

复杂环境下的改进 Camshift 算法研究

陈 炫 颜学龙

(桂林电子科技大学 电子工程与自动化学院 桂林 541004)

摘要: 针对 Camshift 算法要求相邻两帧之间的目标必须具有较大的颜色相似性,对于目标在高速运动情况下和跟踪目标受到遮挡的情况下容易出现目标丢失,并且在目标遇到相似颜色干扰的时候容易受到干扰的问题,提出一种基于加权颜色概率分布,引入局部搜索引导机制,结合 kalman 滤波预测的改进 camshift 算法,实验证明,该算法在复杂背景,物体高速运动或遇到遮挡的情况下都有较好的跟踪效果,并且跟踪效率得到的提升。

关键词: Camshift;局部跟踪引导;kalman 滤波

中图分类号: TP391.4 **文献标识码:** A **国家标准学科分类代码:** 520.6040

Study of improved Camshift algorithm in complex environment

Chen Xuan Yan Xuelong

(School of Electronic Engineering and Automation, Guilin University of Electronic Technology, Guilin 541004, China)

Abstract: Camshift algorithm requires the target have a large color similarity between the adjacent frames, the target is easy to get lost when moving in high speed or occlusion appears. Aiming at this, an improved Camshift algorithm is proposed, which contains a weighted color probability distribution and local-search-guiding mechanism, combining with the Kalman filter forecasting model. The experiments show that, this algorithm have a good tracking effect in complex background when object moves in high-speed or it is occluded.

Keywords: camshift; local-search-guiding mechanism; kalman filter

1 引言

目前在视频目标跟踪领域流行着各种各样的算法,大体分类有基于物体模型、区域、轮廓、物体特征以及主动轮廓的跟踪,其中每种算法都有其优点和缺点。

Camshift 算法是一种无参的密度估计方法,因具有计算量小,实时性高的特点被广泛应用,但是由于直方图是一种相对较弱的对目标特征的描述方法,且算法收敛于局部极值,因此遇到相似色干扰,运动速度快,遮挡的情况跟踪容易失效。

针对以上情况,文献[1]采用融合运动信息的直方图和粒子滤波的方法提高了稳健性,但是由于粒子滤波器大大增加了运算量,不满足实时性要求,文献[4]采用结合 kalman 滤波器的方法,但是在滤波器无法进行正确的运动预判情况下(如物体由静止状态忽然高速运动)效果欠佳;文献[5]采用了最小二乘法拟合运动轨迹在遮挡出现后重新捕获目标,但是效果不如卡尔曼滤波,并且同样不适用于物体运动不规则的情况。

根据以上情况,本文从跟踪算法的智能化、高效率、高鲁棒性和实时性要求这个角度出发,对基于颜色特征进行跟踪

的 Camshift 算法进行改进,本文研究的关键在于从算法原理中找出算法在各种情况下跟踪失效的原因,难点在于找到改进的方法并且验证改进算法的可行性与实际效果。

2 Camshift 算法

2.1 建立目标模型

目标模型是在选定的运动目标区域内建立的直方图,该直方图以 Hue 信息为统计量,因此,应首先把图像转化为 HSV 颜色空间。

2.2 反向投影

将目标模型归一化,得到 $I(x, y)$,对于上一步骤提取出来的 Hue 图像上的每一个像素,用其在 $I(x, y)$ 上的值替代,便得到反向投影图。

2.3 meanshift 计算

设搜索框中心坐标为 (x_0, y_0) ,计算搜索框零阶矩:

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

x 和 y 的一阶矩:

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

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

搜索框即目标的质心为：

$$x_c = \frac{M_{01}}{M_{00}}, y_c = \frac{M_{10}}{M_{00}}$$

以搜索框的几何中心为起点，以其质心终点的向量称为 *meanshift* 向量，每次搜索框都沿着该向量移动，当该向量的长度小于某一特定值或者迭代次数达到要求后，停止计算。

该算法的不足：

- 1) 易受相似颜色背景的干扰，不适应遮挡环境；
- 2) 对高速运动的物体容易出现跟踪丢失的情况；
- 3) 缺乏目标模型实时更新，不适应目标运动过程中发生旋转、翻滚、颜色信息等的变化。

3 算法改进

3.1 加权的颜色概率分布图

位于目标边缘的像素相对于中心像素更容易受到遮挡或背景像素的干扰，可靠性相对较低，相反，目标中心区域的像素点不易受到影响，更能代表目标的特性，所以不同位置的像素各自对目标的描述能力是有强弱之分的，因此，可以通过加权的方式使不同位置像素的贡献值得以更准确地体现，这样能对目标的区域特性进行更合理地描述。基于这个思想，本文对颜色概率分布图进行加权处理，对于一个半径为 r 的圆形区域，其权值函数可设为：

$$k(r) = \begin{cases} 1-r, & r < 1 \\ 0, & \text{其他} \end{cases} \quad (1)$$

这样，当采用 Epanechnikov^[3]核函数，加权后的目标颜色概率分布图表达式为：

$$\hat{q}_u = \sum_{i=1}^n k\left(\left\|\frac{x_i - x_0}{h}\right\|\right) \delta[b(x_i) - u] \quad (2)$$

式中： h 是搜索区域的半径，这样， \hat{q}_u 即可作为目标模板进行相关的相似度匹配运算。

3.2 实时的目标模板更新

在目标运动过程中，可能发生旋转、翻滚状态的变化，其尺寸，颜色等信息也可能会随着运动而改变，为了适应这些状态的变化，可对目标模板进行实时地更新。

设第 k 帧图像中的目标模板用上述形式表示为 $\hat{q}_{u,k}$ ，第 $k-1$ 帧的目标模板为 $\hat{q}_{u,k-1}$ ，候选目标分布为 $p_{u,k-1}$ ，得到目标模板的更新方式为：

$$\hat{q}_{u,k} = (1-\beta)\hat{q}_{u,k-1} + \beta p_{u,k-1}, \beta \text{ 一般取 } 0.5 \sim 0.8.$$

3.3 局部跟踪引导机制

为了搜寻更好的候选目标模型，将原 *Camshift* 算法的

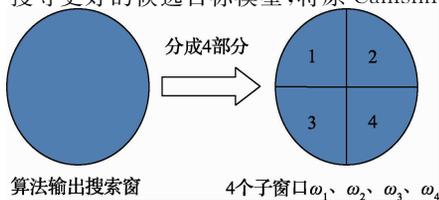


图 1 子搜索窗口划分

输出搜索窗口分为 4 部分，如图 1 所示。

令 $p = \{p_u\}_{u=1}^m$ 和 $\hat{q} = \{\hat{q}_u\}_{u=1}^m$ 分别表示候选目标以及目标本身的颜色直方图，则它们的 Bhattacharyya 系数的表达式为：

$$\rho(p, \hat{q}_u) = \sum_{u=1}^m \sqrt{p_u \hat{q}_u} \quad (3)$$

原 *Camshift* 算法停止迭代运算的条件是： $|x_0 - x_c| < \epsilon$ 且 $|y_0 - y_c| < \epsilon$ 或者迭代次数超过了某个预先设定的阈值 T ，达到这一条件以后 *Camshift* 算法会返回一个搜索框 ω ，这里，用 ρ_ω 代表 *Camshift* 算法返回的窗口 ω 与目标之间的巴氏系数，若 ρ_ω 大于某一阈值 α ，则直接把 ω 作为候选目标 ω_{did} ，否则，执行以下操作：

1) 将窗口分为 4 个子窗口；

2) 计算 $\rho_{\omega_1}, \rho_{\omega_2}, \rho_{\omega_3}, \rho_{\omega_4}$ 。令 $\rho_{\omega_m} = \max \rho_{\omega_i}, K = 1, 2, 3, 4$ 。（由于 4 部分子窗口的直方图出现相同的概率较小，故不考虑这种情况。） ρ_{ω_m} 对应的搜索框是 ω_m ；

3) 若 $\rho_{\omega_m} < \rho_\omega$ ，则停止局部子模块匹配；

4) 否则，将 ω_m 的几何中心坐标 (x_m, y_m) 修正 kalman 滤波器的预测值。

3.4 kalman 滤波

kalman 滤波算法包含两个模型：

信号模型：

$$\mathbf{X}_k = \mathbf{A}_k \mathbf{X}_{k-1} + \mathbf{W}_k \quad (4)$$

式中： \mathbf{X}_k 是一个向量，代表第 k 帧的状态，矩阵 \mathbf{A}_k 代表两帧状态的转变关系， \mathbf{W}_k 是系统噪声。

观测模型：

$$\mathbf{Z}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{V}_k \quad (5)$$

式中： \mathbf{Z}_k 是要观测的数据， \mathbf{H}_k 表示当前状态和要观测的数据之间的关系， \mathbf{V}_k 是零均值白噪声。

在实际应用中，将跟踪目标的位置的坐标，运动速度的水平和垂直分量放到状态向量里，即令 $\mathbf{X}_k = [x_k, y_k, v_{xk}, v_{yk}]^T$ ，把想要获得的位置信息放到观测向量里，即令 $\mathbf{Z}_k = [x_k, y_k]^T$ ，在实际应用中信号模型为：

$$\begin{bmatrix} x_k \\ y_k \\ v_{xk} \\ v_{yk} \end{bmatrix} = \begin{bmatrix} 1 & 0 & t & 0 \\ 0 & 1 & 0 & t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ v_{xk-1} \\ v_{yk-1} \end{bmatrix} + \begin{bmatrix} \frac{t^2}{2} \\ \frac{t^2}{2} \\ t \\ t \end{bmatrix} \omega_k \quad (6)$$

观测模型为：

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ v_{xk} \\ v_{yk} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \omega_k \quad (7)$$

在跟踪过程中，将上一帧的状态向量 \mathbf{X}_{k-1} 通过式 (4) 的运算得到 \mathbf{X}_k ，提取 \mathbf{X}_k 中目标的位置，让 *Camshift* 算法在该位置开始运算，将其达到运算结束条件后的输出位置作

为观测值 Z_k , 来修正向量 X_k 。

3.5 遮挡和相似色干扰条件下持续跟踪

3.5.1 遮挡判定

根据文献[3]的论述,目标遭遇遮挡的情况出现满足以下两个条件:

- 1) $\rho(p_a \bar{q}_a) < T_b, T_b \in (0, 1)$
- 2) 当前帧跟踪物体的面积 S_{rc} 与目标面积 S_{obj} 之比小于某一阈值 θ , 即 $\frac{S_{rc}}{S_{obj}} < \theta, \theta \in (0, 1)$ 。

3.5.2 相似色干扰判定

当出现相似色干扰满足以下两个条件:

- 1) $\rho(p_a \bar{q}_a) < T_c, T_c \in (0, 1)$
- 2) 当前帧跟踪物体的面积 S_{obj} 与目标面积 S_{rc} 之比小于某一阈值 η , 即 $\frac{S_{obj}}{S_{rc}} < \eta, \eta \in (0, 1)$ 。

3.5.3 处理方法

当发生遮挡或者相似色干扰的时候,用卡尔曼滤波器的输出区域作为下一帧搜索计算的初始位置,再进行目标跟踪;当没有出现遮挡或相似色干扰的时候,则继续使用上一帧 Camshift 输出的位置作为下一帧的 Camshift 的跟踪初始位置。

算法流程如图 2 所示。

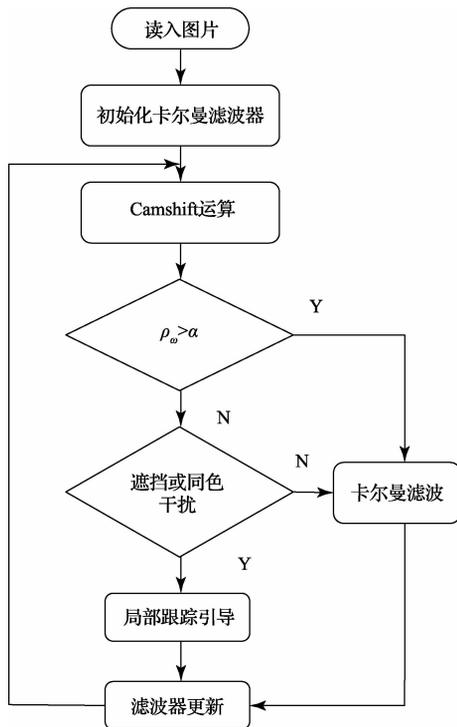


图 2 算法流程

4 实验结果

4.1 相似色干扰实验

如图 3 所示,原算法在遇到相似色干扰的情况下,搜索

框的质心偏离了原目标,导致跟踪失效,而改进的算法由于采用了加权的颜色概率分布,且进行了 kalman 滤波的预测,搜索框与原目标保持一致。



图 3 相似色干扰下原算法与改进算法对比

4.2 静止物体加速运动实验

如图 4 所示,物体由静止状态以较高速度运动,原算法失效,搜索框收成了一个很小的椭圆,原因是运动物体速度较高,前后两帧的搜索区域没有了交集,卡尔曼滤波器不能根据物体静止状态做出运动预判,而改进算则保持了对物体的跟踪,原因是引入了跟踪引导机制,跟踪窗口在下一帧的起始位置会向着跟目标更相似的方向移动,从而保持了对跟踪。

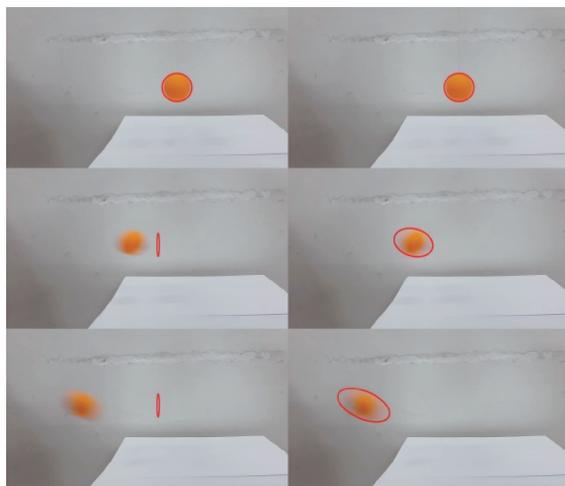


图 4 静止物体加速运动的实验效果对比

4.3 遮挡实验

原算法在遭遇遮挡时目标丢失,不能持续跟踪,原

因是物体再次出现的位置与消失的位置距离较远,超出了搜索的范围,而改进算法在此情况下采用卡尔曼滤波器输出的位置作为下一帧的搜索起始位置,使得前后两帧的搜索区域出现了交集,维持了目标的跟踪,如图5所示。

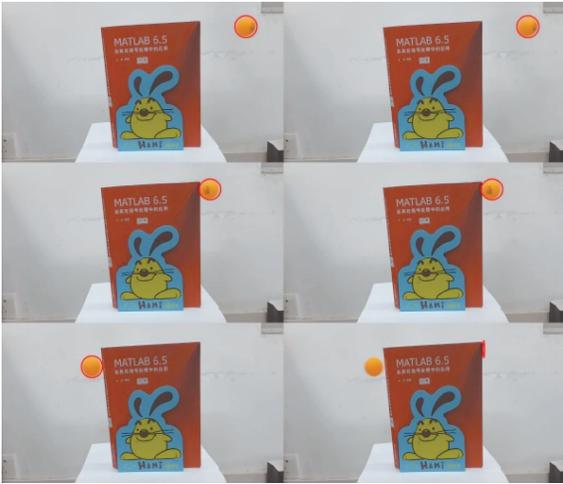


图5 遮挡条件下改进算法与原算法对比

本文算法和原算法在没有遮挡和相似颜色干扰的情况下的处理每帧图像时的迭代次数与时耗测量数据如下,其中虚线是原算法采集得来的数据,实线是改进后算法的测量数据,如图6和7所示。

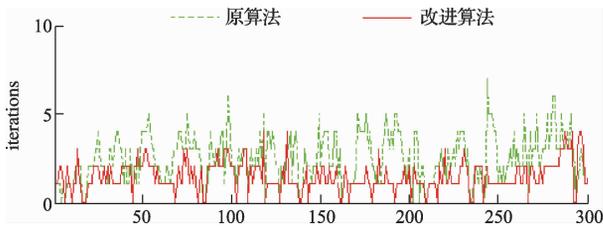


图6 正常跟踪情况下算法迭代次数对比

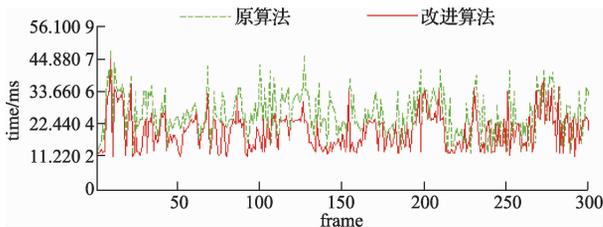


图7 正常跟踪情况下算法耗时对比

可以比较清楚的看到,改进后的算法无论是在处理每帧图像时的迭代次数或是在时耗方面都比原算法要少,原

因是原算法在处理完一帧图像后,下一帧的跟踪起始位置是在上一帧的算法的收敛处,而改进后的算法在下一帧的算法的跟踪起始位置是 kalman 滤波器的输出位置,即对物体的运动进行了预测,因此提高了跟踪的效率。

5 结 论

针对原 Camshift 算法的不足,提出一种基于加权颜色概率分布,在局部进行搜索框分块跟踪引导,结合 kalman 预测模型的改进方法,实验结果表明该算法能较好地适应如相似色干扰,遮挡,物体高速运动等环境。

参考文献

- [1] 王鑫,唐振民. 一种改进的基于 Camshift 的粒子滤波实时目标跟踪算法[J]. 中国图象图形学报, 2010(10):1507-1514.
- [2] 陶立超,赵宇明. 基于分块颜色直方图和粒子滤波的物体跟踪[J]. 计算机工程与用, 2012(7):165-168.
- [3] 欧阳宁,游锦海,莫建文,等. 遮挡情况下的多目标跟踪算法[J]. 计算机应用, 2010(5):1984-1986.
- [4] 闫钧华,陈少华,艾淑芳,等. 基于 Kalman 预测器的改进的 CAMShift 目标跟踪[J]. 中国惯性技术学报, 2014(4):536-542.
- [5] 许晓航,肖刚,云霄,等. 复杂背景及遮挡条件下的运动目标跟踪[J]. 光电工程, 2013(1):23-30.
- [6] 庄严,战洪斌,王伟,等. 基于加权颜色直方图和粒子滤波的彩色物体跟踪[J]. 控制与决策, 2006, 21(8): 868-878.
- [7] 马加庆,韩崇昭. 一类基于信息融合的粒子滤波跟踪算法[J]. 光电工程, 2007(4):22-25.
- [8] 郭大鹏,程卫平,于盛林. 基于帧间差分 and 运动估计的 Camshift 目标跟踪算法[J]. 光电工程, 2010(1): 55-60.
- [9] 张宇. 一种图像确认目标的多目标跟踪方法[J]. 电子测量与仪器学报, 2014, 28(6):617-624.
- [10] 龚卫国,王旭,李正浩. 一种抗遮挡的红外多目标实时检测跟踪算法[J]. 仪器仪表学报, 2014, 35(3): 535-542.
- [11] 李红岩,毛征,袁建建,等. 一种基于算法融合的运动目标跟踪算法[J]. 国外电子测量技术, 2013, 32(12): 36-40.

作者简介

颜学龙,教授,硕士生导师,主要研究电子测量,可测性设计领域的教学和科研工作。

陈炫,硕士研究生,主要研究方向是图像信息处理。