基于核循环结构的自适应特征融合目标跟踪

赵高鹏，沈玉鹏，王建宇
南京理工大学自动化学院，江苏 南京 210094

摘要 视频跟踪中，使用单一特征对目标进行描述难以适应复杂场景的变化，目标的尺度变化、形变、遮挡等因素易导致跟踪失败。为提高跟踪的稳健性，基于核循环结构，提出一种自适应特征融合和模型更新的跟踪方法，并引入尺度更新机制。首先利用目标的灰度特征和局部二值模式特征分别计算滤波响应图，依据响应图的峰值旁瓣比(PSR)自适应地分配权值并加权融合，从而估计目标的最佳位置。然后根据融合后响应图的 PSR 来判断跟踪质量，据此决定是否更新模型。最后在目标位置周围提取方向梯度直方图特征构建尺度金字塔，训练尺度相关滤波器，用来估计目标的最佳尺度。实验选取标准测试数据集中具有光照变化，遮挡和尺度变化的视频序列进行实验，结果表明，该算法能够实现对目标的稳定跟踪，并且在距离精度和成功率上均优于对比算法。

关键词 机器视觉；目标跟踪；核循环结构；特征融合；模型更新；尺度更新
中图分类号 TP391 **文献标识码** A
doi：10.3788/AOS201737.0815001

Adaptive Feature Fusion Object Tracking Based on Circulant Structure with Kernel

Zhao Gaopeng, Shen Yupeng, Wang Jianyu
Department of Automation, Nanjing University of Science and Technology, Nanjing, Jiangsu 210094, *China*

Abstract In video tracking, the use of a single feature to describe the target is difficult to adapt to the changes in complex scenes. Futhermore, the scale change, deformation, occlusion of target and other factors will lead to tracking failure. In order to improve the robustness of tracking, an adaptive feature fusion and model updating tracking algorithm is proposed based on the circulant structure of with kernel, and the scale updating mechanism is also introduced. Firstly, the response maps are calculated using the gray and local binary pattern features of the target respectively and fused by the weights assigned according to the peak to sidelobe ratio(PSR), and the best location is estimated. The PSR of the fused response map is also used to judge the tracking quality to decide whether to update the model. Finally, according to the scale pyramid constructed with the histograms of oriented gradients features extracted around the target location, the scale correlation filter is trained to estimate the optimal scale of the target. The experiment selects the sequences with illumination variations, occlusion and scale changes from the visual tracker benchmark datasets. The results show that the proposed algorithm can track the target robustly in complex scenes, and the distance precision and success rate are also superior to the compared algorithms.

Key words machine vision; object tracking; circulant structure of with Kernel; feature fusion; model update; scale update
OCIS codes 150.0155; 100.1160; 110.1085

1 引 言

目标跟踪是计算机视觉的一个重要分支，在人机交互、视频监控、增强现实、视频压缩等领域都有着广泛的应用[1]。目标跟踪的本质是从连续的视频图像序列中估计出目标的位置和大小。根据外观模型所使用的表达策略，大致可以将跟踪方法分成生成式法和判别式法。生成式法[2-4]在线或离线构建目标的外观模型，然后在候选区域中选择与目标模型最相似的区域作为跟踪结果。判别式法[5-7]将目标的跟踪问题看作是一

收稿日期：2017-02-27；**收到修改稿日期**：2017-04-11
基金项目：江苏省博士后科研资助计划(1601181B)
作者简介：赵高鹏(1983—)，男，博士，讲师，主要从事计算机视觉方面的研究。E-mail：zhaogaopeng@njust.edu.cn

个分类问题，用样本数据训练一个分类器，将前景区域从背景区域中分离出来。

近年来，相关滤波被引入目标跟踪的框架中，并取得了显著的效果。文献[8]提出了最小化输出平方误差和(MOSSE)算法。该算法选定一个目标区域，利用相关滤波器对目标外观建模，并将相似度的计算转移到傅里叶域中，大大提升了跟踪速度。然而 MOSSE 算法也是选择数量有限的候选样本训练分类器模型。在此基础上，文献[9]采用密采样并提出循环结构的检测跟踪（CSK）算法。该算法选定一个目标区域作为基样本并对基样本进行循环移位，构建大量的训练样本来训练正则化最小二乘分类器。同时对候选区域进行循环移位，得到大量检测区域。用密采样方式训练的滤波器更加稳健和精确，而且由于训练和检测过程都是在傅里叶域中进行的，跟踪速度也非常可观。但 CSK 只适用于单通道特征如灰度特征，在选择特征时存在局限性。Henriques 等[10]在 CSK 的基础上，将单通道特征推广到多通道特征，提出了核相关滤波（KCF）算法。文献[11]利用颜色名(CN)特征对 CSK 跟踪器进行拓展，提出了颜色名跟踪(CNT)器。CNT 利用主成分分析将 CN 特征从 11 维降到了 2 维，提高了运行速度和精度。

上述跟踪器都使用单一表观模型对目标进行描述，不仅对目标的先验信息利用率低，容易引入初始模型误差，而且在跟踪过程中难以适应目标表观特征的变化，导致模型误差不断累积进而影响跟踪的稳健性。为此，本文提出一种基于 CSK 框架下的特征融合跟踪算法。依据灰度特征和局部二值模式特征(LBP)[12]计算响应图，根据响应图的峰值旁瓣比(PSR)自适应地分配权值，将响应图加权求和。并根据融合后的响应图的 PSR，自适应地更新表观模型。同时在目标位置处提取不同尺寸图像片的 HOG 特征，用于尺度滤波器的训练和检测，选择最大响应时的尺寸作为估计的最优尺寸。

2　CSK 跟踪器

CSK 算法的核心是利用基样本的循环结构构建出大量样本用于训练和检测。引入核函数计算候选区域和表观模型的相似度，选取相似度最大的位置作为跟踪结果，同时利用离散傅里叶变换降低训练和检测中的计算量。

2.1　正则化最小二乘分类器

CSK 在目标图像样本的基础上通过循环移位得到所有训练样本，用来训练最小二乘分类器。分类器 $f(x)=\boldsymbol{e}^{\mathrm{T}}x$ 的训练过程可描述为

$$\min\sum_i [f(x_i)-y_i]^2+\lambda\|\boldsymbol{e}\|^2, \tag{1}$$

式中 y_i 为样本 x_i 的期望输出，λ 为一个固定的正则化参数。

采用核函数将原始输入 x 映射到特征空间 $\boldsymbol{\varphi}(x)$ 上，定义 $\kappa(x,x')=\boldsymbol{\varphi}^{\mathrm{T}}(x)\boldsymbol{\varphi}(x')$，$\kappa$ 为高斯核函数。由表示定理得 $\boldsymbol{e}=\sum_i\alpha_i\boldsymbol{\varphi}(x_i)$。依据文献[13]，$\boldsymbol{\alpha}$ 的闭式解为

$$\boldsymbol{\alpha}=(\boldsymbol{K}+\lambda\boldsymbol{I})^{-1}\boldsymbol{y}, \tag{2}$$

式中 $\boldsymbol{K}$ 为核矩阵，元素 $\boldsymbol{K}_{ij}=\kappa(x_i,x_j)$，$\boldsymbol{I}$ 为单位矩阵，$\boldsymbol{y}$ 为由 y_i 构成的向量，$\boldsymbol{\alpha}$ 为由 α_i 构成的向量。

文献[9]证明了 $\boldsymbol{K}$ 矩阵具有循环结构，利用这个性质，将(2)式进行傅里叶变换得到

$$\mathscr{F}(\boldsymbol{\alpha})=\frac{\mathscr{F}(\boldsymbol{y})}{\mathscr{F}(\boldsymbol{k})+\lambda}, \tag{3}$$

式中 $\mathscr{F}$ 是傅里叶变换。

2.2　快速检测

对于待检测的图像区域 z，依旧通过密采样产生大量样本。样本与表观模型的相似度为

$$\boldsymbol{y}=\boldsymbol{e}^{\mathrm{T}}z=\sum_i\alpha_i\kappa(z,x_i), \tag{4}$$

式中 x 为目标的表观模型，z 为当前待检测的样本，y 为样本 z 与表观模型的相似度。

同样利用样本循环结构的性质，可得到新输入图像区域 z 在所有候选区域位置处与表观模型相似度的分布为

$$f(z)=\mathscr{F}^{-1}[\mathscr{F}(\boldsymbol{k})\mathscr{F}(\boldsymbol{\alpha})], \tag{5}$$

式中 $\boldsymbol{k}$ 是一个向量，元素 $k_i=\kappa(z,P^ix)$。使得 $f(z)$ 取得最大值的位置就是新一帧中目标位置。

2.3　模型更新

在得到新一帧中目标的位置后，使用线性插值更新分类器的参数和目标的表观模型为

$$x=(1-\beta)x+\beta x_t, \tag{6}$$

$$\alpha=(1-\beta)\alpha+\beta\alpha_t, \tag{7}$$

式中 β 为更新模型的学习率，x 为前一帧用来跟踪的表观模型，x_t 为当前帧获得的表观模型，α 为前一帧分类器的参数，α_t 为当前帧 x_t 训练得到的分类器参数。

CSK 跟踪器只利用了目标的灰度特征，对光线敏感。如图 1(a)所示，当光线发生变化时，CSK 会偏离目标。而且在模型更新时，每一帧的跟踪结果都被用来更新目标的表观模型，当目标被遮挡或跟踪效果不佳时，会将错误的目标信息引入模型当中，使得模型漂移，导致跟踪失败，如图 1(b)所示。同时 CSK 跟踪器没有尺度更新的机制，当目标存在较大尺度变化时，模板更新会引入多余的背景信息，导致跟踪不精确，如图 1(c)所示。

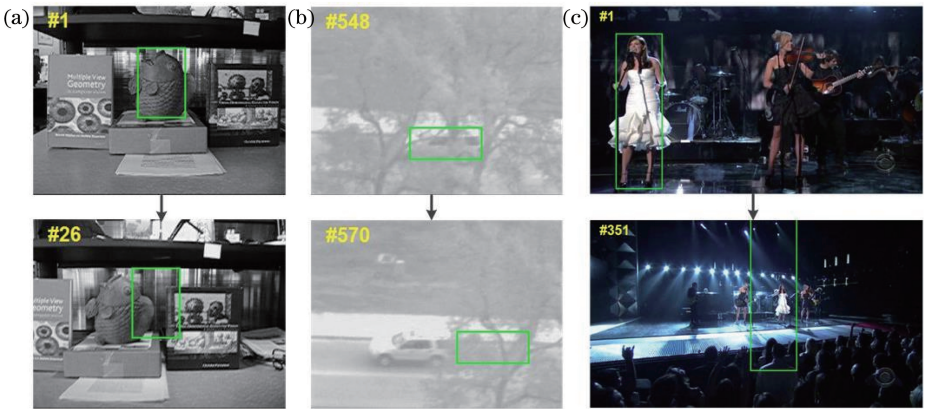

图 1　CSK 算法跟踪结果。(a)光照变化；(b)遮挡；(c)尺度变化

Fig. 1　Tracking results of CSK algorithm. (a) Illumination variation; (b) occlusion; (c) scale variation

3　特征融合和尺度自适应目标跟踪

CSK 跟踪器只利用了目标的灰度特征，对目标的表达能力有限，而且缺少自适应的模型更新和尺度更新的机制，限制了 CSK 跟踪器的性能。本节引入多特征进行跟踪，并在此基础上加入自适应的模型和尺度更新策略，以此来提升 CSK 跟踪器的性能。

3.1　自适应特征融合与模型更新

考虑到颜色特征和纹理特征互补的特性，即颜色特征具有旋转和形变稳健性，纹理特征能适应光照变化，将灰度特征和 LBP 特征的跟踪结果在响应图层面上进行融合。依据各自的跟踪质量，自适应地分配权重，突出更具判别力的特征。

CSK 跟踪器的理想响应是单峰值的高斯分布，且峰值较为突出。当跟踪质量不高时，会导致多个峰值或者峰值不明显。基于此，文献[8]提出一种衡量跟踪质量的指标——PSR。

$$R_{\mathrm{PS}}=\frac{v_{\max}-\mu_{\mathrm{b}}}{\delta_{\mathrm{b}}}, \tag{8}$$

式中 $v_{\max}$ 为响应图的峰值，b 为峰值周围的区域，即旁瓣区域，μ_{b} 和 δ_{b} 是旁瓣区域的均值和标准差。R_{PS} 越大，说明跟踪质量越高。

在用灰度和 LBP 特征跟踪得到各自的响应图时，分别计算灰度特征和 LBP 特征的 R_{PS}，两者中较大的值说明跟踪质量更可靠，在下一帧的检测中更应该依赖这一个特征，因此它的响应图应被赋予更大的权重。

灰度特征和 LBP 特征在 t 帧归一化的权重为

$$\bar{w}_{t,g}=\frac{R_{\mathrm{PS,g}}^2}{R_{\mathrm{PS,g}}^2+R_{\mathrm{PS,L}}^2}, \tag{9}$$

$$\bar{w}_{t,\mathrm{L}}=\frac{R_{\mathrm{PS,L}}^{2}}{R_{\mathrm{PS,g}}^{2}+R_{\mathrm{PS,L}}^{2}},\tag{10}$$

式中 $\bar{w}_{t,\mathrm{g}}$ 和 $\bar{w}_{t,\mathrm{L}}$ 分别为灰度特征和 LBP 特征在 t 帧归一化的权重，$R_{\mathrm{PS,g}}$ 和 $R_{\mathrm{PS,L}}$ 分别为灰度特征和 LBP 特征的 R_{PS} 值。

用 t 帧的权重来更新原有的特征权重

$$w_{t+1,\mathrm{g}}=(1-\tau)\times w_{t,\mathrm{g}}+\tau\times\bar{w}_{t,\mathrm{g}},\tag{11}$$

$$w_{t+1,\mathrm{L}}=(1-\tau)\times w_{t,\mathrm{L}}+\tau\times\bar{w}_{t,\mathrm{L}},\tag{12}$$

式中 τ 为更新系数，实验中取值 0.2。第一帧将灰度和 LBP 特征的权重初始化为 $w_{1,\mathrm{g}}=w_{1,\mathrm{L}}=0.5$。

在 t 帧时，分别用灰度特征和 LBP 特征跟踪得到两个响应图 $M_{t,\mathrm{g}}$ 和 $M_{t,\mathrm{L}}$，融合后的响应图为

$$M_t=w_{t,\mathrm{g}}\times M_{t,\mathrm{g}}+w_{t,\mathrm{L}}\times M_{t,\mathrm{L}},\tag{13}$$

响应图 M_t 的峰值处对应目标的中心。

当跟踪质量较差或者发生遮挡时，跟踪结果不应该用来更新模型。对于相关滤波跟踪，其响应图的形状能够有效地反映跟踪质量。当目标被遮挡或跟踪出现偏差时，滤波的响应峰值不够突出，使得 R_{PS} 较小。预定义阈值 N，当融合后的响应图 M_{t} 的 R_{PS} 值大于 N 时，用(6)式和(7)式来更新模型，否则不更新。

3.2　尺度估计

文献[14]在相关滤波的基础上提出了一种尺度更新的策略，其本质是训练一个一维的相关滤波器。

定义当前算法得到的位置是 x，目标尺寸为 $P\times Q$，尺度金字塔为 $u\in\left\{\lfloor-\frac{s-1}{2}\rfloor,\cdots,\lfloor\frac{s-1}{2}\rfloor\right\}$，$S$ 为尺度金字塔的层数。在 x 周围截取尺寸为 $a^nP\times a^nQ$ 的图像片(a 为尺度因子)，采用双线性插值将所有图像片统一成初始图像片的尺寸并提取 HOG 特征，用于分类器的训练和检测。

训练过程通过最小化代价函数得到最佳的相关滤波器，考虑输入信号 f 具有 d 维特征描述，则优化的目标函数为

$$\varepsilon=\left\|\sum_{l=1}^{d}h^l\bigstar f^l-g\right\|^2+\lambda\sum_{l=1}^{d}\|h^l\|^2,\tag{14}$$

式中 h 为滤波器，g 为理想输出，上标 l 为特征的某一维度，$\bigstar$ 表示循环相关。

根据文献[14]，尺度滤波器的解为

$$H^l=\frac{G^*F^l}{\sum_{k=1}^{d}F^{*k}F+\lambda}=\frac{A_t^l}{B_t+\lambda}\otimes,\tag{15}$$

式中大写字母表示在频域内，G 是理想输出 g 经过离散傅里叶变换后的值，上标 l 和 k 为特征的某一维度，$*$ 表示共轭。为了得到稳健的滤波器，分别对上面的 A_t^l 和 B_t 进行更新

$$A_t^l=(1-\eta)A_{t-1}^l+\eta G_t^*F_t^l,\tag{16}$$

$$B_t=(1-\eta)B_{t-1}+\eta\sum_{k=1}^{d}F_t^{*k}F_t^l,\tag{17}$$

式中 η 为模型更新的学习率，F 是输入信号 f 经过离散傅里叶变换后的值。

在新的一帧中，跟踪器确认位置后，依据尺度金字塔截取样本并提取特征 J，选择最大 y_s 对应的尺度作为结果。

$$y_s=\mathscr{F}^{-1}\left(\frac{\sum_{l=1}^{d}A^{*l}J^l}{B+\lambda}\right)。\tag{18}$$

得到了目标位置和尺度，选取样本重新训练滤波器，并按(16)式和(17)式进行更新。

3.3　算法流程

初始化：确定第一帧中的目标状态，更新阈值 $N=1.6$，初始峰值旁瓣比 $R_{\mathrm{PS},1}=2$。

for $t=1$ to T(T 为序列总帧数)

if $t>1$

//位置估计

根据目标的状态，采样图像块，分别采用 gray 和 LBP 特征表达；

利用$(\alpha_{t-1,\mathrm{g}},x_{t-1,\mathrm{g}})$、$(\alpha_{t-1,\mathrm{L}},x_{t-1,\mathrm{L}})$分别对 gray 和 LBP 特征表达图像块进行滤波，求出响应图 $M_{t,\mathrm{g}}$ 和 $M_{t,\mathrm{L}}$；

根据(13)式计算融合响应图，估计当前帧的位置；

根据(8)式计算融合后响应图的峰值旁瓣比 $R_{\mathrm{PS},t}$。

//尺度估计

根据目标位置，提取 HOG 特征构建尺度金字塔；

根据(A_{t-1},B_{t-1})和(18)式对尺度金字塔滤波，估计当前帧的尺度。

end

//训练和更新

if $R_{\mathrm{ps,t}}>N$

根据目标状态，训练和更新 gray 和 LBP 特征的滤波器模型(α_t',x_t')、尺度滤波模型(A_t,B_t)；

根据(11)式和(12)式更新 gray 和 LBP 特征的融合权重。

end

end

4　实验结果与分析

4.1　实验环境与参数设置

为验证本文算法的性能，进行两组实验。实验一：在不加入尺度更新的情况下，与 CSK 跟踪器对比，用来说明特征融合及模型自适应更新策略的有效性。实验二：加入尺度更新，与 KCF 跟踪器、判别尺度空间跟踪器(DSST)[14]、核的结构化输出跟踪器(STRUCK)[15]、多示例学习跟踪器(MIL)跟踪器[16]、跟踪学习检测跟踪器(TLD)[17]和压缩跟踪器(CT)[18]进行定性和定量比较。测试数据选用公开的标准测试数据集 benchmark[19]中的视频序列。实验一选用的视频序列为 car24、fish 和 suv，实验二选用的视频序列为 bird2、bolt2、human8、jogging2、car1 和 walking2，各视频序列特点如表 1 所示。

表 1　实验视频序列特点

Table 1　Characteristics of video sequences in the experiment

Sequence	Characteristic	Sequence	Characteristic
Car24	Illumination variation	Human8	Illumination variation, scale variation
Fish	Illumination variation	Jogging2	Occlusion
Suv	Occlusion	Car1	Illumination variation, scale variation
Bird2	Occlusion, deformation	Walking2	Scale variation, occlusion
Bolt2	Deformation, background clutters		

所有算法均采用 Matlab 软件实现，相对比的跟踪器均采用相同的算法代码。其中本文算法参数设置为：LBP 特征模型的更新速率 $\beta=0.02$，旁瓣区域设置成目标尺寸的 0.1 倍，预定义的更新阈值 $N=1.6$。

4.2　与 CSK 跟踪器对比

为了说明特征融合的有效性，在不加入尺度更新的情况下，将本文算法与 CSK 跟踪器进行对比。

4.2.1　定性分析

图 2 所示是部分跟踪结果。其中 fish 序列和 car24 序列(选取 1600～1700 帧)分别在第 20 帧和第 1680 帧附近出现明显的光照变化，但 CSK 跟踪器均丢失了目标或者明显偏移了目标，而所提出的经过特征融合后的跟踪器皆能较好地跟踪目标。suv 序列中，目标在第 560 帧附近被树冠遮挡，由于 CSK 没有自适应的模型更新机制，导致遮挡后模型迅速漂移，当目标再出现时无法准确跟踪。加入了上述模型更新机制后，跟踪器能有效地处理遮挡问题。

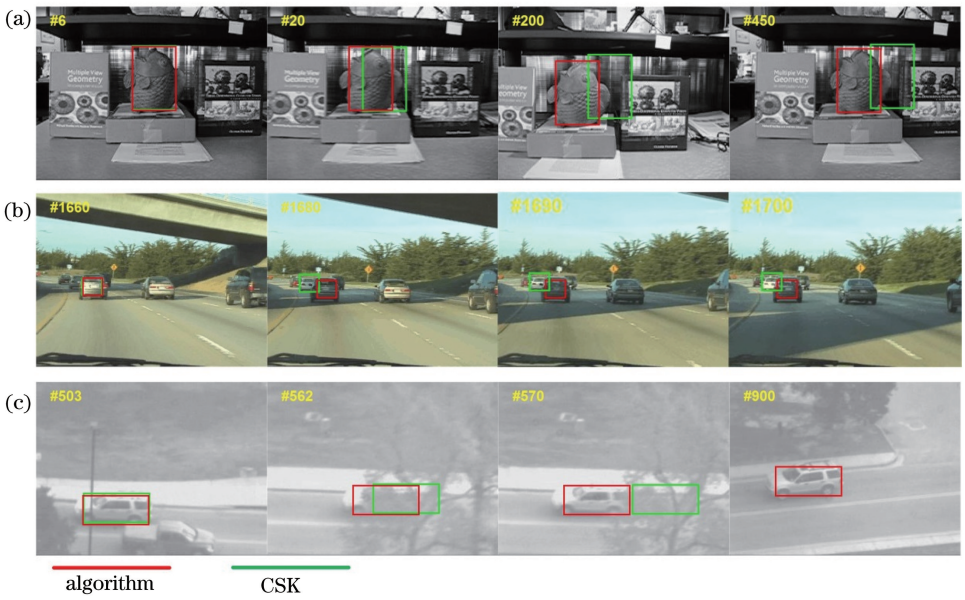

图 2　本文算法与 CSK 算法的跟踪结果。(a) Fish;(b) car24;(c) suv

Fig. 2　Tracking results of the proposed algorithm and CSK algorithm. (a) Fish; (b) car24; (c) suv

4.2.2　定量分析

选取中心位置误差(CLE)来比较本文算法和 CSK 算法的性能。中心位置误差 $c=(\|O_{\mathrm{cal}}-O_{\mathrm{aim}}\|^2)^{\frac{1}{2}}$，其中 O_{cal} 和 O_{aim} 分别为使用算法得到的目标中心点坐标和真实目标中心坐标。中心位置误差越小，表明算法性能越高。

各个序列的中心位置误差如图 3 所示。图中标注的数值是平均中心位置误差(ACLE)。由图 3 可知，在这 3 个序列上，本文算法明显提升了跟踪的精度，改善了跟踪器的性能。

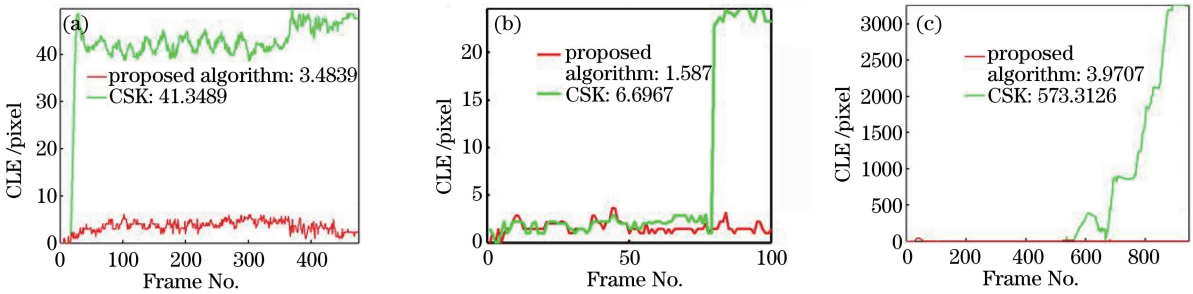

图 3　本文算法和 CSK 算法的中心位置误差对比。(a) Fish;(b) car24;(c) suv

Fig. 3　Center location error comparison of the proposed algorithm and CSK algorithm. (a) Fish; (b) car24; (c) suv

4.3　与各主流跟踪器的比较

4.3.1　定性比较

图(4)是跟踪实验的部分结果，不同算法的结果采用不同颜色的实线矩形框来表示，左上角是图像帧数。图 4(a)是 bird2 视频的部分跟踪结果。目标在第 14 帧左右被部分遮挡，由于纹理特征对遮挡有一定的稳健性，使用 HOG 特征作为表观模型的 DSST、KCF 算法以及本文算法均能够较好地跟踪物体，但其余的算法都出现了偏离。在第 56 帧附近，目标姿态变化较大，基于纹理特征的 KCF 和 DSST 算法偏离了目标，导致后续跟踪不准确甚至失败。在这些主流算法都出现偏离或失败时，本文算法仍然能够较好地跟踪目标。图 4(b)是对 bolt2 视频跟踪的部分结果。目标在第 3 帧附近时，姿态变化很大，KCF、DSST 和 TLD 算法均跟踪失败。而在第 200 帧附近时背景较为复杂，出现了类目标干扰，CT 和 STRUCK 算法偏离目标。只有本文算法和 MIL 算法能准确地跟踪目标。图 4(c)是对 Human8 视频跟踪的部分结果。在第 20 帧附近出现剧烈的光照变化，CT、MIL、STRUCK 和 TLD 算法跟踪失败。KCF 算法由于缺少尺度更新的机制，当目标

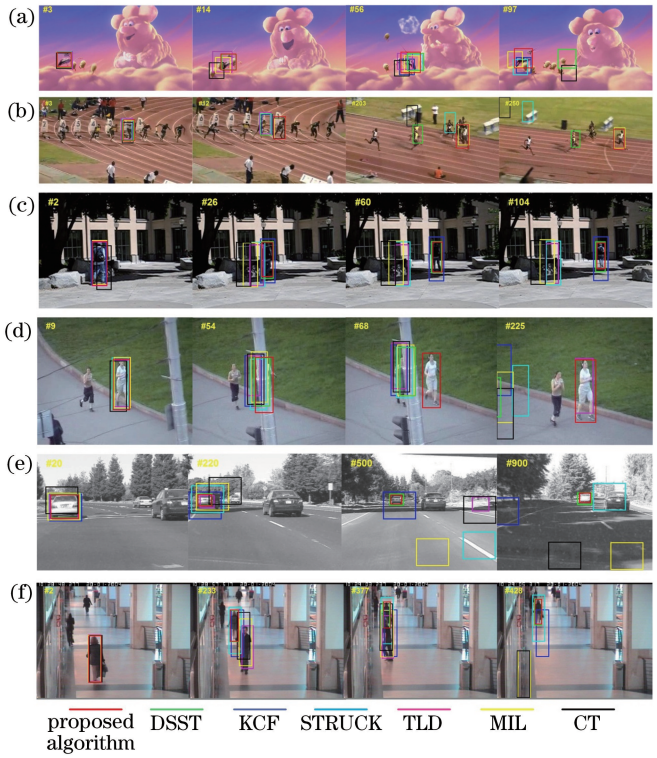

图 4　不同跟踪算法的跟踪结果。(a) Bird2;(b) bolt2;(c) human8;(d) jogging2;(e) car1;(f) walking2

Fig. 4　Tracking results of different tracking algorithms. (a) Bird2; (b) bolt2; (c) human8; (d) jogging2; (e) car1; (f) walking2

尺度变化较大时,跟踪结果不够准确。而本文算法和 DSST 都有很好的跟踪效果。

图 4(d)是 jogging2 视频的部分跟踪结果。在第 54 帧左右,目标被完全遮挡,DSST、KCF、STRUCK、MIL 和 CT 算法均跟踪失败,而本文算法因加入了自适应的目标表观模型的更新策略,能在一定程度上抑制模板的漂移,因此依然能够跟踪目标。TLD 算法虽然也能在跟踪失败后重新找回目标,但总体效果不佳。图 4(e)是对 car1 视频跟踪的部分结果。在目标尺度变化不大时,各算法都能跟踪上。但由于 KCF、STRUCK、MIL 和 CT 算法缺少尺度更新机制,随着目标越来越小,越来越多的无用信息或者错误信息引入目标的表观模型,使得模型不够准确,最终导致跟踪失败。本文算法和 DSST 算法能适应尺度变化,因此具有较好的跟踪效果。从图 4(e)中第 900 帧图像来看,本文算法的尺度比 DSST 要更精确。图 4(f)是 walking2 视频的部分跟踪结果。若没有尺度更新,当目标出现较大尺度变化时,模板会渐渐漂移,使得跟踪的稳健性变差。因此在第 210 帧左右出现类似目标干扰时,KCF、TLD、CT 和 MIL 算法都跟踪失败。STRUCK 算法虽然能够跟住目标,但效果并不好,而本文算法和 DSST 算法都能成功跟踪目标。

4.3.2　定量分析

选取距离精度(DP)和成功率(SR)定量比较各个跟踪器。

1) 距离精度 $p(N_{\mathrm{D}})=\sum_{t=1}^{T}\delta(D_t<N_{\mathrm{D}})/T$,式中 D_t 为第 t 帧的跟踪结果与真实值之间的中心位置误差,N_{D} 为中心位置误差的阈值,T 为视频序列的总帧数。

2) 成功率 $s(N_{\mathrm{Q}})=\sum_{t=1}^{T}\delta(Q_t>N_{\mathrm{Q}})/T$,式中 $Q_t=\frac{\gamma_{\mathrm{aim}}\cap\gamma_{\mathrm{tr}}}{\gamma_{\mathrm{aim}}\cup\gamma_{\mathrm{tr}}}$,为第 t 帧的跟踪结果与真实值之间的重叠率。γ_t 为跟踪结果区域,γ_{aim} 为目标真实区域,N_{Q} 为重叠率阈值,T 为序列的总帧数。

距离精度与中心位置误差的关系如图 5 所示,曲线越陡,表示跟踪精度越高。各个跟踪器的距离精度(中心位置误差阈值为 20)如表 2 所示,对表 2 数据进行分析,加粗代表最优,下划线代表次优。

从图 5 和表 2 中可以看出,本文算法的距离精度在 bolt2 视频序列上是次优的,在其余视频序列上均为最优。

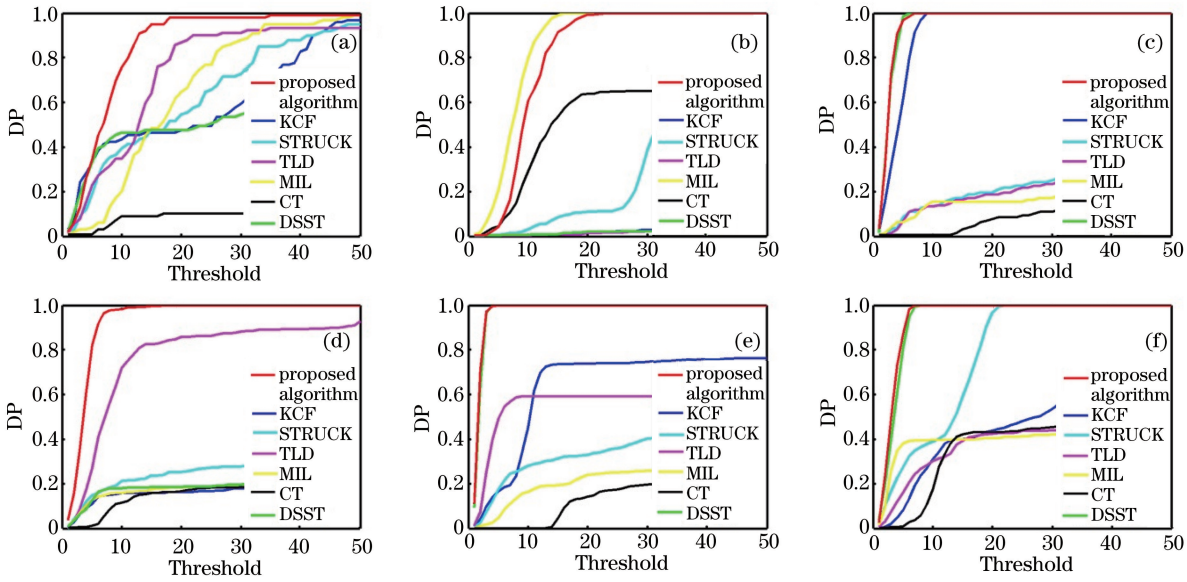

图 5　跟踪算法的距离精度。(a) Bird2;(b) bolt2;(c) human8;(d) jogging2;(e) car1;(f) walking2

Fig. 5　Distance precision of tracking algorithms. (a) Bird2; (b) bolt2; (c) human8; (d) jogging2; (e) car1; (f) walking2

表 2　各跟踪算法的距离精度

Table 2　Distance precision of different tracking algorithms

Sequence	Algorithm						
	KCF	STRUCK	TLD	MIL	CT	DSST	Proposed algorithm
Bird2	0.474	0.545	0.869	0.646	0.101	0.474	**0.979**
Bolt2	0.017	0.109	0.013	**1.000**	0.638	0.020	0.993
Human8	**1.000**	0.195	0.187	0.156	0.078	**1.000**	**1.000**
Jogging2	0.162	0.254	0.856	0.182	0.166	0.185	**1.000**
Car1	0.738	0.330	0.591	0.239	0.141	**1.000**	**1.000**
Walking2	0.438	0.966	0.424	0.406	0.432	**1.000**	**1.000**

图 6 所示为成功率和不同重叠率阈值之间的关系，即重叠率大于特定阈值的视频帧占总帧数的比例。各个跟踪器的成功率(重叠率阈值为 0.5)如表 3 所示，对表中数据分析，加粗代表最优，下划线代表次优。

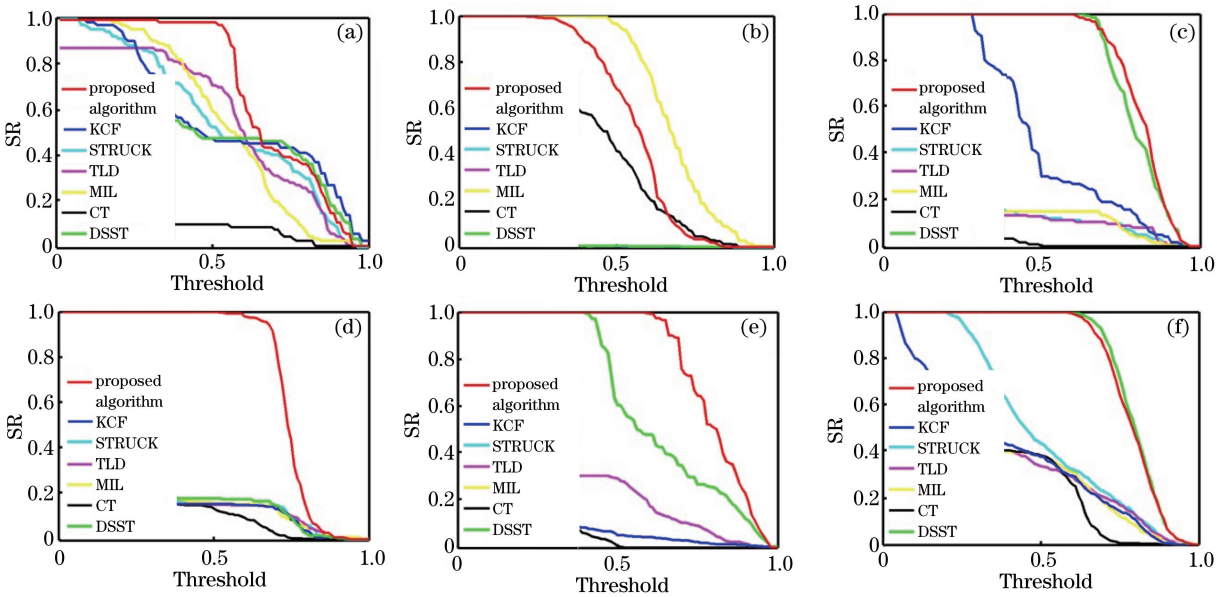

图 6　跟踪算法的成功率。(a) Bird2;(b) bolt2;(c) human8;(d) jogging2;(e) car1;(f) walking2

Fig. 6　Success rate of tracking algorithms. (a) Bird2; (b) bolt2; (c) human8; (d) jogging2; (e) car1; (f) walking2

从图 6 和表 3 中可以看出，本文算法的成功率在 bolt2 序列上是次优的，在其余视频序列上均为最优。

表 3　各跟踪算法的成功率

Table 3　Success rate of different tracking algorithms

Sequence	Algorithm						
	KCF	STRUCK	TLD	MIL	CT	DSST	Proposed algorithm
Bird2	0.474	0.525	0.707	0.595	0.101	0.474	**0.979**
Bolt2	0.006	0.006	0.006	**0.965**	0.419	0.010	0.686
Human8	0.304	0.132	0.132	0.156	0.007	**1.000**	**1.000**
Jogging2	0.159	0.162	0.156	0.162	0.140	0.182	**0.996**
Car1	0.053	0.053	0.293	0.053	0.007	0.604	**1.000**
Walking2	0.378	0.434	0.338	0.380	0.384	**1.000**	**1.000**

4.4　算法复杂度分析

所有仿真实验在中央处理器为 Intel Core i5-3210M，主频为 2.5 GHz，内存为 4 GB 的计算机环境下，基于 MATLAB R2011b 软件平台进行实现。表 4 所示为本文算法与 CSK、DSST 算法在上述 6 个序列上的复杂度值对比。由表 4 可知，本文算法基本能满足实时性的要求。

表 4　复杂度分析

Table 4　Complexity analysis　　frame · s^{-1}

Algorithm	Sequence					
	Bird2	Bolt2	Human8	Jogging	Car1	Walking2
CSK	51.4	130.5	139.1	81.1	175.1	80.7
DSST	7.2	21.6	17.5	10.1	18.0	12.5
Proposed algorithm	20.0	39.0	34.0	28.7	49.6	30.9

5　结　　论

在 CSK 跟踪算法的基础上，从自适应特征融合和自适应模型更新两个方面对 CSK 算法进行了改进，并引入尺度更新的机制。本文算法利用灰度和 LBP 特征的滤波响应图的峰值旁瓣比自适应地分配权值，将两者的响应图加权融合，并依据融合后响应图的峰值旁瓣比决定是否更新模型。同时构建尺度金字塔训练尺度滤波器，用来估计目标的最佳尺度。实验对比表明，在目标发生遮挡、姿态变化、光照变化、尺度变化等复杂场景下，该算法具有更强的稳健性。

参　考　文　献

[1] Zhang Huanlong, Hu Shiqiang, Yang Guosheng. Video object tracking based on appearance models learning[J]. Journal of Computer Research and Development, 2015, 52(1): 177-190.
张焕龙，胡士强，杨国胜. 基于外观模型学习的视频目标跟踪方法综述[J]. 计算机研究与发展，2015，52(1)：177-190.

[2] Sun Rui, Huang Jingru, Ding Wenxiu. A real-time object tracking algorithm based on subspace learning[J]. Opto-Electronic Engineering, 2015, 42(2): 52-58.
孙　锐，黄静茹，丁文秀. 一种基于子空间学习的实时目标跟踪算法[J]. 光电工程，2015，42(2)：52-58.

[3] Yang F, Lu H C, Yang M H. Robust superpixel tracking[J]. IEEE Transactions on Image Processing, 2014, 23(4): 1639-1651.

[4] Liu Wenzhuo, Yuan Guanglin, Xue Mogen. Robust fast visual tracking based on two-stage sparse representation[J]. Acta Optica Sinica, 2016, 36(12): 1215001.
刘文琢，袁广林，薛模根. 基于两阶段稀疏表示的鲁棒快速视觉跟踪[J]. 光学学报，2016，36(12)：1215001.

[5] Babenko B, Yang M H, Belongie S. Robust object tracking with online multiple instance learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(8): 1619-1632.

[6] Xue Yizhe, Wang Tuo. Object-tracking method based on improved cost-sensitive adaboost[J]. Journal of Image and Graphics, 2016, 21(5): 544-555.

薛一哲，王　拓. 基于代价敏感 Adaboost 目标跟踪[J]. 中国图象图形学报，2016，21(5)：544-555.

[7] Li Y, Zhu J K. A scale adaptive kernel correlation filter tracker with feature integration[C]. Proceedings of 13th European Conference on Computer Vision, 2014, 8926: 254-265.

[8] Bolme D S, Beveridge J R, Draper B A, *et al*. Visual object tracking using adaptive correlation filters[C]. Proceedings of 2010 IEEE Conference on Computer Vision and Pattern Recognition, 2010: 2544-2550.

[9] Henriques J F, Caseiro R, Martins P, *et al*. Exploiting the circulant structure of tracking-by-detection with kernels [C]. Proceedings of 12th European Conference on Computer Vision, 2012, 7575: 702-715.

[10] 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.

[11] Danelljan M, Khan F S, Felsberg M, *et al*. Adaptive color attributes for real-time visual tracking[C]. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014: 1090-1097.

[12] Ojala T, Pietikainen M, Harwood D. Performance evaluation of texture measures with classification based on kullback discrimination of distributions[C]. Proceedings of 12th IAPR International Conference on Pattern Recognition, 1994, 1: 582-585.

[13] Rifkin R, Yeo G, Poggio T. Regularized least-squares classification[J]. Nato Science Series Sub Series III Computer and Systems Sciences, 2003, 190: 131-154.

[14] 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, BMVA Press, 2014.

[15] Hare S, Saffari A, Torr P H S. Struck:structured output tracking with kernels[C]. Proceedings of IEEE International Conference on Computer Vision, 2011: 263-270.

[16] Babenko B, Yang M H, Belongie S. Visual tracking with online multiple instance learning[C]. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009: 983-990.

[17] Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(7): 1409-1422.

[18] Zhang K H, Zhang L, Yang M H. Real-time compressive tracking[C]. Proceedings of 12th European Conference on Computer Vision, 2012, 7574: 864-877.

[19] Wu Y, Lim J, Yang M H. Online object tracking: a benchmark[C]. Proceedings of 26th IEEE Conference on Computer Vision and Pattern Recognition, 2013:2411-2418.