当前状态判断  $C_t$ 。当  $C_t$  为1时,表示跟踪状态是可信的,将短期跟踪结果直接输出;当  $C_t$  为0时,表示跟踪状态不可信,检测器也会被激活以处理潜在的目标丢失,执行以下步骤。

### 步骤4:得到相关响应

检测滤波器  $\mathbf{h}_i^{\text{DE}}$  从存储的检测器序列  $\mathbf{H}^{\text{DE}}$  选择出来与从整个图像中提取的特征做相关操作。

### 步骤5:目标定位

将相关响应  $\mathbf{R}^{\text{DE}}$  与运动先验  $\pi(\mathbf{X}_t)$  相乘,得到最大响应的位置作为待检测的估计位置  $\mathbf{X}_t^{\text{DE}}$ ,在  $\mathbf{X}_t^{\text{DE}}$  周围提取特征后与之前存储的所有滤波器分别进行相关操作,跟踪质量  $q_t^{\text{DE}}$  相关响应计算得到,然后根据由短期分量估计的位置和由检测器估计的位置。最终目标位置估计为质量分数较高的位置。

### 步骤6:判断终止条件

判断是否跟踪结束,跟踪未结束则返回步骤1,跟踪结束则完成所有视频序列的目标跟踪。

## 3 实验结果与分析

### 3.1 实验环境及评价参数

1) 实验环境为: MATLAB 2018a, CPU: I7-8700HQ, RAM: 16 GB, tracker\_benchmark\_v1.1 测评工具。

2) 实验采用 UAV123 数据集<sup>[18]</sup>,包含无人机在低空拍摄获取的123个视频序列,用于跟踪测试,其中包括20个长期跟踪序列,称为 UAV20L,用于测试长期跟踪性能。在所有序列中都使用矩形框对目标进行了标注。包括12种视觉属性标注,用于在不同场景下跟踪器的评估:相似目标(similar object, SOB)、相机运动(camera motion, CM)、快速运动(fast motion, FM)、光照变化(illumination variation, IV)、背景干扰(background clutter, BC)、长宽比变化(aspect ratio change, ARC)、部分遮挡(partial occlusion, POC)、完全遮挡(full occlusion, FOC)、移出视野(out-of-view, OV)、低分辨率(low resolution, LR)、尺度变换(scale variation, SV)和视点变化(viewpoint change, VC)。

### 3) 评价参数

a) 精确度指中心位置误差(center location error,  $C_{\text{LE}}$ )小于某一阈值的帧数与总帧数的比值,表

达式为

$$C_{\text{LE}} = \sqrt{(x_t - x_t^{\text{truth}})^2 + (y_t - y_t^{\text{truth}})^2} \quad (41)$$

式中:  $(x_t, y_t)$  为跟踪得到的目标中心,  $(x_t^{\text{truth}}, y_t^{\text{truth}})$  为人工标注目标中心。精确度图可以反映目标跟踪算法的定位目标中心的性能。本文设定阈值为20像素。

b) 成功率指重叠率(overlap rate,  $O_R$ )大于某阈值的帧数与总帧数的比值。 $O_R$  计算式为

$$O_R = \frac{|\mathbf{r}_t \cap \mathbf{r}_t^{\text{truth}}|}{|\mathbf{r}_t \cup \mathbf{r}_t^{\text{truth}}|} \quad (42)$$

式中:  $\mathbf{r}_t$  为跟踪框,  $\mathbf{r}_t^{\text{truth}}$  为人工标注框。成功率图显示的数字代表曲线下面积(area under the curve, AUC)。

### 3.2 自适应时空正则化实验

本文算法用 Our 表示,对比算法为 ECO\_HC<sup>[11]</sup>、STRCF<sup>[12]</sup>、BACF<sup>[10]</sup>、AutoTrack<sup>[15]</sup>、DSST<sup>[7]</sup>、fDSST<sup>[16]</sup>、Staple<sup>[9]</sup>、Staple\_CA<sup>[17]</sup>、LCT<sup>[13]</sup>、KCF<sup>[4]</sup>。

#### 3.2.1 实验参数

相关滤波分类器中,自适应时空正则化参数  $\delta = 0.2$ ,  $\zeta = 13$ ,  $\theta_1 = 0.1$ ,  $\nu = 2 \times 10^{-5}$ ,  $N_r = 100$ , 全局响应阈值  $\phi = 3\ 000$ , ADMM 迭代次数设为4,滤波器更新频率  $N_s = 3$ 。贝叶斯分类器中,颜色直方图划分为32个区间,学习率  $\eta_{\text{hist}} = 0.04$ ,参数  $\varepsilon = 0.01$ 。阈值  $\tau_d = 0.6$ ,  $\tau_h = 25$ ,  $\eta_h = 0.8$ ,融合系数  $a = 0.3$ ,自适应融合系数  $\alpha' = 0.1$ ,目标区域的尺寸被统一缩放到  $150 \times 150$ 。尺度估计中  $a = 1.02$ ,尺度  $s = 33$ 。

#### 3.2.2 定量分析

##### 1) 总体性能评估

本文算法的精确度和成功率分别为0.744和0.511,在比较的跟踪器中排名第一,见图1。实验表明本文算法精确度比 ECO\_HC、AutoTrack、Staple 和 STRCF 分别提升了2.8%、7.8%、11.7%和9.3%,成功率分别提升了4.5%、7.4%、13.6%和6.2%。

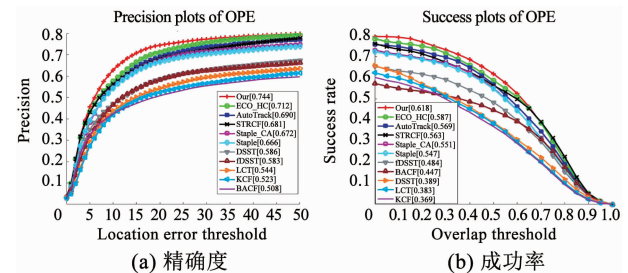


图5 11种算法在UAV123数据集上的结果

Fig. 5 Results of 11 algorithms on UAV123 dataset

本文算法运行速度为64.2帧/s,与其他几种算法相比,本文算法在成功率和精确度得到提升的同时,速度上仍满足实时性。算法速度对比结果见表1。

表 1 11 种算法速度对比

Tab. 1 Comparison of calculation speed of 11 algorithms (帧/s)

| Our  | ECO_HC | AutoTrack | STRCF | Staple_CA | Staple | fDSST | DSST | BACF | LCT  | KCF   |
|------|--------|-----------|-------|-----------|--------|-------|------|------|------|-------|
| 64.2 | 65.8   | 31.1      | 22.0  | 51.1      | 81.3   | 50.2  | 16.6 | 32.4 | 20.2 | 168.2 |

2) 属性性能分析

图 6 和表 2 为 11 种算法在 UAV123 数据集 12 种属性下的成功率。表 2 中加粗代表成功率排名第一, 可知本文算法的成功率在 SOB、CM、FM、IV、BC、ARC、POC、LR、SV 和 VC 10 种属性下排名第一; OV

情况下与 STRCF 并列第一; 在 FOC 情况下排名第二, 但与排名第一的 ECO\_HC 相差较小, 属于误差允许范围内。因为视频属性标注是直接对一整段视频标注, 而有的视频帧与标注属性并不符合, 因此只需在整体数据集和大部分属性上性能优异即可。

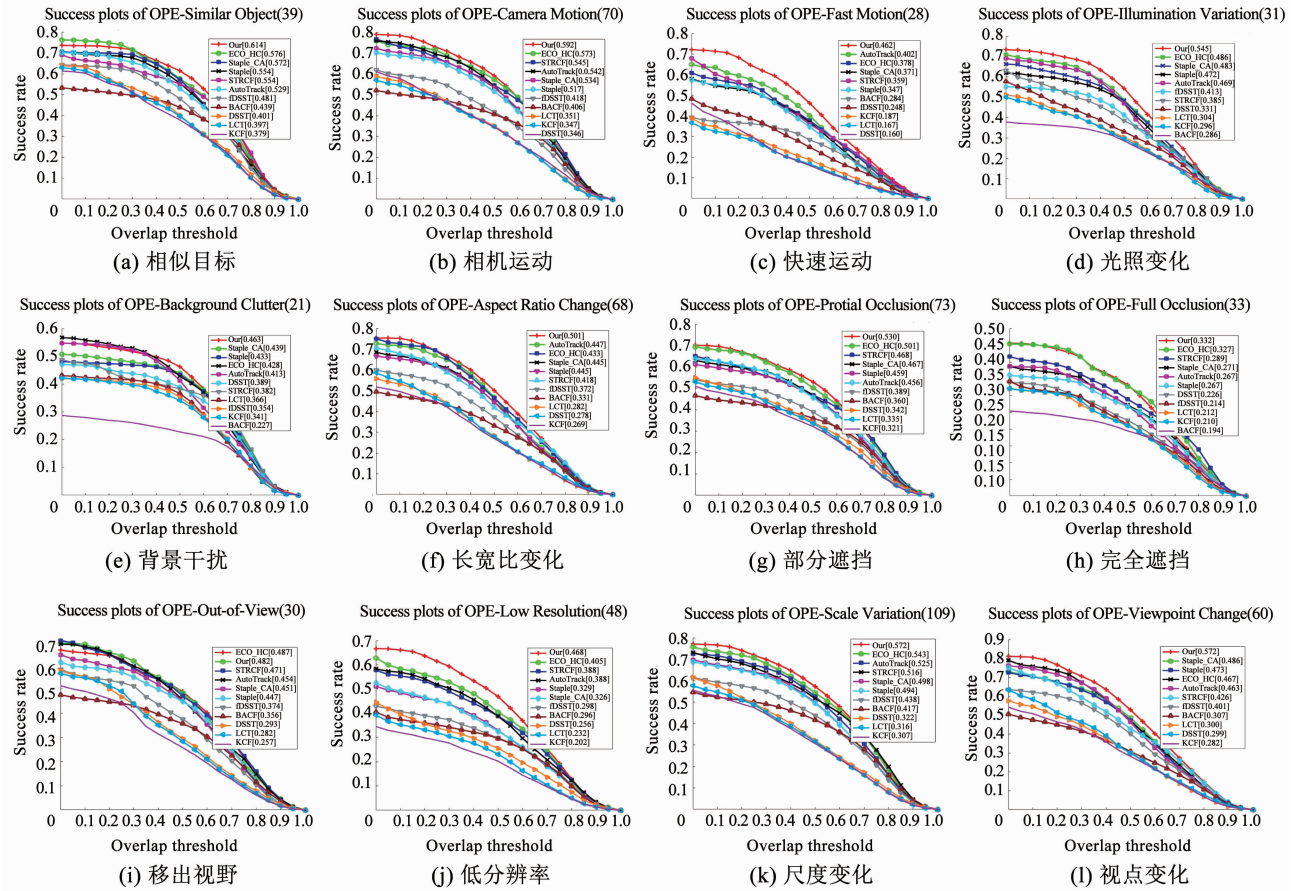


图 6 11 种算法在 UAV123 数据集 12 种属性下成功率的定性分析

Fig. 6 Qualitative analysis of success rate of 11 algorithms on UAV123 dataset under 12 attributes

表 2 11 种算法在 UAV123 数据集 12 种属性下成功率的定量分析

Tab. 2 Quantitative analysis of success rate of 11 algorithms on UAV123 dataset under 12 attributes

| 算法        | SOB          | CM           | FM           | IV           | BC           | ARC          | POC          | FOC          | OV           | LR           | SV           | VC           |
|-----------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Our       | <b>0.494</b> | <b>0.494</b> | <b>0.414</b> | <b>0.456</b> | <b>0.360</b> | <b>0.442</b> | <b>0.440</b> | 0.279        | <b>0.410</b> | <b>0.398</b> | <b>0.482</b> | <b>0.468</b> |
| ECO_HC    | 0.489        | 0.477        | 0.341        | 0.423        | 0.359        | 0.426        | 0.431        | <b>0.283</b> | 0.406        | 0.357        | 0.466        | 0.434        |
| AutoTrack | 0.447        | 0.465        | 0.363        | 0.404        | 0.345        | 0.415        | 0.392        | 0.233        | 0.401        | 0.338        | 0.444        | 0.424        |
| STRCF     | 0.451        | 0.470        | 0.347        | 0.354        | 0.318        | 0.398        | 0.402        | 0.258        | <b>0.410</b> | 0.337        | 0.448        | 0.406        |
| Staple_CA | 0.457        | 0.446        | 0.322        | 0.401        | 0.344        | 0.390        | 0.388        | 0.236        | 0.375        | 0.291        | 0.418        | 0.416        |
| Staple    | 0.450        | 0.435        | 0.318        | 0.384        | 0.334        | 0.386        | 0.380        | 0.226        | 0.369        | 0.290        | 0.414        | 0.410        |
| fDSST     | 0.407        | 0.369        | 0.244        | 0.347        | 0.277        | 0.339        | 0.329        | 0.194        | 0.329        | 0.258        | 0.373        | 0.363        |
| DSST      | 0.362        | 0.322        | 0.186        | 0.307        | 0.314        | 0.286        | 0.306        | 0.200        | 0.289        | 0.228        | 0.312        | 0.304        |
| BACF      | 0.359        | 0.340        | 0.242        | 0.241        | 0.192        | 0.294        | 0.299        | 0.166        | 0.297        | 0.244        | 0.348        | 0.282        |
| LCT       | 0.349        | 0.317        | 0.169        | 0.277        | 0.280        | 0.281        | 0.294        | 0.186        | 0.284        | 0.202        | 0.299        | 0.288        |
| KCF       | 0.342        | 0.310        | 0.184        | 0.270        | 0.272        | 0.267        | 0.282        | 0.185        | 0.256        | 0.180        | 0.291        | 0.274        |

由表 2 可以看出,在完全遮挡情况下,所有跟踪算法成功率都较低,还需要重检测设计。

3.2.3 定性分析

为直观分析跟踪性能,选取 UAV123 数据集中有代表性的视频序对本文与其他 10 种算法进行比较。完全遮挡情况下所有算法跟踪成功率都较低,故将遮挡属性单独定性分析(图 7),而将其他属性一起定性分析(图 8)。

1) 遮挡属性分析

a) 在序列 car9\_1 的 285 帧,目标受相似目标干扰,LCT 算法直接跟踪失败,在 853 帧目标受到部分遮挡后,只有 BACF 和本文算法能跟踪上目标。

b) 在序列 car14\_1 的 828 帧,目标到达图像边缘,经历了 POC、FOC 和 OV,在 930 帧进入视野。随着尺度变化,在 1 248 帧时,此时只有本文算法能够成功跟踪,其他所有算法,随着跟踪过程累积误差造成跟踪失败,并再也无法找回目标。

c) 在序列 group2\_2\_1 的 1 284 帧受背景干扰,BACF 跟踪失败;在 1 476 帧时,目标经历 POC 和 FOC,KCF、AutoTrack、LCT、Staple 跟踪失败;在 1 620 帧时,fDSST 跟踪失败,此时本文算法仍然能正常跟踪。

d) 在序列 person10\_1 的 191 帧目标经历了 POC、FOC 和 OV,在 280 帧时重新回到视野,此时所有算法全部跟踪失败。

e) 在序列 person14\_1\_1 的 151 帧目标经历了 POC 和 FOC,204 帧重新回到视野,此时所有算法跟踪失败。

f) 在序列 person19\_3\_1 的 3 072 帧,目标出现 POC 和 FOC,在 3 246 帧时重新回到视野,此时所有算法全部跟踪失败,并且都不能找回丢失的目标。

本文算法能在 car9\_1、car14\_1、group2\_2\_1 序列跟踪上目标。在 person10\_1、person14\_1\_1、person19\_3\_1 序列中,所有算法都跟踪失败,说明防止退化的方法有一定作用,但针对遮挡和离开视野问题仍需进一步改进。

2) 其他属性分析

a) 序列 boat9 中目标经历了 SV、ARC 和 LR。在 835 帧,KCF 和 LCT 随着模型累计误差,跟踪器完全退化;在 1 309 帧,目标尺度和长宽比变化较大,大多数跟踪器跟踪框与目标相差较多,说明已累积了较多误差。

b) 在序列 car16\_1\_1 的 246 帧中存在 FM 和 SV,大多跟踪器累积了较多误差,与真实目标框相差较多,在 385 帧 6 个跟踪器跟踪失败。

c) 序列 wakeboard5\_1 中目标经历了 POC、SV

和 FM。在 463 帧,Staple、BACF、fDSST、LCT、KCF 跟踪失败;在 792 帧 STRCF 跟踪失败;在 1062 帧 Staple\_CA 跟踪失败。只有本文算法、ECO\_HC、AutoTrack 和 DSST 跟踪成功,但前两种算法跟踪框更接近目标框。

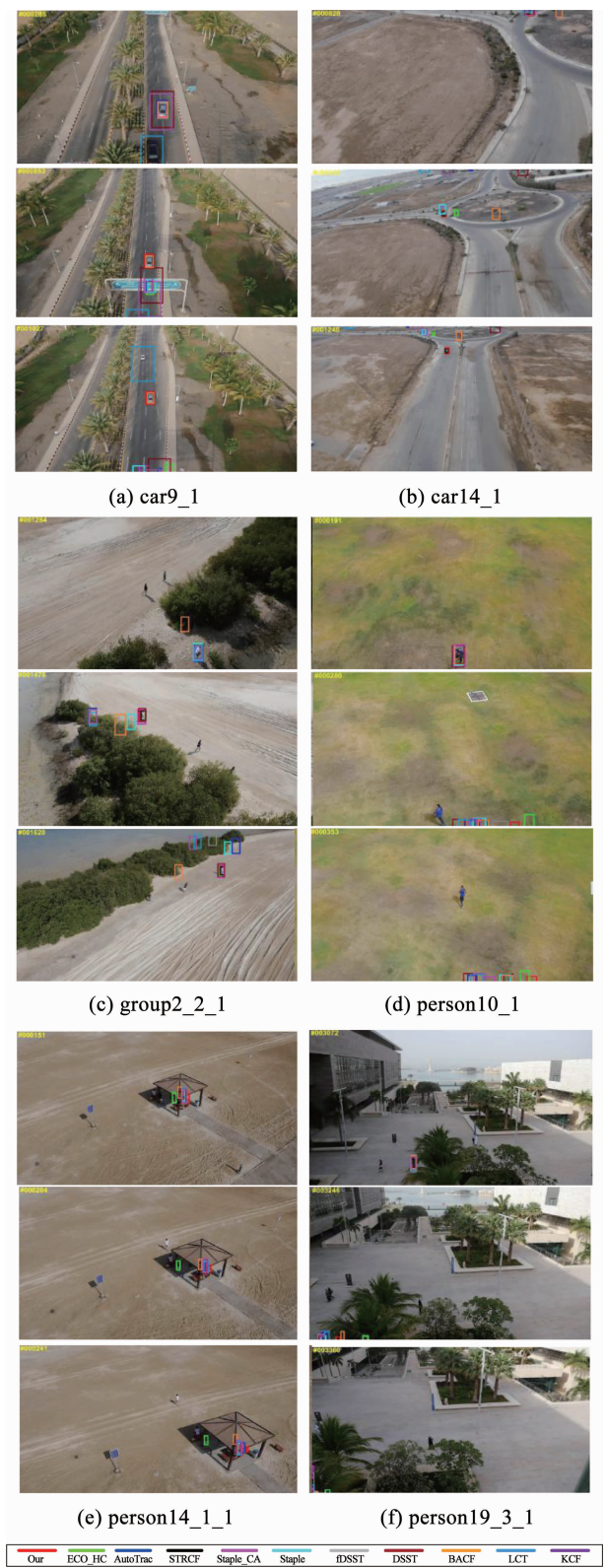


图 7 11 种算法在 6 组序列跟踪结果(遮挡属性)

Fig. 7 Tracking results of 11 algorithms on six sets of sequences (occlusion attribute)

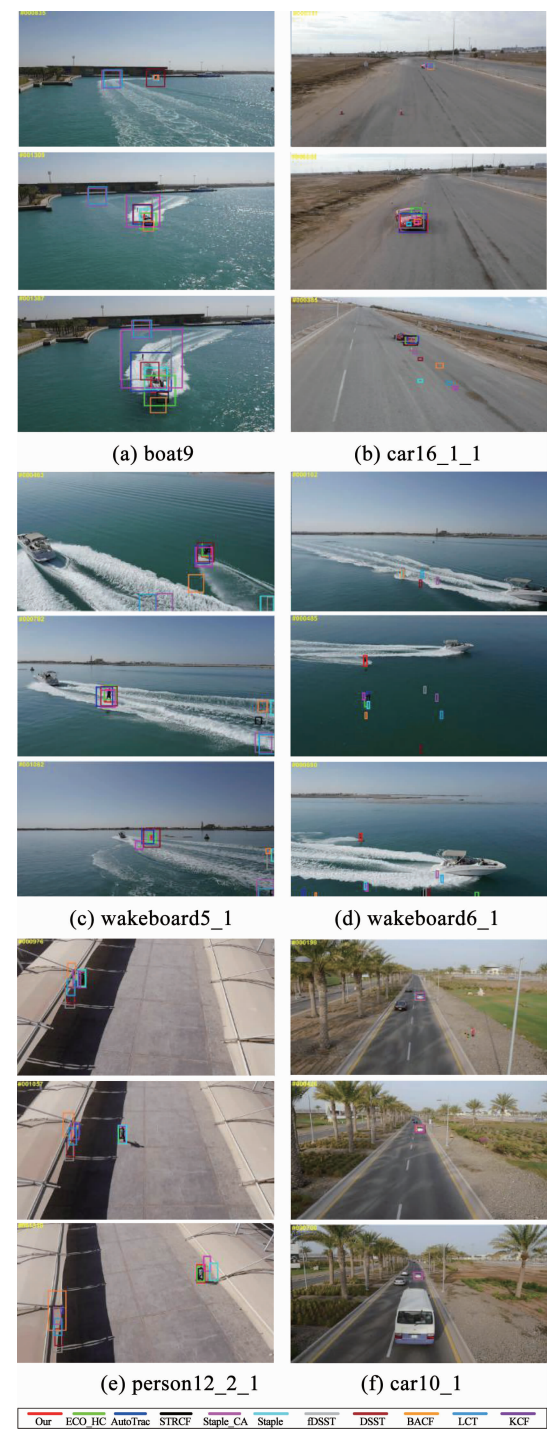


图 8 11 种算法在 6 组序列跟踪结果 (其他属性)

Fig. 8 Tracking results of 11 algorithms on six sets of sequences (other attributes)

d) 序列 wakeboard6\_1 中目标经历了 POC、SV 和 FM。在 102 帧, DSST、LCT、KCF 跟踪失败; 在 485 帧, 除了本文算法, 其他算法都跟踪失败, 并从 690 帧和 1 104 帧可以看出, 后续只有本文算法能正确跟踪目标。

e) 在序列 person12\_2\_1 的 102 帧, 目标受到 IV 影响, BACF、fDSST、KCF、LCT、DSST 跟踪失败, 在 1 057 帧目标由阴影进入阳光下, 此时 AutoTrack 也跟踪失败; 1 319 帧目标被遮挡, Staple\_CA 与 Staple

跟踪失败, 只有本文算法和 ECO\_HC 能成功跟踪目标。

f) 序列 car10\_1 中目标经历了 SV 和 SOB。从 199 帧、426 帧、700 帧可以看出, 环境变化相对简单时所有算法都能较好地跟踪目标。

### 3.3 重检测实验

#### 1) 实验参数

a) 参数设置: 短期跟踪器的参数与前文相同。重检测滤波器数量  $N^{DE} = 5$ , 其中每个重检测滤波器更新频率  $n_i^{DE} = \{0, 1/180, 1/90, 1/15, 1/3\}$ , 尺度因子为  $\{0.5, 0.7, 1, 1.2, 1.5, 2\}$ , 学习率  $\eta = 0.02$ , 随机运动模型尺度增长参数设置为  $a_s = 1.05$ , 不确定性阈值  $\tau_q = 2.7$ , 全局响应阈值  $\phi = 3\ 000$ , 最近帧  $N_q = 90$ 。

b) 对比算法为: ECO\_HC、STRCF、BACF、AutoTrack、DSST、fDSST、Staple、Staple\_CA、LCT、KCF, 短时跟踪算法用 Our 表示, 本节算法用 Our\_LT 表示。

#### 2) 定量分析

a) 总体性能评估。本文方法的精确度和成功率分别为 0.725 和 0.507, 在比较的跟踪器中排名第一, 见图 9。实验结果表明, 加入重检测模块后比 Our 的精确度和成功率分别提升了 18.5% 和 12.4%; 与 ECO\_HC、AutoTrack、Staple 和 STRCF 相比, Our\_LT 算法准确度分别提升了 29.5%、45.3%、53.3% 和 26.1%, 成功率分别提升了 20.7%、46.5%、46.5% 和 23.7%。同时, Our\_LT 速度达到 46.9 帧/s, 满足实时性。

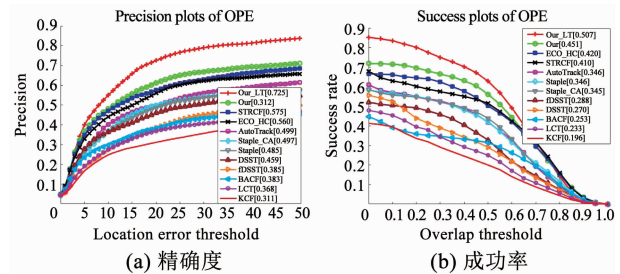


图 9 12 种算法在 UAV20L 数据集上的结果

Fig. 9 Results of 12 algorithms on UAV20L dataset

b) 属性性能分析。图 10 对比了 12 种算法的成功率。由图 10 可以看出, Our\_LT 在部分遮挡、完全遮挡和离开视野 3 个属性上, 比 Our 算法分别提升了 15.5%、75.6%、8.9%, 比 ECO\_HC 算法分别提升了 25.9%、87.8%、15.0%, 比 ECO\_HC 算法分别提升了 24.7%、99.1%、20.8%。这说明加入重检测后, 跟踪成功率得到了很大提升。部分遮挡和离开视野时, 目标仍有一部分在图像中, 故部分鲁棒性

较强的短期算法还可以找到目标。在目标完全遮挡时,几乎所有的短期跟踪算法都失效。因此,在完全遮挡属性下,Our\_LT 得到了显著提升。实验表明重检测能够有效解决目标完全遮挡的问题。

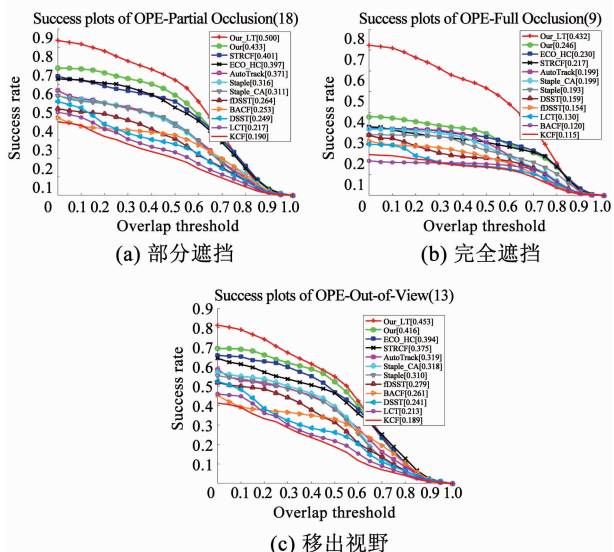


图 10 12 种算法在 3 种属性下成功率

Fig. 10 Success rate of 12 algorithms under three attributes

### 3) 定性分析

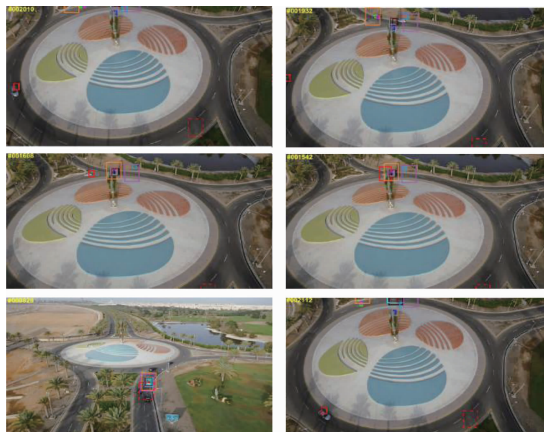
为了进一步验证跟踪性能,在 UAV20L 数据集中选取 4 组具有代表性的视频序列(carl、group3、person14、person19),包含部分遮挡、全部遮挡、离开视野、相似目标等各种复杂变化,来定性验证改进算法的跟踪性能,见图 11。Our\_LT 与其他 11 种跟踪算法相比,取得了更好的跟踪效果。

a) 在序列 carl 的 828 帧,受相似物体干扰,KCF 跟踪失败,随着模型更新,跟踪器完全退化。LCT 和 BACF 模型在跟踪过程中不断的累积误差,在 1 542 帧跟踪器也完全退化。目标在 1 542 帧完全遮挡后,在 1 608 帧时目标回到视野,只有 Our\_LT 算法能成功跟踪,其他算法都跟踪失败。目标在 1 932 帧到达图像边缘,并在后续帧离开视野,在 2 010 帧进入视野,此时仍然只有 Our\_LT 算法能够成功跟踪,其他算法随着累积误差造成跟踪失败,再也无法找回目标。可以看出,Our\_LT 算法能够在部分遮挡、完全遮挡、离开视野情况下稳定跟踪,并在目标丢失后重新找回目标。

b) 在序列 group3 的 1 967 帧,目标受到背景干扰和部分遮挡,只有 Our\_LT 算法能正确跟踪目标,其他算法都跟踪失败;在 2 250 帧时,目标受到完全遮挡,在 2 346 帧再次出现,在 3 978、4 825 帧目标受到部分遮挡,Our\_LT 算法均能够跟踪目标。可以看出,Our\_LT 算法能够在部分遮挡、完全遮挡时稳定跟踪目标。

c) 序列 person14 中的目标经历了相似物体干扰、部分遮挡、完全遮挡和离开视野。目标在 133 帧受到完全遮挡,在 199 帧再次出现,只有 Our\_LT 能够正确跟踪目标,其他算法都失败。目标在 2 040 帧离开视野,在 2 221 帧重新进入视野,Our\_LT 仍可以找回目标并跟踪。目标在 2 833 帧部分遮挡,但 Our\_LT 仍能正确跟踪目标。可以看出,Our\_LT 算法可以在部分遮挡、完全遮挡、离开视野时稳定跟踪,并在目标丢失后重新找回。

d) 序列 person19 中的目标经历了部分遮挡、完全遮挡和离开视野。在 1 002 帧时,目标离开视野,在 1 180 帧重新进入视野,此时只有 Our\_LT、Our 和 LCT 能跟踪上目标。Our 和 LCT 算法跟踪准确的原因是目标从离开视野到重新进入,两个滤波器还未完全退化。之后目标离开视野,1 988 帧再次进入视野时,可以看到 Our\_LT 开始并没有找到目标,在 2 041 帧重新找回目标,此时只有 Our\_LT 和 Our 算法可以跟踪目标,说明防止滤波器退化在一定情况下还可以处理移除视野的问题。但在 3 078 帧目标完全遮挡,Our 算法跟踪失败。在 3 400 帧时目标又被部分遮挡,此时只有 Our\_LT 算法能正确跟踪目标。可以看出,Our\_LT 算法能在部分遮挡、完全遮挡、离开视野时稳定跟踪,且目标丢失后能重新找回。



(a) carl



(b) group3

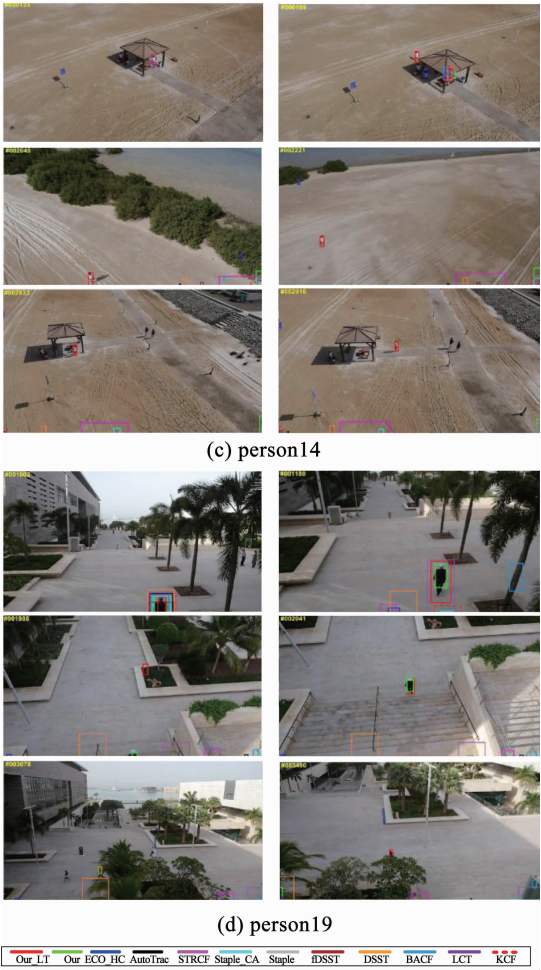


图 11 11 种算法在 4 组序列上的跟踪结果

Fig. 11 Tracking results of 11 algorithms on four sets of sequences

4 结 论

本文提出了一种无人机航拍目标的长期跟踪算法,并从 4 个方面进行改进。实验结果表明,本文算法在目标出现严重遮挡、移出视野等情况时,获得了较好的跟踪效果,且仍满足实时性。但在实际应用中,长期跟踪面临的复杂情况还有很多,因此,下一步还需对重检测模块进行更深入的研究。

参考文献

[1] HUANG He, SONG Jing, GUO Lu, et al. Haze removal method based on a variation function and colour attenuation prior for UAV remote-sensing images [ J ]. Journal of Modern Optics, 2019, 66 ( 12 ): 1282. DOI: 10.1080/09500340.2019.1615141

[2] BOLME D S, BEVERIDGE J R, DRAPER B A, et al. Visual object tracking using adaptive correlation filters[ C ]// IEEE Conference on Computer Vision and Pattern Recognition. San Francisco: IEEE, 2010: 2544. DOI: 10.1109/CVPR.2010.5539960

[3] HENRIQUES J F, CASEIRO R, MARTINS P, et al. Exploiting the circulant structure of tracking-by-detection with kernels [ C ]// European Conference on Computer Vision. Berlin: Springer, 2012: 702. DOI: 10.1007/978-3-642-33765-9\_50

[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, 2014, 37 ( 3 ): 583. DOI: 10.1109/TPAMI.2014.2345390

[5] DALAL N, TRIGGS B. Histograms of oriented gradients for human detection[ C ]//IEEE Conference on Computer Vision and Pattern Recognition. San Diego: IEEE, 2005: 886. DOI: 10.1109/CVPR.2005.177

[6] DANELLJAN M, KHAN F S, FELSBERG M, et al. Adaptive color attributes for real-time visual tracking [ C ]//IEEE Conference on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014: 1090. DOI:10.1109/CVPR.2014.143

[7] DANELLJAN M, HAGER G, KHAN F S, et al. Accurate scale estimation for robust visual tracking [ C ]//British Machine Vision Conference. Nottingham: BMVA Press, 2014: 1. DOI:10.5244/C.28.65

[8] DANELLJAN M, HAGER G, KHAN F S, et al. Learning spatially regularized correlation filters for visual tracking [ C ]//IEEE Conference on Computer Vision and Pattern Recognition. Santiago: IEEE, 2015: 4310. DOI:10.1109/ICCV.2015.490

[9] BERTINETTO L, VALMADRE J, GOLODETS S, et al. Staple: Complementary learners for real-time tracking [ C ]//IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 1401. DOI:10.1109/CVPR.2016.156

[10] GALOOGAHI H K, FAGG A, LUCEY S. Learning background-aware correlation filters for visual tracking[ C ]//IEEE Conference on Computer Vision and Pattern Recognition. Venice: IEEE, 2017: 1135. DOI:10.1109/ICCV.2017.129

[11] DANELLJAN M, BHAT G, KHAN F S, et al. ECO: Efficient convolution operators for tracking [ C ]//IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 6638. DOI: 10.1109/CVPR.2017.733

[12] LI Feng, TIAN Cheng, ZUO Wangmeng, et al. Learning spatial-temporal regularized correlation filters for visual tracking [ C ]// IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 4904. DOI: 10.1109/CVPR.2018.00515

[13] MA Chao, YANG Xiaokang, ZHANG Chongyang, et al. Long-term correlation tracking[ C ]//IEEE Conference on Computer Vision and Pattern Recognition. Boston: IEEE, 2015: 5388. DOI:10.1109/CVPR.2015.7299177

[14] DANELLJAN M, HAGER G, KHAN F S, et al. Accurate scale estimation for robust visual tracking [ C ]//British Machine Vision Conference. Nottingham: BMVA Press, 2014: 1. DOI:10.5244/C.28.65

[15] LI Yiming, FU Changhong, DING Fangqiang, et al. AutoTrack: Towards high-performance visual tracking for UAV with automatic spatio-temporal regularization [ C ]//IEEE Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 11920. DOI: 10.1109/CVPR42600.2020.01194

[16] DANELLJAN M, HAGER G, KHAN F S, et al. Discriminative scale space tracking[ J ]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 39 ( 8 ): 1561. DOI: 10.1109/TPAMI.2014.2345390

[17] MUELLER M, SMITH N, GHANEM B. Context-aware correlation filter tracking [ C ]//IEEE Conference on Computer Vision and

Pattern Recognition. Honolulu: IEEE, 2017; 1396. DOI:10.1109/CVPR.2017.152

[18] MUELLER M, SMITH N, GHANEM B. A benchmark and simulator for UAV tracking[C]//European Conference on Computer Vision. Cham; Springer, 2016; 445. DOI: 10.1007/978-3-319-46448-0\_27

[19] FELZENSZWALB P F, GIRSHICK R B, MCALLESTER D, et al. Object detection with discriminatively trained part-based models[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 32(9): 1627. DOI: 10.1109/TPAMI.2009.167

[20] POSSEGGER H, MAUTHNER T, BISCHOF H. In defense of color-based model-free tracking[C]//IEEE Conference on Computer Vision and Pattern Recognition. Boston; IEEE, 2015; 2113. DOI: 10.1109/CVPR.2015.7298823

(编辑 苗秀芝)

(下接第 145 页)

[13] HAO Cong, DOTZEL J, XIONG Jinjun, et al. Enabling design methodologies and future trends for Edge AI: specialization and co-design[J]. IEEE Design & Test, 2021, 38(4):7. DOI:10.1109/MDAT.2021.3069952

[14] 徐欣, 刘强, 王少军. 一种高度并行的卷积神经网络加速器设计方法[J]. 哈尔滨工业大学学报, 2020, 52(4): 31  
XU Xin, LIU Qiang, WANG Shaojun. A highly parallel design method of convolutional neural network accelerator[J]. Journal of Harbin Institute of Technology, 2020, 52(4): 31. DOI: 10.11918/201812159

[15] WANG Siqi, ANANTHANARAYANAN G, ZENG Yifan, et al. High-throughput CNN inference on embedded ARM Big. LITTLE multicore processors[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2019, 39(10): 2254. DOI:10.1109/TCAD.2019.2944584

[16] SHI Yuanming, YANG Kai, JIANG Tao, et al. Communication-efficient Edge AI: algorithms and systems [J]. IEEE Communications Surveys & Tutorials, 2020, 22(4): 2167. DOI: 10.1109/COMST.2020.3007787

[17] PETERSEN P, VOIGTLAENDER F. Optimal approximation of piecewise smooth functions using deep ReLU neural networks[J]. Neural Networks, 2018, 108: 296. DOI: 10.1016/j.neunet.2018.08.019

[18] LIU W, ANGUELOV D, ERHAN D, et al. SSD: Single shot MultiBox detector [C]//European conference on computer vision. Cham; Springer, 2016; 21. DOI:10.1007/978-3-319-46448-0\_2

[19] LANGER S. Approximating smooth functions by deep neural networks with sigmoid activation function [J]. Journal of Multivariate Analysis, 2021, 182: 104696. DOI: 10.1016/J.JMVA.2020.104696

(编辑 苗秀芝)