

# 改进的多特征粒子滤波目标跟踪算法研究

张蕴琦<sup>1</sup>, 郭发勇<sup>2</sup>, 朱梓清<sup>2</sup>, 王亚民<sup>3</sup>

(1. 常州工程职业技术学院 教学工作部, 江苏 常州 213164;

2. 常州工程职业技术学院 智能制造学院, 轨道交通高端装备学院, 江苏 常州 213164;

3. 常州工程职业技术学院 通识教育教學部, 江苏 常州 213164)

**摘要:** 针对在复杂环境下多特征融合的粒子滤波算法跟踪精确度低的问题, 提出一种改进的多特征融合算法; 该算法采用二阶中心差分卡尔曼滤波方法来实现建议分布函数的优化, 在重要性采样中融入最新的测量信息, 提高了粒子的使用效率, 并引入动态模板更新机制对目标模板实时更新; 在多特征融合策略上利用基于粒子滤波框架下的 EM 算法适用于不同数量样本集的特点求解状态估计, 不仅避免因计算特征权重产生误差, 而且提高了算法的实时性; 滤波器仿真实验结果表明, 在一维非线性模型下对比其它改进粒子滤波算法, 提出的方法性能最优; 在基于视频序列的目标跟踪实验中, 通过比较本文算法在不同特征、不同采样粒子数量条件下的性能对比验证文章算法的有效性; 最后通过一系列不同环境下的跟踪实验证明, 文章所提算法对复杂条件下的目标跟踪具有较高的精度和鲁棒性。

**关键词:** 多特征; 目标跟踪; 中心差分卡尔曼滤波; 粒子滤波; EM 算法

## Research on Target Tracking Algorithm for Improved Multi-feature Particle Filter

ZHANG Yunqi<sup>1</sup>, GUO Fayong<sup>2</sup>, ZHU Ziqing<sup>2</sup>, WANG Yamin<sup>3</sup>

(1. Academic Affairs Office, Changzhou Vocational Institute of Engineering, Changzhou 213164, China;

2. School of Intelligent Manufacturing and Traffic High-tech Equipment, Changzhou Vocational Institute of

Engineering, Changzhou 213164, China;

3. Department of General Education, Changzhou Vocational Institute of Engineering, Changzhou 213164, China)

**Abstract:** Aimed at the problem of low tracking accuracy of particle filter (PF) algorithm for multi-feature fusion under complex environments, an improved multi-feature fusion algorithm was proposed. The algorithm adopted the second-order central difference Kalman filter (SO-CDKF) to optimize the proposal distribution function, the latest observation information was fused into the importance sampling to improve the efficiency of the particles. Meanwhile, the template updating strategy was introduced to update the target template in real time. In the multi-feature fusion strategy, the expectation maximization (EM) algorithm based on PF framework was used to solve the state estimation on the different quantities of particle sets, this not only avoided the errors caused by calculating the weights of multi-feature, but also improved the real-time performance. The filter simulation results show that compared with other improved PF algorithms in one-dimensional nonlinear model, the proposed method has the best performance. Through the target tracking experiments based on video sequence, the effectiveness of the proposed algorithm is verified by comparing the different sampling particle performances. Finally, The target tracking experimental results of different environments show that the proposed algorithm has a high accuracy and robustness for target tracking under complex conditions.

**Keywords:** multi-feature; target tracking; CDKF; PF; EM algorithm

## 0 引言

基于多特征融合的粒子滤波目标跟踪算法一度是视觉跟踪领域的研究热点, 为复杂背景下的目标跟踪提供了可行方案。多特征目标跟踪算法的关键在于选取特征和设计特征融合策略。目标的颜色、纹理、梯度、边缘特征是比较常见的融合特征<sup>[1]</sup>。在融合策略上, 常用的有乘性融合

和加性(加权和)融合。文献[2]、[3]是乘性融合用于多特征跟踪的典型示例, 在粒子滤波框架下, 选取两到三种互补特征用以表征目标, 融合策略上采用带权重的乘积形式实现特征的自适应融合, 其融合结果提高了特征的鉴别能力, 缺点是对噪声敏感, 当背景出现强干扰时容易丢失目标。文献[4-7]采用自适应加性融合策略; 文献[4]使用颜色、边缘、纹理特征进行自适应加权融合, 通过判

收稿日期: 2023-10-09; 修回日期: 2023-10-27。

基金项目: 常州工程职业技术学院校科研基金课题(11130300120010); 江苏省重点研发计划项目(BE2020006-2); 江苏省重点研发计划项目(BE2021016-1)。

作者简介: 张蕴琦(1980-), 女, 硕士, 工程师。

郭发勇(1989-), 男, 博士, 副教授。

王亚民(1966-), 男, 博士, 教授。

引用格式: 张蕴琦, 郭发勇, 朱梓清, 等. 改进的多特征粒子滤波目标跟踪算法研究[J]. 计算机测量与控制, 2023, 31(12): 322-329, 337.

断背景的复杂度对粒子数量进行动态调整增强算法的实时性; 文献 [5] 选取颜色、梯度、纹理特征描述目标, 以单特征和多特征融合状态下的目标估计的欧氏距离为依据计算特征权值; 文献 [6] 提出自适应融合颜色特征和边缘特征的红外目标跟踪算法, 在粒子滤波框架下通过计算不同特征的可区分性和稳定性来设置特征权值; 文献 [7] 融合颜色直方图和方向梯度直方图, 提出一种基于自适应分块的多特征融合粒子滤波跟踪方法。加性融合的结果保留了分布的多样性, 可在一定程度上抑制噪声, 缺点是不能有效提升特征的鉴别性。针对乘性融合和加性融合的缺陷, 文献 [8] 中提出特征不确定性度量的概念, 以粒子空间位置方差和粒子观测概率值熵的乘积作为某种特征的不确定性, 自适应地调节该特征在观测概率中的比重, 把乘性和加性两种融合方式统一到同一自适应框架下, 从而达到取长补短的效果。

上述文献中的多特征融合策略存在三方面缺陷: 1) 标准粒子滤波算法<sup>[9]</sup>选取易于采样的状态转移函数作为建议密度分布函数, 忽略了当前观测值的影响; 2) 同一时刻下多特征融合似然观测模型中各特征采样的粒子数相等, 无法针对实际跟踪场景中各特征的置信度和时间开销的差异设置不同的样本数量, 对实时性造成影响; 3) 特征权值易受随机误差的干扰, 影响多特征融合在提升特征鉴别能力上的优势。

为了克服以上缺陷, 本文采用滤波方法优化抽样函数, 采用 SO-CDKF 来生成粒子滤波的建议密度分布函数, 在粒子滤波框架下利用 EM 算法<sup>[10]</sup>适用于不同数量样本集的优势求解多特征融合下的目标估计, 使外观模型适应目标变化。实验结果表明, 该算法提高了复杂环境下运动目标跟踪的精度和鲁棒性。

## 1 基于 SO-CDKF 的粒子滤波算法

首先定义离散非线性状态空间模型如下所示:

$$\begin{cases} x_k = f(x_{k-1}) + w_{k-1} \\ y_k = h(x_k) + v_k \end{cases} \quad (1)$$

式中,  $f(\cdot)$  和  $h(\cdot)$  表示已知的系统状态转移函数和观测函数,  $w_{k-1}$  和  $v_k$  表示服从均值为 0, 协方差为  $Q_{k-1}$  和  $R_k$  的高斯分布白噪声。二阶中心差分卡尔曼粒子滤波 (SO-CDPF, second-order central difference Kalman particle filter) 的计算步骤如下:

1) 初始化。  $k=0$  时刻, 从先验分布中抽样  $N$  个粒子, 得到初始化粒子集  $\{x_0^i, \omega_0^i\}_{i=1}^N$ , 令  $\omega_0^i = 1/N$ , 求得初始化粒子集的状态均值和协方差。

初始状态均值:

$$\hat{x}_0 = E[x_0^i] \quad (2)$$

初始化协方差:

$$P_{x_0} = E[(x_0^i - \hat{x}_0)(x_0^i - \hat{x}_0)^T] \quad (3)$$

2) 计算权值。对  $n$  维状态向量,  $h$  是中心差分区间的长度, CDKF 的 Sigma 点集个数是  $2n+1$ , 其权值如下:

$$\omega_j^{(m)} = \begin{cases} (h^2 - n)/h^2 & (j = 0) \\ 1/(2h^2) & (j = 1, \dots, 2n) \end{cases} \quad (4)$$

$$\omega_j^{(c_1)} = \frac{1}{4h^2}, \omega_j^{(c_2)} = \frac{h^2 - 1}{4h^4} (j = 1, \dots, 2n) \quad (5)$$

3)  $k=k+1$  时刻, 用 SO-CDKF 生成粒子滤波的建议密度分布函数。

① 确定时间更新的 Sigma 点集:

$$x_{k-1} = [\hat{x}_{k-1} \quad \hat{x}_{k-1} + h \sqrt{P_{x_{k-1}}} \quad \hat{x}_{k-1} - h \sqrt{P_{x_{k-1}}}] \quad (6)$$

② 时间更新:

$$x_{k|k-1}^j = f(x_{k-1}^j) \quad j = 0, \dots, 2n \quad (7)$$

$$\hat{x}_{k|k-1} = \sum_{j=0}^{2n} \omega_j^{(m)} x_{k|k-1}^j \quad (8)$$

$$P_{x_{k|k-1}} = \sum_{j=1}^n [\omega_j^{(c_1)} (x_{k|k-1}^j - x_{k|k-1}^{n+j}) (x_{k|k-1}^j - x_{k|k-1}^{n+j})^T + \omega_j^{(c_2)} (x_{k|k-1}^j + x_{k|k-1}^{n+j} - 2x_{k|k-1}^0) \times (x_{k|k-1}^j + x_{k|k-1}^{n+j} - 2x_{k|k-1}^0)^T] + Q_{k-1} \quad (9)$$

③ 确定测量更新的 Sigma 点集:

$$x_{k|k-1} = [\hat{x}_{k|k-1} \quad \hat{x}_{k|k-1} + h \sqrt{P_{x_{k|k-1}}} \quad \hat{x}_{k|k-1} - h \sqrt{P_{x_{k|k-1}}}] \quad (10)$$

④ 测量更新:

$$y_{k|k-1}^j = h(x_{k|k-1}^j) \quad j = 0, \dots, 2n \quad (11)$$

$$\hat{y}_{k|k-1} = \sum_{j=0}^{2n} \omega_j^{(m)} y_{k|k-1}^j \quad (12)$$

$$P_{y_{k|k-1}} = \sum_{j=1}^n [\omega_j^{(c_1)} (y_{k|k-1}^j - y_{k|k-1}^{n+j}) (y_{k|k-1}^j - y_{k|k-1}^{n+j})^T + \omega_j^{(c_2)} (y_{k|k-1}^j + y_{k|k-1}^{n+j} - 2y_{k|k-1}^0) \times (y_{k|k-1}^j + y_{k|k-1}^{n+j} - 2y_{k|k-1}^0)^T] + R_k \quad (13)$$

$$P_{x_{k+1}|y_{k+1}} = \sqrt{\omega_j^{(c_1)} P_{x_{k|k-1}}} (y_{k|k-1}^j - y_{k|k-1}^{n+j})^T \quad (14)$$

$$K_k = P_{x_{k+1}|y_{k+1}} P_{y_{k|k-1}}^{-1} \quad (15)$$

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_{k|k-1}) \quad (16)$$

$$P_{x_k} = P_{x_{k+1}|y_{k+1}} - K_k P_{y_{k+1}} K_k^T \quad (17)$$

4) 采样:

$$x_k^i \sim q(x_k^i | x_{0:k-1}^i, y_{1:k}^i) \approx N(\hat{x}_k^i, P_{x_k}^i) \quad i = 1, \dots, N \quad (18)$$

5) 计算粒子权重:

$$\omega_k^i \propto \omega_{k-1}^i \frac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{0:k-1}^i, y_{1:k}^i)} \quad i = 1, \dots, N \quad (19)$$

6) 归一化:

$$\bar{\omega}_k^i = \omega_k^i / \sum_{i=1}^N \omega_k^i \quad (20)$$

7) 重采样:

计算有效粒子数,  $N_{\text{eff}} = 1 / \sum_{i=1}^N (\bar{\omega}_k^i)^2$ , 对于给定阈值  $N_{\text{th}}$ , 如有  $N_{\text{eff}} < N_{\text{th}}$ , 令  $\bar{\omega}_k^i = 1/N$ , 对粒子集  $\{x_k^i, \bar{\omega}_k^i\}_{i=1}^N$  进行重采样。

8) 输出滤波分布的均值和协方差:

$$\hat{x}_k = \sum_{i=1}^N \bar{\omega}_k^i x_k^i \quad (21)$$

$$P_{x_i} = \sum_{i=1}^N \omega_k^i (\hat{x}_k - x_k^i)(\hat{x}_k - x_k^i)^T \quad (22)$$

## 2 改进的多特征粒子滤波算法

### 2.1 特征权值

传统多特征粒子滤波算法通过设计自适应权重来调节不同时刻各特征在观测模型中的贡献。文献 [2] 采用最小 Bhattacharyya 距离平方的倒数作为该特征的权重, 但最小距离不能代表其它粒子与目标模板的匹配程度, 当粒子数目较大或分布较为分散时, 特征权值的准确性会降低。文献 [4] 采用平均 Bhattacharyya 距离作为权值, 当粒子出现样本匮乏时同样会产生偏差。文献 [5] 选择单特征目标状态与多特征融合目标状态的相似度来计算自适应权值, 对初始目标提取特征的鲁棒性要求较高, 若首帧的目标状态估计发生偏差, 后续跟踪的误差将被持续放大。文献 [8] 从粒子空间分布和特征的鉴别能力两方面分析特征的不确定性, 提出粒子空间分布越集中, 似然观测概率越尖锐, 此种状况下不确定性越小, 跟踪效果越好, 反之亦然。但在实际应用中, 当不确定性较小时也会出现跟踪失败; 或不确定性较大时也能成功跟踪到目标, 实验效果如图 1 所示。

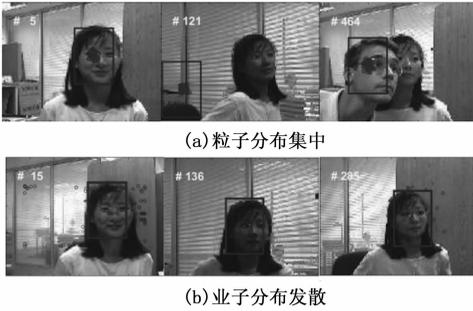


图 1 两种不确定性下的跟踪效果

视频序列 “Girl” 中跟踪目标为女性面部, 根据实验结果可以看出: 图 1 (a) 中当目标跟踪被背景干扰发生偏差时, 粒子呈现出不确定性较小的集中分布; 图 1 (b) 中粒子状态发散, 不确定性较大, 但通过加权求和同样能计算出较为精确的目标状态。

从上文可以看出, 在多特征融合策略下计算自适应特征权值易受随机误差的干扰。为了解决这一问题, 本文在粒子滤波框架下引入 EM 算法求解最终状态估计, 避免因计算特征权重产生误差。

### 2.2 粒子滤波框架下的 EM 算法 (EM-PF)

已知  $\{X_k^{i,j}\}_{j=1}^{n_i}$  是  $k$  时刻下第  $i$  个特征的状态数据集, 样本数量是  $n_i$ ,  $(X_k^{i,j} | \mu_k^i, \sigma_k^i) \sim N(\mu_k^i, \sigma_k^i)$ ,  $\mu_k^i$  和  $\sigma_k^i$  表示当前特征的均值和方差, 且  $\mu_k^i \sim N(\mu_k, \tau_k^2)$ ,  $\mu_k$  和  $\tau_k$  表示特征融合条件下的状态均值和方差。记:

$$\begin{cases} \theta = (\mu, \log \sigma^1, \dots, \log \sigma^i, \log \tau) \\ X = \{X^{i,j}\} \\ Z = (\mu^1, \dots, \mu^i) \end{cases} \quad i = 1, \dots, m, j = 1, \dots, n_i \quad (23)$$

根据文献 [11], 式 (23) 中  $\theta$  是未知数据, 令  $i = 1, \dots, m$ , 取其先验分布为:

$$p(\mu, \log \sigma^1, \dots, \log \sigma^i, \log \tau) \propto \tau \quad (24)$$

根据贝叶斯定理得到后验分布:

$$p(\mu^1, \dots, \mu^m, \mu, \log \sigma^1, \dots, \log \sigma^i, \log \tau | X) \propto p(\mu, \log \sigma^1, \dots, \log \sigma^i, \log \tau) \cdot \prod_{i=1}^m p(\mu^i | \mu, \tau) \prod_{i=1}^m \prod_{j=1}^{n_i} p(X^{i,j} | \mu^i, \sigma^i) \quad (25)$$

把  $Z$  看作潜在数据, 添加  $Z$  后得到关于  $\theta$  的后验分布, 将其展开取对数可得:

$$\begin{aligned} \log p(\mu, \log \sigma^1, \dots, \log \sigma^i, \log \tau | X, Z) &\propto \\ \log p(\mu^1, \dots, \mu^m, \mu, \log \sigma^1, \dots, \log \sigma^i, \log \tau | X) &\propto - \\ \sum_{i=1}^m n_i \log \sigma^i - (m-1) \log \tau - \frac{1}{2\tau^2} \sum_{i=1}^m (\mu^i - \mu)^2 - \\ \sum_{i=1}^m \left[ \frac{1}{2(\sigma^i)^2} \sum_{j=1}^{n_i} (\mu^i - X^{i,j})^2 \right] \end{aligned} \quad (26)$$

直接对  $p(\theta | X)$  进行极大似然估计求未知变量  $\theta$  是很困难的, 把  $Z$  作为潜在数据, 便可采用 EM 算法来实现它的极大化。

E 步:

已知在  $k$  时刻的估计值  $\theta_k = (\mu_k, \sigma_k^1, \dots, \sigma_k^i, \tau_k)$ ,  $i = 1, \dots, m$  在给定  $X$  和  $\theta_k$  的条件下求式 (26) 的期望, 方差已知时正态总体均值  $\mu^i$  的共轭先验分布是正态分布, 可得:

$$(\mu^i | \theta_k, X) \sim N(\hat{\mu}_k^i, V_k^i) \quad (27)$$

其中:

$$\hat{\mu}_k^i = [\mu_k / (\tau_k)^2 + (n_i \cdot \bar{X}_k^i) / (\sigma_k^i)^2] \cdot V_k^i \quad (28)$$

$$V_k^i = \left( \frac{1}{(\tau_k)^2} + \frac{n_i}{(\sigma_k^i)^2} \right)^{-1} \quad (29)$$

式 (28) 中,  $\bar{X}_k^i$  是  $k$  时刻第  $i$  个特征的状态均值。

$$\bar{X}_k^i = \frac{1}{n_i} \sum_{j=1}^{n_i} X_k^{i,j} \quad (30)$$

对于任意与  $\mu^i$  无关的变量  $\zeta$ ,

$$\begin{aligned} E[(\mu^i - \zeta)^2 | \theta_k, X] &= [E(\mu^i | \theta_k, X) - \zeta]^2 + \\ \text{Var}(\mu^i | \theta_k, X) &= (\hat{\mu}_k^i - \zeta)^2 + V_k^i \end{aligned} \quad (31)$$

分别令  $\zeta$  等于  $\mu$  和  $X^{i,j}$ , 得到式 (26) 在  $X$  和  $\theta_k$  给定下的期望。

$$\begin{aligned} Q(\theta | \theta_k, X) &= - \sum_{i=1}^m n_i \log \sigma^i - (m-1) \log \tau - \\ &\frac{1}{2\tau^2} \sum_{i=1}^m [(\mu^i - \mu)^2 + V_k^i] - \\ &\sum_{i=1}^m \left[ \frac{1}{2(\sigma^i)^2} \sum_{j=1}^{n_i} (\mu^i - X^{i,j})^2 + V_k^i \right] + \xi \end{aligned} \quad (32)$$

其中:  $\xi$  是与  $\theta$  无关的变量。

M 步:

对  $Q(\theta | \theta_k, X)$  进行极大似然估计只需对  $\mu, \sigma^i, \tau$  分别求导得:

$$\mu_{k+1} = \frac{1}{m} \sum_{i=1}^m \hat{\mu}_k^i \quad (33)$$

$$\sigma_{k+1}^i = \left\{ \frac{1}{n_i} \sum_{j=1}^{n_i} [(X_k^{i,j} - \mu_k^i)^2 + V_k^i] \right\}^{1/2} \quad (34)$$

$$\tau_{k+1} = \left\{ \frac{1}{m-1} \sum_{j=1}^m [(\mu_k^j - \mu_k)^2 + V_k^j] \right\}^{1/2} \quad (35)$$

### 2.3 算法实现

本文提出的基于多特征融合的 SO-CDPF 算法的实现步骤如下:

1) 初始化。k=0 时刻, 在初始帧中手动选择目标区域, 采用颜色和纹理特征描述目标, 建立初始模板  $q_0^u$  (u 是特征直方图的索引区间)。根据先验分布采样第 i 个特征的  $n_i$  个粒子, 设置初始化粒子集为  $\{X_0^{i,j}, \omega_0^{i,j}\}_{j=1}^{n_i}$ , 令  $\omega_0^{i,j} = 1/n_i$ ;

2) 粒子传播。k=k+1 时刻, 根据 SO-CDKF 生成建议密度分布函数, 得到采样粒子集  $\{X_k^{i,j}\}_{j=1}^{n_i}$ ;

3) 计算粒子权值并归一化。根据式 (19) 更新粒子的权值, 式 (20) 对权值进行归一化;

4) 对第 i 个特征进行重采样, 生成新的粒子集  $\{X_k^{i,j}, \omega_k^{i,j}\}_{j=1}^{n_i}$ ;

5) 目标模板更新。通过度量 k 时刻第 i 个特征候选区域  $p_k^{i,u}$  与初始模板  $q_0^{i,u}$  和前一时刻目标模板  $q_{k-1}^{i,u}$  特征直方图的 Bhattacharyya 距离作为模型更新的依据, 选择与候选区域相似度更高的模板进行跟踪<sup>[12]</sup>。目标模型的更新公式如下:

$$q_k^{i,u} = \begin{cases} q_0^{i,u}, d(p_k^{i,u}, q_0^{i,u}) < d(p_k^{i,u}, q_{k-1}^{i,u}) \\ (1-\alpha)q_{k-1}^{i,u} + \alpha p_{k|X_k}^{i,u}, \text{otherwise} \end{cases} \quad (36)$$

$E_{[X_k]}$  是 k 时刻第 i 个特征的平均状态,  $\alpha$  表示平均状态的特征直方图在目标特征直方图中的权重, 本文中  $\alpha$  取值 0.2。

6) 通过 EM-PF 算法求取特征融合下的状态估计:

- ①E 步骤: 根据式 (28)、(29) 计算  $\mu_k^i, V_k^i$ ;
- ②M 步骤: 根据式 (33) ~ (35) 计算  $\mu_{k+1}^i, \sigma_{k+1}^i, \tau_{k+1}$ ;
- ③判断 EM 迭代是否继续;
- 7) 特征融合下的状态估计:  $\hat{X}_k = \mu_k$ ;
- 8) 判断是否已到尾帧, 否的话返回步骤 (2)。

### 3 实验结果与分析

实验基于 Matlab R2019 的仿真环境, 计算机配置为处理器 IntelCore (T-M) i5-9400F、主频 2.90 GHz、内存 8 GB。为详细验证算法的有效性进行 3 项实验: 1) 通过一维非线性模型进行仿真, 对采用扩展卡尔曼滤波 (EKF, extended Kalman filter)<sup>[13]</sup>、无迹卡尔曼滤波 (UKF, unscented Kalman filter)<sup>[14]</sup>、容积卡尔曼滤波 (CKF, cubature Kalman filter)<sup>[15]</sup> 优化建议分布的扩展卡尔曼粒子滤波 (EPF, extended Kalman particle filter)、无迹卡尔曼粒子滤波 (UPF, unscented Kalman particle filter)、容积卡尔曼粒子滤波 (CPF, cubature Kalman particle filter) 和标准 PF 及 CDPF 这 5 种滤波算法的性能进行验证; 2) 验证本文算法在不同粒子数量下的性能; 3) 通过一系列不同环境下的跟踪实验证明本文算法的有较性。

#### 3.1 不同滤波算法的性能对比

模型的离散状态方程和测量方程如下所示:

$$x(k) = 1 + \sin(0.04\pi k) + 0.5x(k-1) + w(k-1) \quad (37)$$

$$y(k) = \begin{cases} 0.2x^2(k) + v(k), k \leq 30 \\ 0.5x - 2 + v(k), k > 30 \end{cases} \quad (38)$$

过程噪声  $w(k-1)$  是符合  $\Gamma(3,2)$  分布的随机噪声矩阵, 观测噪声  $v(k)$  的协方差  $R_k$  是  $1 \times 10^{-5}$ , k 是单次蒙特卡洛仿真的运行步长,  $k \in [1, 2, \dots, 60]$ , 状态初值  $x_0 = 0.1$ , 状态估计初值  $x_{0|0} = 1$ , 方差估计初值  $P_{0|0} = 5$ , PF 算法粒子数取值 100, 仿真 200 次。

图 2 是 200 次蒙特卡洛仿真的平均滤波状态估计, 可以看出除去 PF 算法, EPF 与真实状态的误差最大, UPF 的滤波效果优于 EPF, CPF 和 CDPF 最接近真实状态。因为 EKF 是通过泰勒级数展开式一阶线性化截断来转化非线性问题, 当高阶项无法忽略时会导致滤波发散, 精度严重降低。UKF、CKF、CDKF 本质上都是对状态的先验分布抽取一组数量、位置分布和权值确定的 Sigma 采样点, 通过对 Sigma 点经非线性函数传递后的结果进行加权计算, 以二阶泰勒精度来逼近非线性状态的后验分布, 能有效克服 EKF 的缺陷。

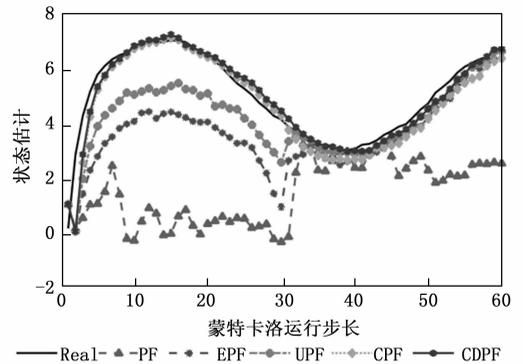


图 2 系统平均状态估计

图 3 是上述 5 种算法的精确度比较, 滤波估计精度衡量指标采用均方根误差 (RMSE, root mean square error) 计算:

$$RMSE_k = \sqrt{\frac{1}{M} \sum_{m=1}^M \|\hat{x}_k^m - x_k\|^2} \quad (39)$$

式中, M 为蒙特卡洛运行次数,  $\hat{x}_k^m$  和  $x_k$  分别为第 m 次运行 k 时刻的目标估计状态和真实状态。精确度和实时性比较采用 200 次蒙特卡洛仿真的平均值。

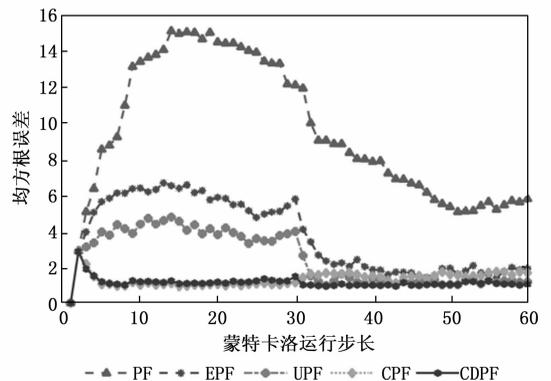


图 3 平均精确度比较

图 4 是其实时性比较, 时间单位是秒 (s)。在图 4 中, 除去 PF 算法, EPF 的计算时间最短, CPF 和 CDPF 的计算时间相近, UPF 耗时最多。

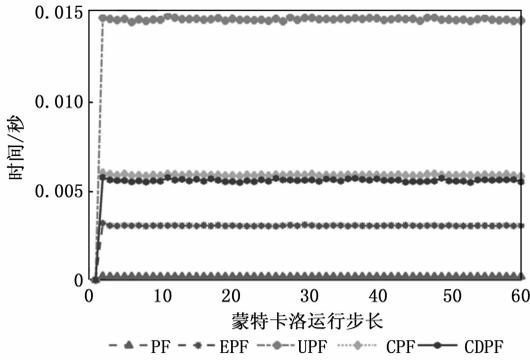


图 4 平均实时性比较

表 1 从精确度和实时性两方面对 5 种算法的性能做定量分析。可以看出, CDPF 的性能稍微优于 CPF: 在精确度上, CDPF 的平均均方根误差比 CPF 减少 15.79%, 在实时性上, CDPF 的平均时间比 CPF 减少 4.90%

表 1 性能比较

算法	平均均方根误差	平均时间/ $\times 10^{-2}$ s
PF	9.47	1.13
EPF	3.72	17.88
UPF	2.52	85.86
CPF	1.33	34.49
CDPF	1.12	32.80

### 3.2 单特征与多特征算法的性能对比

视频序列 Girl, 像素  $128 \times 96$ , 共 500 帧, 选取复杂背景下的快速运动的人脸作为跟踪目标。为了验证本文算法在特征融合上的有效性, 采用本文算法和颜色、纹理单特征 SO-CDPF 算法作对比。为了更符合人体视觉感知、提高对噪声的抵抗能力, 颜色特征选取 HSV, 纹理特征选取 MB-LBP<sup>[20]</sup>, 目标模板动态更新。测试目标的初始位置手动给定, HSV 和 MB-LBP 特征的粒子数均为 50, HSV 特征方差取值 0.1, MB-LBP 特征方差取值 0.03。

#### 3.2.1 定性分析

视频序列中, 目标受旋转、尺度变化、遮挡等干扰因素的影响。目标发生平面外旋转时, 初始目标区域几乎完全消失, 模板自动更新机制确保跟踪的有效性。当目标面部特征消失时, HSV 特征的跟踪效果比 MB-LBP 特征算法的精确度高; 目标发生平面内旋转和尺度变化时, 3 种算法在该干扰下都能跟踪到目标; 目标被男性面部遮挡时, 由于目标与干扰物的 HSV 特征相似, 两者交叉时 HSV 特征跟踪框发生一定程度的偏移, 而 MB-LBP 能捕捉局部特征, 遮挡阶段 MB-LBP 特征的跟踪效果较之 HSV 特征更精确。本文算法虽然在个别帧上出现偏差, 但可以即时恢复跟踪, 比两种单特征算法的跟踪效果更稳定、精确。图 5 显示了

以上 3 种算法在第 108、150、240、311、328、439、466、491 帧上的跟踪效果。

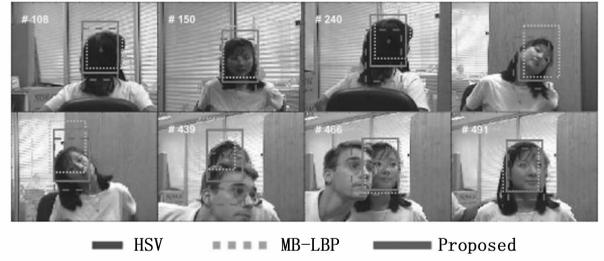


图 5 “Girl” 序列中 3 种算法的跟踪结果

#### 3.2.2 定量分析

本文对以上 3 种算法采用中心位置误差和运行速率两方面对精确度和实时性做评估。中心位置误差表示目标中心位置与真实中心位置间的误差, 其值越小跟踪精确度越高, 计算方法如式 (40) 所示, 单位是像素:

$$d_k = \sqrt{\|\hat{x}_k - x_k\|^2} \quad (40)$$

$d_k$  为第  $k$  帧上的中心位置误差,  $\hat{x}_k$  和  $x_k$  分别为目标的中心位置和真实中心位置。对比式 (39) 可知, 中心位置误差就是蒙特卡洛运行 1 次的均方根误差。图 6 表示该 3 种算法的中心位置误差。运行速率是某算法每秒运行的帧数, 单位是帧/秒。

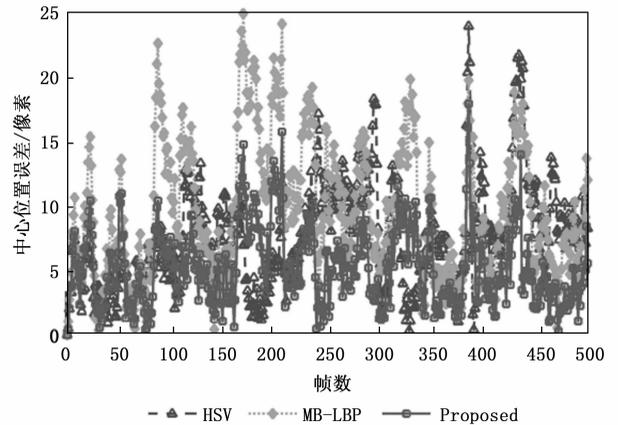


图 6 “Girl” 序列中 3 种算法的中心位置误差

表 2 显示以上 3 种算法在 “Girl” 序列上的平均性能比较结果。在精确度上, 本文算法的平均中心位置误差对比改进 HSV、MB-LBP 单特征 SO-CDPF 算法分别减少 29.96%、45.21%。在实时性上, 因计算复杂度本文算法的平均运行速率与改进 HSV 特征 SO-CDPF 算法相比减少 45.31%, 与改进 MB-LBP 特征 SO-CDPF 算法相比减少 51.68%, 但能够达到平均 7 frame/s 以上, 基本满足跟踪实时性要求。

表 2 “Girl” 序列中 3 种算法平均性能比较

算法	平均中心位置误差/像素	平均运行速率/(帧/秒)
HSV	7.51	13.13
MB-LBP	9.60	14.86
Proposed	5.26	7.18

### 3.3 不同采样粒子数下的性能对比

上一节中分析了“Girl”序列中两种单特征在不同干扰因素下的表现效果。假设 HSV 特征的粒子数是  $N_1$ , MB-LBP 特征的粒子数是  $N_2$ , 使用 4 种不同的粒子数量分配方案在进行精确度和实时性对比。本文算法在不同粒子数下的中心位置误差如图 7 所示。

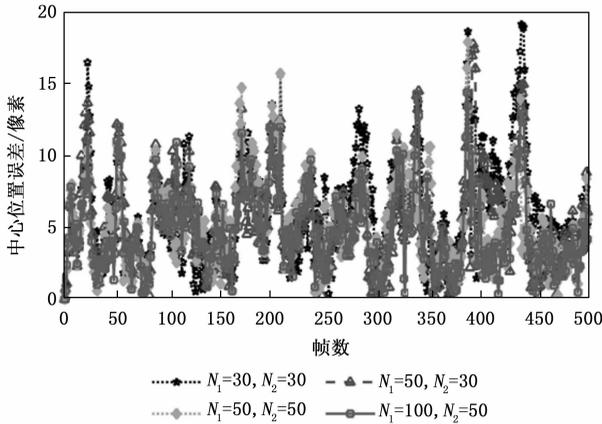


图 7 “Girl”序列中本文算法不同粒子数的中心位置误差

表 3 显示本文算法在不同粒子数下的平均性能比较结果。可以看出, 粒子数量的增加能提高算法的精确度, 但会产生较大的时间开销, 影响实时性。以“ $N_1=30, N_2=30$ ”和“ $N_1=100, N_2=50$ ”两种分配方案为例, 后者的平均中心位置误差比前者减少 20.76%, 后者的平均运行速率比前者减少 57.11%。对比 4 种方案的平均性能可知, “ $N_1=50, N_2=30$ ”的粒子数量分配方案在兼顾精确度和实时性方面最优。

表 3 “Girl”序列中本文算法不同粒子数的平均性能比较

粒子数量	平均中心位置误差/像素	平均运行速率/(帧/秒)
$N_1=30, N_2=30$	6.07	9.35
$N_1=50, N_2=30$	5.32	9.06
$N_1=50, N_2=50$	5.26	7.18
$N_1=100, N_2=50$	4.81	4.01

### 3.4 不同视频环境下的跟踪实验

本次实验中, 笔者把本文算法与基于相关滤波的 KCF<sup>[16]</sup>、DSST<sup>[17]</sup>、BACF<sup>[18]</sup>、SAMF<sup>[19]</sup>、Staple<sup>[20]</sup> 作性能对比。在特征选取上, KCF、DSST 和 BACF 使用单一 HOG 特征。SAMF 采用灰度、CN、HOG 三种特征融合, Staple 使用颜色直方图和 HOG 特征。在尺度变化上, 除 KCF 外, 其它 4 种相关滤波算法都具备尺度变化。

#### 3.4.1 定性分析

本文选取 5 个具有代表性的视频序列, 对 6 种算法进行实验验证, 本文算法的特征方差与 4.2 节相同, 粒子数量根据场景复杂度设置。5 种相关滤波算法使用默认参数。视频序列的部分跟踪结果对比分析如图 8 所示, 每一帧中左上角为帧序号, 为了更直观地显示本文算法的精确度, 跟

踪目标的真实中心位置用○符号标注, 本文算法的中心位置用与跟踪框同色的×符号标注。视频序列的主要属性如表 4 所示。

表 4 视频序列参数表

序列名称	像素	帧数	干扰因素
Skating2	640×352	473	遮挡、变形、平面外旋转、尺度变化、快速运动
Jogging	514×420	307	遮挡、变形、平面外旋转
Soccer	560×316	392	光照、尺度变化、遮挡、运动模糊
Diving	400×224	215	尺度变化、变形、平面内旋转
ClifBar	320×240	472	尺度变化、遮挡、运动模糊、快速运动、平面内旋转、出视野、背景杂波

1) “Skating2”序列, 由于相似干扰物和背景颜色变化的影响, 使得目标的颜色和 HOG 特征受到干扰。在两个运动目标多次相互交叉时, 5 种对比算法都受到相似干扰物的影响, 而纹理特征能够有效地鉴别目标, 因而本文算法的跟踪性能相对稳定。

2) “Jogging”序列, 目标在 51 帧被完全遮挡后, KCF、DSST、BACF、Staple 因采用固定的模型更新速率导致目标被遮挡后持续学习不良样本, 出现跟踪失败。SAMF 的模型更新速率最小, 所以对短时间的遮挡不敏感。本文算法得益于模板更新机制, 出现短暂漂移后在 69 帧重新捕捉到目标中心。

3) “Soccer”视频序列, 从 71 帧起目标发生快速模糊运动, 采用 HOG 特征的 DSST 和 KCF 先后丢失目标; 从 106 帧起目标被红色彩带遮挡, Staple、BACF、SAMF 因模型更新的误差积累导致后续跟踪向红色物体漂移。在尺度变化上, DSST 和 Staple 均能响应尺度变化; BACF 和 SAMF 由于目标特征快速变化引发跟踪框不断扩大。本文算法得益于多特征融合和模板更新机制, 在目标发生运动模糊时出现偏差后能很快寻回目标, 但在遮挡阶段因目标与背景颜色相似, 导致颜色直方图跟踪效果变差, 发生较长时间的跟踪漂移, 对其它干扰因素的影响不敏感。

4) “Diving”视频序列, 本文算法充分发挥了颜色直方图在快速变形上的优势, 对比 5 种相关滤波算法鲁棒性更高。174 帧到尾帧, 除本文算法外, 其它 5 种对比算法全部跟踪失败。Staple 虽然同样具备颜色直方图特征, 但在多特征融合时使用常数作为特征权重, 无法充分发挥颜色特征的优势。

5) “ClifBar”视频序列, 目标从 71 帧起发生模糊运动和尺度变化, 本文算法的颜色和纹理特征在运动模糊的干扰下出现漂移乃至跟踪失败。224 帧, 目标在前两种干扰因素下又加入背景杂波, SAMF 由于模型更新速率无法适应目标外观变化, 和 KCF 一并持续跟踪失败。后续跟踪中, 本文算法对背景杂波影响下的尺度变化不敏感。330 帧目标出现位置突变, Staple 跟踪失败。DSST、BACF 和本文算法持续稳定跟踪直到尾帧。

从定性分析可以看出, 本文算法基于 EM-PF 框架进行特征融合, 可充分发挥颜色直方图的全局表达能力和纹理直方图的区域性特征的互补优势, 有效抵抗形变、局部遮挡、快速运动、旋转及一定程度的光照变化。与同样不具备尺度变化的 KCF 相比, 本文算法得益于颜色直方图对尺度变化不敏感, 对目标非剧烈的尺度变化表现出较高的鲁棒性, 缺点是无法抵抗运动模糊的干扰, 但在动态模板更新机制下可实现偏差校正。

### 3.4.2 定量分析

本文对该 6 种算法的性能从中心位置误差、重叠率和跟踪速度三方面进行评估。

1) 中心位置误差: 所有算法在 5 个视频序列上的中心位置误差结果如图 9 所示。经过比较, 本文算法的中心位置误差处于较低水平。

6 种算法在 5 个视频序列上的平均中心位置误差如表 5 所示。

表 5 平均中心位置误差比较

算法	平均中心位置误差/像素				
	Skaing2	Jogging	Soccer	Diving	ClifBar
KCF	30.76	144.47	15.37	42.57	36.72
DSST	34.67	150.84	20.29	73.61	5.33
BACF	45.10	189.31	59.38	40.59	9.64
SAMF	48.16	3.39	68.48	70.38	25.24
Staple	23.07	146.52	65.65	76.52	29.16
Proposed	20.70	6.16	15.80	14.81	12.71

在“Skating2”、“Diving”序列中, 本文算法排名最优, 对比次优算法 Staple、BACF 的中心位置误差分别减少 10.27%、63.51%; 在“Jogging”、“Soccer”序列中, 本文算法排名次优, 对比最优算法 SAMF、KCF 的中心位置误差分别相差 81.71%、2.80%; 在“ClifBar”序列中, 本文算法的精确度排名第三, 对比最优算法 DSST 相差 138.46%, 对比次优算法 BACF 相差 31.85%。

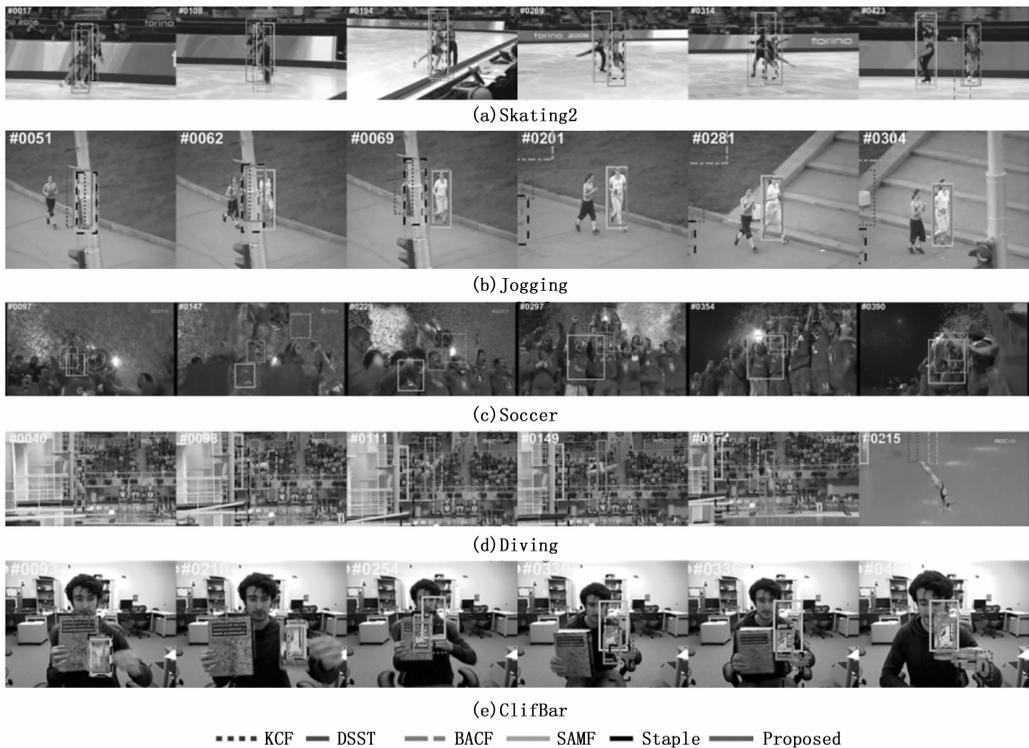


图 8 不同算法的部分跟踪结果对比

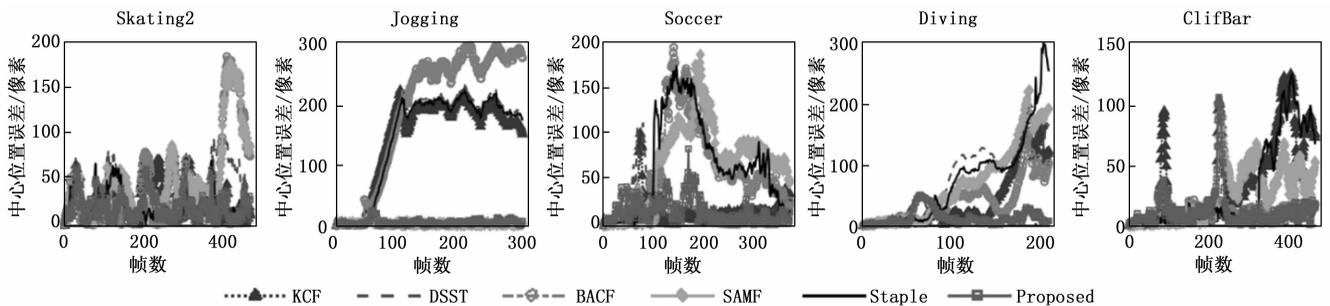


图 9 5 个视频序列上不同算法的中心位置误差

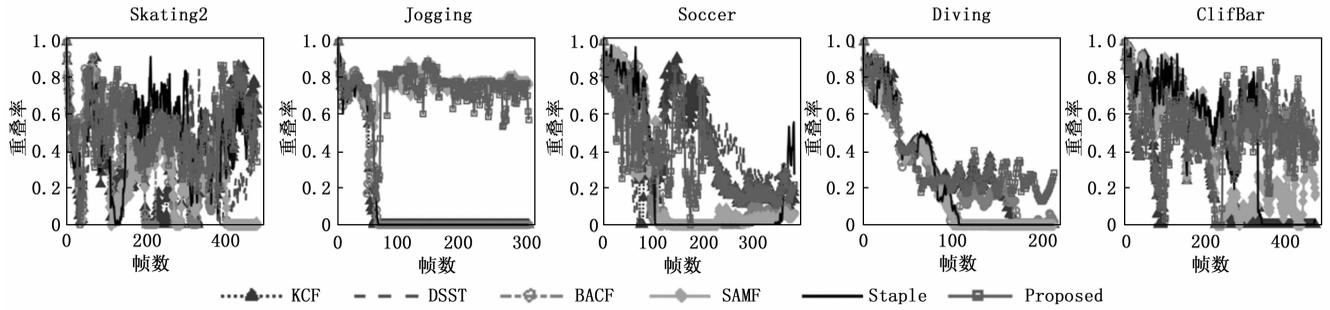


图 10 5 个视频序列上不同算法的重叠率

2) 重叠率:

重叠率是跟踪算法得到的目标区域面积与真实目标区域面积重叠部分的占比, 计算公式如下所示:

$$O_k = \frac{A_k \cap A_{gt\_k}}{A_k \cup A_{gt\_k}} \quad (41)$$

式中,  $A_k$  是跟踪算法在第  $k$  帧得到的目标区域面积,  $A_{gt\_k}$  是数据集人工标注的第  $k$  帧的真实区域面积。重叠率  $O_k$  越大表示算法的跟踪效果越好。所有算法在 5 个视频序列上的重叠率结果如图 10 所示。

经过比较, 本文算法的重叠率在大部分时间内处于较高水平。6 种算法在 5 个视频序列上的平均重叠率如表 6 所示。

表 6 平均重叠率比较

算法	平均重叠率				
	Skaing2	Jogging	Soccer	Diving	ClifBar
KCF	0.35	0.12	0.42	0.32	0.26
DSST	0.38	0.14	0.43	0.21	0.65
BACF	0.32	0.13	0.22	0.27	0.57
SAMF	0.30	0.77	0.20	0.23	0.38
Staple	0.48	0.14	0.22	0.24	0.46
Proposed	0.52	0.74	0.36	0.35	0.48

在“Skating2”、“Diving”序列中, 本文算法获得最高平均重叠率, 对比次优算法 Staple、KCF 分别提高 8.33%、9.36%; 在“Jogging”序列中本文算法排名次优, 比最优算法 SAMF 相差 3.90%; “Soccer”序列中, 本文算法排名第三, 对比最优算法 DSST 和次优算法 KCF 分别相差 16.28%、14.29%; 在“ClifBar”序列中, 本文算法排名第三, 对比最优算法 DSST 和次优算法 BACF 分别相差 26.15%、15.79%。

3) 运行速率: 6 种算法在 5 个视频序列上的平均运行速率如表 7 所示。

表 7 平均跟踪速度比较

算法	平均运行速率/(帧/秒)				
	Skating2	Jogging	Soccer	Diving	ClifBar
KCF	200.51	162.36	153.70	290.64	369.59
DSST	8.68	28.72	24.85	42.16	66.17
BACF	21.10	22.93	25.93	30.79	37.79
SAMF	24.71	20.14	12.66	30.48	40.54
Staple	5.14	5.05	5.07	4.13	4.65
Prposed	5.16	7.75	7.90	7.52	8.64

由表 7 可得, KCF 在全部序列上平均运行速率最高; DSST 的平均运行速率受跟踪目标区域面积影响较大; Staple 的实时性最差。本文算法的时间开销主要用于 SO-CDKF 计算 Sigma 点集的均值和协方差, 总体时间开销与粒子数量和目标区域大小相关。笔者利用 Matlab 支持混和编程的特点, 对主要时间开销部分使用 C 语言优化, 可把运行速率至少提高到 70 帧/秒以上。

4 结束语

针对传统多特征融合的粒子滤波算法存在的不足之处, 本文提出一种改进的多特征融合算法, 采用颜色和纹理两种具有互补性的特征直方图用以表征目标, 利用 SO-CDKF 算法优化建议分布函数, 在采样粒子集中融入最新的测量信息, 有效提升采样精度, 并采用 EM-PF 算法进行多特征融合, 根据场景复杂度分配各特征的采样粒子数, 最大程度地兼顾算法的实时性。实验数据表明, 该算法可有效抵抗多种干扰因素的影响, 具有较高的精确性和鲁棒性。

参考文献:

- [1] 咎孟恩, 周航, 韩丹, 等. 粒子滤波目标跟踪算法综述 [J]. 计算机工程与应用, 2019, 55 (5): 8-17, 59.
- [2] 姜华, 范勇. 多线索融合和区域划分的粒子滤波跟踪算法 [J]. 计算机工程与应用, 2013, 49 (19): 186-190.
- [3] 刘明华, 汪传生, 王宪伦. 基于特征可分性和稳定性度量的多特征融合目标跟踪算法 [J]. 控制与决策, 2016, 31 (7): 1231-1236.
- [4] 杨欣, 刘加, 周鹏宇, 等. 基于多特征融合的粒子滤波自适应目标跟踪算法 [J]. 吉林大学学报 (工学版), 2015, 45 (2): 533-539.
- [5] DOU J F, LI J X. Robust visual tracking base on adaptively multi-feature fusion and particle filter [J]. Optik, 2014, 125 (5): 1680-1686.
- [6] CHONG Y, WANG Z, RONG C, et al. A particle filter infrared target tracking method based on multi-feature adaptive fusion [J]. Geomatics & Information Science of Wuhan University, 2016, 41 (5): 598-604.
- [7] BIE X, LIU H, CHANG F, et al. Multi-target tracking method based on the adaptive fragment and multi-feature fusion [J]. Journal of Xidian University, 2017, 44 (2): 151-157.

(下转第 337 页)