

基于多模型的实时压缩跟踪算法

张建明* 金晓康 吴宏林 伍悠

(长沙理工大学综合交通运输大数据智能处理湖南省重点实验室 长沙 410114)

(长沙理工大学计算机与通信工程学院 长沙 410114)

摘要: 目标跟踪易受光照、遮挡、尺度、背景及快速运动等因素的影响,还要求较高的实时性。目标跟踪中基于压缩感知的跟踪算法实时性好,但目标外观变化较大时跟踪效果不理想。该文基于压缩感知的框架提出多模型的实时压缩跟踪算法(MMCT),采用压缩感知来降低跟踪过程产生的高维特征,保证实时性能;通过判断前两帧的分类器最大分类分数的差值来选择最合适的模型,提高了定位的准确性;提出新的模型更新策略,按照决策分类器的不同采用固定或动态的学习率,提高了分类精度。MMCT引入的多模型没有增加计算负担,表现出优异的实时性能。实验结果表明,MMCT算法能够很好地适应光照、遮挡、复杂背景及平面旋转的情况。

关键词: 目标跟踪; 压缩感知; 实时; 多模型; 动态学习率

中图分类号: TP391

文献标识码: A

文章编号: 1009-5896(2018)10-2373-08

DOI: 10.11999/JEIT171128

Multi-model Real-time Compressive Tracking

ZHANG Jianming JIN Xiaokang WU Honglin WU You

(Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation,
Changsha University of Science and Technology, Changsha 410114, China)

(School of Computer and Communication Engineering, Changsha University of Science and
Technology, Changsha 410114, China)

Abstract: Object tracking is easily influenced by illumination, occlusion, scale, background clutter, and fast motion, and it requires higher real-time performance. The object tracking algorithm based on compressive sensing has a better real-time performance but performs weakly in tracking when object appearance is changed greatly. Based on the framework of compressive sensing, a Multi-Model real-time Compressive Tracking (MMCT) algorithm is proposed, which adopts the compressive sensing to decrease the high dimensional features for the tracking process and to satisfy the real-time performance. The MMCT algorithm selects the most suitable classifier by judging the maximum classification score difference of classifiers in the previous two frames, and enhances the accuracy of location. The MMCT algorithm also presents a new model update strategy, which employs the fixed or dynamic learning rates according to the differences of decision classifiers and improves the precision of classification. The multi-model introduced by MMCT does not increase the computational burden and shows an excellent real-time performance. The experimental results indicate that the MMCT algorithm can well adapt to illumination, occlusion, background clutter and plane-rotation.

Key words: Object tracking; Compressive Sensing (CS); Real-time; Multi-model; Dynamic learning rate

1 引言

目标跟踪是计算机视觉领域中的一个基本问

题,具有广阔的应用前景^[1]。目标跟踪存在着诸多无法预料的因素干扰着目标的观测,例如光照的变化、遮挡、姿态与视角的变化、相似物体及所处的复杂背景的干扰等^[2]。根据对目标建模的不同,目标跟踪算法主要分为两大类:生成式方法与判别式方法。

生成式跟踪算法主要是对前景进行表观建模,通过最小重构误差搜索候选区域,找到最佳匹配位置,利用在线学习机制更新目标模型。Collins等人^[3]在Mean-Shift方法中引入可导的尺度空间滤波器实

收稿日期: 2017-11-30; 改回日期: 2018-07-16; 网络出版: 2018-07-25

*通信作者: 张建明 jmzhang@csust.edu.cn

基金项目: 国家自然科学基金(61402053, 61772454, 61811530332),
湖南省教育厅科学研究重点项目(16A008)

Foundation Items: The National Natural Science Foundation of China (61402053, 61772454, 61811530332), The Scientific Research Fund of Hunan Provincial Education Department (16A008)

现了目标跟踪的尺度自适应(SMS); Adam等人^[4]提出一种基于局部块的目标表观建模方法(Fragment-based, Frag), 将目标划分为非重叠的子块, 通过融合不同子块的模板直方图确定跟踪目标; Oron等人^[5]提出了局部无序跟踪算法(Locally Orderless Tracking, LOT), 采用局部子块的无序匹配策略, 匹配最为相似的子块, 提升跟踪精度。基于生成式的跟踪方法根据提取的目标特征与模型相匹配实现目标的跟踪, 忽略了背景信息在场景中的作用, 在目标自身变化剧烈或者被遮挡时容易产生漂移。

相比于生成式跟踪方法, 判别式跟踪方法能够最大化的分离目标和背景, 将跟踪问题转化为二分类的问题。其核心思想是如何更新判别式分类器从而减少跟踪的漂移, 包括多实例学习跟踪算法(Multiple Instance Learning, MIL)^[6]、在线AdaBoost跟踪算法(On-line AdaBoost, OAB)^[7]、跟踪-学习-检测的跟踪算法(Tracking-Learning-Detection, TLD)^[8]、压缩跟踪算法(Compressive Tracking, CT)^[9]、混合决策压缩跟踪算法(Mixed Decision Compressive Tracking, MDCT)^[10]等。当前流行的深度学习的方法属于判别式方法。Held等人^[11]提出了将回归网络应用于通用目标跟踪, 达到实时的性能, 但其精度需要进一步提高; Nam等人^[12]提出了多域网络跟踪算法进行端到端的学习, 大幅提升了跟踪的性能; Song等人^[13]将卷积残差网络应用于跟踪, 利用单层的卷积操作模拟相关滤波操作, 提升了跟踪的精度。目标跟踪仅提供第1帧的信息, 而深度学习往往需要大量的标注数据进行训练, 且在使用GPU加速的情况下实时的性能依然不尽理想, 因此使得深度学习在目标跟踪领域的应用受到一定的限制。

尽管上述提出了诸多优秀的算法, 但到目前为止, 仍然没有单一的跟踪算法能同时解决跟踪中各项因素的干扰, 只是在某些方面具有较好的鲁棒性。针对CT算法的实时性能好但适应外观变化的能力较弱, 以及MDCT对遮挡、复杂背景的鲁棒性好但模型更新过快导致信息丢失较多的优缺点, 提出了多模型的实时压缩跟踪算法。通过判断前两帧的最大分类分数的差值, 在多个分类器模型中选择最合适的模型, 并提出了新的模型更新策略及使用不同的学习率对多模型进行更新。在Wu等人^[14]提出的目标跟踪Benchmark的实验表明, MMCT算法提高了在遮挡、光照变化以及复杂背景中跟踪的鲁棒性, 且具有较好的实时性。

2 压缩跟踪算法

2.1 原始压缩跟踪算法

CT算法^[9]利用压缩感知理论将高维的图像特征

进行压缩, 大幅降低了多尺度特征的维度, 减少了跟踪过程中的数据量, 实现目标有效且实时的跟踪。 t 帧时, 提取正、负样本的多尺度特征, 通过一个符合RIP(Restricted Isometry Property)原则的稀疏测量矩阵 $\mathbf{R} \in \mathbb{R}^{n \times m}$ ($n \ll m$, n 和 m 分别为 \mathbf{R} 的行数和列数), 将高维特征 $\mathbf{x} \in \mathbb{R}^m$ 投影到低维空间得到特征 \mathbf{v} , 如式(1):

$$\mathbf{v} = \mathbf{R}\mathbf{x} \quad (1)$$

根据压缩感知理论, 通过稀疏测量矩阵将 \mathbf{x} 投影到低维空间的 $\mathbf{v} = (v_1, v_2 \dots v_n)^T \in \mathbb{R}^n$, 利用最小重构误差从 \mathbf{v} 中高概率的恢复出高维向量 \mathbf{x} 。

对于样本 \mathbf{x} (m 维), 它的低维表示是 \mathbf{v} (n 维)。假定低维向量 \mathbf{v} 中的特征元素符合正态分布且符合贝叶斯原理, 利用朴素贝叶斯建模得到分类器 $H(\mathbf{v})$ 。

$$H(\mathbf{v}) = \lg \left\{ \frac{\prod_{i=1}^n p(v_i|y=1)p(y=1)}{\prod_{i=1}^n p(v_i|y=0)p(y=0)} \right\} \\ = \sum_{i=1}^n \lg \left\{ \frac{p(v_i|y=1)}{p(v_i|y=0)} \right\} \quad (2)$$

其中, $y \in \{0, 1\}$ 代表样本的标签, 设定 $y=0$ 表示负样本, $y=1$ 表示正样本。Diaconis等人^[15]已经证明: 大多数高维空间的随机投影总是服从高斯分布。据此, 假定构建的分类器 $H(\mathbf{v})$ 中的条件概率 $p(v_i|y=1)$ 和 $p(v_i|y=0)$ 满足高斯分布, 即

$$\left. \begin{aligned} p(v_i|y=1) &\sim N(\mu_i^1, \sigma_i^1) \\ p(v_i|y=0) &\sim N(\mu_i^0, \sigma_i^0) \end{aligned} \right\} \quad (3)$$

其中, μ_i^1 、 σ_i^1 和 μ_i^0 、 σ_i^0 分别为正样本和负样本的第 i 个特征的均值和方差。在分类器构建和更新阶段, 利用学习率 $\lambda > 0$ 按照式(4)更新 $\{\mu_i^1, \sigma_i^1\}$, $\{\mu_i^0, \sigma_i^0\}$ 采用相同方法更新。

$$\mu_i^1 \leftarrow \lambda \mu_i^1 + (1 - \lambda) \mu^1, \\ \sigma_i^1 \leftarrow \sqrt{\lambda(\sigma_i^1)^2 + (1 - \lambda)(\sigma^1)^2 + \lambda(1 - \lambda)(\mu_i^1 - \mu^1)^2} \quad (4)$$

$t+1$ 帧时, 采样、降维后的特征使用 t 帧更新好的分类器进行分类, 通过选取分类器 $H(\mathbf{v})$ 中分类分数最大的 $H(\hat{\mathbf{v}})$ 对应的区域作为目标位置。该区域的特征 $\hat{\mathbf{v}}$ 的定义如式(5):

$$\hat{\mathbf{v}} = \underset{\mathbf{v}}{\arg \text{Max}}(H(\mathbf{v})) \quad (5)$$

2.2 混合分类决策压缩跟踪算法

MDCT算法^[10]认为, CT算法采用式(5)确定目标位置及固定的学习率更新分类器的参数, 当目标受到严重遮挡或光照的变化, 跟踪算法仍然将目标

框中的外观已经变化的目标作为真正目标，始终以恒定的学习率更新分类器，导致分类器的性能降低，目标区域漂移甚至丢失。MDCT引入混合决策模型，定义两个分类器，如式(6)：

$$\begin{cases} H_{ocp} = \{\mu_o^1, \sigma_o^1, \mu_o^0, \sigma_o^0\} \\ H_{ncp} = \{\mu_p^1, \sigma_p^1, \mu_p^0, \sigma_p^0\} \end{cases} \quad (6)$$

其中， H_{ocp} 为原始分类器， H_{ncp} 为新的分类器。当原始分类器的最大分类分数 $H_{ocp}(\hat{v})$ 小于0，初始化一个新分类器 H_{ncp} ，这样获得两个分类器。在后续的跟踪中，有选择地使用这两个分类器对目标进行跟踪，其分类器的选择条件如式(7)：

$$L_t = \begin{cases} \text{Loc}(H_{ncp}(\hat{v})), H_{ocp}(\hat{v}) < 0 \\ \text{Loc}(H_{ocp}(\hat{v})), H_{ocp}(\hat{v}) > 0 \end{cases} \quad (7)$$

其中， $L_t = \text{Loc}(H_{ncp}(\hat{v}))$ 表示第 t 帧时，将 H_{ncp} 分类器的最大分类分数 $H_{ncp}(\hat{v})$ 所在的位置作为目标的最终位置 L_t 。MDCT算法为了减少更新分类器的影响，设立了一个新的不断增长的学习率，按照特定的方式进行分类器更新，有效防止了跟踪目标的特征被覆盖的对象特征所取代。

3 基于多模型的实时压缩跟踪算法

跟踪是动态样本状态估计问题，往往会涉及到模型的更新。原始CT算法使用单一的分类器来确定目标位置和一个恒定不变的学习率来更新分类器，使得目标外观变化较大时，分类器的最大分类分数可能急剧减少，继续将分类器的最大分类分数作为目标位置是不合适的，在发生漂移的目标周围采集正、负样本更新分类器，会造成误差的累积，使得跟踪任务的失败。而MDCT通过设定分类器的最大分类分数0为界限，利用不同的分类器进行分类，在遮挡、复杂背景下提高了跟踪的鲁棒性，但

简单的以 $H_{ocp}(\hat{v})$ 小于0就利用新分类器进行定位，判断条件过于简单，导致模型的交替过快，目标的细节信息会丢失较多，对跟踪准确率的提升有一定的限制。针对上述算法的不足，MMCT通过提出了多模型的分类算法，以及新的多模型更新策略，较大程度地提高了在复杂背景下、遮挡、光照、旋转等因素下目标跟踪的鲁棒性和准确性。

3.1 多模型分类算法

为了克服CT算法只使用单一分类器适应能力不强，以及MDCT的混合分类器交替过快丢失过多信息的缺陷，MMCT通过引入阈值 θ ，设计了一种新的基于多个分类器模型的跟踪决策算法，在第1帧中引入多个分类器对目标的表现进行建模。而阈值 θ 在一定程度上能够反映在跟踪过程中，目标所受到的各种因素干扰的程度，因此通过判断前两帧的最大分类分数的差值与 θ 的大小关系，在多个分类器中选择出当前帧的最合适模型，这样在保证实时性的同时，加强了分类器在外观较大变化的适应能力，提高了在各种因素干扰下的分类精度；并且在多个模型的选择中，利用 θ 限制了模型的过快交替，使得保留较多的细节信息。多模型算法的框架如图1所示。

MMCT通过判断前两帧的最大分类分数的差值与 θ 的不同情况，定义了3个模型的状态，其模型的状态如式(8)：

$$\begin{cases} H_o = \{\mu_o^1, \sigma_o^1, \mu_o^0, \sigma_o^0\} \\ H_f = \{\mu_f^1, \sigma_f^1, \mu_f^0, \sigma_f^0\} \\ H_b = \{\mu_b^1, \sigma_b^1, \mu_b^0, \sigma_b^0\} \end{cases} \quad (8)$$

图1中所示为一种决策情况。其中 H_* 代表 H_o, H_b, H_f 中的某一个， H_*^t 表示第 t 帧选择的分类器 H_* 。在跟踪的过程中，在第1帧真实的目标周围

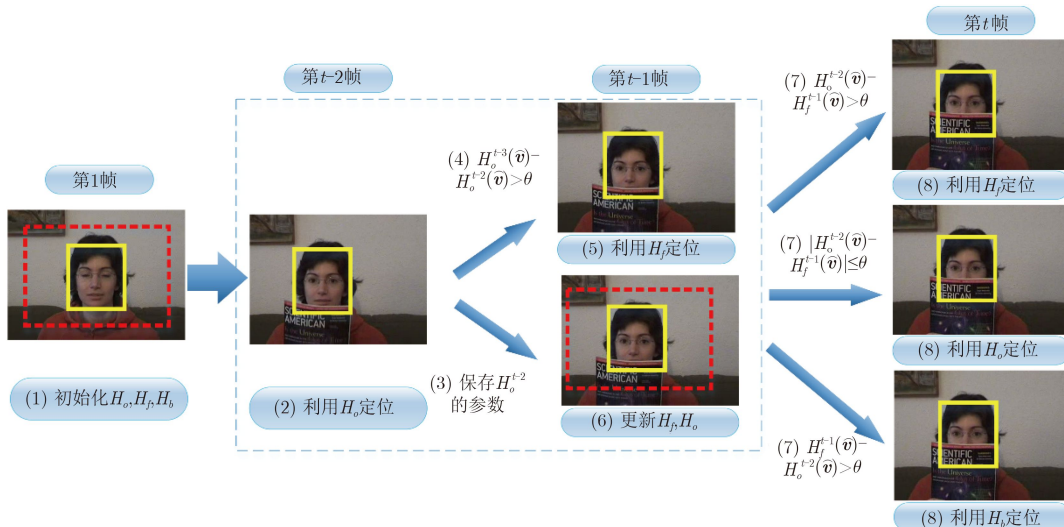


图1 多模型压缩跟踪算法框架(t 帧多模型选择策略)

采样, 降维后利用式(2)建立多分类器模型 H_o , 将其参数保存至 H_f 和 H_b , 实现多模型的初始化。第2帧时, 利用 H_o 确定目标位置, 利用式(4)更新 H_o 分类器参数。

第3帧开始, 使用多分类器模型进行跟踪。确定第 t 帧目标位置时, 判断 $t-2$ 帧的分类器最大分类分数 $H_*^{t-2}(\hat{\mathbf{v}})$ 与 $t-1$ 帧的分类器最大分类分数 $H_*^{t-1}(\hat{\mathbf{v}})$ 的差值: (1)若没有超过阈值 θ , 利用 H_o 分类器及正样本参数 $\{\mu_o^1, \sigma_o^1\}$ 和负样本参数 $\{\mu_o^0, \sigma_o^0\}$ 进行目标的确定, 该参数能有效适应跟踪任务。在分类器更新之前, 把 H_o 参数保存至 H_f 及 H_b ; (2)若超过了阈值 θ , 分为两种情况: (a) $t-2$ 帧的分类分数更大, 这表明目标对象被某些因素所影响, 在这种情况下, H_o 分类器已经不能准确估计目标的真实位置。为了跟踪被干扰的目标, 使用第 $t-1$ 帧已经更新的 H_f 的正样本参数 $\{\mu_f^1, \sigma_f^1\}$ 和负样本参数 $\{\mu_f^0, \sigma_f^0\}$ 进行目标位置的确定, 并更新 H_o 和 H_f 两个分类器模型; (b) $t-1$ 帧的分类分数更大, 表明目标外观同样发生了变化, 目标的分类效果变好, 干扰的情况减少, 使用 $t-1$ 帧已经更新的 H_b 的正样本参数 $\{\mu_b^1, \sigma_b^1\}$ 和负样本参数 $\{\mu_b^0, \sigma_b^0\}$ 进行目标位置的确定, 并更新 H_o 和 H_b 两个分类器模型。

当分类器在第 t 帧利用 H_o 不能准确地定位目标位置时, 使用分类器 H_f 及 H_b 来估计 t 帧的目标位置。不同的分类器模型根据前两帧的最大分类分数差值来定位跟踪的目标。多模型的决策方式如式(9):

$$L_t = \begin{cases} \text{Loc}(H_f(\hat{\mathbf{v}})), & H_*^{t-2}(\hat{\mathbf{v}}) - H_*^{t-1}(\hat{\mathbf{v}}) > \theta \\ \text{Loc}(H_o(\hat{\mathbf{v}})), & |H_*^{t-2}(\hat{\mathbf{v}}) - H_*^{t-1}(\hat{\mathbf{v}})| \leq \theta \\ \text{Loc}(H_b(\hat{\mathbf{v}})), & H_*^{t-1}(\hat{\mathbf{v}}) - H_*^{t-2}(\hat{\mathbf{v}}) > \theta \end{cases} \quad (9)$$

3.2 分类器更新策略

现阶段的绝大多数检测跟踪算法使用自适应的方式进行模型更新^[16]。分类器模型的更新方式对分类器的分类能力以及跟踪算法的性能至关重要。学习率是衡量分类器新鲜程度的参数, 其值越大, 样本训练时对分类器的影响越小, 反之亦然。当样本发生变化时, 可通过调节学习率减少或增加样本对分类器的影响。因此, MMCT算法通过采用不同的学习率对多模型进行更新, 更好地适应了跟踪过程中各个因素的影响。不同的学习率根据式(10)进行更新。

$$\left. \begin{aligned} \lambda_o &\leftarrow K \\ \lambda_f &\leftarrow \lambda_f + (1 - \lambda_f) \frac{\varepsilon}{\varepsilon + 1} \\ \lambda_b &\leftarrow p\lambda_o - (1 - p) \frac{\varepsilon}{\varepsilon + 1} \end{aligned} \right\} \quad (10)$$

其中, $\lambda_o, \lambda_f, \lambda_b$ 是多模型的学习率, K 表示固定的学习率, ε 表示前两帧分类器最大分类分数差值超过 θ 的数量, p 表示权值。当前两帧的最大分类分数变化在 θ 之内, 使用学习率 λ_o 估计目标的状态。如果目标被长时间遮挡, 即 $t-2$ 帧的分类分数较大, 新的学习率 λ_f 会随着 ε 增大, 有效防止目标的特征被覆盖的对象目标特征所取代。当 $t-1$ 帧的分类分数大, 新学习率 λ_b 会降低, 有效恢复了原始的目标特征, 减少跟踪漂移。MMCT通过结合flag标志以及不同的条件控制, 实现了有效的多模型参数更新方法。

3.3 多模型压缩跟踪算法流程

MMCT通过建立多个分类器模型, 使用不同的学习率对多模型进行参数的更新, 更加鲁棒地适应光照、遮挡、平面旋转及复杂背景下的跟踪。由于只是加入了多模型的选择条件, 并没有增加计算的复杂度, MMCT算法达到了实时的效果。表1所示是基于多模型的实时压缩跟踪算法流程及分类器参数的更新策略。

4 实验评估与分析

为了验证本文提出的MMCT算法的有效性、鲁棒性及实时性, 选择与6个算法进行对比。所有算法均在Wu等人^[14]提出的目标跟踪的Benchmark上统一配置, GroundTruth文件中的第1行数据作为初始位置。选取数据集OTB2013中具有代表性的6组图像序列, 包括了目标遮挡、平面旋转、姿态变化、复杂背景等诸多挑战。实验平台为Windows7, 64位操作系统, Matlab 2015b, 双核CPU 2.7 GHz, 内存10 GB。

4.1 评价指标

目标跟踪中为了评价算法各种跟踪性能, 采用跟踪成功率SR(Success Rate)衡量跟踪的稳定性, 其表现为目标的真实包围框与算法所得包围框的重叠率。若重叠率大于0.5, 则认为当前帧跟踪成功; 否则, 跟踪失败。将跟踪成功的帧数除以总帧数得到跟踪成功率SR。

采用准确率衡量跟踪的准确性, 其表现为中心点的位置误差CLE(Center Location Error), 即真实目标的中心坐标与算法跟踪中心坐标的欧式距离。所有帧的欧式距离的平均值即为中心位置偏差。

采用每秒运行帧数FPS(Frame Per Second)衡量跟踪的耗时情况。

4.2 参数设置

实验中具有较多的预设参数。利用压缩感知降维后的矩阵维度设定 $n = 50$ 。式(10)设定 $\lambda_o = K$ 是

表1 基于多模型的实时压缩跟踪算法

输入: 第 t 帧的图像
输出: 第 t 帧的目标位置
(1) if $H_*^{t-2}(\hat{v}) - H_*^{t-1}(\hat{v}) > \theta$ && flag=0
(2) 利用 H_f 及参数 $\{\mu_f^1, \sigma_f^1, \mu_f^0, \sigma_f^0\}$ 进行定位 $L_t = \text{Loc}(H_f(\hat{v}))$, 初始化学习率 λ_f ;
(3) 采集正、负样本, 利用 λ_o 更新 H_f 分类器, λ_f 更新 H_o 分类器;
(4) 设置flag=1, 转入第(16)步处理第 $t+1$ 帧;
(5) end if
(6) if $H_*^{t-1}(\hat{v}) - H_*^{t-2}(\hat{v}) > \theta$ && flag=0
(7) 利用 H_b 及参数 $\{\mu_b^1, \sigma_b^1, \mu_b^0, \sigma_b^0\}$ 进行定位 $L_t = \text{Loc}(H_b(\hat{v}))$, 初始化学习率 λ_b ;
(8) 采集正、负样本, 利用 λ_o 更新 H_b 分类器, λ_b 更新 H_o 分类器;
(9) 设置flag=1, 转入第(16)步处理第 $(t+1)$ 帧;
(10) end if
(11) if $ H_*^{t-2}(\hat{v}) - H_*^{t-1}(\hat{v}) \leq \theta$ && flag=0
(12) 将 H_o 参数保存至 H_f 及 H_b ;
(13) 在第 t 帧确定目标的位置为 $L_t = \text{Loc}(H_o(\hat{v}))$, 使用学习率 λ_o 按照式(4)更新 H_o 分类器;
(14) 转入第(1)步处理第 $t+1$ 帧;
(15) end if
(16) if $H_*^{t-2}(\hat{v}) - H_*^{t-1}(\hat{v}) > \theta$ && flag=1
(17) 利用 H_f 及 $\{\mu_f^1, \sigma_f^1, \mu_f^0, \sigma_f^0\}$ 定位 $L_t = \text{Loc}(H_f(\hat{v}))$;
(18) 采集正、负样本, 利用 λ_o 更新 H_f 分类器;
(19) 按照式(10)更新学习率 λ_f , 使用学习率 λ_f 更新 H_o 分类器;
(20) 转入第(16)步处理第 $t+1$ 帧;
(21) end if
(22) if $H_*^{t-1}(\hat{v}) - H_*^{t-2}(\hat{v}) > \theta$ && flag=1
(23) 利用 H_b 及 $\{\mu_b^1, \sigma_b^1, \mu_b^0, \sigma_b^0\}$ 定位 $L_t = \text{Loc}(H_b(\hat{v}))$;
(24) 采集正、负样本, 利用 λ_o 更新 H_b 分类器;
(25) 按照式(10)更新学习率 λ_b , 使用学习率 λ_b 更新 H_o 分类器;
(26) 转入第(16)步处理第 $t+1$ 帧;
(27) end if
(28) if $ H_*^{t-1}(\hat{v}) - H_*^{t-2}(\hat{v}) \leq \theta$ && flag=1
(29) 将 H_o 参数保存至 H_f 及 H_b ;
(30) 确定目标位置为 $L_t = \text{Loc}(H_o(\hat{v}))$, 利用学习率 λ_o 按照式(4)更新 H_o 分类器;
(31) 设置学习率 $\lambda_f = 0$ 和 $\lambda_b = \lambda_o$, 将flag设置为0, 转到第(1)步处理第 $t+1$ 帧;
(32) end if

一个常数, $K = 0.80$ 。式(10)中, 设定权值 $p = 0.9$ 。由于阈值 θ 控制了多个模型的转换, 其值的设定最为关键。通过选取不同的 θ 进行对比实验。根据 θ 的不同取值, 按照曲线与坐标轴围成的面积值(Area Under Curve, AUC)进行降序排列, 其结果如图2所示。因此, 在分类器的选择分支中, 将阈值 θ 设为95。

4.3 定量分析

MMCT通过与6个跟踪算法在6个图像序列上得到表2跟踪成功率对比。其中, 加粗为效果最好, 下划线为效果次好。从表中可以看出, MMCT算法在处理遮挡及光照变化问题上优势较为明显。相较于原始的CT算法, MMCT算法具有明显的提高, 在平均跟踪率上居于第2。图3所示的跟踪成功率与不同干扰因素下的曲线图, MMCT算法在图3(a)光照变化中取得最好的效果, 图3(b)的遮挡和图3(c)的复杂背景中居于第2。在加入了多个分类器的情况下, 相对原始CT算法有较大提升。

表3所示的是中心偏差率及平均帧率的对比。MMCT算法在处理遮挡、光照变化剧烈、复杂背景等问题上优势具有明显的优势, 且达到实时的处理速度。图4所示的是跟踪准确率与中心偏差率阈值的关系, MMCT在光照变化, 遮挡以及复杂背景中表现较好, 都居于第2, 表明多模型分类策略对于提高跟踪的准确率以及鲁棒性具有一定的效果, 一定程度上可以推广至其他算法。

4.4 算法性能分析

为了更加直观地分析MMCT跟踪算法的实验效果, 将不同算法的跟踪结果在6组不同场景下的图像序列进行可视化。

图5所示的是不同因素干扰的情况下各个算法的跟踪结果。各个算法的初始位置保持一致, 在目标运动过程中, 表现了不同算法的差异性。Fish序列包含光照变化、尺度变换干扰的影响, MMCT算法能够较好地适应光照变化, 从始至终能稳定地跟踪。FaceOcc1序列包含严重遮挡的因素, MMCT算法对遮挡情况表现出优异的效果, 不仅能够鲁棒准确地跟踪, 且几乎没有漂移。Boy序列包含快速移动、平面旋转等干扰因素, 前期各算法的效果差别不大, 但是后期MMCT的跟踪漂移加大, 渐渐的丢失目标的特征。Sylvester序列包含平面旋转、光照变化以及复杂背景等因素, MMCT算法能够较好地跟踪。Football和Basketball序列受到快速运动、复杂背景以及相似物体遮挡等因素的干扰, 跟踪的难度较高, MMCT的偏移较大, 效果不尽理想。由此可见, MMCT需要提高目标在快速运动情况下的跟踪效果。

5 结束语

MMCT算法将多分类器模型引入原始压缩跟踪算法, 是一种更加鲁棒、准确的实时目标跟踪算法。通过结合前两帧的最大分类分数, 对多分类器进行决策, 优化分类结果, 并创新性的提出使用不同学习率的多分类器参数更新策略, 使得分类器的

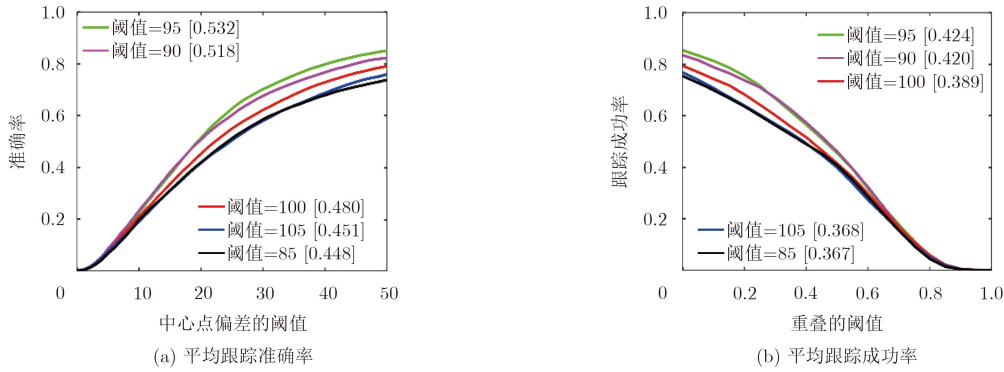


图2 选取不同阈值的跟踪准确率与成功率

表2 跟踪成功率

图像序列	TLD ^[8]	LOT ^[5]	OAB ^[7]	CT ^[9]	MDCT ^[10]	MMCT
Fish	0.49	0.25	0.44	0.21	<u>0.48</u>	0.45
Sylvester	0.63	0.52	0.57	<u>0.60</u>	0.49	0.57
Faceocc1	0.55	0.40	0.17	<u>0.68</u>	0.70	0.66
Boy	<u>0.57</u>	0.41	0.75	0.30	0.28	0.24
Basketball	0.07	0.45	0.05	0.13	0.11	<u>0.28</u>
Football	<u>0.56</u>	0.54	0.41	0.09	0.27	0.64
平均成功率	0.48	0.43	0.40	0.34	0.39	<u>0.47</u>

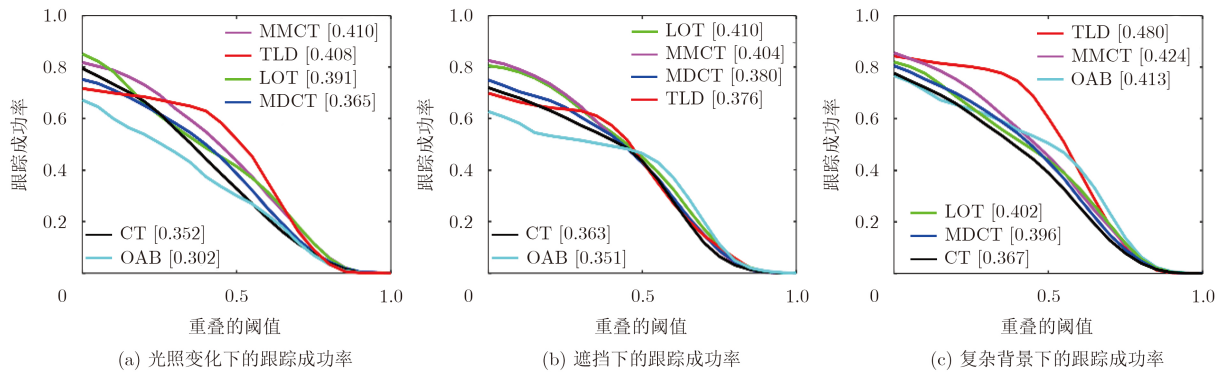


图3 不同算法的跟踪成功率

表3 中心偏差率及平均帧率

图像序列	TLD	LOT	OAB	CT	MDCT	MMCT
Fish	13.20	29.30	<u>22.50</u>	41.85	26.41	24.69
Sylvester	10.39	13.10	13.40	<u>12.12</u>	18.64	14.00
Faceocc1	30.63	34.20	89.34	<u>21.81</u>	21.16	23.92
Boy	<u>7.91</u>	72.34	3.70	20.21	60.66	66.65
Basketball	127.97	<u>69.28</u>	141.02	91.79	145.18	45.77
Football	13.40	<u>9.21</u>	22.63	227.90	23.59	7.4
平均偏差率	<u>33.92</u>	37.91	48.77	69.30	49.27	30.41
平均帧率	15.92	0.48	5.26	74.70	72.85	<u>74.60</u>

分类能力进一步提升。与常见的6种跟踪算法在6个视频序列上进行实验对比，表明MMCT提出的多模型跟踪算法能够更好地适应目标遮挡、光照变

化、平面旋转、复杂背景等因素的干扰，提高了算法的跟踪稳定性。同时，加入的多分类器模型没有增加算法的计算负担，平均处理速度高达74.60帧/s，

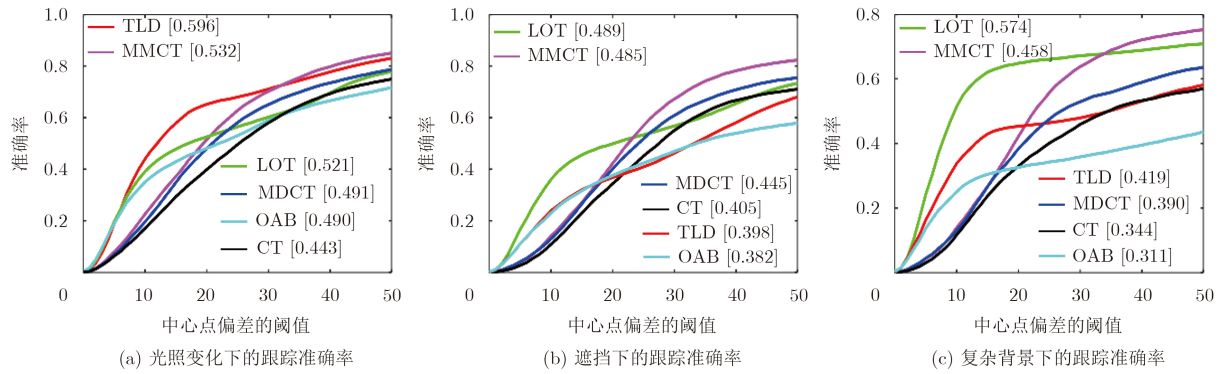


图 4 不同算法的跟踪准确率

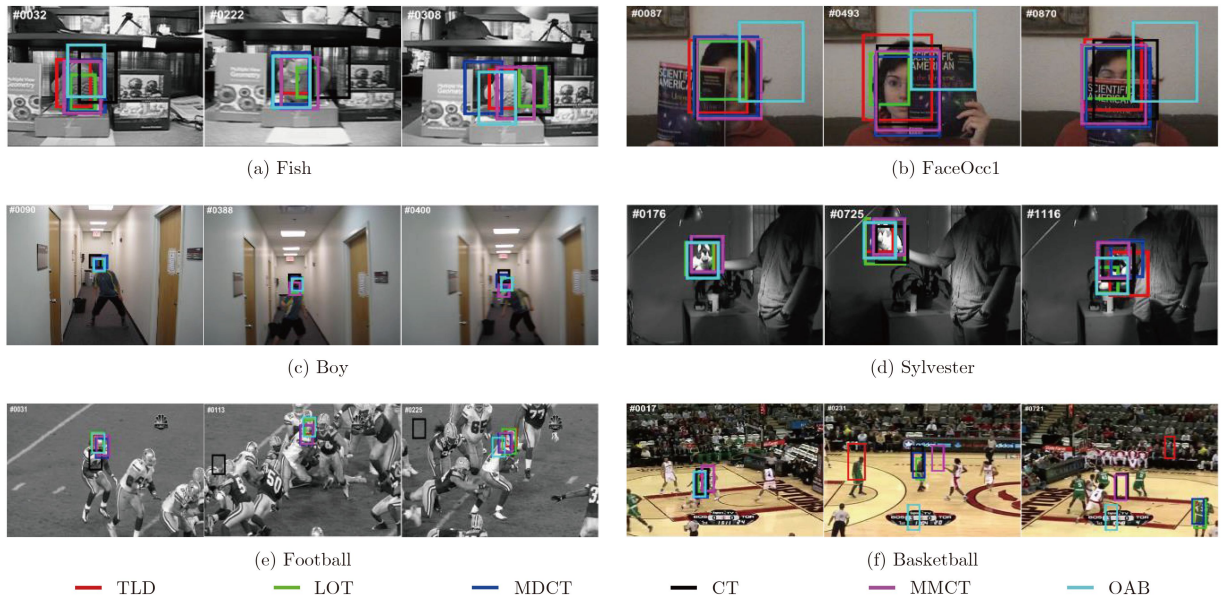


图 5 不同跟踪算法显示的结果

超过了30帧/s的实时处理要求。因此，MMCT算法能够应用于智能视频监控中，提高监控系统的智能监控能力。基于相关滤波的跟踪算法在当前的跟踪领域很流行，下一步的工作是推广多模型算法，将其应用于相关滤波中，提高的响应值确定目标的准确性，提高算法的跟踪性能。

参 考 文 献

[1] 郁道银, 王悦行, 陈晓冬, 等. 基于随机投影和稀疏表示的跟踪算法[J]. 电子与信息学报, 2016, 38(7): 1602–1608. doi: [10.11999/JEIT151064](https://doi.org/10.11999/JEIT151064).
YU Daoyin, WANG Yuexing, CHEN Xiaodong, et al. Visual tracking based on random projection and sparse representation[J]. *Journal of Electronics & Information Technology*, 2016, 38(7): 1602–1608. doi: [10.11999/JEIT151064](https://doi.org/10.11999/JEIT151064).

[2] 田鹏, 吕江花, 马世龙, 等. 基于局部差别性分析的目标跟踪算法[J]. 电子与信息学报, 2017, 39(11): 2635–2643. doi: [10.11999/JEIT170045](https://doi.org/10.11999/JEIT170045).
TIAN Peng, LÜ Jianghua, MA Shilong, et al. Robust object

tracking based on local discriminative analysis[J]. *Journal of Electronics & Information Technology*, 2017, 39(11): 2635–2643. doi: [10.11999/JEIT170045](https://doi.org/10.11999/JEIT170045).

[3] COLLINS R T. Mean-shift blob tracking through scale space[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Madison, USA, 2003, 2: 234–240.

[4] ADAM A, RIVLIN E, and SHIMSHONI I. Robust fragments-based tracking using the integral histogram[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, USA, 2006, 1: 798–805. doi: [10.1109/CVPR.2006.256](https://doi.org/10.1109/CVPR.2006.256).

[5] ORON S, BAR-HILLEL A, LEVI D, et al. Locally orderless tracking[J]. *International Journal of Computer Vision*, 2015, 111(2): 213–228. doi: [10.1007/s11263-014-0740-6](https://doi.org/10.1007/s11263-014-0740-6).

[6] BABENKO B, YANG M H, and BELONGIE S. Robust object tracking with online multiple instance learning[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, 33(8): 1619–1632. doi: [10.1109/TPAMI.2010.226](https://doi.org/10.1109/TPAMI.2010.226).

- [7] GRABNER H, GRABNER M, and BISCHOF H. Real-time tracking via on-line boosting[C]. Proceedings of the British Machine Vision Conference, Edinburgh, UK, 2006: 47–56.
- [8] KALAL Z, MIKOLAJCZYK K, and MATAS J. Tracking-learning-detection[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(7): 1409–1422. doi: [10.1109/TPAMI.2011.239](https://doi.org/10.1109/TPAMI.2011.239).
- [9] ZHANG Kaihua, ZHANG Lei, and YANG M H. Real-time compressive tracking[C]. Proceedings of the European Conference on Computer Vision, Florence, Italy, 2012: 864–877. doi: [10.1007/978-3-642-33712-3_62](https://doi.org/10.1007/978-3-642-33712-3_62).
- [10] SUN Hang, LI Jing, CHANG Jun, *et al.* Efficient compressive sensing tracking via mixed classifier decision[J]. *Science China Information Sciences*, 2016, 59(7): 072102. doi: [10.1007/s11432-015-5424-5](https://doi.org/10.1007/s11432-015-5424-5).
- [11] HELD D, THRUN S, and SAVARESE S. Learning to track at 100 fps with deep regression networks[C]. Proceedings of the European Conference on Computer Vision, Amsterdam, Netherlands, 2016: 749–765. doi: [10.1007/978-3-319-46448-0_45](https://doi.org/10.1007/978-3-319-46448-0_45).
- [12] NAM H and HAN B. Learning multi-domain convolutional neural networks for visual tracking[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016: 4293–4302. doi: [10.1109/CVPR.2016.465](https://doi.org/10.1109/CVPR.2016.465).
- [13] SONG Yibin, MA Chao, GONG Linjun, *et al.* CREST: Convolutional residual learning for visual tracking[C]. Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 2017: 2555–2564. doi: [10.1109/ICCV.2017.279](https://doi.org/10.1109/ICCV.2017.279).
- [14] WU Yi, LIM J, and YANG M H. Online object tracking: A benchmark[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, USA, 2013: 2411–2418. doi: [10.1109/CVPR.2013.312](https://doi.org/10.1109/CVPR.2013.312).
- [15] DIACONIS P and FREEDMAN D. Asymptotics of graphical projection pursuit[J]. *Annals of Statistics*, 1984, 12(3): 793–815. doi: [10.1214/aos/1176346703](https://doi.org/10.1214/aos/1176346703).
- [16] HARE S, GOLODETZ S, SAFFARI A, *et al.* Struck: Structured output tracking with kernels[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016, 38(10): 2096–2109. doi: [10.1109/TPAMI.2015.2509974](https://doi.org/10.1109/TPAMI.2015.2509974).
- 张建明: 男, 1976年生, 副教授, 博士, 研究方向为计算机视觉、智能交通系统。
- 金晓康: 男, 1993年生, 硕士生, 研究方向为计算机视觉、深度学习。
- 吴宏林: 男, 1982年生, 讲师, 博士, 研究方向为压缩感知。
- 伍悠: 女, 1995年生, 硕士生, 研究方向为计算机视觉、深度学习。