

# 基于位置纠正和 Kalman 滤波算法的目标跟踪方法<sup>①</sup>



陈杨梅, 赵传森, 邹 斌

(湖北大学 数学与统计学学院, 武汉 430062)  
通信作者: 陈杨梅, E-mail: cym5991@163.com

**摘 要:** 本文基于判别尺度空间跟踪算法, 将位置纠正方法和卡尔曼滤波算法应用于行人跟踪中. 为解决行人因形变和环境变化导致的跟踪不准确的问题, 本文充分利用 *fhog* 特征在行人跟踪上的优势, 以判别尺度空间算法中的位置滤波器所计算的位置为中心, 再次提取行人的 *fhog* 特征并将其与位置滤波器模板做相关运算, 以此纠正行人位置. 其次, 利用卡尔曼滤波算法对纠正后的行人位置进行预测和再次纠正, 最终在双重纠正的位置上训练新的位置滤波器模板. 本文选取 OTB-100 中的行人数据集对该方法进行性能测试, 实验结果表明, 在原算法位置上, 再次提取 *fhog* 特征进行相关运算能够纠正行人的位置, 同时卡尔曼滤波对纠正位置进行预测和再次纠正, 可使行人的定位精度再次提升.

**关键词:** 卡尔曼滤波; 判别尺度空间跟踪; 位置纠正; 位置滤波器

引用格式: 陈杨梅, 赵传森, 邹斌. 基于位置纠正和 Kalman 滤波算法的目标跟踪方法. 计算机系统应用, 2023, 32(6): 149-158. <http://www.c-s-a.org.cn/1003-3254/9136.html>

## Target Tracking Method Based on Position Correction and Kalman Filtering Algorithm

CHEN Yang-Mei, ZHAO Chuan-Sen, ZOU Bin

(Faculty of Mathematics and Statistics, Hubei University, Wuhan 430062, China)

**Abstract:** On the basis of the discriminative scale space tracking algorithm, the position correction method and Kalman filtering algorithm are applied to pedestrian tracking in this study. Due to deformation and environmental changes, pedestrians cannot be accurately tracked. To solve the problem, this study makes full use of the advantage of the *fhog* feature in pedestrian tracking and takes the position calculated by the position filter in the discriminative scale space algorithm as the center. It extracts the *fhog* feature of pedestrians again and correlates it with the position filter template to correct the pedestrian position. Then, the Kalman filtering algorithm is used to predict and correct the corrected pedestrian position again, and finally, a new position filter template is trained in the twice-corrected position. In this study, the pedestrian data set in OTB-100 is selected to test the performance of the method. The experimental results show that in the original algorithm position, the *fhog* feature is extracted again for correlation operations to correct the position of pedestrians. At the same time, the Kalman filtering predicts and corrects the corrected position, which can improve the positioning accuracy of pedestrians again.

**Key words:** Kalman filtering; discriminative scale space tracking; correction of position; position filter

<sup>①</sup> 收稿时间: 2022-12-07; 修改时间: 2023-01-06; 采用时间: 2023-01-19; csa 在线出版时间: 2023-04-23  
CNKI 网络首发时间: 2023-04-24

## 1 介绍

目标跟踪作为计算机视觉领域最具挑战的关键技术之一,在机器人视觉导航,无人驾驶,医学诊断等多个领域有着广泛的应用<sup>[1]</sup>.在实际跟踪中,由于场景的多变性和复杂性,实现高效目标跟踪仍具挑战性,尤其是目标的跟踪定位问题.目前国内外研究者为解决跟踪中的定位问题已做出了大量研究.

1975年在基于概率密度梯度函数的计算中,首次定义了均值偏移 (meanshift) 概念<sup>[2]</sup>,1995年 Cheng 阐述了均值偏移的应用范畴<sup>[3]</sup>,部分学者在其之后也对均值偏移问题运用在跟踪方法中.2009年,多实例目标跟踪算法通过分类的方式对目标实施追踪<sup>[4]</sup>.2010年,平均误差的最小输出和 (minimum output sum of squared error, MOSSE)<sup>[5]</sup> 作为跟踪界相关滤波器的开山之作,该算法采用了最小化均方差方式去训练目标的位置滤波器,并以此确定目标的位置.此后,一些基于 MOSSE 的追踪算法 (循环结构内核 (circulant structure kernel, CSK)<sup>[6]</sup>、核相关滤波 (kernel correlation filter, KCF)<sup>[7]</sup>、判别尺度空间跟踪 (discriminative scale space tracker, DSST)<sup>[8]</sup>) 等新算法也陆续诞生,速率都能够超过每秒数百帧,从而能够在实时的追踪系统中普遍地使用.

而部分性能更优良的改进追踪算法也随之提出.其中较经典的算法有卡尔曼滤波 (Kalman filter) 算法<sup>[9]</sup>,粒子滤波 (particle filter) 算法<sup>[10]</sup> 以及基于峰值旁瓣比的抗遮挡目标跟踪方法.例如文献 [11,12] 通过卡尔曼

滤波器来预测车辆在下一帧中的运动状态来解决车辆的定位问题,但这种方法在复杂行车场景中具有局限性.李军等<sup>[13]</sup> 采用峰值旁瓣比并结合粒子滤波来进行状态估计以解决目标框丢失问题,但这种方法只能处理短暂的完全遮挡,无法解决目标精准定位的问题.张红颜等<sup>[14]</sup> 将目标的总体色彩特征划分为局部色彩特征进而提高算法的抗遮挡能力,但这种方法在目标颜色相近的情况下容易跟错目标.文献 [15-17] 通过卷积神经网络提取目标深度特征,将深度特征融入相关滤波,并采用高置信度更新的方式提高抗遮挡能力,但这种方法局限于复杂的计算量.宦若虹等<sup>[18]</sup> 基于积分直方图进行特征提取,对不同的遮挡情况采取不同跟踪机制,但这种方法没有有效解决目标准确定位问题.

在相关滤波算法中,判别尺度空间算法是将尺度和位置估计相结合的算法,其引入快速傅里叶变换使跟踪速度有了很大提升,并能准确评估目标的位置和尺度,因此其在目标跟踪、智能监控方面应用广泛<sup>[19]</sup>.但将其应用于行人跟踪时,该算法对因行人形变、环境变化导致的定位不准确问题的处理能力较弱.

为解决行人形变、环境变化导致的定位不准确问题,本文在 DSST 的基础结构下,利用 *fhog* 特征<sup>[20]</sup> 在行人跟踪上的优势,以原算法所计算的行人位置为中心,再次提取行人的 *fhog* 特征,将其与位置滤波器模板做相关运算来纠正行人位置.同时利用 Kalman 滤波算法对纠正后的行人位置进行预测和再次纠正.整个算法的流程图如图 1 所示.

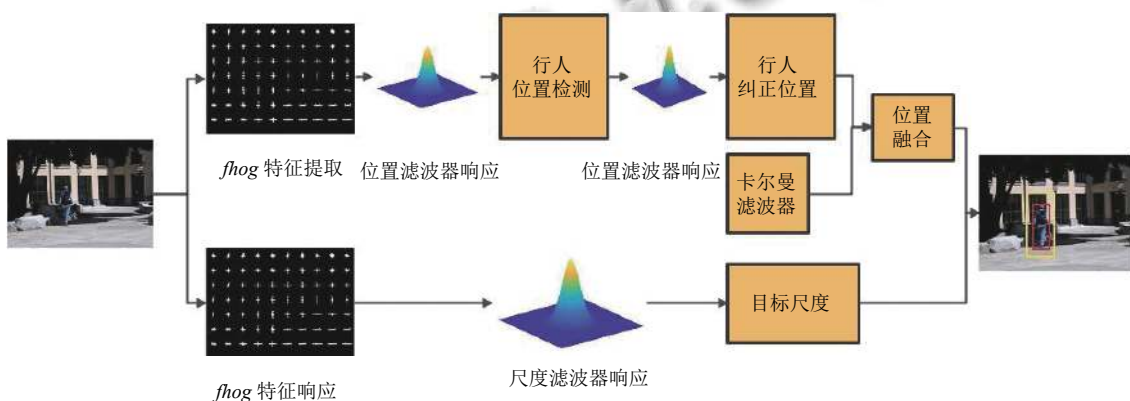


图 1 改进算法流程图

本文的主要贡献如下.

(1) 首次将位置纠正方法应用在行人跟踪上. 通过

再次提取行人的 *fhog* 特征并进行相关运算,能够扩大行人的检索范围并提取出较完整的行人特征描述子,

从而对行人的位置进行纠正. 同时, 也能够在不大幅度降低算法速度的前提下, 训练纠正位置的滤波器模板, 提高跟踪性能.

(2) 将 Kalman 滤波器的预测和纠正功能应用在行人跟踪中. Kalman 滤波器的预测和纠正模型将行人纠正的位置再次进行了纠正, 实现了对行人位置的双重纠正, 从而再次提高跟踪算法的性能.

本文的其余部分组织如下: 第 2 节详细介绍了 DSST 算法, 第 3 节详细介绍了本文所提出的改进方法, 第 4 节展示了跟踪评价指标和对行人的跟踪实验结果, 最后对全文进行了总结.

## 2 判别尺度空间算法

相关滤波<sup>[21-23]</sup>是当前研究的热点, 其源于信号处理领域. 在众多相关滤波算法中, 判别尺度空间跟踪算法 (DSST) 是将尺度和位置估计相结合的算法, 其可以对目标尺度进行精确估计. 首先其通过所训练的位置滤波器确定目标所在的位置, 在已确定的位置基础上, 再由不同尺度的样本所训练出的尺度滤波器对该目标尺度进行估计, 两种滤波器的设计思想相同.

以下分别介绍了判别尺度空间算法的位置滤波器和尺度滤波器模型的详细构建策略.

### 2.1 位置滤波器模型

首先以跟踪目标为中心, 提取 3 倍于目标大小的图像块的 *fhog* 特征, 采用正则化最小二乘模型构造最小代价函数, 即每个通道的最优滤波器由式 (1) 构造:

$$\varepsilon = \left\| \sum_{l=1}^d h^l \otimes f^l - g \right\|^2 + \lambda \sum_{l=1}^d \|h^l\|^2 \quad (1)$$

其中,  $f$  为训练样本特征, 特征通道  $l \in \{1, \dots, d\}$ ,  $f^l$  为第  $l$  维的样本特征,  $h^l$  为第  $l$  个通道对应的滤波器,  $\lambda$  为正则化参数,  $g$  为样本对应的高斯函数输出.

为提高运算速度, 将式 (1) 转换到频域计算, 根据极值点求导法则可得滤波器式 (2):

$$H^l = \frac{\overline{G}F^l}{\sum_{k=1}^d \overline{F^k}F^k + \lambda} \quad (2)$$

其中,  $\overline{G}$  为  $g$  经过离散傅里叶变换后的复共轭,  $\overline{F^k}$  是  $f^k$  离散傅里叶变换后的复共轭.

为将前后时刻的目标外观相联系, 需要利用式(3)

进行迭代更新:

$$\begin{cases} A_t^l = (1-\eta)A_{t-1}^l + \eta\overline{G}F_t^l \\ B_t = (1-\eta)B_{t-1} + \eta \sum_{k=1}^d \overline{F_t^k}F_t^k \end{cases} \quad (3)$$

其中,  $A_t^l$  是滤波器  $H^l$  的分子,  $B_t$  是滤波器  $H^l$  的分母,  $\eta$  是算法的学习率.

对于新的一帧, 在前一帧位置基础上提取 3 倍于目标大小的图像块特征, 经过离散傅里叶变换后得到  $Z_t^l$ , 利用式 (4) 得到响应:

$$Y_t = \frac{\sum_{l=1}^d \overline{A_{t-1}^l} Z_t^l}{B_{t-1} + \lambda} \quad (4)$$

对式 (4) 进行反傅里叶变换, 其响应最大值所对应的区域  $y_t$  即为当前目标区域.

$$y_t = F^{-1}(Y_t) \quad (5)$$

### 2.2 尺度滤波器模型

为确定跟踪目标更精确的尺度, 通过不同尺寸的目标样本特征加以训练构造出尺度滤波器, 最终将两个滤波器分别得到的位置和尺度作为跟踪结果.

以位置滤波器所获取的目标的位置为中心, 利用:  $a^n P \times a^n R, n \in \left\{ \left[ -\frac{S-1}{2} \right], \left[ -\frac{S-2}{2} \right], \dots, \left[ \frac{S-1}{2} \right] \right\}$  获取  $S$  个不同尺度的样本, 其中  $P, R$  分别是前一帧目标的宽, 高,  $a$  为尺度因子,  $S$  为尺度个数. 其次将尺度样本用于尺度滤波器的训练, 所构造的尺度滤波器的方式和位置滤波器相同. 最后通过当前帧所提取的样本的通道特征  $Z_{s,t}^l$ , 利用式 (6) 得到尺度响应:

$$Y_{s,t} = \frac{\sum_{l=1}^d \overline{A_{t-1}^l} Z_{s,t}^l}{B_{t-1} + \lambda} \quad (6)$$

对式 (6) 进行反傅里叶转换, 响应最大值所对应的尺度即是当前目标尺寸.

DSST 算法将跟踪任务看作目标定位和尺度估计两个独立的问题. 先使用跟踪物体的 *fhog* 特征训练位置滤波器, 从而判断目标中心位置, 接着利用目标的 33 个尺度样本特征训练另一个尺度滤波器, 来检测目标的尺度变化, 通过两个滤波器的相互作用, DSST 算法能够解决目标定位和尺度自适应的问题, 其算法流程图如图 2 所示.



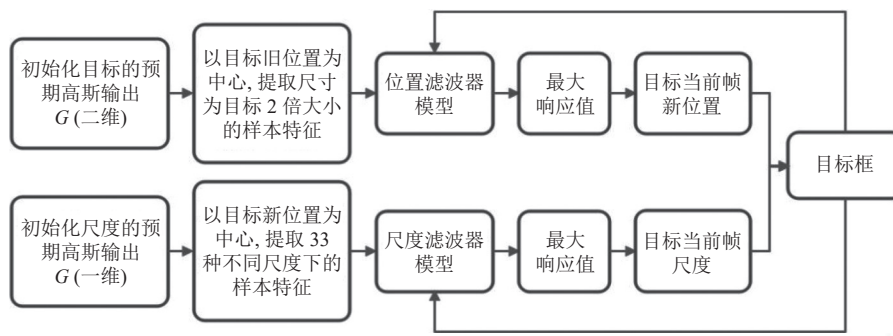


图2 DSST 算法流程图

判别尺度空间跟踪算法虽然可以较准确评估目标的位置和尺度, 不过该算法对行人行走过程中因形变和环境变化而导致的定位不准确的问题的处理能力较弱. 特别在行人快速行走过程中, 因滑动窗口提取的特征不完善, 易造成跟踪模型被污染、行人最终定位精度不高的问题.

### 3 改进方法

#### 3.1 基于 fhog 特征对原跟踪算法的位置进行纠正

在机器学习算法中, 特征工程是十分关键的. 相比于其他的特征描述方式, 原跟踪算法的 fhog 特征有许多优势. 因为 fhog 特征主要是对图像的局部区域通过处理得到的, 因此其在图像几何、光学变化方面都具

有很好的不变性. 另外, 在精确的方向抽样和较强的局部光学归一化等条件下, 若人的姿态一直是直立的状态, 如果做了微小的身体动作, 那么这种微小的姿态差异就不会影响行人检测的效果, 所以 fhog 特征描述子在行人检测领域特别适用.

为解决行人在行走过程中因形变和环境变化而导致的跟踪定位不准确问题, 本文在不改变判别尺度空间算法特征的前提下, 充分利用 fhog 特征在行人检测中的优势, 以原跟踪算法的位置滤波器所计算的位置为中心, 再次提取出目标的 fhog 特征并将其与位置滤波器模板进行相关运算, 最终目标位置在原算法基础上得到了纠正, 位置滤波器的模板也再次得到了更新. 其中, 算法改进后流程图如图 3 所示.

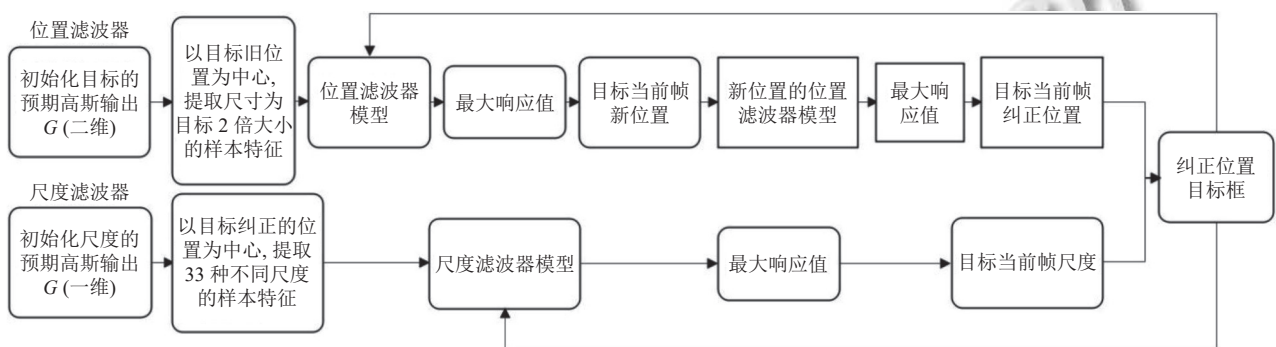


图3 基于 fhog 特征改进算法流程图

#### 3.2 基于纠正位置, Kalman 滤波算法对其预测和再次纠正

现实世界具有不确定性, 而系统的不确定性主要体现在 3 个方面: 不存在完美的数学模型、系统的扰动是不可控的、测量传感器会存在误差. 在具有不明确信号的系统中, Kalman 滤波对系统接下来的状态可以做出较有根据的估计, 即便系统伴随着各种干扰,

Kalman 滤波总能指出真实发生的情况.

Kalman 滤波基本方程是时域内的递推形式, 运算流程包含预测和纠正两个阶段, 解决问题时不需要储存大量信息. 如果观察到新数据就可算新滤波值, 占用存储空间小且速度快, 满足实时性, 适用于嵌入式系统, 可处理平滑随机过程又能处理多维问题, 所以 Kalman 思想当前被广泛用于图像信息处理领域.

### 3.2.1 Kalman 滤波简述

Kalman filter 假设:

1) 观测噪声和过程噪声都是满足高斯分布的。  
2) 状态转移方程和观测方程都是线性的, 预测和更新都建立在线性计算基础上的。

3) 先验估计和后验估计误差都符合期望为 0 的高斯分布。

4) 观测噪声和模型噪声分布并不随系统状态改变而发生变化。即过程噪声协方差矩阵  $Q$ , 观测噪声协方差矩阵  $R$  不变。

5) 初始状态下, 状态向量不受噪声影响。

定义 1. 定义一个随机离散时间过程的状态向量, 该过程可用一个离散随机差分方程式表述:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (7)$$

其中,  $x_k$  ( $n$ 维向量) 表示  $k$  时刻系统状态变量,  $x_{k-1}$  ( $n$  维向量) 表示  $k-1$  时刻系统状态变量,  $A$  ( $n \times n$ ) 表示状态转移矩阵, 即将  $k$  时刻和  $k-1$  时刻状态联系起来,  $B$  表示控制矩阵, 用来衡量控制量  $u$  如何作用于系统当前状态,  $w_{k-1}$  ( $n$ 维向量) 表示噪声 (对应  $x_k$  中每个分量噪声,  $w_k \sim N(0, Q)$ ,  $Q$  表示协方差矩阵)

定义 2. 定义一个观测变量:

$$Z_k = Hx_k + V_k \quad (8)$$

其中,  $Z_k$  ( $m$ 阶向量) 表示观测值,  $x_k$  ( $n$ 阶向量) 表示状态变量,  $H$  ( $m \times n$ 阶矩阵) 表示状态变量  $x_k$  对测量变量  $Z_k$  的增益,  $V_k$  表示观测噪声,  $V_k \sim (0, R)$ ,  $R$  表示协方差矩阵。

#### 3.2.1.1 Kalman 滤波预测阶段

根据前一刻后验估计值估计当前时刻的状态, 从而得到当前时刻的先验估计值:

$$\begin{cases} \hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \\ P_k^- = AP_{k-1}A^T + Q \end{cases} \quad (9)$$

其中,  $\hat{x}_k^-$ :  $k$  时刻的先验状态估计值, 即根据  $k-1$  时刻最优估计预测的  $k$  时刻的结果。  $A$ : 状态转移矩阵, 即通过  $k-1$  时刻如何推测  $k$  时刻状态的方法。  $B$ : 控制矩阵, 即控制量  $u$  如何作用于当前状态。  $\hat{x}_{k-1}$ :  $k-1$  时刻的后验状态估计值, 是滤波后结果, 即最优估计状态。  $P_{k-1}$ :  $k-1$  时刻的后验估计协方差矩阵, 用来衡量状态不确定度, 是滤波结果。  $P_k^-$ :  $k$  时刻的先验估计协方差矩阵, 是滤波中间结果。  $Q$ : 噪声协方差矩阵, 即状态转移矩阵与实际过程的误差。

#### 3.2.1.2 Kalman 滤波纠正阶段

通过当前时刻的观测值更正预测阶段的估计值, 即通过先验估计值得到当前的后验估计值, 其中卡尔曼滤波器状态的更新方程式如下所示:

$$\begin{cases} K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \\ \hat{x}_k = \hat{x}_k^- + K_k (Z_k - H\hat{x}_k^-) \\ P_k = (I - K_k H) P_k^- \end{cases} \quad (10)$$

其中,  $\hat{x}_k^-$ :  $k$  时刻的先验状态估计值, 根据  $k-1$  时刻最优估计  $k$  时刻的结果。  $P_k^-$ :  $k$  时刻的先验估计协方差矩阵, 是滤波中间结果。  $P_k$ :  $k$  时刻的后验估计协方差矩阵, 是滤波结果。  $H$ : 状态变量到观测的转换矩阵。  $R$ : 测量噪声协方差, 是与仪器特性相关数值。  $K_k$ : 卡尔曼系数。  $\hat{x}_k$ :  $k$  时刻的后验状态估计值, 是滤波结果。  $Z_k$ : 观测值, 是滤波输入值。  $Z_k - H\hat{x}_k^-$ : 实际观测与预测的残差, 用卡尔曼系数修正先验, 得到后验。

#### 3.2.2 Kalman 滤波算法在跟踪上的综合应用

本文将 DSST 算法作为行人的基础跟踪器, 除了将原算法的位置滤波器所计算的行人位置进行了纠正, 本文在纠正位置的基础上还引入 Kalman 滤波器。基于纠正位置, 通过 Kalman 滤波器的预测功能, 使得纠正位置再次得到了纠正, 最终使得行人的定位更加准确。

首先, 将第 1 帧中行人的位置初始化为  $(x_1, y_1)$ , 并将第 1 帧  $x$  轴、 $y$  轴 ( $v_{x1}, v_{y1}$ ) 的速度和  $(a_{x1}, a_{y1})$  的加速度设置为  $(0, 0)$ 。其次, 通过式 (11) 和式 (12) 可以估计 3 个连续帧之间的瞬时速度和加速度, 利用这些参数通过式 (13) 在第 2 帧中预测目标可能存在的区域。然后使用 DSST 跟踪算法的位置滤波器检测目标的位置  $(x_2, y_2)$ , 通过式 (11) 更新  $(v_{x2}, v_{y2})$ 。同理, 加速度  $(a_{x2}, a_{y2})$  可以由式 (12) 进行更新。最后, 用相同的方式不断地预测和检测视频序列中每一帧的目标的位置。

$$\begin{cases} v_{xk} = x_k - x_{k-1} \\ v_{yk} = y_k - y_{k-1} \end{cases} \quad (11)$$

$$\begin{cases} a_{xk} = v_{xk} - v_{xk-1} \\ a_{yk} = v_{yk} - v_{yk-1} \end{cases} \quad (12)$$

$$\begin{cases} P_{xk} = x_{k-1} + v_{xk-1} \cdot \Delta k + 0.5 \cdot a_{xk-1} \cdot \Delta k^2 \\ P_{yk} = y_{k-1} + v_{yk-1} \cdot \Delta k + 0.5 \cdot a_{yk-1} \cdot \Delta k^2 \end{cases} \quad (13)$$

其中,  $\Delta k$  表示时间间隔为 1,  $(x, y)$  表示 DSST 算法检测的结果,  $(P_x, P_y)$  表示通过目标的状态式 (13) 所预测的结果。

卡尔曼滤波器作为目标状态的估计策略, 假设目标跟踪算法的运动模型为以上匀加速模型, 则运动目

标和观测向量可以用如下线性方程进行描述:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (14)$$

$$Z_k = Hx_k + V_k \quad (15)$$

在以上两个方程中,  $x_k$  为视频序列第  $k$  帧的目标状态, 在式(14)中,  $A$  和  $B$  是卡尔曼模型的基本参数. 在式(15)中,  $Z_k$  为第  $k$  帧目标状态的观测值,  $H$  则为测量系统的基本参数. 在以上方程中,  $w_{k-1}$  和  $V_k$  依次为过程和测量噪声, 并且假定它们都是高斯噪声, 并假定它不随系统状况而变化,  $Q$  和  $R$  依次表示过程噪声和测量噪声的协方差.

### 3.2.2.1 Kalman 滤波算法预测阶段

对于满足上述条件的系统, 卡尔曼滤波器是最优的信息融合处理器. 首先, 利用目标的运动模型, 预估目标在下一帧状态下的位置; 其次, 式(16)可以根据目标的状态向量  $\hat{x}_{k-1}$  预测当前的位置; 最后, 当前协方差  $P_k^-$  可以通过式(17)进行更新.

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (16)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (17)$$

其中,  $\hat{x}_k^-$  为目标的当前预测位置,  $\hat{x}_{k-1}$  为  $k-1$  时刻目标位置优化的结果. 在式(17)中,  $P_k^-$  是相对于  $\hat{x}_k^-$  的协方差,  $P_{k-1}$  是相对于  $\hat{x}_{k-1}$  的协方差, 由匀加速运动模型可得  $A$ ,  $B$  均为二阶单位矩阵,  $A^T$  为  $A$  的转置,  $Q$  是在第 1 帧中所设置的运动模型的协方差.

### 3.2.2.2 Kalman 滤波算法纠正阶段

以 DSST 的位置滤波器所检测到的目标位置作为测量值  $Z_k$ , 根据匀加速运动模型所预测结果的  $\hat{x}_k^-$ 、测量值  $Z_k$  以及式(18)所估计的卡尔曼增益, 再利用式(19)就可以得到目标当前的最优的状态估计  $\hat{x}_k$ , 为了使卡尔曼滤波器一直运行到视频序列的最后一帧, 可以通过式(20)来更新每一帧所对应的协方差  $P_k$ .

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (18)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(Z_k - H\hat{x}_k^-) \quad (19)$$

$$P_k = (I - K_k H) P_k^- \quad (20)$$

其中,  $K_k$  为当前帧中的卡尔曼增益,  $I$ ,  $H$  表示单位矩阵,  $R$  表示测量误差. 简而言之,  $Q$  和  $R$  也能分别作为预测值和测量值的置信度, 当  $R$  越大, 测量值的置信度就越小, 当  $Q$  越大, 预测值的置信度就越大. 通常来说, 过程噪声  $Q$  准确获取较困难, 可通过控制变量法进行对比实验获取. 测量的噪声  $R$  则与传感器检测准确度有关, 可通过对传感器传输的数据长期统计分析以确定其实测方差, 其与传感器实际的方差接近.

卡尔曼预测阶段负责目标位置预测和噪声协方差更新. 首先, 设定前一帧中所使用的目标运动状态向量为  $(v_{x_{k-1}}, v_{y_{k-1}}, a_{x_{k-1}}, a_{y_{k-1}})$ , 通过式(13)来预测下一帧中目标的位置  $(P_{x_k}, P_{y_k})$ , 同时利用二维卡尔曼位置滤波器来消除高斯噪声, 然后通过 DSST 中的位置相关滤波器来检测目标的位置, 再利用其检测的结果  $(x_k, y_k)$  通过式(11)和式(12)对平面内运动模型状态向量  $(v_{x_k}, v_{y_k}, a_{x_k}, a_{y_k})$  进行更新. 采用同样的方法可以不断地预测和检测当前帧目标的状态向量, 并且不断更新当前帧目标的最优状态估计  $\hat{x}_k$ , 卡尔曼增益  $K_k$  和协方差矩阵  $P_k$ , 直至视频序列的最后一帧.

本文采用 DSST 算法作为行人跟踪器, 将纠正位置方法和 Kalman 滤波器融合到其中, 融合之后的跟踪算法结构如前文图 1 所示. 其中除正则化项  $\lambda$ 、搜索填充区域和尺度因子  $\alpha$  外, 所有参数均与标准的 DSST 算法相同. 在所提出的跟踪器中, 正则化参数  $\lambda$  设置为 0.02, 填充区域设置为 2.0, 尺度因子设置为 1.02. 对于二维位置卡尔曼滤波器, 初始时的过程噪声和测量噪声的协方差均设置为  $Q=[25, 10, 1]$ ,  $R=25$ . 当过程噪声  $Q$  和测量噪声  $R$  取不同的值时, 会对跟踪结果产生不同影响.

## 4 算法改进前后结果分析

### 4.1 跟踪评价指标

在国际标准跟踪评测中, 跟踪器的性能通常由中心位置误差 (center location error,  $CLE$ ) 和距离精确度 (distance precision,  $DP$ ) 和重叠精确度 (overlap precision,  $OP$ ) 这 3 个度量指标来评价<sup>[24]</sup>.

#### 4.1.1 中心位置误差 ( $CLE$ ) 和距离精度 ( $DP$ )

对一个长度为  $T$  的视频序列, 如果假定目标在某一帧  $t$  ( $t=1, \dots, T$ ) 的标记的真实值的中心点位置坐标是  $(x_t^s, y_t^s)$ , 而跟踪框的中心点位置是  $(x_t^o, y_t^o)$ , 那么中心位置误差就定义为二者之间的欧式距离:

$$CLE = \sqrt{(x_t^s - x_t^o)^2 + (y_t^s - y_t^o)^2} \quad (21)$$

$CLE$  主要测量的是目标的跟踪框坐标与标记的真实坐标之间的像素差异, 能够在一定程度上反应跟踪算法的精确度.  $CLE$  值越小, 则表示跟踪算法对该帧目标的跟踪精度越高. 为获取算法在整个视频序列中的总体性能, 可统计其在整个视频序列中的中心位置误差, 然后再取平均值得到平均中心位置误差. 一般来说, 算法在视频序列上的平均中心位置误差越小, 跟踪算



法的性能越好. 虽然  $CLE$  简单常用, 但它没有考虑目标的旋转和尺度变化, 因此当目标在跟踪过程中出现较大尺度或旋转变化时, 它并不能精确地评价跟踪算法的精度. 所以引入了距离精度  $DP$ , 其表示  $CLE$  低于某一个阈值范围的帧数占视频序列的总帧数的比例, 其中实验的阈值选为 20 pixels.

#### 4.1.2 重叠精度 ( $OP$ )

重叠精度  $OP$  是另一个被广泛使用的评价标准, 反映的是每一帧预测的目标区域  $B'_i$  和真实标记  $B'_j$  之间交集与并集的比值:

$$OP = \frac{Area(B'_i \cap B'_j)}{Area(B'_i \cup B'_j)} \quad (22)$$

其中,  $B'_i \cap B'_j$  和  $B'_i \cup B'_j$  如图 4 所示.

由上图可知重叠精度的计算结果在越接近 1, 说明跟踪算法预测的目标区域与实际的目标区域重合度越高, 算法的跟踪精度越高. 同样的, 可以计算跟踪算法在测试视频所有帧的重叠精度的平均值 (即平均重叠精度) 来评价其在视频序列上的整体表现, 平均重叠精度越大, 算法的性能越好. 重叠精度考虑了目标旋转、尺度变换等因素, 能够更好地反映跟踪结果

的精确度, 所以综合以上 3 种评价方法可以更好地评价跟踪结果.

#### 4.2 算法改进前后结果对比分析

本文挑选 OTB-100 中的行人数据集在 Matlab 软件进行实验. 在这些视频序列中, 选择了 8 组数据集进行效果展示, 其中改进前后的跟踪效果如图 5 所示, 序列中绿色的跟踪框表示原算法的跟踪效果, 红色的跟踪框表示改进后的跟踪效果, 图片的左上角数字表示视频序列的帧数.

为了更精确验证本文所提出策略的有效性, 将标准 DSST 算法与位置经过双重纠正后的 DSST 算法通过距离精度  $DP$ , 重叠精度  $OP$ , 中心位置误差  $CLE$  这 3 个指标进行比较, 比较数据如图 6 和表 1 所示.



图 4 目标区域和真实标记区域的并集与交集



图 5 行人数据集

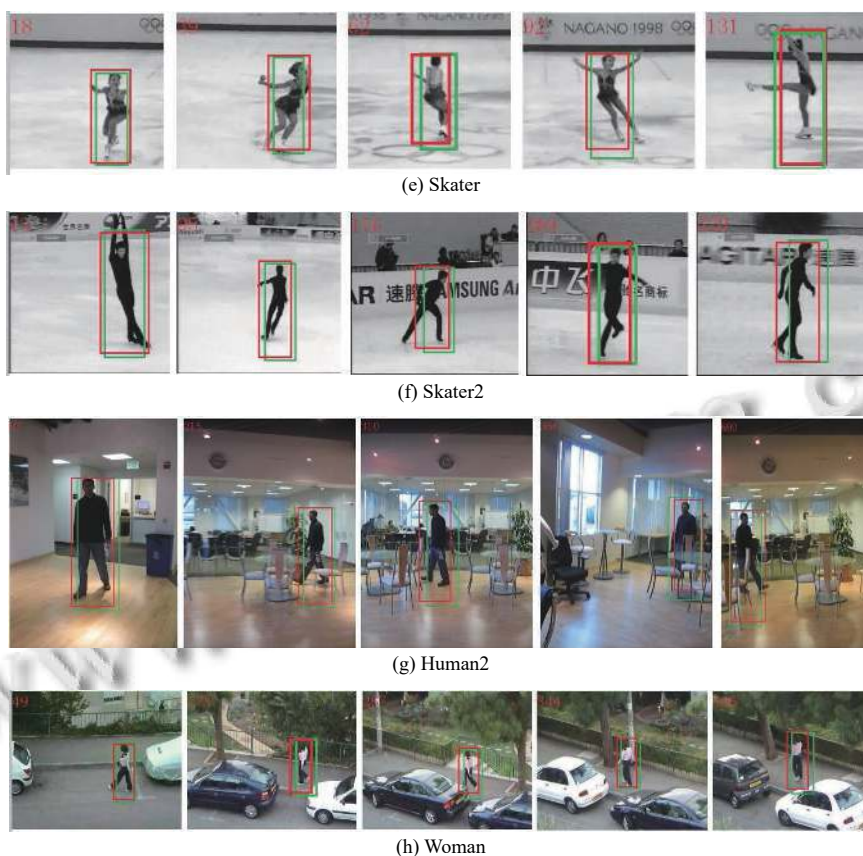


图5 行人数据集(续)

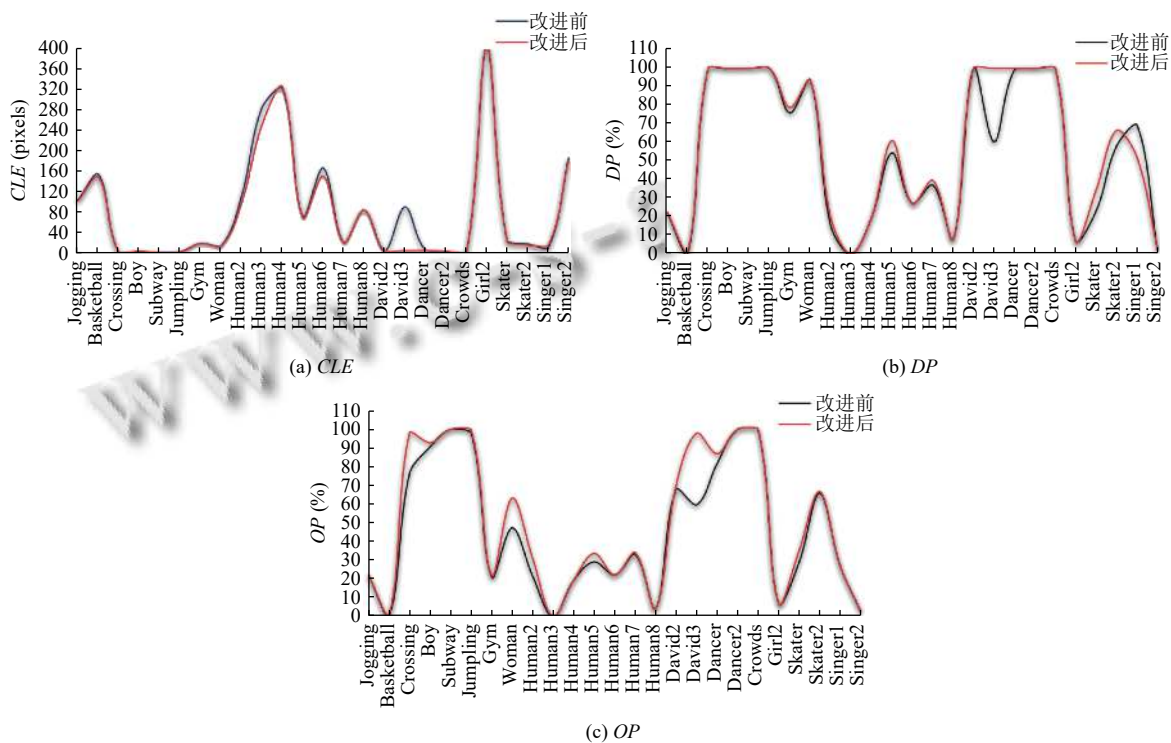


图6 改进前后比较数据



表1 改进前后比较数据

数据集	CLE (pixels)		DP (%)		OP (%)	
	改进前	改进后	改进前	改进后	改进前	改进后
Jogging	106	104	22.8	23.1	22.5	22.5
Basketball	156	151	1.1	2.48	1.1	1.66
Crossing	4.01	3.41	100	100	76.7	98.3
Boy	5.5	5.31	99.8	100	90.7	92.9
Subway	3.22	2.15	100	100	100	100
Jumpling	4.77	3.73	100	100	98.1	99.7
Gym	21.3	20.6	76.1	79	21.6	22.4
Woman	16.1	15.3	93.6	93.8	47.6	63.5
Human2	113	99.9	15.9	20.3	22	31.4
Human3	286	252	0.648	0.648	0.353	0.412
Human4	324	327	20.2	20.2	19.8	19.8
Human5	74	71.8	54.7	61.3	29.6	34.1
Human6	168	152	27.7	27.4	22.6	22.6
Human7	23.3	22.5	37.6	40	33.6	34.4
Human8	86.2	85.9	8.59	8.59	4.69	4.69
David2	6.94	6.68	100	100	68.2	70.4
David3	92	6.89	60.7	100	59.9	98
Dancer	8.08	7.39	99.1	100	81.8	87.1
Dancer2	5.28	5.57	100	100	100	100
Crowds	3.97	2.95	100	100	100	100
Girl2	404	405	7.53	7.53	7.47	7.4
Skater	25.8	24.6	23.8	35	29.4	35.6
Skater2	20.1	18.2	58.4	66.7	66	66.9
Singer1	12.1	17.2	69.5	52.4	27.6	27.6
Singer2	188	183	3.01	3.28	3.28	3.55

表1和图6表明,与标准的DSST算法相比,所提改进方案的跟踪性能得到了提升,尤其在数据集 Crossing, David3, Dancer, Crowds, Skater, Skater2, Woman, Human2和Human5中改进方案的跟踪性能显著提升.由于行人快速移动和图像出现模糊时,利用 $fhog$ 特征在行人检测上的优势,通过在原算法位置上再次提取 $fhog$ 特征并进行相关运算,能够使得跟踪目标的搜索范围再次扩大从而提取了更精确、更完整的行人特征描述子,最后通过卡尔曼滤波对行人位置的预测和纠正功能,使得位置在原算法的基础上得到了双重纠正,最终所更新的位置滤波器模板也更大程度上使得跟踪性能得到提高.

## 5 总结

本文针对判别尺度空间跟踪算法在目标跟踪中出现的问题,对算法进行了以上方式的改进.首先,通过 $fhog$ 特征适用于行人检测的优势<sup>[25]</sup>,在不改变该算法特征和不大幅度降低算法速度的前提下,以位置滤波器计算的位置为中心,使用滑动窗口再次提取行人的 $fhog$ 特征,并将其与位置滤波器模板进行相关运算后,

得到纠正后的行人位置,然后以纠正位置为中心训练位置滤波器模型,从而使得行人的定位精度在原算法基础上得到了提升.其次,针对行人行走过程中出现快速移动和环境变化的问题,本文引入了具有数据融合功能的Kalman滤波算法,通过其加入,行人的定位再次得到了纠正,最终跟踪精度也随之提升.

## 参考文献

- Collins RT, Lipton AJ, Kanade T, et al. A system for video surveillance and monitoring. Technical Report, Pittsburgh: Carnegie Mellon University, 2002.
- Fukunaga K, Hostetler L. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory, 1975, 21(1): 32–40. [doi: 10.1109/TIT.1975.1055330]
- Cheng YZ. Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 17(8): 790–799. [doi: 10.1109/34.400568]
- Babenko B, Yang MH, Belongie S. Robust object tracking with online multiple instance learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(8): 1619–1632. [doi: 10.1109/TPAMI.2010.226]

- 5 Bolme DS, Beveridge JR, Draper BA, *et al.* Visual object tracking using adaptive correlation filters. Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Francisco: IEEE, 2010. 2544–2550.
- 6 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: Springer, 2012. 702–715.
- 7 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)]
- 8 Danelljan M, Häger G, Kan FS, *et al.* Accurate scale estimation for robust visual tracking. Proceedings of the 2014 British Machine Vision Conference. BMVA Press, 2014. [doi: [10.5244/C.28.65](https://doi.org/10.5244/C.28.65)]
- 9 Kalman RE. A new approach to linear filtering and prediction problems. Journal of Basic Engineering, 1960, 82(1): 35–45. [doi: [10.1115/1.3662552](https://doi.org/10.1115/1.3662552)]
- 10 李由, 张恒, 李立春. 基于多测量融合的粒子滤波跟踪算法. 国防科技大学学报, 2007, 29(5): 26–30. [doi: [10.3969/j.issn.1001-2486.2007.05.007](https://doi.org/10.3969/j.issn.1001-2486.2007.05.007)]
- 11 李文姣, 秦勃. 基于卡尔曼滤波器的抗遮挡车辆跟踪算法. 计算机应用, 2012, 32(S2): 174–176, 179.
- 12 Ettliger A, Neuner H, Burgess T. Development of a Kalman filter in the Gauss-Helmert model for reliability analysis in orientation determination with smartphone sensors. Sensors, 2018, 18(2): 414. [doi: [10.3390/s18020414](https://doi.org/10.3390/s18020414)]
- 13 李军, 曹梦婷. 基于置信图峰值旁瓣比优化的遮挡目标跟踪方法. 中国, 108986142A. 2018-12-11.
- 14 张红颖, 王赛男, 胡文博. 基于分块颜色直方图的MS跟踪算法. 中国民航大学学报, 2019, 37(1): 39–45. [doi: [10.3969/j.issn.1674-5590.2019.01.008](https://doi.org/10.3969/j.issn.1674-5590.2019.01.008)]
- 15 火元莲, 李明, 曹鹏飞, 等. 基于深度特征与抗遮挡策略的运动目标跟踪. 西北师范大学学报(自然科学版), 2020, 56(4): 49–56. [doi: [10.16783/j.cnki.nwnuz.2020.04.008](https://doi.org/10.16783/j.cnki.nwnuz.2020.04.008)]
- 16 Ma C, Huang JB, Yang XK, *et al.* Hierarchical convolutional features for visual tracking. Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015. 3074–3082.
- 17 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556, 2015.
- 18 宦若虹, 王楚, 陶一凡, 等. 一种抗遮挡的基于积分直方图的粒子滤波目标跟踪方法. 中国, 107507222A. 2021-01-01.
- 19 Haritaoglu I, Harwood D, Davis LS.  $W^4$ : Real-time surveillance of people and their activities. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8): 809–830. [doi: [10.1109/34.868683](https://doi.org/10.1109/34.868683)]
- 20 Dalal N, Triggs B. Histograms of oriented gradients for human detection. Proceedings of the 2015 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego: IEEE, 2005. 886–893.
- 21 张云洲, 郑瑞, 暴吉宁, 等人. 基于多个相关滤波器的行人跟踪尺度算法. 东北大学学报(自然科学版), 2019, 40(9): 1228–1233, 1239.
- 22 Zhang YH, Yang YJ, Zhou WN, *et al.* Motion-aware correlation filters for online visual tracking. Sensors, 2018, 18(11): 3937. [doi: [10.3390/s18113937](https://doi.org/10.3390/s18113937)]
- 23 Xu LY, Luo HB, Hui B, *et al.* Real-time robust tracking for motion blur and fast motion via correlation filters. Sensors, 2016, 16(9): 1443. [doi: [10.3390/s16091443](https://doi.org/10.3390/s16091443)]
- 24 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)]
- 25 Felzenszwalb PF, Girshick RB, McAllester D, *et al.* Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(9): 1627–1645. [doi: [10.1109/TPAMI.2009.167](https://doi.org/10.1109/TPAMI.2009.167)]

(校对责编: 牛欣悦)