

DOI: 10.13382/j.jemi.2017.03.009

结合 SURF 与 Kalman 滤波的 CAMShift 跟踪算法*

张磊 彭力

(江南大学 物联网学院 无锡 214122)

摘要:针对传统的 CAMShift 目标跟踪算法,在出现颜色干扰,遮挡等复杂背景中容易跟丢的问题,提出了一种结合 SURF 特征匹配与 Kalman 滤波的 CAMShift 跟踪算法。该算法利用 CAMShift 算法跟踪得到的候选目标与模板目标的色度和梯度方向的综合直方图比较计算得到的 Bhattacharyya 系数作为判定依据,当系数大于给定阈值时,采用 SURF 算法对搜索窗口和上一帧跟踪结果进行特征匹配,重新计算目标的大小和位置。同时为了避免目标快速运动时跟踪失败和减少 SURF 匹配的计算量,利用 Kalman 滤波对运动目标窗口进行预测更新以确定下一帧搜索窗口的中心位置。实验表明,该算法在图像背景复杂,出现颜色干扰以及部分遮挡时能够稳定跟踪,其跟踪速度与结合 SURF 的 CAMShift 算法相比有显著提高。

关键词: 目标跟踪; CAMShift 算法; Kalman 滤波; SURF 算法; Bhattacharyya 系数

中图分类号: TP391.4; TN911.73 **文献标识码:** A **国家标准学科分类代码:** 510.40

Camshift tracking algorithm of combined with SURF and Kalman filter

Zhang Lei Peng Li

(Institute of Networking Engineering, Jiangnan University, Wuxi 214122, China)

Abstract: In this paper, a tracking algorithm based on CAMShift which combined with SURF feature matching and Kalman filter is proposed to deal with the problems in traditional CAMShift algorithm, such as tracking failure under color interference or occlusion. The algorithm calculates the Bhattacharyya coefficient of integrated histogram composed of chroma feature and gradient direction feature between candidate target and template target as judging basis, it uses CAMShift algorithm. As the coefficient more than the threshold, SURF algorithm will be used to match the search window and the tracking result of the previous frame, then recalculate the target's size and position by the matching result. To avoid tracking failure by fast-moving of target and reduce the computation of SURF matching, the center position of moving target in the next frame will be predicted by Kalman predictor. The experimental results show that the new algorithm can achieve stable tracking object against complex backgrounds, color interference or occlusion, and have higher tracking speed than CAMShift algorithm combined with SURF.

Keywords: object tracking; CAMShift algorithm; Kalman filtering; SURF algorithm; Bhattacharyya coefficient

1 引言

目前,运动目标的检测和跟踪已成为计算机视觉领域最重要的应用之一。尽管已经出现了非常多的方法,但是如何快速、有效地跟踪到运动目标,仍然是一个非常值得研究的课题。先进的图像处理技术和一些开源代码

使得许多跟踪算法得以改进并实现, CAMShift (continuously adaptive mean-shift) 算法就是其中的一种^[1]。

传统的 CAMShift 算法是一种基于颜色信息的半自动检测算法。该算法原理简单,具有稳定性、实时性和较强的鲁棒性,在简单背景中有较好的跟踪效果。但 CAMShift 算法跟踪开始前需要手动的选择图像中的一块

区域作为实时连续跟踪的感兴趣区域。而且当背景中出现遮挡,相似颜色干扰,或在目标快速运动时容易跟丢。根据以上存在的问题,许多人对 CAMShift 算法做了大量改进。为了解决目标在快速运动或者被遮挡时的跟踪失败问题,Huang 等人^[2]提出利用 Kalman 滤波器对当前帧中运动目标的位置和速度进行预测。但是此种方法当目标和背景颜色相近时,或者光照变化大时跟踪效果却不理想。为了解决相近颜色干扰问题,Li^[3]提出将 SURF (speed-up robust features) 特征点匹配融入到 CAMShift 算法中,弥补了传统 CAMShift 算法只以颜色为关键信息的不足。但是 SURF 算法计算相对复杂,运行耗时过长不能满足实时跟踪的要求且当目标出现快速运动时容易跟踪失败^[4]。

针对以上问题本文给出了在 CAMShift 算法基础上结合 SURF 特征匹配和 Kalman 滤波的目标跟踪解决方案。根据 CAMShift 跟踪结果与目标模板之间色度与梯度方向综合直方图的比较,来判断 CAMShift 的跟踪结果是否准确。如果比较结果大于给定阈值则认为跟踪失败,然后对 Kalman 预测得到的当前帧的搜索窗口和上一帧跟踪得到的目标窗口利用 SURF 算法进行特征匹配,计算出目标在搜索窗口中的位置和大小,并将结果更新给 Kalman 滤波器。实验证明本文提出的方法能有效地跟踪运动目标,并达到了实时性的要求。

2 算法分析

2.1 利用 CAMShift 算法进行初步跟踪

CAMShift 算法将连续视频序列中运动目标的颜色信息作为主要的跟踪依据^[5]。算法的主要思想如下:

- 1) 初始化搜索窗口的大小和位置;
- 2) 计算搜索窗口的反向投影图;
- 3) 利用 Meanshift 算法进行迭代,获得新的搜索窗口的位置和大小;
- 4) 重复以上过程,实现对目标的连续跟踪。

CAMShift 算法将视频图像的所有帧作 Mean-Shift 运算,并将上一帧的结果作为下一帧 Mean-Shift 算法的搜索窗口的初始值,并一直迭代下去,当目标的大小发生改变时,该算法可以自适应调整目标跟踪区域的大小^[6]。但 CAMShift 算法将颜色作为目标的主要特征,忽略了其空间分布特征,当图像背景复杂,出现相似目标颜色像素干扰或者目标发生转动和形变时,将导致跟踪失败^[7]。为了有效地解决这一问题,本文算法对 CAMShift 的初步跟踪结果进行了判定,如果判定结果认为 CAMShift 算法跟踪失败则采用改进的 SURF 算法重新跟踪。

2.2 CAMShift 跟踪结果判定

CAMShift 算法在运动目标颜色特征与背景颜色特征

差别不大或目标附近有与目标色度相近的物体时,往往会失去跟踪目标或者跟踪目标不准确。因此判断跟踪结果是否准确仅依靠颜色特征是不够的。本文受参考文献^[8]启发,在 CAMShift 跟踪结果的判定上给出了一种利用图像色度特征和梯度方向特征综合直方图的 Bhattacharyya 距离作为参考目标区域与候选区域相似性程度的解决方案^[8]。

1) 色度直方图

CAMShift 算法将图像由 RGB 颜色空间转化到 HSV 颜色空间^[3],并仅提取出其中的色度 H 分量,将其划分为 16 级,每级 16 个色度,共 48 个子特征。根据 Mean-Shift 跟踪算法原理^[9-10],目标区域的颜色特征直方图分布为:

$$\hat{P}_u(y) = C_h \sum_{i=1}^{n_h} k \left(\left\| \frac{y - x_i}{h} \right\| \right)^2 \delta[b_1(x_i) - u] \quad (1)$$

$1 \leq u \leq 48$

式中: $b_1(x_i)$ 为像素 x_i 颜色空间的特征值,归一化常数 C_h 可由 $\sum_{i=1}^{n_h} \hat{P}_u(y) = 1$ 推出, $\delta[b_1(x_i) - u]$ 为单位脉冲函数。

2) 梯度方向直方图

梯度方向可以用来描述图像的结构信息。当两个物体重叠时,边界处会出现较大的梯度差异,颜色和梯度方向的结合能够提高匹配准确性。将灰度梯度方向空间等分为 16 级,每 $\pi/8$ 角度为一级。

要求取梯度方向直方图,首先要计算目标区域的色度图像,然后计算该色度图内所有像素点 $I(h,w)$ 的梯度 $d(h,w)$ 和方向 $ori(h,w)$,计算公式如下^[11]:

$$\begin{cases} d_x = I(h,w+1) - I(h,w-1) \\ d_y = I(h+1,w+1) - I(h,w-1) \\ d(h,w) = \sqrt{(d_x)^2 + (d_y)^2} \\ ori(h,w) = \text{atan2}(d_y, d_x), ori \in [-\pi, \pi] \end{cases} \quad (2)$$

式中: d_x 表示像素点 $I(h,w)$ 水平方向两相邻点色度值之差, d_y 表示像素点 $I(h,w)$ 垂直方向两相邻点色度值之差; $ori(h,w)$ 大小范围为 $[-\pi, \pi]$, $\text{atan2}(d_x, d_y)$ 表示以原点为起点, x 轴正方向与射线 (d_x, d_y) 间的角度。目标区域的梯度方向直方图分布为^[12]:

$$\hat{P}_u(y) = C_h \sum_{i=1}^{n_h} k \left(\left\| \frac{y - x_i}{h} \right\| \right)^2 \delta[b_2(x_i) - u] \quad (3)$$

$49 \leq u \leq 64$

式中: $b_2(x_i)$ 为像素 x_i 的灰度梯度方向的特征值。可以获得综合直方图分布:

$$\hat{P}_u(y) = \begin{cases} C_h \sum_{i=1}^{n_h} k \left(\left\| \frac{y - x_i}{h} \right\| \right)^2 \delta[b_1(x_i) - u], 1 \leq u \leq 48 \\ C_h \sum_{i=1}^{n_h} k \left(\left\| \frac{y - x_i}{h} \right\| \right)^2 \delta[b_2(x_i) - u], 49 \leq u \leq 64 \end{cases} \quad (4)$$

得到 CAMShift 跟踪结果和目标模板的综合颜色直方图后计算两者的 Bhattacharyya 距离,如果大于设定阈值,则认为此帧经过 CAMShift 跟踪后得到的结果不准确,而上一帧是准确的。此时采用 SURF 算法将上一帧的准确跟踪结果和此帧图像进行特征匹配,对此帧图像中目标重新定位跟踪。

2.3 SURF 算法重定位

SURF 算法是一种尺度不变特征变换算法,可用来处理两幅图片之间的匹配问题,对图像的旋转、平移、尺度缩放、亮度变化、遮挡、和噪声等具有良好的不变性^[13]。利用 SURF 算法可以有效定位运动目标在下一帧中的位置。

SURF 特征提取主要分为 5 个主要步骤:构建 Hessian 矩阵,构建图像尺度空间,精确定位特征点,确定特征点主方向,生成特征点描述子^[14]。特征提取完成后,将上一帧运动目标所在窗口图像,与当前帧图像根据优化后的特征点描述子进行匹配,对当前帧图像中的目标进行定位。

SURF 算法作为 SIFT 算法的加速版,虽然已经比 SIFT 算法快了 3 倍以上^[15]。但是在对较大尺寸图像中的目标进行跟踪时,由于算法的复杂度,依然无法做到实时跟踪。为了解决这一问题,本文将 Kalman 滤波算法和 SURF 算法相结合,利用 Kalman 算法预测运动目标在下一帧中的位置,只对预测位置附近的图像与上一帧图像中目标所在窗口中图像进行 SURF 特征匹配,精确定位目标在预测窗口中的位置,从而减小 SURF 算法处理图像尺寸的大小,进一步增强 SURF 跟踪算法的稳定性和快速性,定位效果如图 1 所示。



图 1 SURF 算法重定位结果

Fig. 1 The results of relocation using SURF algorithm

由图 1 可以看出,当 CAMShift 算法在图像背景出现相似颜色干扰而导致跟踪失败时,利用 SURF 算法对目标特征点进行匹配,可以得到目标在当前帧搜索窗口中的位置和大小,有效地将跟踪继续下去。

2.4 利用 Kalman 滤波器预估目标位置

Kalman 滤波是一种基于误差方差最小准则的递推线性最小方差估计算法^[16]。它利用线性系统状态方程,通过输入输出的观测数据,对系统的状态进行最优估计。系统状态方程和观测方程分别为:

$$\mathbf{X}_k = \mathbf{A}_{k,k-1}\mathbf{X}_{k-1} + \mathbf{V}_{k-1} \quad (5)$$

$$\mathbf{Y}_k = \mathbf{H}_k\mathbf{X}_k + \mathbf{W}_{k-1} \quad (6)$$

式中: $\mathbf{A}_{k,k-1}$ 表示系统状态转移矩阵, \mathbf{H}_k 为系统的观测矩阵, \mathbf{X}_k 、 \mathbf{X}_{k-1} 分别是 k 时刻、 $k-1$ 时刻系统的状态向量,过程噪声向量 \mathbf{V} 和观测噪声向量 \mathbf{W} 均为均值为 0 的白噪声序列,它们互不相关,协方差矩阵分别为 \mathbf{Q} 和 \mathbf{R} 。

由于在进行目标跟踪时 2 帧之间时间间隔较短,目标运动状态不会发生较大改变,所以可以将目标的状态近似为匀速运动。系统的状态向量设为 $\mathbf{X}_k = (x_{sx}, y_{sk}, x_{vx}, y_{vk})$ 。其中 x_{sk} 、 y_{sk} 表示目标质心在 X 轴和 Y 轴上的坐标, x_{vk} 、 y_{vk} 分别表示目标质心在 X 轴和 Y 轴上的速度。定义观测向量 $\mathbf{Y}_k = [x_{tk}, k, y_{tk}]$ 。具体推导公式为:

$$\mathbf{X}_k = \mathbf{A}_{k,k-1}\mathbf{X}_{k-1} \quad (7)$$

$$\mathbf{X}_k = \mathbf{X}_{k,k-1} + \mathbf{K}_k[\mathbf{Y}_k - \mathbf{H}_k\mathbf{X}_{k,k-1}] \quad (8)$$

$$\mathbf{K}_k = \mathbf{P}_{k,k-1}\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_{k,k-1}\mathbf{H}_k^T + \mathbf{R})^{-1} \quad (9)$$

$$\mathbf{P}_{k,k-1} = \mathbf{A}_{k,k-1}\mathbf{P}_{k-1}\mathbf{A}_{k,k-1}^T + \mathbf{Q} \quad (10)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k,k-1} \quad (11)$$

式中: $\mathbf{X}_{k,k-1}$ 为状态的预测值, \mathbf{X}_k 为 $\mathbf{X}_{k,k-1}$ 的修正值, \mathbf{K}_k 为 Kalman 滤波增益矩阵, $\mathbf{P}_{k,k-1}$ 为 $\mathbf{X}_{k,k-1}$ 的协方差矩阵, \mathbf{P}_k 为 \mathbf{X}_k 的协方差矩阵^[17]。

基于 Kalman 滤波预测器对图像序列中运动目标估计过程如下:

- 1) 利用目标初始位置 x_0 初始化滤波器,该值为手动选择得到的目标或检测得到的目标的位置和速度,若速度未知,可设为 0,并记录当前时刻;
- 2) 利用当前的运动状态和误差协方差,预测目标在下一帧中的位置和速度,获得先验估计值;
- 3) 利用反馈的方法,结合新的实际观测值与先验估计值获取后验估计,并将其作为本帧的目标位置和速度信息,更新滤波器,返回步骤 2)。

Kalman 滤波器可以根据目标的速度和位置信息预测下一帧目标可能出现的位置,不仅缩小了搜索范围,减少了运算量,同时解决了目标快速运动时跟踪不准确的问题。

3 算法实现

本文提出的改进算法在开始跟踪时采用 CAMShift 算法,根据 CAMShift 跟踪结果与目标模板之间色度与梯度方向综合直方图的比较,来判断 CAMShift 的跟踪结果是否准确。如果大于给定阈值则认为 CAMShift 跟踪失败。然后对 Kalman 预测得到的当前帧的搜索窗口和上一帧跟踪得到的目标窗口利用 SURF 算法进行特征匹配,计算出目标的位置和大小,并将结果更新给 Kalman 滤波器,循环实现目标的连续跟踪,算法具体实现步骤如下。

- 1) 读取视频帧,将图像由 RGB 转化到 HSV 色彩空间。

2) 选取图像中运动目标所在的感兴趣区域, 设定初始跟踪窗口作为模板目标。

3) 利用模板目标跟踪窗口中心位置初始化 Kalman 滤波器。

4) 将上一帧计算得到运动目标的大小和利用 Kalman 滤波器预测出的当前帧运动目标的位置作为当前帧 CAMShift 算法的输入窗口, 选取比当前输入窗口稍大的窗口作为搜索窗口, 计算搜索窗口内图像 H 分量的反向投影图, 利用 CAMShift 算法跟踪得到候选目标的大小和位置。

5) 获取候选目标窗口和模板目标窗口颜色和梯度方向上的综合直方图, 并比较两直方图得到其 Bhattacharyya 距离, 当 Bhattacharyya 距离大于给定阈值时, 认为 CAMShift 跟踪结果不准确进入步骤 6), 小于阈值时认为跟踪结果准确进入步骤 7)。

6) 利用 SURF 算法对上一帧的跟踪结果和当前帧搜索窗口进行检测, 得到两幅图像的特征向量。利用 FLANN 匹配器对两组特征向量进行匹配, 计算匹配特征点间的最小距离, 并去除距离大于 3 倍最小距离的特征点, 然后用 RANSAC 算法对匹配的特征点进行优化。最终得到的特征点即为相对准确的特征点。利用优化后的特征点求两幅图像的单应性矩阵, 并将上一帧跟踪结果图像的 4 个角点(左上角, 右上角, 左下角, 右下角) $(x_{(k-1)_i}, y_{(k-1)_i}), i = 1, 2, 3, 4$ 和中心点 $(x_{(k-1)_c}, y_{(k-1)_c})$ 通过仿射变换找到其在当前搜索窗口中的位置为 $(x_{k_i}, y_{k_i}), i = 1, 2, 3, 4$ 和 (x_{k_c}, y_{k_c}) 。假定优化得到的上一帧目标的特征点集为:

$$P = \{(x_i, y_i), V_i\} \quad i = 1, 2, \dots, N \quad (12)$$

当前帧搜索窗口特征点集为:

$$P' = \{(x'_i, y'_i), V'_i\} \quad i = 1, 2, \dots, N \quad (13)$$

本文采取平均值的方法, 计算搜索窗口中候选目标位置与上一帧目标位置的中心偏移量, 如式(20)所示。

$$(\Delta x, \Delta y) = \frac{1}{N} \sum_{i=1}^N [(x'_i, y'_i) - (x_i, y_i)] \quad (14)$$

当前帧候选目标的位置取上一帧目标中心位置加上计算出的中心偏移量后与当前帧 SURF 算法匹配结果的中心位置的平均值:

$$(X_k, Y_k) = \left(\frac{(X_{k-1} + \Delta x) + (x_{k_s} + u)}{2}, \frac{(Y_{k-1} + \Delta y) + (y_{k_s} + v)}{2} \right) \quad (15)$$

式中: (X_{k-1}, Y_{k-1}) 为上一帧目标的中心位置。 u 、 v 分别为当前搜索窗口左上角点的横、纵坐标。当前帧候选目标的大小为:

$$(W_k, H_k) = \left(\frac{(x_{k_2} - x_{k_1}) + (x_{k_4} - x_{k_3})}{2}, \right.$$

$$\left. \frac{(y_{k_3} - x_{k_1}) + (y_{k_4} - y_{k_2})}{2} \right) \quad (16)$$

式中: W_k 、 H_k 分别为候选目标在当前帧图像中的宽和高。

7) 将 CAMShift 算法或 SURF 算法计算得到的当前帧候选目标的大小和位置作为当前帧运动目标的大小和位置。并将候选目标的位置更新给 Kalman 滤波器, 将候选目标的大小作为下一帧 CAMShift 算法输入窗口的大小。

8) 标记出运动目标在当前帧图像中的大小和位置, 并进入步骤 4) 直到跟踪结束。

4 实验结果及分析

本算法在 Intel i3-4130、内存 4G 的 PC 上用 VS2010 加装 OpenCV 4.11 实现, 对摄像头采集到的视频数据分别采用 CAMShift 算法、文献[3]中 CAMShift 与 Kalman 结合算法、本文算法进行目标跟踪, 跟踪结果如图 2 所示。

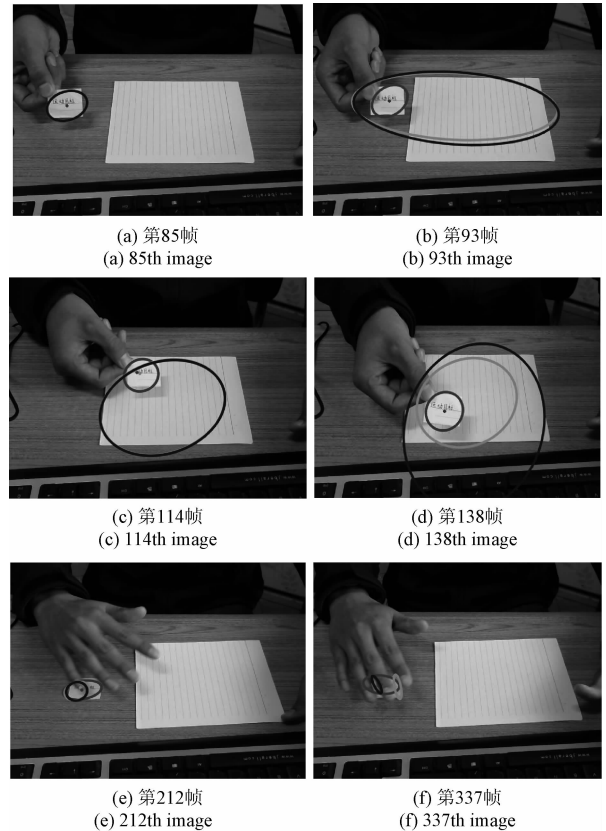


图 2 3 种算法背景颜色干扰下跟踪结果对比
Fig. 2 Comparison of the tracking results of the three algorithms under the background color interference

其中黑色椭圆为传统 CAMShift 算法的跟踪结果, 浅

灰色椭圆为 CAMShift 和 Kalman 结合算法的跟踪结果,深灰色椭圆是本文算法的跟踪结果。图 2(a)~(d) 为运动目标穿越相同背景色下的跟踪结果,可以看出本文算法跟踪效果明显好于其他两种算法。图 2(e)、(f) 为目标被遮挡时 3 种算法的跟踪结果,其中本文算法可以完整的跟踪到整个运动目标的位置,且当遮挡面积占比小于 80% 或完全被遮挡但遮挡时间较短时,跟踪准确率可以达到 90% 以上。而其他两种算法在被遮挡时则只能跟踪到部分区域,当遮挡面积占比超过 50%,跟踪失败。为了更精确体现本算法的优势,现以背景复杂,目标运动复杂的测试视频 lemming(分辨率为 640×480) 为样本连续采集 1 336 帧图像,进行跟踪,得到跟踪结果如图 3 所示。



图 3 3 种算法复杂背景复杂运动下的跟踪结果对比
Fig. 3 Comparison of the tracking results of the three algorithms under complex background and complex motion.

在图 3 中,图 3(a)、(e) 为目标与背景色相似时跟踪结果,图 3(b) 为目标快速运动时的跟踪结果,图 3(c)、(d) 为目标被遮挡时的跟踪结果,图 3(f) 为目标颜色与背景色相差较大时的跟踪结果。可以看出在复杂背景下目标发生缩放,快速运动,被遮挡,相似颜色干扰时本文算法的跟踪结果明显优于其他两种算法。

另外本文对视频 lemming 分别以文献 [3] 中

CAMShift 与 Kalman 滤波相结合的算法、文献 [14] 中 CAMShift 与 SURF 相结合的算法,以及本文算法进行跟踪,并计算每种算法的运算时间、平均误差、准确率进行横向对比。其中算法运算时间是指处理每帧图像所消耗的时间;准确率是指算法跟踪效果的准确度;平均误差是指运动目标实际位置与 3 种算法的跟踪位置之间以像素点为单位的距离差的平均值。结果如表 1 所示。

表 1 三种目标跟踪算法比较

Table 1 Comparison of three target tracking algorithms

算法	帧数	运算时间 (s/帧)	平均误差/ pixel	准确率/%
CAMShift_Kalman	1336	0.103	123.526	64.24
CAMShift_SURF	1336	0.267	69.673	93.95
本文算法	1336	0.122	72.305	92.43

由比较结果可知,CAMShift_Kalman 算法由于机理简单,运算量少,所以单帧运算时间最短,但是由于算法仅依赖颜色特征,因此平均误差相对较大,准确率低。CAMShift_SURF 算法对每帧图像都需要利用 SURF 算法进行特征匹配,虽然准确率高,但是运算速度慢,无法达到实时性的要求。本文算法结合了两种算法的优势,虽然在运算时间上略长于 CAMShift_Kalman 算法,但是准确率却有了较大的提升,相比于 CAMShift_SURF 算法虽然准确率略有降低,但运行时间却缩短了一半以上,并满足了实时性的要求。

5 结论

本文提出了一种结合 SURF 和 Kalman 滤波的 CAMShift 运动目标跟踪算法。利用候选目标和模板目标色度和梯度方向上综合直方图之间的 Bhattacharyya 系数来判断 CAMShift 算法的跟踪结果是否准确,如果不准确则采用 SURF 算法进行跟踪,同时为了提高跟踪准确性和减少运算量,结合 Kalman 滤波来预测运动目标下一帧的位置。试验证明该算法跟踪准确性高,计算时间短,是一种可以广泛应用于实时单目标跟踪的算法。

参考文献

- [1] HSIEH C J, LUM K Y. Automated image tracking based on the Camshift algorithm with adaboost and target trajectory and size estimation [C]. 11th IEEE International Conference on Control & Automation (ICCA), 2014: 918-923.
- [2] HUANG S, HONG J. Moving object tracking system based on camshift and Kalman filter [C]. International Conference on Consumer Electronics, Communications

- and Networks (CECNet), 2011: 1423-1426.
- [3] LI J. Research on moving objects detection and target tracking based in the video [D]. Chengdu: Southwest Jiao Tong University, 2010.
- [4] 任克强, 胡梦云. 基于改进 SURF 算子的彩色图像配准算法 [J]. 电子测量与仪器学报, 2016, 30 (5): 748-756.
- REN K Q, HU M Y. Color image registration algorithm based on improved SURF [J]. Journal of Electronic Measurement and Instrumentation, 2016, 30 (5): 748-756.
- [5] 徐萧萧, 杨浩, 陈宗海. 一种基于改进 Camshift 的目标跟踪算法 [J]. 系统仿真技术及其应用, 2009 (11): 851-854.
- XU X X, YANG H, CHEN Z H. An object tracking algorithm based on improved Camshift [J]. System Simulation Technology and its Application, 2009 (11): 851-854.
- [6] 徐超, 高梦珠, 查宇锋, 等. 基于 HOG 和 SVM 的公交乘客人流量统计算法 [J]. 仪器仪表学报, 2015, 36 (2): 446-452.
- XU CH, GAO M ZH, ZHA Y F, et al. Bus passenger flow calculation algorithm based on HOG and SVM [J]. Chinese Journal of Scientific Instrument, 2015, 36 (2): 446-452.
- [7] 万中田, 洗钟业, 胡明宇, 等. 基于 Kalman 预测器的多特征 CAMShift 运动目标跟踪算法 [J]. 武汉大学学报: 工学版, 2015, 48 (5): 712-716, 722.
- WAN ZH T, XIAN ZH Y, HU M Y, et al. Moving target tracking based on Kalman predictor and multi-featured Camshift algorithm [J]. Journal of Wuhan University : Engineering Science Edition, 2015, 48 (5): 712-716, 722.
- [8] 唐继勇, 仲元昌, 张校臣, 等. 融合梯度和色度信息的分块均值漂移目标跟踪 [J]. 电讯技术, 2015, 55 (9): 1019-1024.
- TANG J Y, ZHONG Y CH, ZHANG X CH, et al. Block Mean Shift tracking combing gradient with hue information [J]. Telecommunication Engineering, 2015, 55 (9): 1019-1024.
- [9] ZHU S L. Mean Shift and related algorithms in video tracking research [D]. Hangzhou: Zhejiang University, 2006.
- [10] CHANG X K, YE Q X, LIU X X, et al. Moving object tracking based on histograms of oriented grads and colors [J]. Journal of Henan University: Natural Science Edition, 2007, 37 (6): 629-634.
- [11] LU X, LEI H, HE Z B. Automatic Camshift tracking algorithm based on multi-feature [J]. Computer Applications, 2010, 30 (3): 650-652.
- [12] MA Z H, GU S H, RONG H L. Camshift moving object tracking algorithm based on SIFT feature points matching [J]. Computer Science, 2014, 41 (6): 291-294.
- [13] TAO L, JING X, SUN S, et al. Combining SURF with MSER for image matching [C]. IEEE International Conference on Granular Computing (GrC), 2013: 286-290.
- [14] 路宁. 基于 SURF 和 Camshift 的物体跟踪方法 [J]. 微型机与应用, 2012, 31 (21): 40-43.
- LU N. A tracking method using SURF and Camshift [J]. Microcomputer and Application, 2012, 31 (21): 40-43.
- [15] LI Y, LI JY, XU Z P. Moving target fast tracking using SURF and cluster analysis method [J]. Liquid Crystal and Display, 2011, 26 (4): 544-550.
- [16] 余彦霖, 祖家奎, 廖智麟. 基于卡尔曼滤波的 MUH 姿态信号融合算法研究 [J]. 电子测量技术, 2015, 38 (5): 106-110.
- YU Y L, ZU J K, LIAO ZH L. The research on attitude determination algorithm based on Kalman filter for unmanned helicopter [J]. Electronic Measurement Technology, 2015, 38 (5): 106-110.
- [17] LI Y H, PENG Y G, LI Z X, et al. An intelligent tracking technology based on kalman and mean shift algorithm [C]. 2nd International Conference on Computer Modeling and Simulation, IEEE, 2010: 107-109.

作者简介



张磊, 2014 年于江南大学获得学士学位, 现为江南大学硕士研究生, 主要研究方向为传感网与模式识别。

E-mail: 632881377@qq.com

Zhang Lei received B. Sc. from Jiangnan University in 2014. Now he is a M. Sc. candidate in Jiangnan University. His present research interest includes visual sensor networks and pattern recognition.



彭力, 2002 年于北京科技大学获得博士学位, 现任江南大学物联网工程学院教授。目前主要研究方向为智能控制与计算机仿真。

E-mail: penglimail2002@163.com

Peng Li received Ph. D. from the University of Science and Technology Beijing in 2002. Now, he is a professor and Ph. D. supervisor of Jiangnan University. His major research interests include intelligent control and computer simulation.