

基于 Kalman 滤波的 Camshift 运动跟踪算法

翟卫欣¹ 程承旗^{2,†}

1. 北京大学遥感与地理信息系统研究所, 北京 100871; 2. 北京大学工学院空天信息工程研究中心, 北京 100871;

† 通信作者, E-mail: ccq@pku.edu.cn

摘要 提出将 Camshift 与 Kalman 滤波相结合的方法。首先, 通过二次搜索来调整搜索窗口的位置和大小, 保证 Camshift 跟踪的可靠性; 然后, 在 Camshift 算法的基础上, 通过卡尔曼滤波对搜索窗口进行运动预测, 保证实时跟踪。实验结果表明, 在图像背景复杂且目标不规则运动的情形下, 采用此方法仍能有效地跟踪到目标。在真实视频数据上的实验结果表明该方法具有很好的应用前景。

关键词 Camshift; Kalman 滤波; 目标跟踪

中图分类号 P208

A Camshift Motion Tracking Algorithm Based on Kalman Filter

ZHAI Weixin¹, CHENG Chengqi^{2,†}

1. Institute of Remote Sensing and GIS, Peking University, Beijing 100871; 2. Aerospace Information Engineering Research Center, Peking University, Beijing 100871; † Corresponding author, E-mail: ccq@pku.edu.cn

Abstract This paper describes a method which combines Camshift and Kalman filter. Firstly, this method adaptively adjusts the position and size of the window by the two search to ensure the reliability of Camshift tracking; then, on the basis of Camshift algorithm, by Kalman filter, it makes prediction for motion search and assists to improve the efficiency of the tracking algorithm. The experimental results show that the new algorithm can effectively track the target in the target and irregular motion complex background situation. Experimental results on real video data show that proposed algorithm has good foreground to illustrate the method.

Key words Camshift; Kalman filtering; object tracking

对于运动物体的跟踪, 一直以来在计算机视觉领域都是一个热点, 在安全监控、运动检测等方面有较大的应用前景。运动物体跟踪包含运动目标的检测和跟踪。运动目标检测的方法有帧间差分法^[1-2]、背景差分法^[3-4]和光流法^[5]。帧间差分法是比较传统的方法, 实现较为简单, 对于动态环境效果较好, 但是分割运动对象的效率不高。背景差分法提取目标的结果比较好, 缺点是容易受光照及其他外部条件的干扰, 尤其对于变化较大的场景比较敏感。光流法能够检测独立运动的对象, 不需要先验信息, 缺点是对光线依赖程度大, 容易丢失目标, 实现过程复杂, 同时光流法需要特定的硬件支持,

否则很难达到一定的效果。物体的跟踪是利用运动物体的特征, 选择适当的匹配与搜索算法, 寻找与目标相似的位置, 实现对运动物体的预测跟踪。常见的物体跟踪算法有基于区位的物体跟踪算法^[6]和基于特征的物体跟踪算法^[7]。

Bradski^[8]提出的 Camshift 算法是以颜色直方图为目标模式的目标跟踪算法, 能够快速解决一般情况下的目标变形和遮挡的问题。对于 Camshift 算法的改进有很多种, 如 Li 等^[9]提出将 Camshift 算法与 Surf 算法相结合进行目标跟踪, Zulkifley 等^[10]利用颜色恒常的办法提高了 Camshift 算法的性能; Youf 等^[11]提出利用端对端技术对 Camshift 算法

进行改进, 并通过 PTZ 摄像机对算法进行了验证。但是 Camshift 算法是基于颜色模型的跟踪方法, 因此对于光线等条件很敏感、相同颜色分量相近的情况, 难以实现正确跟踪; 对于加速和高速运动的物体, 跟踪的鲁棒性较差, 尤其是目标受到干扰时, 不能很好地跟踪目标, 难以解决遮挡后跟踪丢失等问题。为了解决上述问题, 本文以 Camshifit 算法为基础, 将 Kalman 滤波的方法引入运动预测中。实验结果表明, 改进的方法能够有效解决运动物体被遮挡后仍然能正确跟踪运动物体的情况, 并且跟踪的实时性较好。

1 基本算法

1.1 Camshift 算法

由于 RGB 颜色空间对光照亮度变化比较敏感, 为了减少亮度变化对跟踪效果的影响, Camshift 算法将图像由 RGB 颜色空间转化到 HSV 颜色空间进行后续处理。Camshift 的算法流程如图 1 所示。

首先选择初始搜索窗口, 使窗口恰好包含整个跟踪目标。然后对窗口中每个像素 H 通道上的值采样, 从而得到目标的色调(Hue)直方图。将该直方图保存下来, 作为目标的颜色直方图模型。在跟踪过程中, 对视频图像处理区域中的每一个像素, 通过查询目标的颜色直方图模型, 得到该像素为目标像素的概率, 图像处理区域之外的其他区域作为概率为 0 的区域。经上述处理, 视频图像转换为目标颜色概率分布图, 也称为目标颜色投影图。为便于显示, 将投影图转化为 8 位的灰度投影图, 概率为 1 的像素值设为 255, 概率为 0 的像素值设为 0, 其他像素也转换为相应的灰度值。因此灰度投影图中越亮的像素为目标像素的可能性越大。

图 1 中设置搜索中心是 Camshift 算法的关键。设 (x, y) 为搜索窗口中的像素位置, $I(x, y)$ 是投影

图 (x, y) 处的像素值。

定义搜索窗口的零阶矩 M_{00} 和一阶矩 M_{10} 、 M_{01} :

$$M_{00} = \sum_x \sum_y I(x, y),$$

$$M_{10} = \sum_x \sum_y xI(x, y),$$

$$M_{01} = \sum_x \sum_y yI(x, y),$$

求得搜索窗口内的质心位置为

$$(x_c, y_c) = (M_{01} / M_{00}, M_{10} / M_{00}),$$

然后根据质心 M_{00} 调整搜索窗口大小。如果移动距离大于预设的固定阈值, 则重新计算调整后的窗口质心, 进行新一轮的窗口位置和尺寸调整, 直到窗口中心与质心间的移动距离小于预设的固定阈值, 或者循环运算的次数达到某一最大值, 认为收敛条件满足, 进入下一帧图像进行新的目标搜索。在新的图像中, 利用上一帧图像中最终得到的窗口质心位置和零阶矩 M_{00} , 设置新的搜索窗口的位置和尺寸。Camshift 跟踪算法在图像帧内与帧间都根据上一步得到的 M_{00} , 自适应调整搜索窗口的大小, 因而可以适应跟踪目标动态变形的情形。

1.2 Kalman 滤波

Kalman 滤波算法是一种快速、高效的递推估计算法, 能比较准确地预测出目标的速度和位置。Kalman 滤波利用状态方程来描述状态的转移过程^[12]。系统的状态方程及观测方程分别为

$$X_k = A_{k,k-1} X_{k-1} + V_{k-1},$$

$$Y_k = H_k X_k + W_{k-1},$$

其中 X_k 和 X_{k-1} 分别是 k 和 $k-1$ 时刻的状态变量, $A_{k,k-1}$ 为相应的状态变化矩阵; H_k 为 k 时刻的观测矩阵, V 和 W 分别代表状态噪声和观测噪声矩阵, 相应的方差矩阵分别为 Q 和 R 。设状态向量 $X_k = [x_{sk}, y_{sk}, x_{vk}, y_{vk}]$ 的各个值分别代表在 x 和 y 方向的

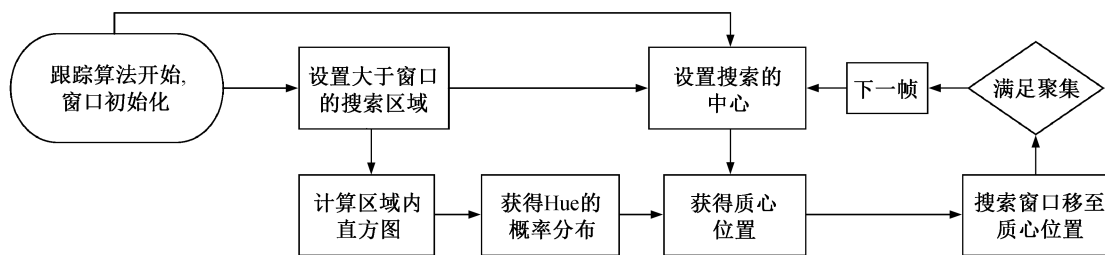


图 1 Camshift 算法详细流程
Fig. 1 Detailed algorithm procession of Camshift

位置和速度, 观测向量 $Y_k = [x_{tk}, k, y_{tk}]$ 代表目标的观测位置, 具体推导公式如下。

$$\begin{aligned} X_k &= A_{k,k-1} X_{k-1}, \\ X_k &= X_{k,k-1} + K_k [Y_k - H_k X_{k,k-1}], \\ K_k &= P_{k,k-1} H_k^T (H_k P_{k,k-1} H_k^T + R), \\ P_{k,k-1} &= A_{k,k-1} P_{k-1} A_{k,k-1}^T + Q, \\ P_k &= (I - K_k H_k) P_{k,k-1}. \end{aligned}$$

其中 $X_{k,k-1}$ 为状态的预测值, X_k 为 $X_{k,k-1}$ 的修正值, K_k 为 Kalman 滤波增益矩阵, $P_{k,k-1}$ 为 $X_{k,k-1}$ 的协方差矩阵, P_k 为 X_k 的协方差矩阵^[13]。

1.3 Kalman 滤波的运动估计策略

用 Kalman 滤波预测某一帧图像中目标质心在下一帧中可能出现的位置, 再进行搜索匹配, 完成对目标的提取和跟踪。Kalman 滤波的运动估计策略分以下几个步骤: 1) 用前景图像与背景图像进行差分运算, 得到监测区域及其图形属性, 最终获取其质心坐标; 2) 计算检测区域并提取目标, 得到质心最大的目标位置, 输入 Kalman 滤波器; 3) 基于 Kalman 滤波器预测下一帧图像中的质心位置; 4) 得到最佳的匹配位置; 5) 重复上述步骤直至视频结束。

2 基于 Kalman 滤波的 Camshift 算法

基于 Kalman 滤波的 Camshift 算法同样是基于颜色概率模型的跟踪方法, 即在跟踪前手工选择 ROI (region of interest) 以提取跟踪对象的颜色分布, 然后通过不断改变搜索窗口的中心位置来获得物体的质心, 实现对物体的跟踪。

为了减少高速运动或者遮挡等对物体跟踪造成的影响, 本文对 Camshift 算法进行改进。目标跟踪时, 以 Camshift 算法对搜索窗口进行运动检测。目标跟踪丢失后, 扩大窗口尺寸进行二次寻找, 如果二次寻找后仍搜索不到运动物体, 则维持搜索窗口在一定大小, 保持搜索状态继续搜索。当在后续帧

中再次出现跟踪目标时, 重新进行搜索跟踪, 然后以 Camshift 算法为基础, 配合 Kalman 滤波器, 实现同时基于色彩特征和运动特征的物体追踪。Camshift 算法会产生一个窗口, 对搜索窗口进行调整, 同时利用 Kalman 滤波对运动目标的状态进行估计, 得到更精确的目标位置。从原理上讲, 可以认为是两种算法的线性拟合。

$$X_{k+1} = \alpha X_{k+1,k} + (1-\alpha)Y,$$

其中, X_{k+1} 为对目标在 $k+1$ 时刻的预测位置, $X_{k+1,k}$ 为 Kalman 滤波器的预测位置, Y 为 Camshift 算子的预测位置, 比例因子 α 会根据当时目标受干扰程度的不同而发生改变。此算法提高了搜索窗口的位置精度, 能够实现对运动物体的精确实时跟踪。组合算法的框图如图 2 所示。

改进的基于 Kalman 预测的算法是在原有的基础上加入 Kalman 滤波的目标状态估计算法, 能够对整体效果进行修正, 如图 3 所示。

3 实验结果与分析

3.1 实验设计

为了将新的 Camshift 算法与传统的 Camshift 算法相对比, 本文设计一组对照试验来测试两类算法的适应情况。算法基于 Visual Studio 2012 编译器和 Opencv 库 2.4.7 进行, 以道路上的汽车为主要跟踪对象, 选取在市内道路和十字路口的汽车运行视频数据, 比较两类算法中对于汽车跟踪的情况。本实验抽取 80 组汽车的目标数据, 分别采用传统的 Camshift 算法和本文提出的结合 Kalman 滤波器的 Camshift 算法进行跟踪操作。

3.2 数据分类

实验选取的对象分为 4 类: 低速无遮挡的汽车、低速有遮挡的汽车、高速无遮挡的汽车和高速有遮挡的汽车, 每个类别有 20 组数据。

本文区别高速和低速的标准是判断车速是否大

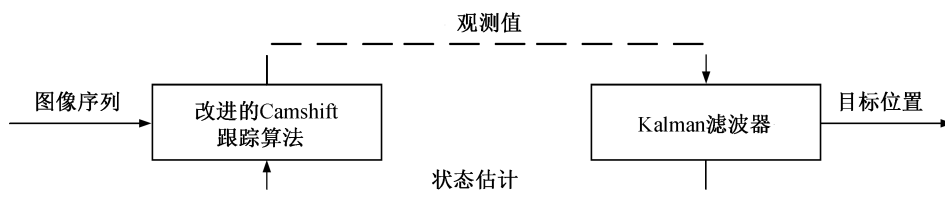


图 2 Camshift 与 Kalman 结合算法
Fig. 2 Combination of Camshift and Kalman filtering

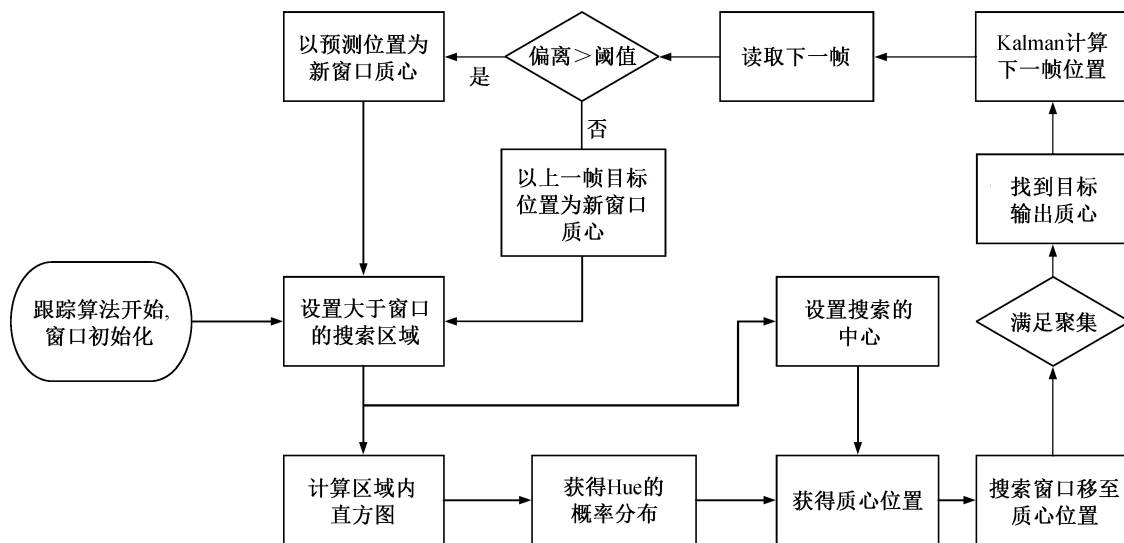


图 3 Camshift 与 Kalman 结合算法详细流程
Fig. 3 Detailed algorithm procession of Camshift based on Kalman filtering

于 10 m/s, 有些汽车在视频中有明显加速或者减速运动, 那么将这样的汽车行驶数据分成两部分: 一部分为高速行驶, 另一部分为低速行驶。本文区别是否被遮挡的方法是, 观测汽车在行驶路程中是否有在视野中大于 1/4 的部分未露出来的情况, 如果有, 则认为被遮挡, 否则认为未被遮挡。

3.3 阈值选取

实验中视频来源的摄像头在道路的前上方, 拍摄方向正对道路, 高度为 5 m, 视频分辨率 a 为 24 fps, 单位像素所代表的地面距离 r 约为 0.015 m。理论上两帧间的移动距离为 $v/(a \cdot r)$ 。 r 随距离远近有所变化, v 在每组实验中各不相同。本文实验选取高斯分布的 95% 置信区间的上界作为阈值, 则高速和低速两类的阈值分别为 80 像素和 220 像素。实验结果如表 1 所示。

本文实验结果表明, 结合 Kalman 滤波器的 Camshift 算法能够在原有的 Camshift 算法基础上,

大大提高跟踪效果, 尤其在目标高速运动或者目标有遮挡时更为明显。图 4 中的实验结果反映在目标高速运动且有遮挡的条件下, 两种算法的区别。图 4(a)为相遇前的情况, 图 4 (b)为相遇时的情况, 图 4(c)为利用 Camshift 算法进行追踪的相遇后的图像, 图 4(d)为利用 Camshift 与 Kalman 结合算法进行追踪的相遇后的图像, 可以看到, 第一种算法的追踪出现错误, 第二种并没有类似错误。

当目标像素在整幅图像中所占比重不大, 而帧间位移较大时, Camshift 算法中根据上一帧图像跟踪结果得到的新搜索窗口可能与目标没有联接, 导致跟踪丢失。尤其当目标帧间位移过大时, 如果不对目标进行一定的运动估计, 则可能会失效。改进算法运用得到的预测值对搜索窗口进行平移, 增大搜索窗口与目标联接的可能性, 提高跟踪的可靠性。Camshift 算法能够很好地对低速运动且无遮挡的目标进行实时跟踪, 但是当目标提速或者被遮挡

表 1 两种算法实验结果
Table 1 Experimental results of two kinds of algorithms

算法	识别结果(成功个数/实验总数)			
	低速无遮挡	低速有遮挡	高速无遮挡	高速有遮挡
Camshift 算法	20/20	10/20	16/20	4/20
Camshift & Kalman 算法	20/20	14/20	19/20	13/20

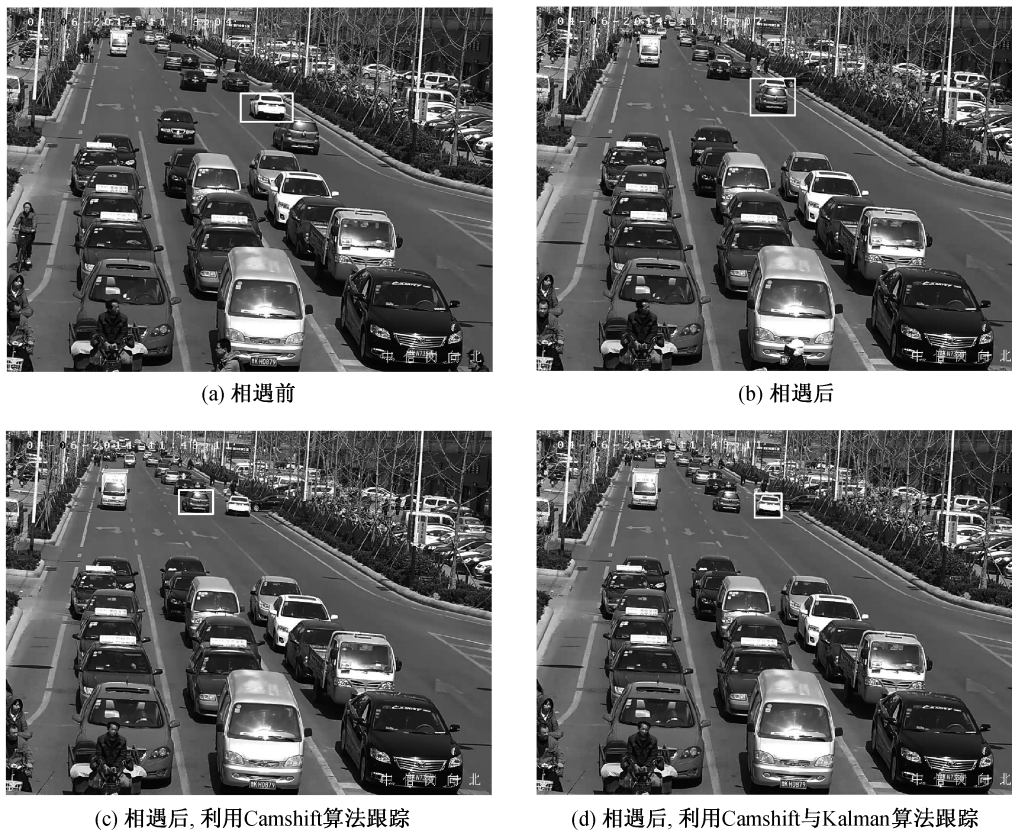


图 4 两种算法跟踪效果的区别

Fig. 4 Differences between two algorithms

时,跟踪的正确率会下降。结合 Kalman 滤波的 Camshift 算法一直保持较高的正确率,即很难在某个时间点离开跟踪目标,能够保持实时跟踪。

另外,采用原有的算法,在跟踪过程中,由于物体与背景过于相似,那么物体被遮挡或者经过某些特定地段时,原有的跟踪会失效,跟踪框此时会离开原有目标而转向其他目标。在 Kalman 滤波器加入 Camshift 后,由于对运动存在一定的预期,并不完全依靠原有的基于图像特征进行的匹配,所以跟踪效率会提高。

4 总结

本文提出一种改进的结合 Kalman 滤波的 Camshift 算法,对运动物体进行实时跟踪。在 Camshift 算法的基础上,通过二次搜索和 Kalman 滤波来保障运动物体跟踪的可靠性和实时性。实验表明,该方法具有较好的实时性和鲁棒性,尤其在追踪物体高速运动或者在复杂背景以及遮挡等情况下,跟踪效果显著提高。但是在遮挡的时间较长或

者物体出现较大形变时,本文算法仍然不能很好地解决跟踪失效或者错误的问题,这方面的研究仍需继续。

参考文献

- [1] Anderson C H, Burt P J, van der Wal G S. Change detection and tracking using pyramid transform techniques // International Society for Optics and Photonics. Cambridge, MA, 1985: 72-78
- [2] 邬大鹏,程卫平,于盛林. 基于帧间差分和运动估计的 Camshift 目标跟踪算法. 光电工程, 2010, 37(1): 55-60
- [3] Wildes R P, Asmuth J C, Green G L, et al. A system for automated Iris recognition // Applications of Computer Vision, Proceedings of the Second IEEE Workshop on. Sarasota, 1994: 121-128
- [4] Beauchemin S S, Barron J L. The computation of optical flow. ACM Computing Surveys (CSUR), 1995,

- 27(3): 433–466
- [5] 计文平, 郭宝龙, 于贵广. 基于虚拟线圈的光流法车流量检测. 计算机仿真, 2004, 21(1): 109–110
- [6] Bascle B, Deriche R. Region tracking through image sequences // Proceedings of the Fifth International Conference on Computer Vision. Cambridge, MA, 1995: 302–307
- [7] Hwang V S S. Tracking feature points in time-varying images using an opportunistic selection approach. Pattern Recognition, 1989, 22(3): 247–256
- [8] Bradski G R. Computer vision face tracking for use in a perceptual user interface // Proc IEEE Workshop Applications of Computer Vision. Princeton, 1998: 214–219
- [9] Li J, Zhang J, Zhou Z, et al. Object tracking using improved Camshift with SURF method // Open-Source Software for Scientific Computation (OSSC), 2011 International Workshop on. Beijing, 2011: 136–141
- [10] Zulkifley M A, Mustafa M M, Hussain A. On improving CAMSHIFT performance through colour constancy approach // Computer & Information Science (ICCIS), 2012 International Conference on. Kuala Lumpur, 2012, 1: 375–378
- [11] Yousf W M, Elmowafy O M, Abdl-Dayem I A. C18. Modified CAMShift algorithm for adaptive window tracking // Radio Science Conference (NRSC), 2012 29th National. Cairo, 2012: 301–308
- [12] Kalman R E. A new approach to linear filtering and prediction problems. Journal of Fluids Engineering, 1960, 82(1): 35–45
- [13] Li Yinghong, Pang Yigui, Li Zhengxi, et al. An intelligent tracking technology based on kalman and mean shift algorithm // Computer Modeling and Simulation, 2010 Second International Conference on. Sanya, 2010: 107–109