

基于通道权重融合特征的目标跟踪算法^①



郭利¹, 周盛宗²

¹(福建师范大学 数学与信息学院, 福州 350117)

²(中国科学院 福建物质结构研究所, 福州 350002)

通讯作者: 周盛宗, E-mail: zhousz@fjirsm.ac.cn

摘要: 目标跟踪是机器视觉领域的一个研究热点, 如何提高复杂场景下的跟踪水平是一个挑战性的问题. 以往的研究表明, 如何有效使用特征是实现跟踪的关键. 因此, 提出一种基于通道融合特征的目标跟踪算法. 该方法基于多通道相关滤波框架, 引入特征通道权重, 根据通道对响应值的贡献度调整权重, 从而构建实时特征组合. 该算法能够快速捕捉目标状态变化, 有效跟踪目标. 为了验证算法跟踪的有效性, 我们在公开数据集 OTB-2015 上测试算法性能, 并与多种跟踪算法进行比较. 实验结果显示, 该算法在跟踪精度、成功率上都取得较好的结果, 整体性能优于对比算法.

关键词: 机器视觉; 通道权重; 融合特征; 相关滤波; 目标跟踪

引用格式: 郭利, 周盛宗. 基于通道权重融合特征的目标跟踪算法. 计算机系统应用, 2020, 29(12): 178-186. <http://www.c-s-a.org.cn/1003-3254/7702.html>

Object Tracking Algorithm with Fusion Feature Based on Channel Weight

GUO Li¹, ZHOU Sheng-Zong²

¹(College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350017, China)

²(Fujian Institute of Research on the Structure of Matter, Chinese Academy of Sciences, Fuzhou 350002, China)

Abstract: Target tracking is a research hotspot in the field of machine vision. How to improve the tracking level in complex scenarios is a challenging problem. Previous studies have shown that how to use features effectively is the key to tracking. Therefore, a target tracking algorithm based on channel fusion features is proposed. Based on the multi-channel correlation filtering framework, the method introduces the feature channel weight, adjusts the weight according to the contribution of the channel to the response value, and then constructs the real-time feature combination. The algorithm can capture the state change of the target quickly and track the target effectively. In order to verify the effectiveness of the algorithm tracking, we test the algorithm performance on the open dataset OTB2015 and compare it with a variety of tracking algorithms. Experimental results show that the algorithm has sound tracking accuracy and success rate, and the overall performance is better than the compared algorithm.

Key words: machine vision; channel weight; fusion features; correlation filter; object tracking

近年来, 目标跟踪在视频监控、运动分析、人机交互、自动驾驶及交通安防等领域获得了长足发展. 但大多数目标跟踪算法不但在面对光照变化、运动模

糊、快速运动、背景混乱、遮挡等复杂环境下跟踪失败, 而且在目标变形、旋转、姿势变化等剧烈变化时也表现欠佳. 因此, 有必要继续探索健壮的目标跟踪算法.

① 基金项目: 中国科学院 STS 计划配套项目 (2019T3008, 2019T3009)

Foundation item: Supporting Project for CAS Science and Technology Service Network Initiative (STS Initiative) (2019T3008, 2019T3009)

收稿时间: 2020-04-28; 修改时间: 2020-05-27; 采用时间: 2020-06-01; csa 在线出版时间: 2020-11-30

以往研究表明,有效的特征一般能较好的表示跟踪目标并快速运算,代表性的有颜色特征^[1]、纹理特征^[2]、灰度特征^[3]、超像素特征^[4]、Lab 特征^[5]等.得益于特征引入频域,将矩阵乘法运算转化为点乘运算,基于相关滤波的目标跟踪算法获得了长足发展.譬如:基于单通道灰度特征的 MOSSE^[6]和 CSK^[7]、基于 11 通道颜色特征的 CN^[8]、基于 31 通道 HOG 特征的 KCF^[9]、基于 42 通道特征(31 维 HOG 特征、10 维颜色特征、1 维灰度特征)的 SAMF^[10]、基于 HOG 特征和颜色特征的 Staple^[11]、基于 HOG 特征和 Lab 特征的特征 SACFT^[12]等.这些算法一般将不同特征简单叠加或以固定比例组合,未考虑实时特征组合对目标跟踪的影响.针对该问题,赵高鹏等^[13]基于峰值旁瓣比融合灰度特征和 LBP 特征,陈法领等^[14]基于峰值旁瓣比和相邻帧间的响应一致性融合纹理特征和颜色特征,尹宽等^[15]基于融合特征器的可信度选择合适线性组合方式,常敏等^[16]基于平均峰值-相关能量方法融合不同特征.与几位学者从特征类别层面融合特征的做法不同,本文算法从特征通道层面融合特征,该算法通过各通道对响应值的贡献度调整权重,构建实时特征组合,进而控制特征融合,从而提升跟踪效果.

本文算法的亮点主要表现在以下几个方面:(1)引入通道权重矩阵,不破坏循环矩阵性质,不影响闭合求解;(2)先对特征矩阵和权重矩阵点乘,不影响框架将矩阵乘运算转化到傅里叶域下求点乘;(3)基于各通道响应值构建权重更新策略,及时捕捉目标运动状态,灵活构建特征组合.

相关实验结果表明:通道权重在特征融合过程中,发挥着很大的作用;本文提出的特征融合算法在跟踪目标时能有效提升精确度和成功率.

1 基于通道权重融合特征的目标跟踪算法

为了能够使用多维度特征,Galoogahi 等^[17]提出多通道相关滤波算法.该算法设定:

$$E(h) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^D \left\| y_i(j) - \sum_{k=1}^K h^{(k)T} * x_i^k [\Delta\tau_j] \right\|_2^2 + \frac{\lambda}{2} \sum_{k=1}^K \|h^{(k)}\|_2^2 \quad (1)$$

式中, N 是样本数目, D 是循环移位的位数, $y_i(j)$ 是期望相关响应的第 j 个元素, K 是特征通道数目, $h^{(k)}$ 是第

k 通道的过滤器, x_i^k 是第 i 帧第 k 通道的向量化, $\Delta\tau_j$ 表示循环位移操作, $x_i^k [\Delta\tau_j]$ 是对 x_i^k 做 j 步的循环位移, λ 是规范化系数.

在跟踪过程中,各个通道对响应值的贡献度不同,而该算法将各通道等同看待.因此,提出基于通道融合特征的目标跟踪算法.该算法的目标函数如下:

$$E(h) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^D \left\| y_i(j) - \sum_{k=1}^K h^{(k)T} * (ratio_i^k * x_i^k) [\Delta\tau_j] \right\|_2^2 + \frac{\lambda}{2} \sum_{k=1}^K \|h^{(k)}\|_2^2 \quad (2)$$

式中, N 、 D 、 $y_i(j)$ 、 K 、 $ratio_i^k$ 、 $h^{(k)}$ 、 x_i^k 、 $\Delta\tau_j$ 、 λ 同式(1)中变量意义相同, $(ratio_i^k * x_i^k) [\Delta\tau_j]$ 是对 $(ratio_i^k * x_i^k)$ 做 j 步的循环位移.

1.1 引入特征权重

与式(1)相比,式(2)中添加了 $ratio_i^j$.该项通过各通道响应值占比(见式(3))构建,与 $x_i^k [\Delta\tau_j]$ 相乘实现自适应特征组合,一定程度上发挥着特征权重的作用.具体实现如下:

$$ratio_i^k = \begin{cases} f_i^k / \sum_{k=1}^K f_i^k, & \sum_{k=1}^K f_i^k > 0, 0 < j < K+1 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

式中, $ratio_i^k$ 是第 i 帧第 k 通道的响应值占比, f_i^k 表示第 i 帧第 k 通道的响应值, K 是特征通道数目, $\sum_{k=1}^K f_i^k$ 表示第 i 帧的所有通道响应值之和.

1.2 引入区域评估机制

为保证 $ratio_i^j$ 的准确性,对式(3)中的 f_i^k 引入区域评估机制.该机制基于一个前提和一个区域特性.该前提设定如下:

$$(pos_m, pos_n) = \arg \max_{(pos_m, pos_n)} \sum_{k=1}^K f_i^k(m, n), \quad (4)$$

$$0 < m < M+1, 0 < n < N+1$$

式中, (pos_m, pos_n) 是最佳坐标位置, M 是搜索区域的宽度, N 是搜索区域的高度, $f_i^k(m, n)$ 是第 i 帧第 k 通道 (m, n) 处的响应值.

由于 $\sum_{k=1}^K f_i^k(pos_m, pos_n)$ 最大,并不能保证 $f_i^k(pos_m, pos_n)$ 在第 i 帧第 k 通道上也是最大值.但是,如果 f_i^k

(pos_m, pos_n) 在第 i 帧第 k 通道上, 小于甚至等于 (pos_m, pos_n) 周围 8 个位置上的最大响应值, 那么该通道上的位置选择一定是错误的. 所以, 该通道上的响应值 $f_i^k(pos_m, pos_n)$ 对特征组合的贡献度应该为零, 这就是有效通道的区域特性. 而该特性也在一定程度上保证了模型推测的位置至少是在各个通道上是局部最优, 也就是 $f_i^k(pos_m, pos_n)$ 大于 (pos_m, pos_n) 周围 8 个位置上的最大响应值. 该区域特性具体实现如下:

$$f_i^k = \begin{cases} f_{\max} = \arg \max(f_i^k(pos_{m-1:m+1}, pos_{n-1:n+1})) \\ f_i^k(pos_m, pos_n), f_i^k(pos_m, pos_n) > f_{\max} \\ 0, \text{ otherwise} \end{cases} \quad (5)$$

式中, f_i^k 表示第 i 帧第 k 通道的响应值, (pos_m, pos_n) 表示模型推测的位置, $(pos_{m-1:m+1}, pos_{n-1:n+1})$ 表示 (pos_m, pos_n) 的上、下、左、右、左上、右上、左下和右下共 8 个位置, f_{\max} 表示第 i 帧第 k 通道 (pos_m, pos_n) 周围 8 个位置的最大响应值.

1.3 引入滑动窗口机制

式 (2) 的 $(ratio_i^k * x_i^k) [\Delta \tau_j]$ 将被拓展转化为 X_β (见式 (7) 有关说明), 构成模型 $\hat{\alpha}_i$ 的组成部分. 而 $\hat{\alpha}_i$ 的更新被设定为线性递增, 为了保持 $ratio_i^k$ 与 $\hat{\alpha}_i$ 的同步, $ratio_i^k$ 也被设定为线性递增.

由于式 (5) 中的 f_i^k 均大于等于 0, 造成 $ratio_i^k$ 均大于等于 0. 长期跟踪过程中, 一些通道的权重值将迅速累加, 势必造成模型严重依赖个别通道, 这对模型非常不利.

为缓解这一问题, 引入滑动窗口机制. 该机制定模型仅依据当前帧及其以前的部分帧, 来更新通道权重. 在减缓通道特征累加风险的同时, 捕捉最近的模型变化, 舍弃较久远的变化. 该机制设定如下:

$$ratio_{I+1}^k = \begin{cases} ratio_0^k + ratio^w * \sum_{i=1}^I ratio_i^k, 0 < I < n + 1 \\ ratio_0^k + ratio^w * \sum_{i=I-99}^I ratio_i^k, n < I \end{cases} \quad (6)$$

式中, $ratio_{I+1}^k$ 是第 $I+1$ 帧第 k 通道的权重, $ratio_0^k$ 是第 k 通道的初始值, $ratio^w$ 是权重学习率, $ratio_i^k$ 是第 i 帧第 k 通道的权重, n 是参与权重更新的帧数目.

2 算法实现

目标函数式 (2) 可以化简为:

$$\frac{1}{2} \|y - X_\beta \omega\|_2^2 + \frac{\lambda}{2} \|\omega\|_2^2 \quad (7)$$

式中, y 是回归目标 y_i 的集合, X_β 是 $((X_\beta^1)^T (X_\beta^2)^T \dots (X_\beta^k)^T \dots (X_\beta^k)^T)^T$, (X_β^k) 是 $(ratio_i^k * x_i^k) [\Delta \tau_j]$ 的循环矩阵, ω 是线性回归方程的投影矩阵.

2.1 求解线性回归

求解式 (7) 得:

$$\omega = \left((X_\beta)^T X_\beta + \lambda I \right)^{-1} (X_\beta)^T y \quad (8)$$

式中, X 、 X_β 、 y 同式 (7) 中对应变量意义相同, I 是单位矩阵.

在复数空间的傅里叶域下:

$$\omega = \left((X_\beta)^H X_\beta + \lambda I \right)^{-1} (X_\beta)^H y \quad (9)$$

式中, X^H 是 X 的复共轭转置, $X^H = (X^*)^T$.

2.2 求解非线性回归

引入核机制^[9], 将线性问题的输入映射到非线性空间 $\varphi(x)$, 则 ω 可以表示为:

$$\omega = \sum_i \alpha_i \varphi(x_i) \quad (10)$$

式中, α 是对偶空间.

将 $f(z) = \omega^T z = \sum_i \alpha_i k(z, x_i)$ 带入式 (7), 求解得:

$$\alpha = (K + \lambda I)^{-1} y \quad (11)$$

式中, $K_{ij} = k((x_\beta)_i, (x_\beta)_j) = \varphi^T((x_\beta)_i) \varphi((x_\beta)_j)$.

在傅里叶域下, 有:

$$\hat{\alpha} = \frac{\hat{y}}{\hat{k}^{x_\beta x_\beta} + \lambda} \quad (12)$$

式中, $\hat{\cdot}$ 表示对应的离散傅里叶变换.

2.3 核函数选择

选择高斯核^[9]:

$$k^{x_\beta x_\beta'} = \exp \left(-\frac{1}{\delta^2} (\|x_\beta\|^2 + \|x_\beta'\|^2) - 2\mathcal{F}^{-1} \left(\sum_{j=1}^C (\hat{x}_\beta^j)^* \odot (\hat{x}_\beta'^j) \right) \right) \quad (13)$$

选择线性核^[9]:

$$k^{x_\beta x_\beta'} = \exp \left(-2\mathcal{F}^{-1} \left(\sum_{j=1}^C (\hat{x}_\beta^j)^* \odot (\hat{x}_\beta'^j) \right) \right) \quad (14)$$

式中, C 表示特征通道数目.

2.4 检测阶段

选择高斯核^[9]:

$$k^{x_\beta z_\beta} = \exp \left(-\frac{1}{\delta^2} (\|x_\beta\|^2 + \|z_\beta\|^2) - 2\mathcal{F}^{-1} \left(\sum_{j=1}^C (\hat{x}_\beta^j)^* \odot (\hat{z}_\beta^j) \right) \right) \quad (15)$$

选择线性核^[9]:

$$k^{x\beta z\beta} = \exp\left(-2\mathcal{F}^{-1}\left(\sum_{j=1}^C (\hat{x}_\beta^j)^* \odot \hat{z}_\beta^j\right)\right) \quad (16)$$

其中, z 为样本的特征矩阵, $z_\beta = z \odot \beta$, z_β^j 表示对 z 的第 j 通道施加特征权重, C 表示特征通道数目.

$$f(z) = \mathcal{F}^{-1}\left(\hat{k}^{x\beta z\beta} \odot \hat{a}\right) = \mathcal{F}^{-1}\left(\frac{\hat{k}^{x\beta z\beta} \odot \hat{y}}{\hat{k}^{x\beta} x_\beta + \lambda}\right) \quad (17)$$

式中, $f(z)$ 表示样本特征 z 的响应值, \mathcal{F}^{-1} 表示反傅里叶变换, $\hat{\cdot}$ 表示对应的离散傅里叶变换.

2.5 更新阶段

根据式 (3)、式 (5)、式 (6) 更新 $ratio_i^k$, 并将 $ratio_i^k$ 代入式 (12) 得到 \hat{a}_i , 进一步更新模型:

$$\hat{a}_i = (1 - lr) * \hat{a}_{i-1} + lr * \hat{a}_i \quad (18)$$

式中, \hat{a}_i 是跟踪第 i 帧后更新的模型, \hat{a}_{i-1} 是跟踪第 $i-1$ 帧后更新的模型, lr 是模型学习率.

2.6 算法流程

具体算法流程如算法 1 所示.

算法 1. 本文算法

输入: 视频序列 $Frame_1, Frame_2, \dots, Frame_N$, 回归目标 $y \in R^{m \times n}$, 初始帧的目标中心位置 cen_0 .

输出: 当前帧 $curframe$ 的目标中心位置 $cen_{curframe}$.

For 1:N

IF $frame > 0$

① 在前一帧 $preframe$ 目标中心位置 $cen_{preframe}$ 的周围, 划定检测区域并采集样本;

② 对采集到的样本提取纹理特征 $Z_{hog} \in R^{m \times n \times 31}$ 、颜色特征 $Z_{color} \in R^{m \times n \times 10}$ 和灰度特征 $Z_{grey} \in R^{m \times n \times 1}$, 对特征 $Z_{cur} = [Z_{hog}, Z_{color}, Z_{grey}] \in R^{m \times n \times 42}$ 执行 $Z_\beta = Z_{cur} \odot \beta$ 操作, 再对 Z_β 做 FFT, 得到频域特征 \hat{Z}_β ;

③ 使用高斯核时, 利用式 (13)、式 (15)、式 (17) 求响应值 $f(z)$; 使用线性核时, 利用式 (14)、式 (16)、式 (17) 求响应值 $f(z)$;

④ $cen_{curframe} = \arg \max_{(pos_m, pos_n)} (f(z))$

End

⑤ 利用式 (3)、式 (5)、式 (6) 更新通道权重;

⑥ 使用高斯核时, 利用式 (15)、式 (12)、式 (18) 更新外观模型; 使用线性核时, 利用式 (16)、式 (12)、式 (18) 更新外观模型;

End

3 实验结果与分析

3.1 实验数据

实验选取了公开数据集 OTB-2015^[18] 的 100 组视频序列, 围绕光照变化 (Illumination Variation, IV)、尺度变化 (Scale Variation, SV)、遮挡 (Occlusion, OCC)、变形 (Deformation, DEF)、运动模糊 (Motion Blur, MB)、快速运动 (Fast Motion, FM)、平面内旋转 (In-Plane

Rotation, IPR)、平面外旋转 (Out-of-Plane Rotation, OPR)、超出视野范围 (Out-of-View, OV)、背景杂乱 (Background Clutters, BC)、低分辨率 (Low Resolution, LR) 等 11 种挑战情况进行展开.

3.2 评价指标

本文主要用到 3 个评价指标, 分别是精确度 (precision)、成功率 (success rate) 和跟踪速度 (speed). 精确度是在跟踪序列中准确跟踪到的帧所占的比重, 这些帧的目标中心与真实位置的距离小于某个阈值, 该指标越大越好; 成功率是在跟踪序列中准确跟踪到的帧所占的比重, 这些帧的跟踪框与真实框的重叠率大于某个阈值, 该指标越大越好; 跟踪速度是每秒跟踪的帧数, 一般采用帧/秒 (fps) 表示.

(1) 精确度

精确度求解如下:

$$pre(j) = \frac{\sum_{i=1}^{frames} 1 \left\{ \sqrt{(x_i^T - x_i^G)^2 + (y_i^T - y_i^G)^2} \leq \varepsilon(j) \right\}}{frames} \quad (19)$$

式中, $pre(j)$ 是阈值取 $\varepsilon(j)$ 时的精确度. $1\{bool\}$ 在 $bool$ 等于 true 时为 1, $bool$ 等于 false 时为 0. $\sqrt{(x_i^T - x_i^G)^2 + (y_i^T - y_i^G)^2}$ 表示第 i 帧的目标中心位置 (x_i^T, y_i^T) 与其真实位置 (x_i^G, y_i^G) 的欧式距离. $\sum_{i=1}^{frames} 1 \left\{ \sqrt{(x_i^T - x_i^G)^2 + (y_i^T - y_i^G)^2} \leq \varepsilon(j) \right\}$ 表示所有欧式距离小于 $\varepsilon(j)$ 的帧数, $frames$ 表示某视频序列的帧数.

(2) 成功率

成功率求解如下:

$$suc(j) = \frac{\sum_{i=1}^{frames} 1 \left\{ \xi(j) \leq P(A_i^T \cap A_i^G) / P(A_i^T \cup A_i^G) \right\}}{frames} \quad (20)$$

式中, $suc(j)$ 是阈值取 $\xi(j)$ 时的精确度, $1\{bool\}$ 在 $bool$ 等于 true 时为 1, $bool$ 等于 false 时为 0, A_i^T 表示第 i 帧的目标框, A_i^G 表示第 i 帧的真实目标框, \cap 表示交集, \cup 表示并集, $P(B)$ 为 B 框内的像素个数, $\sum_{i=1}^{frames} 1 \left\{ \xi(j) \leq P(A_i^T \cap A_i^G) / P(A_i^T \cup A_i^G) \right\}$ 表示像素比大于 $\xi(j)$ 的帧数, $frames$ 表示某视频序列的帧数.

3.3 实验环境及参数设置

实验电脑的处理器的 Intel(R) Core(TM) i7-7500U CPU(2.9 GHZ), 内存是 8 GB, 开发软件为 Matlab

R2014b. 算法参数设置如下: 高斯核的方差 δ 为 0.5, Hog 特征的细胞单元大小为 4×4 , Hog 特征的方向为 9, Hog 特征 31 通道, gray 特征 1 通道, 颜色特征 10 通道, 模型的学习率 lr 为 0.01, 第 k 通道的初始值 $ratio_0^k = 1$, 通道的权重学习率 $ratio^w = 0.22$.

3.4 实验结果分析

(1) 定量分析

为了定量评估本文算法 (WSAMF) 的有效性, 我们对比了 WSAMF、SAMF、KCF、CN 在 OTB-2015^[18] 数据集的 100 组视频序列的表现. 我们做了两组实验, 第一组实验取距离阈值 $0 \leq \varepsilon(j) \leq 50$ 像素、重叠率阈值 $0 \leq \xi(j) \leq 1$, 第二组实验取距离阈值 $0 \leq \varepsilon(j) \leq 15$ 像素、重叠率阈值 $0.5 \leq \xi(j) \leq 1$.

第一组实验用于评估算法的综合性能, 第二组实

验用于测试算法的有效性. 因为在实际的目标跟踪过程中, 目标中心位置与实际中心位置的距离超过 15 像素, 算法基本跟踪失败; 目标框与真实框的重叠率小于 0.5 时候, 算法基本无效; 所以, 我们在第一组实验的基础上, 做了第二组实验.

第一组实验: 距离阈值 $0 \leq \varepsilon(j) \leq 50$ 像素、重叠率阈值 $0 \leq \xi(j) \leq 1$. 结果如图 1 所示.

图 1(a) 为算法执行一次的精度图, 该图显示算法随着距离阈值 $\varepsilon(j)$ 从 0 像素到 50 像素的精确度变化. 图 1(b) 为算法执行一次的成功率图, 该图显示算法随着重叠率阈值 $\xi(j)$ 从 0 到 1 的成功率变化. 通过图 1 可以看出, 相比 SAMF、KCF、CN, WSAMF 在精确度上分别提升 0.78%、8.5%、27.3%, 成功率则分别提升 1.3%、21.5%、42.9%.

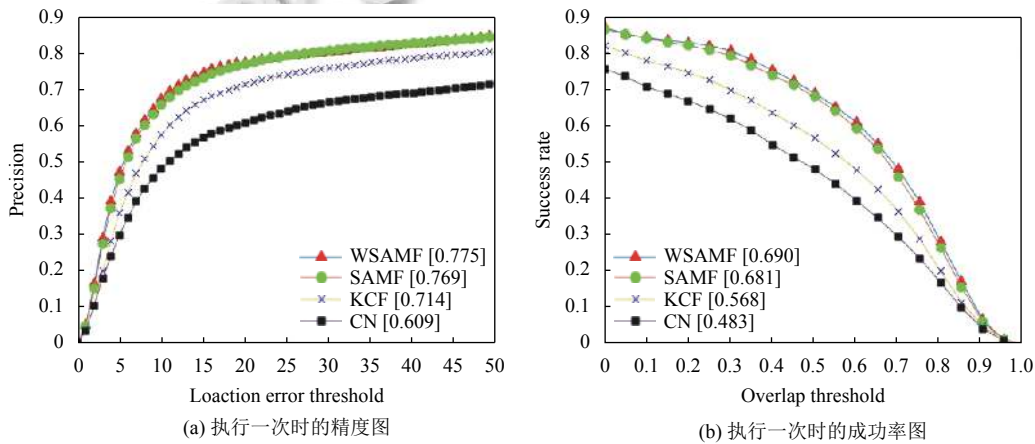


图 1 4 种算法的跟踪情况

为了综合评估本文算法 (WSAMF) 的性能, 在公开数据集 OTB-2015 的 100 个视频序列上, 取距离阈值 $0 \leq \varepsilon(j) \leq 50$ 像素、重叠率阈值 $0 \leq \xi(j) \leq 1$, 对比

WSAMF、SAMF、KCF、CN 在不同属性下精确度和成功率, WSAMF 都取得相对不错的跟踪效果, 具体结果见表 1、表 2.

表 1 4 种算法在 11 种属性下的精确度

算法	IV	DEF	SV	OCC	MB	FM	IPR	OPR	OV	BC	LR
WSAMF	0.707	0.697	0.738	0.740	0.680	0.696	0.727	0.760	0.787	0.711	0.789
SAMF	0.706	0.698	0.719	0.735	0.688	0.703	0.732	0.746	0.711	0.687	0.739
KCF	0.736	0.624	0.659	0.652	0.638	0.667	0.713	0.691	0.584	0.712	0.615
CN	0.591	0.531	0.560	0.574	0.523	0.543	0.662	0.622	0.538	0.601	0.547

表 2 4 种算法在 11 种属性下的成功率

算法	IV	DEF	SV	OCC	MB	FM	IPR	OPR	OV	BC	LR
WSAMF	0.642	0.603	0.623	0.675	0.665	0.624	0.649	0.656	0.589	0.638	0.545
SAMF	0.630	0.614	0.590	0.668	0.658	0.625	0.565	0.540	0.531	0.609	0.343
KCF	0.563	0.508	0.437	0.529	0.582	0.568	0.516	0.461	0.471	0.483	0.329
CN	0.472	0.441	0.378	0.448	0.466	0.465	0.642	0.665	0.643	0.653	0.568

第二组实验: 距离阈值 $0 \leq \varepsilon(j) \leq 15$ 像素、重叠率阈值 $0.5 \leq \xi(j) \leq 1$. 结果如图2所示.

图2(a)为算法执行一次的精度图, 该图显示算法随着距离阈值 $\varepsilon(j)$ 从0像素到15像素的精确度变化. 图2(b)为算法执行一次的成功率图, 该图显示算法随着重叠率阈值 $\xi(j)$ 从0.5到1的成功率变化. 通过图2可见, 相比SAMF、KCF、CN, WSAMF在精确度上分

别提升2.9%, 27.6%, 52.6%, 成功率则分别提升5.6%, 34.6%, 63.8%.

为了测试本文算法(WSAMF)的有效性, 在公开数据集OTB-2015的100个视频序列上, 取距离阈值 $0 \leq \varepsilon(j) \leq 15$ 像素、重叠率阈值 $0.5 \leq \xi(j) \leq 1$, 对比WSAMF、SAMF、KCF、CN在不同属性下精确度和成功率, WSAMF都取得不错的跟踪效果, 具体结果见表3、表4.

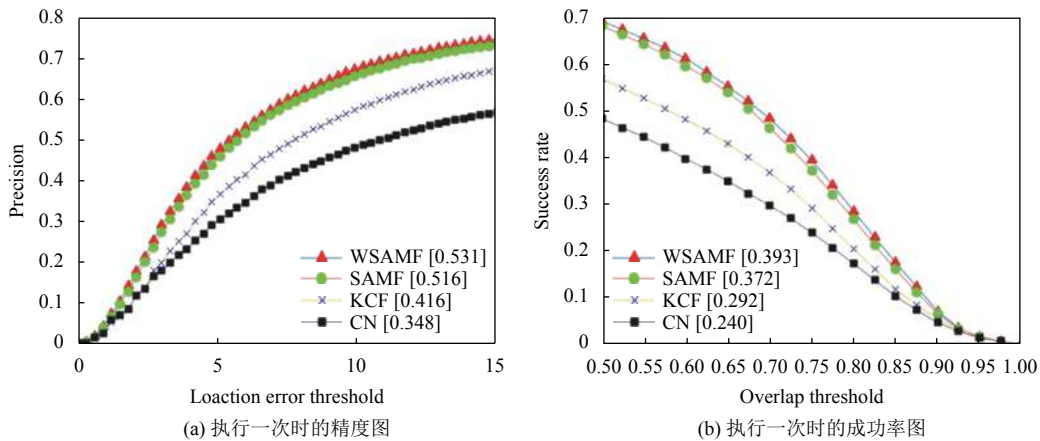


图2 4种算法的跟踪情况

表3 4种算法在11种属性下的精确度

算法	IV	DEF	SV	OCC	MB	FM	IPR	OPR	OV	BC	LR
WSAMF	0.519	0.432	0.497	0.521	0.428	0.413	0.436	0.487	0.448	0.509	0.625
SAMF	0.505	0.428	0.469	0.498	0.406	0.396	0.427	0.471	0.406	0.504	0.607
KCF	0.402	0.350	0.356	0.370	0.341	0.356	0.360	0.370	0.367	0.437	0.436
CN	0.342	0.273	0.305	0.320	0.250	0.285	0.355	0.332	0.335	0.357	0.400

表4 4种算法在11种属性下的成功率

算法	IV	DEF	SV	OCC	MB	FM	IPR	OPR	OV	BC	LR
WSAMF	0.367	0.300	0.292	0.360	0.444	0.387	0.318	0.341	0.411	0.406	0.199
SAMF	0.335	0.293	0.256	0.335	0.396	0.360	0.305	0.316	0.361	0.385	0.192
KCF	0.283	0.223	0.178	0.261	0.335	0.310	0.244	0.236	0.325	0.311	0.150
CN	0.212	0.182	0.147	0.228	0.240	0.234	0.231	0.199	0.248	0.222	0.127

通过以上两组实验可以看出, 本文算法在综合性能、有效性上, 都有一定提升.

(2) 定性分析

为了更加直观的评估算法性能, 我们对KCF、CN、SAMF和WSAMF在basketball(DEF)、Jogging-2(OCC)、Car4(IV)、Singer1(IV)、boy(OPR)、doll(SV)和Walking2(SV、OCC)等7个视频序列上进行了定性分析. 为了区分4种算法, 我们选用不同的颜色跟踪框表示. 其中, 红色实线为KCF算法, 绿色虚横线框为CN算法, 蓝色点横线框为SAMF算法, 黑色虚点线框

为WSAMF算法.

图3中, 目标在跟踪过程中发生了形变, KCF、CN在第259帧和第266帧只定位目标局部, SAMF在第271帧和第547帧则包含了相对多一点的背景信息, 只有WSAMF能够更好的跟踪这些目标.

图4中, 目标在跟踪过程中发生了遮挡, KCF、CN在第59帧、第60帧、第61帧和第64帧发生漂移, SAMF则包含了相对多一点的背景信息, 只有WSAMF能够更好的跟踪这些目标.

图5中, 目标在跟踪过程中光照变暗, KCF、CN、

SAMF 在第 202 帧、第 203 帧、第 218 帧和第 220 帧包含了相对多一点的背景信息, 只有 WSAMF 能够更好的跟踪这些目标.

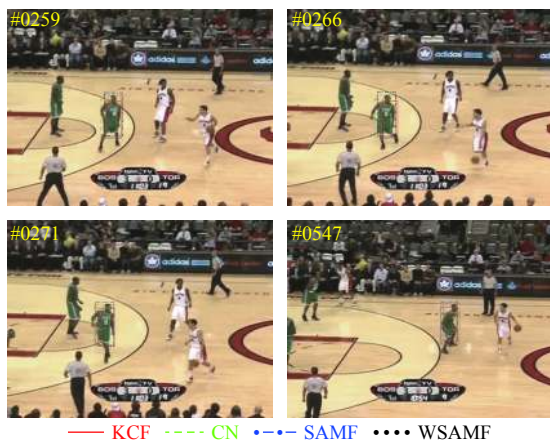


图 3 4 种算法在 Basketball 序列上的跟踪情况对比



图 4 4 种算法在 Jogging-2 序列上的跟踪情况对比

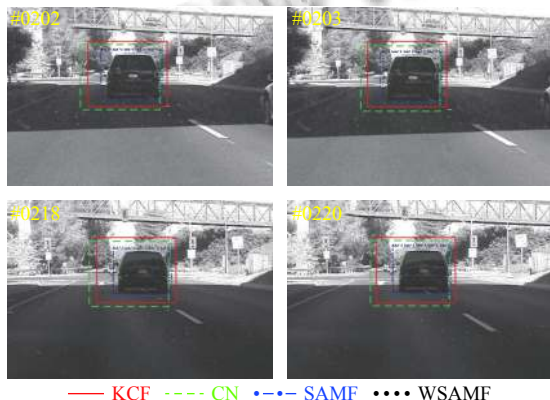


图 5 4 种算法在 Car4 序列上的跟踪情况对比

图 6 中, 目标在跟踪过程中光照增强, KCF、CN 在第 41 帧、第 75 帧、第 86 帧和第 131 帧包含了相对多一点的背景信息, SAMF 在第 131 帧包含了相对多一点的背景信息, 只有 WSAMF 能够更好地跟踪这些目标.

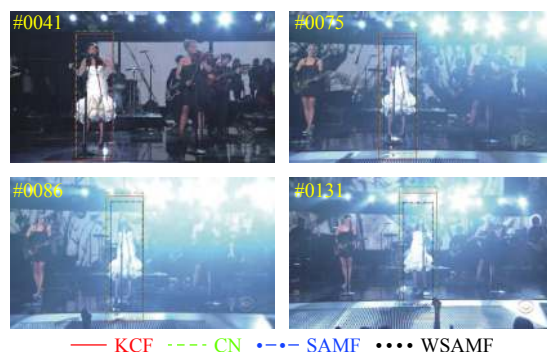


图 6 4 种算法在 Singer1 序列上的跟踪情况对比

图 7 中, 目标在跟踪过程中发生球面外旋转, KCF 和 CN 在第 459 帧、第 462 帧、第 584 帧和第 596 帧只跟踪到目标的一部分, 而 SAMF 则跟踪到多一点的背景信息, 只有 WSAMF 能够更好的跟踪这些目标.

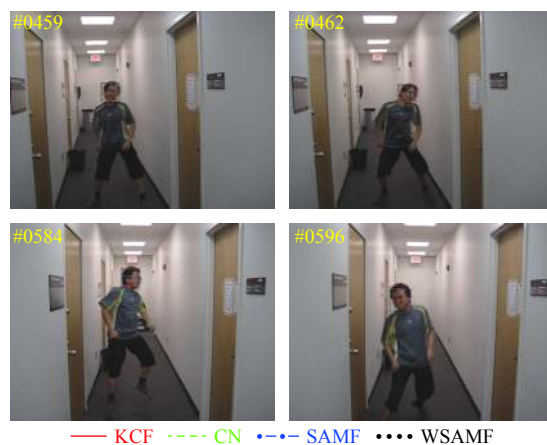


图 7 4 种算法在 boy 序列上的跟踪情况对比

图 8 中, 目标在跟踪过程中发生尺度变化, KCF、CN 在第 3526 帧、第 3669 帧、第 3721 帧和第 3737 帧发生不同程度的漂移, SAMF 跟踪到少一点的目标信息, 只有 WSAMF 能够更好的跟踪这些目标.

图 9 中, 目标在跟踪过程中发生遮挡、尺度变化, KCF、CN 在第 375 帧、第 487 帧和第 489 帧发生不同程度的漂移, SAMF 在第 206 帧、487 帧和 489 帧跟踪到多一点的背景信息, 只有 WSAMF 能够更好的跟踪这些目标.

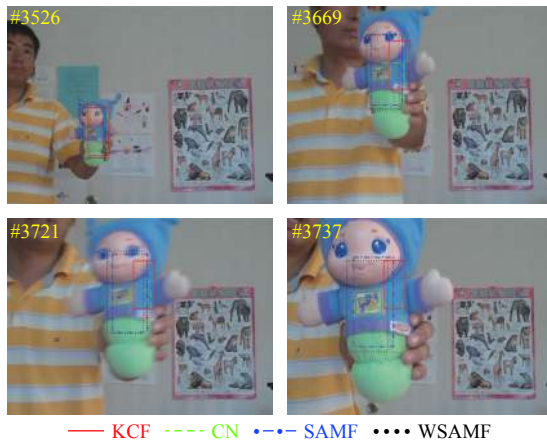


图8 4种算法在doll序列上的跟踪情况对比

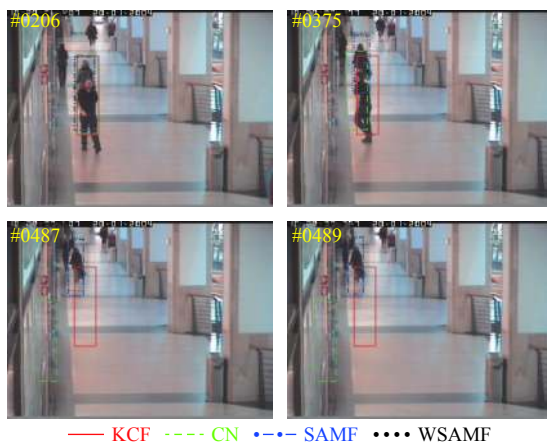


图9 4种算法在walking2序列上的跟踪情况对比

(3) 跟踪速度

为了直观的展示跟踪速度,在OTB-2015数据集上任意选取basketball、Jogging-2、Car4、Singer1、boy、doll和Walking2等7个视频序列上,对跟踪情况进行对比,具体情况见表5。

表5 4种算法的跟踪速度(fps)

视频序列	WSAMF	SAMF	KCF	CN
Basketball	3.8665	6.6457	3.5163	93.4951
Jogging-2	3.3483	7.9191	5.0138	52.8304
Car4	2.9183	7.6156	13.1487	45.9857
Singer1	3.4195	7.5834	4.5436	10.4701
Boy	4.8583	7.2777	3.4719	157.6025
Doll	6.5207	12.276	23.6181	70.2455
Walking2	4.4065	9.4809	5.8025	74.2437

单从表中数据来看,4种算法都达不到实时要求,这主要是受电脑硬件条件限制。在硬件条件获得一定提升后,WSAMF应该可以满足实时性跟踪要求。

另外,需要补充说明的是,同一种算法在不同数据集下跟踪速度有一定差异,这主要是因为不同数据集的跟踪目标、搜索区域数据复杂度不同,导致计算量不同,进而导致跟踪速度有差异。

4 结语

本文算法根据各通道对响应值的贡献度调整权重,构建实时特征组合,进而控制特征融合。与现有的基于类别融合特征的算法不同,该算法从通道层面融合特征。实验表明该算法能够有效提升目标跟踪的精确度、成功率,整体性能优于对比算法。实验中发现,在目标趋近于完全遮挡的过程过慢时,由于算法未对目标中背景剔除,则容易出现不同程度的跟踪漂移甚至失败。因此,如何剔除目标中的背景信息或有效区分目标、背景,将是下一步研究的工作重点。

参考文献

- Comaniciu D, Ramesh V, Meer P. Real-time tracking of non-rigid objects using mean shift. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Hilton Head Island, SC, USA. 2000. 142-149.
- Dalal N, Triggs B. Histograms of oriented gradients for human detection. Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego, CA, USA. 2005. 886-893. [doi: 10.1109/CVPR.2005.177]
- Ross DA, Lim J, Lin RS, et al. Incremental learning for robust visual tracking. International Journal of Computer Vision, 2008, 77(1): 125-141.
- Wang S, Lu HC, Yang F et al. Supersixel tracking. Proceedings of 2011 International Conference on Computer Vision. Barcelona, Spain. 2011. 1323-1330.
- 许莉,王敏,温月.基于Lab颜色空间的运动目标检测.华中科技大学学报(自然科学版),2013,41(S1):219-222.
- Bolme DS, Beveridge JR, Draper BA, et al. Visual object tracking using adaptive correlation filters. Proceedings of 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Francisco, CA, USA. 2010. 2544-2550.
- Henriques JF, Caseiro R, Martins P, et al. Exploiting the circulant structure of tracking-by-detection with kernels. Proceedings of the 12th European Conference on Computer Vision. Florence, Italy. 2012. 702-715.
- Danelljan M, Khan FS, Felsberg M, et al. Adaptive color

- attributes for real-time visual tracking. Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA. 2014. 1090–1097.
- 9 Henriques JF, Caseiro R, Martins P, *et al.* High-speed tracking with kernelized correlation filters. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(3): 583–596. [doi: [10.1109/TPAMI.2014.2345390](https://doi.org/10.1109/TPAMI.2014.2345390)]
- 10 Li Y, Zhu JK. A scale adaptive kernel correlation filter tracker with feature integration. Proceedings of European Conference on Computer Vision. Zurich, Switzerland. 2014. 254–265.
- 11 Bertinetto L, Valmadre J, Golodetz S, *et al.* Staple: Complementary learners for real-time tracking. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA. 2016. 1401–1409.
- 12 李聪, 鹿存跃, 赵珣, 等. 特征融合的尺度自适应相关滤波跟踪算法. 光学学报, 2018, 38(5): 0515001.
- 13 赵高鹏, 沈玉鹏, 王建宇. 基于核循环结构的自适应特征融合目标跟踪. 光学学报, 2017, 37(8): 0815001.
- 14 陈法领, 丁庆海, 常铮, 等. 自适应特征融合的多尺度核相关滤波目标跟踪. 光学学报, 2020, 40(3): 0315001.
- 15 尹宽, 李均利, 李丽, 等. 复杂情况下自适应特征更新目标跟踪算法. 光学学报, 2019, 39(11): 1115002.
- 16 常敏, 沈凯, 张学典, 等. 自适应特征融合和模型更新的相关滤波跟踪. 光学学报, 2019, 39(9): 0915001.
- 17 Galoogahi HK, Sim T, Lucey S. Multi-channel correlation filters. Proceedings of 2013 IEEE International Conference on Computer Vision. Sydney, NSW, Australia. 2013. 3072–3079.
- 18 Wu Y, Lim J, Yang MH. Object tracking benchmark. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1834–1848. [doi: [10.1109/TPAMI.2014.2388226](https://doi.org/10.1109/TPAMI.2014.2388226)]