

基于背景感知与快速尺寸判别的相关滤波跟踪算法

王永雄^{1,2}, 冯 汉^{1,2}

(1. 上海理工大学光电信息与计算机工程学院, 上海, 200093; 2. 上海康复器械工程技术研究中心, 上海, 200093)

摘要: 背景感知相关滤波 (Background-aware correlation filters, BACF) 算法有效地解决了相关滤波类跟踪算法中的边界效应问题, 提升了训练样本集的质量和数量, 能够精确估计目标的位置变化, 从而提高了跟踪器的性能。然而为了检测尺度变化, BACF 算法通过多次重复计算不同尺度的目标区域, 严重影响了跟踪速度。本文在 BACF 算法的基础上, 采用平移加尺度滤波的思想, 设计独立的一维尺度滤波器, 与 BACF 算法无缝结合。只需预测一次目标的位置变化, 再利用尺度滤波器预测目标尺度变化。因为两个滤波器单独训练、局部优化, 尺度滤波器计算量远小于 BACF 算法, 所以本文算法在保证精准预测目标尺度变化的同时极大提升目标的跟踪速度。实验结果表明: 与 BACF 算法相比, 本文算法在不损失跟踪精度的基础上提高约 75% 的跟踪速度。

关键词: 目标跟踪; 相关滤波; 尺度自适应; 边界效应

中图分类号: TP391.7 **文献标志码:** A

Object Tracking Based on Kernelized Correlation Filter with Background-Aware and Fast Discriminative Scale Space

WANG Yongxiong^{1,2}, FENG Han^{1,2}

(1. School of Optional-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, 200093, China; 2. Shanghai Rehabilitation Equipment Engineering Technology Research Center, Shanghai, 200093, China)

Abstract: To deal with the limitation of boundary effect and improve tracking speed in the traditional object tracking based on correlation filter, a fast discriminative scale space with background-aware correlation filters (BACF) algorithm is proposed. The BACF algorithm effectively solves the boundary effect caused by cyclic shift, greatly increases the quality and quantity of training samples, and thus improves the performance of the tracker. However, because of its disadvantage in scale detection strategy, it seriously affects its tracking speed. The we design a one-dimensional scale filter. The scale filter and translation filter are trained and optimized independently. The proposed algorithm greatly improves tracking speed while ensuring scale and translation estimation. The experimental results show that compared with BACF algorithm, the proposed algorithm can improve the tracking speed by about 75% without losing the tracking accuracy.

Key words: object tracking; kernelized correlation filters (KCF); adaptive scale; boundary effect

引言

相关滤波(Correlation filters, CF^[1])首次被应用到目标跟踪就取得很好的跟踪效果,产生了极大的反响。Henriques等在此基础上引入循环移位操作和核技巧(Circulant structure of tracking-by-detection with kernel, CSK^[2])极大增加了训练样本集并增强滤波器的性能,在后续的工作中他们提出了核相关滤波跟踪(High-speed tracking with kernelized correlation filters, KCF)算法^[3]使用了更具鉴别力的快速梯度直方图特征(Fast HOG, FHOG^[4])替代原先的灰度特征,进一步提高跟踪精度。Danelljan等^[5]则利用颜色属性特征(Color name, CN^[6])提出颜色自适应的思想,选取最有鉴别力的颜色特征进行跟踪,同样能取得较好的跟踪效果。另外近几年深度学习与相关滤波的结合在跟踪领域也取得越来越多的成果^[7-9]。

相关滤波类跟踪算法在取得飞速发展的同时也有一些难点问题亟待解决,边界效应就是其中之一。边界效应的产生是由于循环移位过程中除了最原始的样本以外其余样本都是循环移位合成的,这样真实样本占比太少会导致训练结果不佳。实际应用中普遍通过扩大目标检测区域并且添加余弦窗抑制边界效应,这样只要在循环移位过程中保持目标的完整即可认为该样本合理。但是添加余弦窗会过滤掉了大量背景信息,极大降低了跟踪器的性能。Galoogahi等提出的背景感知相关滤波(Background-aware correlation filters, BACF)^[10]算法巧妙地设计了一个裁剪矩阵 P ,在扩大检测区域的基础上进行循环移位操作,最后裁剪出合适的目标和背景信息。假设检测区域扩大后矢量化的维度为 T ,原始目标区域矢量化的维度为 D ,则被循环移位操作污染的样本只占 $\frac{D-1}{T}$,当 $T \gg D$ 时污染样本可以忽略不计。

BACF算法通过裁剪操作使得训练样本集的质量和数量都有极大的提高,能够精准预测目标位置变化。但是目标在移动过程中除了位置变化外,还会伴随着尺度变化,BACF算法通过多次重复计算多个尺度的目标区域获得最大响应,求得目标预测的位置及尺度。假设BACF算法单个尺度的位置预测运算时间为 T ,则 N 个尺度的计算时间约为 $N \cdot T$,该尺度检测策略严重影响目标的跟踪速度。本文针对此问题采用了Danelljan等^[11]提出的平移加尺度滤波思想,设计了一个独立的一维尺度滤波器,充分利用了BACF算法的解决边界效应精准估计目标位置的优势,只计算单次BACF算法的目标位置估计,然后再利用尺度滤波器预测目标尺度变化。由于两个滤波器单独训练、局部优化,且尺度滤波器计算量远小于完整的BACF算法,假设尺度滤波器单次计算时间为 T_s ($T_s < T$),则本文算法单次计算时间约为 $T + T_s$,所以本文算法能在保证精准预测和目标尺度变化的同时极大提升目标的跟踪速度,与BACF算法相比,本文算法在不损失跟踪精度的基础上提高约75%的跟踪速度。

1 相关滤波器模型

1.1 传统相关滤波器模型

空间域相关滤波跟踪模型是通过最小化损失函数来找到适合的滤波器 h ,具体损失函数定义为

$$E(h) = \frac{1}{2} \left\| \mathbf{y} - \sum_{k=1}^K \mathbf{h}_k \circ \mathbf{x}_k \right\|_2^2 + \frac{\lambda}{2} \sum_{k=1}^K \|\mathbf{h}_k\|_2^2 \quad (1)$$

式中: $\mathbf{y} \in \mathbf{R}^D$ 代表目标期望输出; K 表示目标特征通道数;“ \circ ”表示空间相关运算符; λ 为正则化系数; $\mathbf{x}_k \in \mathbf{R}^D$, $\mathbf{h}_k \in \mathbf{R}^D$,分别表示第 k 个通道矢量化的训练样本和第 k 个通道的滤波器, D 表示原训练样本单通道的维度。由于式(1)只针对一帧图像做优化,往往得不到鲁棒性高的滤波器,通过循环移位操作可以极大扩充训练样本集,提高滤波器的鲁棒性。添加循环移位操作后的损失函数定义为

$$E(\mathbf{h}) = \frac{1}{2} \sum_{j=1}^D \left\| \mathbf{y}(j) - \sum_{k=1}^K \mathbf{h}_k^T \mathbf{x}_k[\Delta\tau_j] \right\|_2^2 + \frac{\lambda}{2} \sum_{k=1}^K \|\mathbf{h}_k\|_2^2 \quad (2)$$

式中: $[\Delta\tau_j]$ 代表第 j 次循环移位操作, $\mathbf{y}(j)$ 代表期望输出的第 j 个元素, 上标 T 代表转置操作。

1.2 背景感知相关滤波器模型

循环移位操作可以极大增加训练样本集, 提高分类器性能, 通过利用频域的相关性质^[12]进行快速求解。通过引言部分介绍可知边界效应会极大削弱跟踪器的性能。不同于文献[11]采用 1.1 节所示的平移滤波器带有边界效应, 本文采用 BACF 算法的思想, 利用基于背景感知的平移滤波器, 削弱边界效用, 提升跟踪器性能。

利用基于背景感知的平移滤波器通过添加一个裁剪矩阵 \mathbf{P} 裁剪出合适的训练样本, 既能增加训练样本数量, 又能极大提高训练样本的质量。添加裁剪矩阵 \mathbf{P} 后的损失函数定义为

$$E(\mathbf{h}) = \frac{1}{2} \sum_{j=1}^T \left\| \mathbf{y}(j) - \sum_{k=1}^K \mathbf{h}_k^T \mathbf{P} \mathbf{x}_k[\Delta\tau_j] \right\|_2^2 + \frac{\lambda}{2} \sum_{k=1}^K \|\mathbf{h}_k\|_2^2 \quad (3)$$

式中: \mathbf{P} 表示 $D \times T$ 的二值矩阵, T 表示扩充后训练样本单通道的维度, 其中 $T \gg D$; $\mathbf{x}_k \in \mathbf{R}^D$ 表示第 k 个通道训练样本。由于 $T \gg D$, 所以滤波器 \mathbf{h} 的尺寸远小于训练样本 \mathbf{x} 的尺寸。

2 频域优化求解

2.1 频域投影

式(3)的求解计算量为 $O(D^3K^3)$, 不利于跟踪的实时性。通过利用复频域的性质可以有效减少计算量, 式(3)在复频域的表达方式为

$$\begin{aligned} E(\mathbf{h}, \hat{\mathbf{g}}) &= \frac{1}{2} \|\hat{\mathbf{y}} - \hat{\mathbf{X}} \hat{\mathbf{g}}\|_2^2 + \frac{\lambda}{2} \|\hat{\mathbf{h}}\|_2^2 \\ \text{s.t. } \hat{\mathbf{g}} &= \sqrt{T} (\mathbf{F} \mathbf{P}^T \otimes \mathbf{I}_K) \mathbf{h} \end{aligned} \quad (4)$$

式中: 上标 \wedge 运算符代表傅里叶变换, \otimes 运算符表示为克罗内克积, \mathbf{I}_K 为 $K \times K$ 的单位矩阵, \mathbf{F} 为 $T \times T$ 的傅里叶变换系数矩阵, $\hat{\mathbf{g}}$ 表示辅助变量, $\hat{\mathbf{X}} = [\text{diag}(\hat{\mathbf{x}}_1)^T, \dots, \text{diag}(\hat{\mathbf{x}}_K)^T]$, $\mathbf{h} = [\mathbf{h}_1^T, \dots, \mathbf{h}_K^T]^T$, $\hat{\mathbf{g}} = [\hat{\mathbf{g}}_1^T, \dots, \hat{\mathbf{g}}_K^T]^T$ 。

为了求解式(4), 通过添加拉格朗日向量 $\hat{\boldsymbol{\zeta}}$, 并利用增广拉格朗日方法(Augmented Lagrangian method, ALM)得到式(5)所示的增广拉格朗日式。

$$\begin{aligned} L(\hat{\mathbf{h}}, \hat{\mathbf{g}}, \hat{\boldsymbol{\zeta}}) &= \frac{1}{2} \|\hat{\mathbf{y}} - \hat{\mathbf{X}} \hat{\mathbf{g}}\|_2^2 + \frac{\lambda}{2} \|\hat{\mathbf{h}}\|_2^2 + \\ &\hat{\boldsymbol{\zeta}}^T (\hat{\mathbf{g}} - \sqrt{T} (\mathbf{F} \mathbf{P}^T \otimes \mathbf{I}_K) \mathbf{h}) + \\ &\frac{\mu}{2} \|\hat{\mathbf{g}} - \sqrt{T} (\mathbf{F} \mathbf{P}^T \otimes \mathbf{I}_K) \mathbf{h}\|_2^2 \end{aligned} \quad (5)$$

式中: μ 为惩罚系数, 拉格朗日向量 $\boldsymbol{\zeta} = [\hat{\boldsymbol{\zeta}}_1^T, \dots, \hat{\boldsymbol{\zeta}}_K^T]^T$ 大小为 $KT \times 1$, $\hat{\boldsymbol{\zeta}}_k$ 为第 k 个通道的拉格朗日向量的傅里叶变换形式。

2.2 优化求解

由于式(5)引入了裁剪矩阵 \mathbf{P} 导致没有解析解, 把原问题的求解过程转化为求解滤波器 \mathbf{h} 和辅助变量 \mathbf{g} 的两个子问题, 再采用交替方向乘子法(Alternating direction method of multipliers, ADMM)求解此

优化问题。

Subproblem \mathbf{h}^*

$$\begin{aligned} \mathbf{h}^* = \arg \min_{\mathbf{h}} & \left\{ \frac{\lambda}{2} \|\mathbf{h}\|_2^2 + \hat{\boldsymbol{\zeta}}^\top (\hat{\mathbf{g}} - \sqrt{T} (FP^\top \otimes I_K) \mathbf{h}) + \right. \\ & \left. \frac{\mu}{2} \|\hat{\mathbf{g}} - \sqrt{T} (FP^\top \otimes I_K) \mathbf{h}\|_2^2 \right\} = \\ & \left(\mu + \frac{\lambda}{\sqrt{T}} \right)^{-1} (\mu \mathbf{g} + \boldsymbol{\zeta}) \end{aligned} \quad (6a)$$

Subproblem \mathbf{g}^*

$$\begin{aligned} \mathbf{g}^* = \arg \min_{\mathbf{g}} & \left\{ \frac{1}{2} \|\hat{\mathbf{y}} - \hat{\mathbf{X}} \hat{\mathbf{g}}\|_2^2 + \right. \\ & \hat{\boldsymbol{\zeta}}^\top (\hat{\mathbf{g}} - \sqrt{T} (FP^\top \otimes I_K) \mathbf{h}) + \\ & \left. \frac{\mu}{2} \|\hat{\mathbf{g}} - \sqrt{T} (FP^\top \otimes I_K) \mathbf{h}\|_2^2 \right\} \end{aligned} \quad (6b)$$

式(6a)中: $\mathbf{g} = \frac{1}{\sqrt{T}} (PF^\top \otimes I_K) \hat{\mathbf{g}}, \boldsymbol{\zeta} = \frac{1}{\sqrt{T}} (PF^\top \otimes I_K) \hat{\boldsymbol{\zeta}}$,其计算量为 $\mathcal{O}(KT \log(T))$ 。式(6b)的计算量 $\mathcal{O}(T^3 K^3)$,不能满足跟踪实时性要求。但是由于 $\hat{\mathbf{X}}$ 是稀疏的,可以把子问题 \mathbf{g} 拆分为 T 个独立的目标函数来求解

$$\begin{aligned} \hat{\mathbf{g}}(t)^* = \arg \min_{\hat{\mathbf{g}}(t)} & \left\{ \frac{1}{2} \|\hat{\mathbf{y}}(t) - \hat{\mathbf{x}}(t) \hat{\mathbf{g}}(t)\|_2^2 + \right. \\ & \hat{\boldsymbol{\zeta}}(t)^\top (\hat{\mathbf{g}}(t) - \hat{\mathbf{h}}(t)) + \\ & \left. \frac{\mu}{2} \|\hat{\mathbf{g}}(t) - \hat{\mathbf{h}}(t)\|_2^2 \right\} \end{aligned} \quad (7)$$

式中: $t = 1, \dots, T, \hat{\mathbf{x}}(t) = [\hat{x}_1(t), \dots, \hat{x}_K(t)]^\top, \hat{\mathbf{g}}(t) = [\text{conj}(\hat{g}_1(t)), \dots, \text{conj}(\hat{g}_k(t))]^\top$, $\text{conj}(\bullet)$ 运算符表示复共轭运算, $\hat{\mathbf{h}}(t) = [\hat{h}_1(t), \dots, \hat{h}_K(t)]^\top, \hat{\mathbf{h}}_K(t) = \sqrt{D} FP^\top \mathbf{h}_{k_0}$ 。利用谢尔曼-莫里森公式求得式(7)的解为

$$\begin{aligned} \hat{\mathbf{g}}(t)^* = & \frac{1}{\mu} (T\hat{\mathbf{y}}(t)\hat{\mathbf{x}}(t) - \hat{\boldsymbol{\zeta}}(t) + \mu\hat{\mathbf{h}}(t)) - \\ & \frac{\hat{\mathbf{x}}(t)}{\mu b} (T\hat{\mathbf{y}}(t)\hat{s}_x(t) - \hat{s}_\zeta(t) + \mu\hat{s}_h(t)) \end{aligned} \quad (8)$$

式中: $\hat{s}_x(t) = \hat{\mathbf{x}}(t)^\top \hat{\mathbf{x}}(t), \hat{s}_\zeta(t) = \hat{\mathbf{x}}(t)^\top \hat{\boldsymbol{\zeta}}, \hat{s}_h(t) = \hat{\mathbf{x}}(t)^\top \hat{\mathbf{h}}, b = \hat{s}_x(t) + T\mu$ 。式(8)计算量为 $\mathcal{O}(TK)$ 。

通过上述求解过程可知,计算量主要集中在两个子问题的优化过程。为了得到尺度变化,原BACF算法针对 N 个尺度求解 N 次优化问题将耗费大量的计算时间,因此原BACF算法跟踪速度较低。

3 尺度滤波器融合

上述介绍了BACF算法的建模及求解过程,解决了跟踪过程中的边界效应,可以精准估计目标位置变化。由于目标运动过程中还会伴随着尺度变化,如果不能精准且快速估计目标尺度变化仍会导致跟踪器不够完善。不同于原BACF算法利用平移滤波器在多尺度缩放的目标区域进行检测,需要对每个尺度做一次相关运算求解,导致消耗大量计算时间。本文将采用平移加尺度滤波的思想,额外训练一个一维尺度滤波器与平移滤波实现无缝融合,精准快速预测目标尺度变化。

3.1 尺度滤波器模型

尺度滤波器的建模与1.1节中传统核相关滤波器类似,通过计算尺度滤波器 h_s 与训练样本 f 的相关运算结果,然后计算与期望输出 g_s 的均方误差,最后通过均方误差最小化求得最优尺度滤波器 h_s ,如式(9)所示。

$$\varepsilon = \left\| g_s - \sum_{l=1}^d h_s^l \circ f^l \right\|^2 + \lambda_s \sum_{l=1}^d \|h_s^l\|^2 \quad (9)$$

式中:上标 l 代表第 l 个通道, λ_s 为正则化常数, g_s 表示一维的高斯输出。式(9)是一个最小线性二乘问题,通过帕斯瓦尔定理转移到频域可以快速求解得到

$$H_s^l = \frac{\overline{G}_s F^l}{\sum_{k=1}^d \overline{F}^k F^k + \lambda_s} \quad l=1, \dots, d \quad (10)$$

式中: H_s, G_s, F 分别代表滤波器 h_s ,期望输出 g_s 和训练样本 f 的傅里叶变换。求得滤波器 h_s 后,采用如下更新策略进行更新

$$\begin{aligned} A_t^l &= (1 - \eta) A_{t-1}^l + \eta \overline{G} F_t^l \quad l=1, \dots, d \\ B_t &= (1 - \eta) B_{t-1} + \eta \sum_{k=1}^d \overline{F}^k F_t^k \end{aligned} \quad (11)$$

式中: A_t, B_t 分别代表式(10)的分子、分母部分, η 代表学习速率。式(10)给出了尺度滤波器 h_s 的表达式,应用该滤波器到当前第 t 帧目标区域上进行相关运算将得到尺度响应输出 y_s, y_s 在复频域的表达式为

$$Y_s = \frac{\sum_{l=1}^d \overline{A}_{t-1}^l Z_t^l}{B_{t-1} + \lambda_s} \quad (12)$$

式中: d 代表特征维度,下标 $t, t-1$ 分别代表当前帧和上一帧, Z_t^l 代表当前帧目标区域 z_t^l 的傅里叶变换,可以利用傅里叶反变换求得响应输出 $y_s, y_s = \mathcal{F}^{-1}\{Y_s\}$ 。设 $P \times R$ 表示目标在第 $t-1$ 帧尺度的宽和高,通过找到 y_t 最大值所在的位置为 n ,则目标在第 t 帧尺度为: $a^n P \times a^n R, a$ 为尺度因子。

3.2 尺度划分与降维

上节给出了尺度输出响应的表达式 $y_s = \mathcal{F}^{-1}\{Y_s\}$,其中 Y_s 如式(12)所示。设 $P \times R$ 表示目标在前一帧的尺度大小, S 表示划分尺度的个数。通过对目标中心区域选取尺寸为 $a^n P \times a^n R$ 的训练样本 f_n ,其中 a 表示尺度因子,上标 $n \in \left\{ \left\lfloor -\frac{S-1}{2} \right\rfloor, \dots, \left\lfloor \frac{S-1}{2} \right\rfloor \right\}$ 。本文选取尺度个数 S 为17,找出尺度响应输出 y_s 中最大元素所在的位置 i ,确定尺度系数为 a^{n_i} ,其中 $n_i = \frac{-S-1}{2} + i$,则目标预测的尺度大小为 $a^{n_i} P \times a^{n_i} R$ 。

由于尺度系数 a^n 是指数函数,为非线性增长,可以实现对较大尺度进行粗检测,对较小尺度进行细检测。通过上述分析可知,仅通过对式(12)进行一次求解就能够预测目标的尺度变化。不同于式2.2节中平移滤波器需要大量优化迭代求解预测位置变化,式(12)具备解析解,能够快速求解尺度输出响应。如图1所示,图1(a)为原图,跟踪目标为行人;图1(b)利用平移滤波器在不同尺度上进行计算得到预测的尺度加位置,所需时间 $T_{\text{total}} = 5 \times T$ (T 为一次平移滤波器计算的时间,假设有5个尺度);图1(c)在利用平移滤波器预测得到位置的基础上,再利用尺度滤波器预测目标尺度变化,所需时间 $T_{\text{total}} = T + T_s$ (T 为一次平移滤波器计算的时间, T_s 为预测尺度所需的时间)。由前面分析可知 $T_s \ll T$,所以平移加滤波

的思想能够节省大量运算时间。

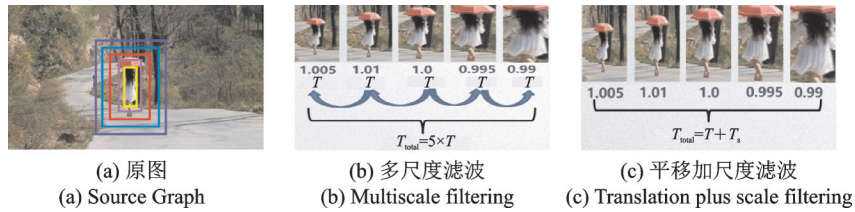


图1 尺度检测对比图

Fig.1 Scale detection comparison

式(12)具备解析解可以快速预测目标的尺度变化,由于目标每一维的特征维度大小为 $P \times R$,但是需要划分的特征尺度个数为 S ,且 $P \times R \gg S$,所以不同特征尺度之间有着极强的相关性,导致数据冗余,增添不必要的计算负担。为了削弱数据特征相关性,减小数据冗余,本文利用主成分分析法(Principal component analysis, PCA^[13])把特征向量降维为 S 维,再进行计算,从而极大减小了特征向量的维度,从而提高尺度检测速度。

4 实验结果与分析

为了有效评估本文算法的性能,实验中采用了VOT2014数据集,该数据集共有50段不同类型的跟踪场景,包含了光照变化、快速运动、运动模糊、尺度变化、背景相似干扰等常见的跟踪问题,同时采用视觉跟踪中最常用的3种评估方法来评估本文算法的性能指标:中心位置误差法(Center location error, CLE),距离精度法(Distance precision, DP)和重叠精度法(Overlap precision, OP)。CLE评价方法即

计算目标预测位置 (x_p, y_p) 和真实位置 (x_r, y_r) 的平均欧式距离 $d = \frac{\sum_{p,r} \sqrt{(x_p - x_r)^2 + (y_p - y_r)^2}}{N}$, N 为样本帧数;DP评价方法即计算CLE数据中小于一定阈值的帧数占总帧数的百分比,本实验取该阈值为20像素;OP评价方法即计算目标预测位置及大小与实际位置及大小的重叠率大于一定阈值的帧数占总帧数的百分比,该阈值一般选取PASCAL^[14]的评价标准为0.5。同时为了直观对比本文算法的性能,本文选取了近几年优秀的几种算法做对比实验,包括以尺度变换为代表的DSST算法,解决边界效应的BACF算法,以循环移位操作和核技巧代表的KCF算法。各算法选取的特征均为FHOG^[4]特征。

4.1 实验环境及参数设置

本实验的硬件平台为Inter(R) Core(TM) i7 2.90 GHz 处理器,8 GB 内存,软件平台为Matlab R2015a。第2节和第3节均选取FHOG特征作为目标的表现形式,FHOG特征cell大小为4,梯度直方图方向个数为9,2.2节中的ADMM优化过程中迭代次数设为2,惩罚因子 μ 设为1,本文的正则化系数 λ 和 λ_s 为0.001,第3节更新速率 η 为0.025,3.2节中尺度因子 a 为1.02。

4.2 定性分析

为了方便显示与对比算法的性能优劣,本实验首先选取实验视频集中6类代表不同特性的视频序列做定性分析,其中每个视频序列对应如表1所示的特性。

为了显示对比算法与本文算法的跟踪结果,选取表1中每个视频序列的3帧照片,如图2所示,其中图1(a)中不同颜色框代表不同跟踪算法。

表 1 视频序列
Table 1 Video Sequence

视频名称	帧数	视频主要特性	视频名称	帧数	视频主要特性
Coke	291	完全遮挡, 物体旋转	Woman	597	运动模糊, 部分遮挡
Couple	140	视频抖动, 低分辨率	Jumping	313	运动模糊, 姿态变化
Tiger1	354	部分遮挡, 姿态变化	Basketball	725	快速运动, 姿态变化

从图 2 可以直观看出本文算法有较高的鲁棒性, 在目标经过遮挡、形变、运动模糊之后能持续跟准目标, KCF 算法和 DSST 算法都有不同程度上的偏移, BACF 算法鲁棒性也很高, 但从 Basketball 和 Coke 序列中可以看出, 经过姿态变化和大面积遮挡之后由于本文算法独立的尺度检测策略使得跟踪器能更好捕捉到目标的尺度变化, 进而学习到更合适的背景与目标的占比, 所以本文算法较 BACF 算法有更高的鲁棒性。

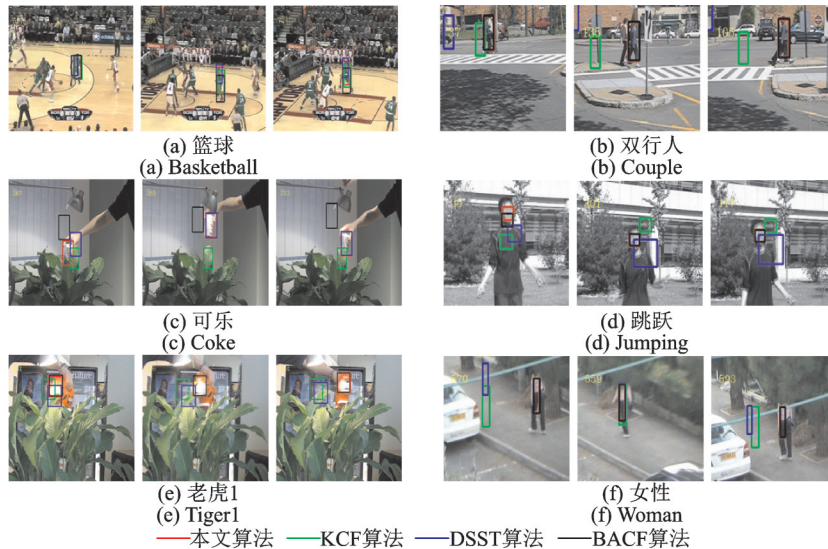


图 2 定性分析图

Fig.2 Qualitative analysis

4.3 定量分析

通过 4.2 的定性分析, 可以直观粗略看出各算法性能, 接下来将进行更细致的分析, 通过计算上述提到的 3 种性能评估指标 (CLE、DP、OP) 来定量对比各算法性能差异, 具体数据如表 2 所示。表 2 加粗部分数据为各对比算法中最优性能数据, 从表 2 可以看出本文算法和 BACF 算法在 CLE、DP、OP 3 个指标中都能取得很好的成绩; 从最后一行均值数据可以看出本文算法性能略优于 BACF 算法, 而 KCF^[3] 算法和 DSST 算法效果欠佳。

表 2 只给出了在 20 个像素阈值下的 DP 值, 为了看出各对比算法的 DP 变化曲线以了解各跟踪器的鲁棒性, 图 3 给出了在 1~50 个像素阈值下各个算法的 DP 曲线。从图 3 可以直观看出随着阈值的增加本文算法和 BACF 法 DP 曲线值趋近于 1, 性能明显强于 KCF 和 DSST^[10] 算法。

在上述实验中为了具体直观对比显示只取了其中 6 个代表性视频序列做实验。为了不失一般性,

表2 多种不同方法的性能指标对比

Table 2 Multi algorithm performance comparison

视频名称	CLE/像素				DP/%				OP/%			
	BACF	KCF	DSST	本文	BACF	KCF	DSST	本文	BACF	KCF	DSST	本文
Basketball	7.87	7.89	10.09	5.32	80.83	92.30	80.60	92.00	80.28	90.10	68.00	90.50
Coke	20.47	18.70	12.80	12.00	86.60	83.18	91.80	93.13	75.95	72.90	82.80	82.13
Couple	3.66	47.60	126.00	3.71	100.00	25.70	10.70	100.00	89.29	24.30	10.70	86.43
Jumping	4.35	26.12	36.87	4.37	98.72	33.87	5.10	98.70	92.65	28.10	4.79	95.21
Tiger1	10.46	47.40	41.40	10.36	91.78	35.30	29.60	90.96	91.51	36.40	29.60	92.05
Woman	2.41	10.10	9.96	2.47	99.83	93.80	93.80	99.83	93.30	93.60	93.30	93.30
均值	8.20	26.30	39.52	6.37	92.96	60.69	51.93	95.77	87.25	57.57	48.20	89.49

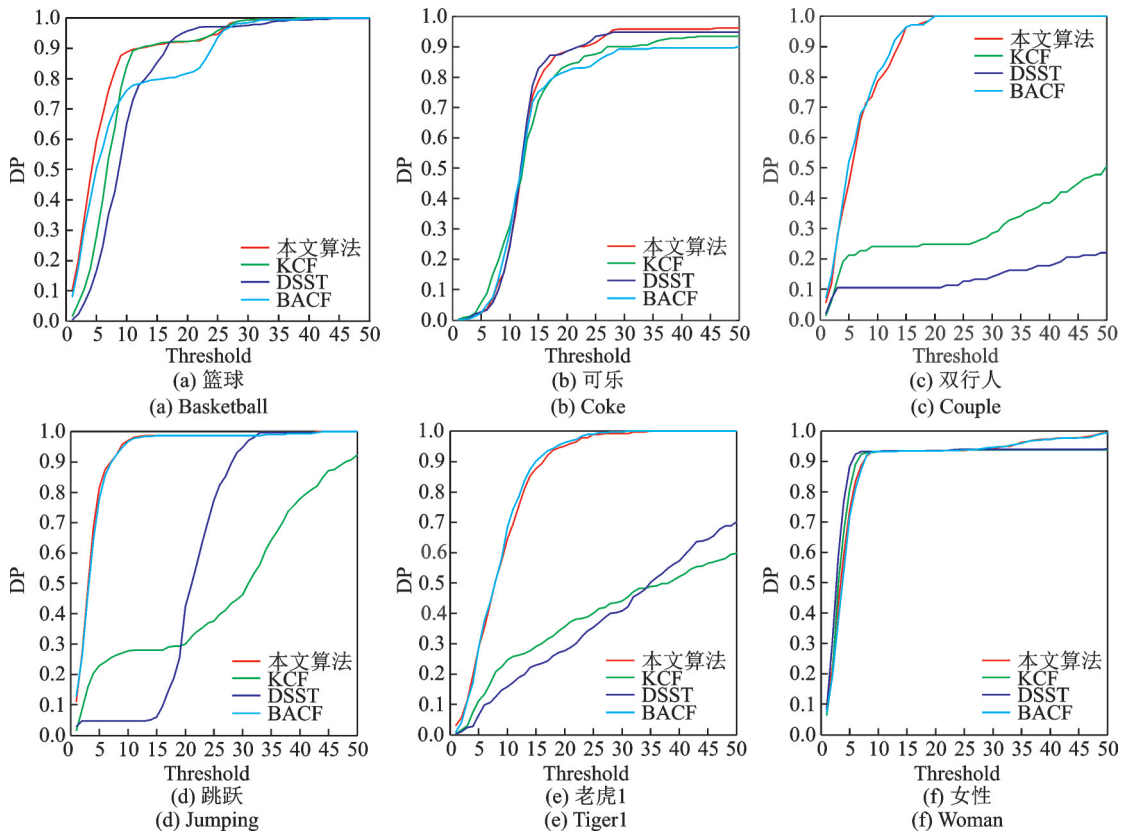


图3 在1~50个像素阈值下各个算法的DP曲线

Fig.3 DP curves of different algorithms under 1—50 thresholds

本文对数据集中所有50个视频序列做了对比实验并求出CLE, DP, OP值。同时为了便于查看,给出了各算法3个性能指标均值,实验结果数据如表3所示。

从表3可以看出,在CLE和OP性能均值指标上本文算法略低于BACF算法1个像素和1个百分点,在DP性能均值指标上高于BACF算法2.6个百分点,KCF和DSST算法性能均远弱于本文算法和BACF算法。

表3 性能平均指标对比

Table 3 Performance average index comparison

性能指标	实验算法			
	KCF	DSST	BACF	本文算法
CLE/像素	32.34	38.05	27.76	28.79
DP/%	76.35	75.52	80.27	82.89
OP/%	65.87	69.24	79.54	78.53

4.4 速度对比

本算法的最大优势在于不损失跟踪精度的前提下,极大地提升算法的跟踪速度,前几节主要以CLE,DP,OP三个性能指标来分析跟踪精度,可以看出本文算法与BACF算法在跟踪精度上差异不大,在CLE和OP性能均值指标上略低于BACF算法1个像素和1个百分点,在DP性能均值指标上高于BACF算法2.6个百分点。接下来通过实验数据分析各算法的跟踪速度,由于同一算法的跟踪速度在不同目标跟踪过程中的差异性并不大,本实验通过将表1中6个视频序列的跟踪速度的平均值作为最终结果再作对比,结果如表4所示。

表4 跟踪速度对比

Table 4 Tracking speed comparison

视频名称	实验算法			
	KCF/(f·s ⁻¹)	DSST/(f·s ⁻¹)	BACF/(f·s ⁻¹)	本文算法/(f·s ⁻¹)
Basketball	125	19	12	21
Coke	110	14	10	20
Couple	137	31	10	20
Jumping	203	40	15	24
Tiger1	93	11	11	18
Woman	107	28	13	22
均值	129	24	12	21

从表4的跟踪速度可以看出,KCF算法的跟踪速度最快达到129 f/s,但是从前几节的对比实验可知其跟踪精度远低于本文算法和BACF算法。在跟踪速度上本文算法为21 f/s,而BACF算法为12 f/s,速度超出BACF算法75%,完全可以满足跟踪的实时性要求并且并不损失跟踪精度。

5 结束语

本文通过在传统BACF算法基础上融合尺度滤波器,既利用BACF算法的优势解决边界效应,精准估计目标位置变化,又利用尺度滤波器快速精准估计目标尺度变化,在不损失跟踪精度的同时极大提升了目标的跟踪速度。通过对比实验可以看出,本文算法能达到既快又好的跟踪效果。但是在跟踪过程中还有难点亟待解决,比如:长时间跟踪模板污染的问题,姿态、快速运动模板更新不及时,目标丢失再跟踪等,将是今后的研究方向。

参考文献:

- [1] BOLME D S, BEVERIDGE J R, DRAPER B A, et al. Visual object tracking using adaptive correlation filters[C]//

- Proceedings of 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. [S.l.]: IEEE, 2010: 2544-2550.
- [2] HENRIQUES J F, CASEIRO R, MARTINS P, et al. Exploiting the circulant structure of tracking-by-detection with kernels [C]//Proceedings of the European Conference On Computer Vision. Berlin, Heidelberg: Springer, 2012: 702-715.
- [3] 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.
- [4] 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, 2010, 32(9): 1627-1645.
- [5] DANELLJAN M, SHAHBAZ KHAN F, FELSBERG M, et al. Adaptive color attributes for real-time visual tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l.]: IEEE, 2014: 1090-1097.
- [6] VAN DE WEIJER J, SCHMID C, VERBEEK J, et al. Learning color names for real-world applications[J]. IEEE Transactions on Image Processing, 2009, 18(7): 1512-1523.
- [7] DANELLJAN M, ROBINSON A, KHAN F S, et al. Beyond correlation filters: Learning continuous convolution operators for visual tracking[C]//Proceedings of the European Conference on Computer Vision. Cham: Springer, 2016: 472-488.
- [8] DANELLJAN M, BHAT G, SHAHBAZ KHAN F, et al. Eco: Efficient convolution operators for tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l.]: IEEE, 2017: 6638-6646.
- [9] GUO Q, FENG W, ZHOU C, et al. Learning dynamic siamese network for visual object tracking[C]//Proceedings of the IEEE International Conference on Computer Vision. [S.l.]: IEEE, 2017: 1763-1771.
- [10] GALOOGAHI H K, FAGG A, LUCEY S. Learning background-aware correlation filters for visual tracking[C]//Proceedings of the IEEE International Conference on Computer Vision. [S.l.]: IEEE, 2017: 1135-1143.
- [11] DANELLJAN M, HÄGER G, KHAN F, et al. Accurate scale estimation for robust visual tracking[C]//Proceedings of the British Machine Vision Conference. Nottingham: BMVA Press, 2014.
- [12] KUMAR A, ZHANG D. Personal authentication using multiple palmprint representation[J]. Pattern Recognition, 2005, 38(10): 1695-1704.
- [13] SHLENS J. A tutorial on principal component analysis[M]. New york: New york University, 2014.
- [14] EVERINGHAM M, VAN GOOL L, WILLIAMS C K I, et al. The pascal visual object classes (VOC) challenge[J]. International Journal of Computer Vision, 2010, 88(2): 303-338.

作者简介:



王永雄(1970-),男,教授,博士生导师,研究方向:机器学习 and 机器人智能算法、机器视觉和图像处理, E-mail: wyxiong@usst.edu.cn。



冯汉(1992-),男,硕士研究生,研究方向:机器视觉、目标检测与跟踪, E-mail: 167760672@st.usst.edu.cn。

(编辑:夏道家)