

# 基于视频的运动目标跟踪算法

刘俊杰<sup>1a</sup>, 杨 勇<sup>1a</sup>, 才 华<sup>1b</sup>, 曲福恒<sup>1a</sup>, 李双鑫<sup>2</sup>

(1. 长春理工大学 a. 计算机科学技术学院; b. 电子信息工程学院, 长春 130022; 2. 东北师范大学 图书馆, 长春 130024)

**摘要:** 为更加准确、快速地检测与跟踪运动目标, 将背景差分法和帧间差分法相融合对 CAMSHIFT (Continuously Adaptive Mean-SHIFT) 算法进行改进。首先, 通过背景差分法和帧间差分法相融合确定目标所在区域, 然后结合 CAMSHIFT 迭代算法实现目标跟踪。实验结果表明, 该方法改变了传统 CAMSHIFT 算法需手动选定目标和跟踪窗容易发散的局限性, 并提高了跟踪的准确性与稳定性。

**关键词:** CAMSHIFT 算法; 目标跟踪; 背景差分法; 帧间差分法

中图分类号: TP391.41

文献标识码: A

DOI:10.19292/j.cnki.jdxp.2017.03.019

## Algorithm of Moving Object Tracking Based on Video Images Sequence

LIU Junjie<sup>1a</sup>, YANG Yong<sup>1a</sup>, CAI Hua<sup>1b</sup>, QU FuHeng<sup>1a</sup>, LI Shuangxin<sup>2</sup>

(1a. School of Computer Science and Technology; 1b. School of Electronic and Information Engineering,

Changchun University of Science and Technology, Changchun 130022, China;

2. Library Northeast Normal University, Changchun 130024, China)

**Abstract:** Moving target tracking in video is an important research direction in the field of computer vision. In order to detect and track the moving objects more accurately and quickly, we propose an improved CAMSHIFT (Continuously Adaptive Mean-SHIFT) algorithm based on background subtraction and frame difference algorithm. First of all, the target area is determined by the background difference method and the frame difference method, then combining with the CAMSHIFT iterative algorithm the target tracking is realized. Experiments show that this method has changed the traditional CAMSHIFT algorithm which needs manually selecting the target, has overcome the limitations of diverging, and has improved the accuracy and stability of tracking.

**Key words:** continuously sdaptive mean-SHIFT( CAMSHIFT); target tracking; background subtraction; frame difference method

## 0 引 言

基于视频图像序列的跟踪技术是实现目标搜索功能和跟踪功能的关键技术, 其主要根据在每帧图像之间确定目标相关信息参数的相互关系, 通过前后帧的匹配关系确定目标轨迹等信息, 从而实现对目标的跟踪。通常情况下, 目标的检测与跟踪可分为两类: 1) 摄像头固定的情况下, 只能跟踪在其视场内运动的目标; 2) 摄像头可上下左右进行平移旋转, 前景目标一直在摄像机的视场范围内运动。笔者主要研究处于静态背景环境下的 CAMSHIFT(Continuously Adaptive Mean-SHIFT) 目标跟踪算法, 首先利用背景差分法和帧间差分法相融合实现对前景目标的快速检测, 然后在确定的运动目标所在区域内通过 CAMSHIFT 迭代算法实现对运动目标的稳定跟踪。

收稿日期: 2016-08-30

基金项目: 吉林省科技发展计划基金资助项目(20130101179JC)

作者简介: 刘俊杰(1989—), 男, 长春人, 长春理工大学助理实验师, 主要从事计算机图像处理研究, (Tel) 86-43578807929 (E-mail) 403192671@qq.com; 通讯作者: 杨勇(1970—), 男, 长春人, 长春理工大学教授, 硕士生导师, 主要从事图形图像处理研究, (Tel) 86-43504323586 (E-mail) yy@cust.edu.cn。

该算法解决了由于光照变化导致背景亮度,色度变化或背景颜色与目标相近而导致无法检测运动目标的问题,改变了一般 CAMSHIFT 算法需手动选取前景目标的缺陷,解决了跟踪过程中搜索窗容易发散的问题,检测速度快,跟踪更加稳定和持久。

## 1 目标检测

### 1.1 背景差分法

当摄像头固定时,对于图像序列中,背景相当于静止不变的部分,所以除了背景部分就是前景,即笔者要检测的运动目标,但由于天气、光照和背景物体的加入和移出,需实时对背景进行更新。其基本思想就是首先建立一个能准确描述背景的背景模型,然后将当前帧图像与背景模型作差分运算,再选定某一阈值。如果像素间的差值大于这一阈值,则判断此像素属于前景目标;如果小于阈值,则属于背景图像<sup>[1]</sup>。因此,背景模型的建立与更新是关键的一步。

1) 背景模型的建立与更新。常用的获得背景图像的方法一般有两种:一种是时间平均法,这种方法容易造成前景目标与背景图像的融合;另一种是均值法,即通过对连续图像序列进行统计,平均获得背景图像,用公式表示为

$$B_n(x, y) = \frac{1}{N} \sum_{i=1}^N \text{image}_i(x, y) \quad (1)$$

其中  $B_n(x, y)$  表示所求背景图像,  $N$  表示一段图像序列中的帧数,  $\text{image}_i(x, y)$  表示第  $i$  帧序列图像中  $(x, y)$  像素点,通过改变  $x$  和  $y$  的值即可获得全部背景图像。所以背景更新公式可设为

$$B_{n+1}(x, y) = aB_n(x, y) + (1 - a)D_n(x, y) \quad (2)$$

其中  $a$  为更新速率,一般取值介于 0.5 ~ 1 之间,  $B_n(x, y)$  为当前时刻的背景图像值,  $D_n(x, y)$  为当前帧的灰度值,  $B_{n+1}(x, y)$  为更新后的背景灰度值。

2) 前景提取。因为前景目标和背景图像在颜色、亮度等方面的不同选取每帧图像,笔者以当前帧图像与背景图像作差分,取结果高于判决门限  $T_n$  为前景目标,若设置为 1,低于判决门限  $T_n$  为背景;若设置为 0,根据式(2)将差分更新到背景中去,通过运算后得到二值图像

$$G_n(x, y) = \begin{cases} 0 & |D_n(x, y) - B_n(x, y)| < T_n \\ 1 & |D_n(x, y) - B_n(x, y)| \geq T_n \end{cases} \quad (3)$$

背景差分法能非常迅速提取出运动目标,而且完整性较好。但容易受到光线变化的影响,而且背景模型的更新是关键,不太适用于摄像头运动的情况。

### 1.2 帧间差分法

基本思想:帧间差分法就是抽取一段图像序列,选取相邻两帧或多帧之间作差分运算;从而直接得到前景目标的轮廓形态。视频序列中出现运动目标后,相邻两帧或多帧之间会出现较大差别,通过两帧或多帧作差分运算,得到每个像素点差值的绝对值,然后将其与选择的阈值作比较,以检测序列图像中是否有运动目标和目标大体轮廓<sup>[2]</sup>。相邻两帧作差分的公式为

$$D_k(x, y) = |I_k(x, y) - I_{k-1}(x, y)| \quad (4)$$

其中  $D_k(x, y)$  是相邻两帧在  $(x, y)$  像素点处作差分运算得到的灰度值,  $k$  代表第  $k$  帧图像与第  $k-1$  帧图像作差分,  $I_k(x, y)$  和  $I_{k-1}(x, y)$  分别是第  $k$  帧图像与第  $k-1$  帧图像在  $(x, y)$  像素点的灰度值,对  $D_k(x, y)$  作二值化运算

$$G_k(x, y) = \begin{cases} 1 & D_k > T_n \\ 0 & D_k \leq T_n \end{cases} \quad (5)$$

其中  $T_n$  为阈值。若两图像间发生位置变化,则像素取值为 1;若两图像间没有发生位置变化,像素取值为 0,便可提取出感兴趣的前景目标。

帧间差分法实现过程相对简单,具有较低的程序设计复杂度和高度动态的环境适应性。对不同场景的变化相对不太敏感。稳定性更好。但该方法提取的目标完整性较差,仅能确定目标的大体轮廓。同时,帧间的时间间隔选择是关键,若运动目标运动过快,将导致前后两帧之间没有重叠,便检测到两

个分开的物体;若当目标移动缓慢,选择的帧间时间间隔过短,将导致相邻两帧间物体近似完全重叠,产生“空洞”现象。

## 2 目标跟踪

### 2.1 传统的 CAMSHIFT 算法

CAMSHIFT 算法是 Meanshift 算法的改进,称为连续自适应的 Meanshift 算法,实际上是对连续的图像序列进行分析<sup>[3]</sup>。其基本思想: CAMSHIFT 算法是将图像用颜色概率分布图表示,然后在当前帧图像中确定目标所在的位置和大小,在视频图像的下一帧中,初始化搜索窗口即为上一帧所得到的目标所在的位置和大小,经过多次迭代搜索最终确定目标所在的区域,从而达到跟踪的目的。传统的 CAMSHIFT 算法的流程大致分为 3 部分。

1) 反向投影。反向投影是将视频中的所有帧图像从 RGB 色彩空间转换到 HSV (Hue, Saturation, Value) 色彩空间,HSV 是根据颜色的最直观特性创建的一种颜色空间,又称为六角锥体模型 (Hexcone Model) (见图 1)<sup>[4]</sup>。它有 3 个颜色参数,色调  $H$ : 用角度衡量,取值范围为  $0^\circ \sim 360^\circ$ ,从红色为起始点按逆时针方向计算,红色为  $0^\circ$ ,绿色为  $120^\circ$ ,蓝色为  $240^\circ$ 。饱和度  $S$ : 取值范围为  $0.0 \sim 1.0$ ;亮度  $V$ : 取值范围为  $0$  (黑色)  $\sim 1.0$  (白色)。转换后对  $H$  分量作直方图统计,由此得到不同  $H$  分量出现的概率,通过直方图可查找大小为  $h$  的概率,最后即可得到一个颜色概率的查找表,将图像中每个像素的值替换成其对应的颜色概率,最终会得到颜色概率分布图,它是一个灰度图像,这个过程称为反向投影。

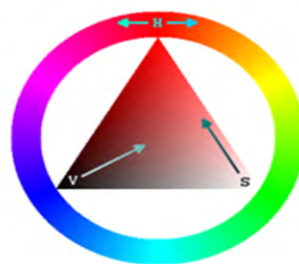


图 1 HSV 颜色模型图

Fig. 1 HSV color model

2) Meanshift 算法。该算法是一种无参数核密度估计算法,最开始时它被称为偏移的均值向量;用于跟踪方面,Meanshift 算法就相当于一个迭代的步骤;偏移均值会通过多次迭代收敛到一定条件结束。一般分为以下几个步骤: ① 在已经得到的颜色概率分布图中选取开始搜索窗  $S$ 。② 计算零阶距  $M_{00}$ , 一阶距  $M_{10}, M_{01}$ ; 计算搜索窗的质心:  $X_c = M_{10}/M_{00}, Y_c = M_{01}/M_{00}$ ; 调整搜索窗大小: 宽度为  $S = \sqrt{M_{00}/256}$ 。③ 将搜索窗的中心移动到上一步求得的质心点的位置,选定相对固定的阈值  $T$ ,如果移动的距离大于  $T$ ,则重复以上步骤,当搜索窗的中心和质心的距离小于  $T$  时结束<sup>[5]</sup>。

3) CAMSHIFT 算法。它实际上是对连续的视频序列中的所有帧图像做 meanshift 迭代运算,首先根据当前帧得到搜索窗的大小和位置,然后利用上一帧得到的结果作为 meanshift 算法下一帧搜索窗的初始值。经过反复的迭代运算后可实现对目标的准确有效的跟踪。传统的 CAMSHIFT 算法需手动选取运动目标所在的区域,而且只能单纯地考虑颜色直方图,在背景或其他目标颜色像素干扰的情况下,会出现跟踪错误或丢失的现象,在摄像机运动的情况下其跟踪效果不好,不适合做多目标跟踪。所以需对其算法进行相应的改进。

### 2.2 基于背景差分法和帧间差分法相结合的 CAMSHIFT 目标跟踪算法

根据背景差分法和帧间差分法的基本原理,可将这两种方法进行融合,使它们的优势得到互补,提高运动检测的效果<sup>[6]</sup>。首先选取一个背景帧,对每个像素点建立单高斯背景模型,然后在图像序列中选取相邻两帧作差分运算,区分出目标区域和非目标区域,将非目标区域更新到背景帧中,这样在当前帧图像中即可清晰分辨目标区域和背景区域,得到目标轮廓和轮廓内相关像素点的信息,在目标区域内建立颜色概率分布图,结合 CAMSHIFT 算法对目标进行跟踪。

算法步骤如下。

- 1) 在视频初始帧时选取只包含背景像素的图像,并对其建立模型,设初始背景图像  $B_n(x, y)$ 。
- 2) 选取当前帧图像  $I_k(x, y)$ , 将其与建立的背景模型之间作差分运算  $D(x, y) =$

$|I_k(x, y) - B_n(x, y)|$  从图像中能得到完整的运动目标。

3) 再选取与其相邻的前一帧图像  $I_{k-1}(x, y)$ , 并通过计算当前帧与前一帧图像之间的差值  $D_k(x, y) = |I_k(x, y) - I_{k-1}(x, y)|$ , 得到目标的变化部分, 将非目标部分更新到背景图像  $B_n(x, y)$  模型中。

4) 根据背景差分法得到的差值  $D(x, y)$  和帧间差分法得到的帧间差  $D_k(x, y)$  的交集得到了一个前景目标所运动的区域的基本轮廓。

5) 对于背景图像的实时变化的情况, 可用均值法

$$B_n(x, y) = \frac{1}{N} \sum_{i=1}^N \text{image}_i(x, y) \quad (6)$$

求取变化后的背景, 再反复 2) ~ 4) 步骤, 直到检测到变化后的目标。

6) 由于确定了运动目标所在的区域, 所以在用 CAMSHIFT 算法进行跟踪时无需手动选取运动目标, 只需将检测的目标所在区域作为初始搜索窗口, 然后通过反向投影建立颜色概率分布图。

7) 在反向投影图中利用 Meanshift 算法迭代搜索, 通过选取相似系数在相邻两帧图像中找到最大的匹配区域, 将搜索窗的中心移动到搜索结果的质心位置, 自适应调整搜索窗口的大小, 即 CAMSHIFT 搜索算法<sup>[7]</sup>。

8) 得到的当前帧搜索窗口的中心和大小作为下一帧 CAMSHIFT 搜索窗的起点, 经过反复迭代, 最终跟踪到运动目标。

改进的 CAMSHIFT 算法流程图如图 2 所示。

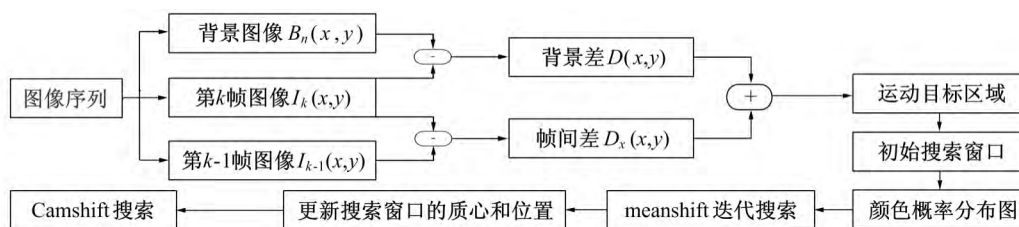


图 2 改进的 CAMSHIFT 算法流程图

Fig. 2 Flow chart of improved CAMSHIFT algorithm

### 3 实验结果及分析

该实验运行的环境是 VS2012 搭配 Opencv2.4.8; VS2012 是微软推出的当下最流行的应用程序开发环境。Opencv 是基于开源发行的跨平台计算机视觉库, 它由一系列 C 函数和 C++ 类构成。VS2012 与 Opencv 配置时需注意 Opencv 需较高版本, 而且只需配置一次即可永久使用<sup>[8]</sup>。

实验对象为一段小车运动的视频, 根据小车自身的颜色特征进行跟踪, 首先提取出一帧背景图像 (见图 3), 然后通过背景差分法和帧间差分法相融合检测运动目标, 从而提取运动目标的轮廓 (见图 4), 并用椭圆框标记出运动目标 (见图 5), 检测出小车所在的区域和大体的轮廓后, 用 CAMSHIFT 算法进行跟踪。在小车所在目标区域内建立的颜色概率分布直方图如图 6 所示。传统 CAMSHIFT 目标跟踪算法跟踪效果图如图 7 ~ 图 11 所示。



图 3 背景帧图像

Fig. 3 Image of background



图 4 运动目标的大体轮廓

Fig. 4 The outline of the motion target



图5 用椭圆框标记出运动目标

Fig.5 The oval box marking target

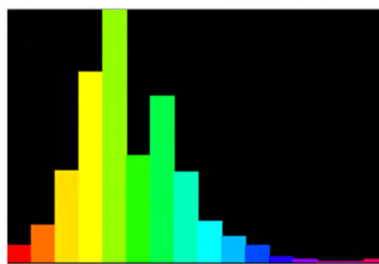


图6 颜色概率分布直方图

Fig.6 Color probability distribution histogram



图7 第10帧图像

Fig.7 The tenth frame image



图8 第30帧图像

Fig.8 The thirtieth frame image



图9 第50帧图像

Fig.9 The fiftieth frame image



图10 第70帧图像

Fig.10 The seventieth frame image



图11 第100帧图像

Fig.11 The 100th frame image

基于背景差分法和帧间差分法相融合的 Camshift 目标跟踪算法跟踪效果图如图12~图16所示。



图12 第10帧图像

Fig.12 The tenth frame image



图13 第30帧图像

Fig.13 The thirtieth frame image



图14 第50帧图像

Fig.14 The fiftieth frame image

实验表明,对同一段视频的图像序列用基于背景差分法和帧间差分法相融合的 CAMSHIFT 算法与传统的 CAMSHIFT 算法进行实验对比分析的结果表明,利用传统的 CAMSHIFT 目标跟踪算法进行跟踪,当运动小车完整地出现在摄像机的视场范围内时,需手动选择一个初始搜索窗;而使用改进后的算法,首先用背景差分法和帧间差分法相结合可完整地检测运动的人体目标及其所在区域<sup>[9]</sup>,无需手动选取运动目标,只需要把检测的目标所在区域作为初始搜索窗口,即椭圆所标记的区域,然后在此区域内通过反向投影建立颜色概率分布直方图(见图6),再结合 CAMSHIFT 迭代搜索算法,可发现,在整个跟踪过程中,在第50帧左右时,两种算法均可跟踪到运动的小车目标,跟踪效果较为稳定,当跟踪到100帧左右,



小车逐渐接近摄像机的视场边缘时,可以发现传统的 CAMSHIFT 算法搜索窗已经发散,跟踪结果失败,而改进后的算法仍然可准确地跟踪到运动目标,没有出现目标丢失和跟踪窗发散的现象,跟踪仍然持久和稳定。



图 15 第 70 帧图像



图 16 第 100 帧图像

Fig. 15 The seventieth frame image      Fig. 16 The 100th frame image

表 1 表示跟踪过程中,算法改进前后目标的中心点位置与真实位置之间的绝对误差,此绝对误差是两点之间的绝对距离误差。在前景目标运动过程中,使用传统的 CAMSHIFT 算法跟踪误差急剧变大<sup>[10]</sup>,而使用背景差分法和帧间差分法相融合的 CAMSHIFT 改进算法则保持了跟踪误差较小。

表 1 算法改进前后的目标中心点坐标绝对误差

Tab.1 The absolute error of the target center point coordinates before and after the improvement of the algorithm

	目标真实中 心点坐标	算法改进前目 标中心点坐标	算法改进后目 标中心点坐标	算法改进前绝 对误差/像素	算法改进后绝 对误差/像素
第 30 帧	( 68 ,95)	( 64.6 ,90)	( 67.8 ,94)	6.05	1.02
第 50 帧	( 80 ,102)	( 72 ,90.2)	( 78 ,98.6)	14.26	3.94
第 70 帧	( 112 ,108)	( 102 ,96)	( 108 ,104.2)	15.62	5.52
第 100 帧	( 138 ,114)	( 125 ,102.8)	( 133.8 ,108)	17.16	7.32

4 结 语

运动目标跟踪是计算机视觉领域的一个重要的方面,运动目标检测与跟踪的算法必须具有实时性好,跟踪准确度高,性能具有鲁棒性。笔者分析了背景差分法,帧间差分法和传统 CAMSHIFT 算法的基本原理,提出了一种基于背景差分法和帧间差分法相结合的 CAMSHIFT 算法,通过背景差分法与帧间差分法的融合,能快速和准确地检测运动目标,对目标的几何特征提取的误差小,改变了 CAMSHIFT 算法需手动选取目标区域的问题,对背景不断变化有很强的适应性,改进后的 CAMSHIFT 算法取得了良好的跟踪效果,实现了目标跟踪时间持久和稳定,具有很强的适应性和鲁棒性。

参考文献:

[1] 卢章平,孔德飞,李小蕾,等. 基于差分与三帧差分结合的运动目标检测算法 [J]. 计算机测量与控制,2013, 21( 12): 3315-3318.  
LU Zhangping, KONG Defei, LI Xiaolei, et al. Moving Object Extraction Based on Background Difference and Frame Difference Method [J]. Computer Measurement and Control, 2013, 21( 12): 3315-3318.

[2] 张小建,徐慧. 基于视频处理的运动车辆检测算法的研究 [J]. 液晶与显示,2012, 27( 1): 108-113.  
ZHANG Xiaojian, XU Hui. Moving Vehicle Detection Algorithm Based on Video Processing [J]. Chinese Journal of Liquid Crystals and Displays, 2012, 27( 1): 108-113.

[3] 卢璇,雷航,郝宗波. 联合多特征的自动 CamShift 跟踪算法 [J]. 计算机应用,2010, 30( 3): 650-652.  
LU Xuan, LEI Hang, HAO Zongbo. Automatic CamShift Tracking Algorithm Based on Multi-Feature [J]. Journal of Computer Applications, 2010, 30( 3): 650-652.

[4] 樊兴,刘祯,武云鹏,等. 基于视频图像的目标识别与跟踪算法 [J]. 火力与指挥控制,2014, 39( 6): 116-119.  
FAN Xing, LIU Zhen, WU Yunpeng, et al. Study on Algorithms of Target Recognition and Target Tracking Based on Video Sequence [J]. Fire Control & Command Control, 2014, 39( 6): 116-119.

- [5] 张娟,毛晓波,陈铁军. 运动目标跟踪算法研究综述 [J]. 计算机应用研究, 2009, 26(12): 4407-4410.  
ZHANG Juan, MAO Xiaobo, CHEN Tiejun. Survey of Moving Object Tracking Algorithm [J]. Application Research of Computers, 2009, 26(12): 4407-4410.
- [6] 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 [J]. 光电工程, 2010, 37(1): 55-60.  
WU Dapeng, CHENG Weiping, YU Shenglin. Camshift Object Tracking Algorithm Based on Inter-Frame Difference and Motion Prediction [J]. Opto-Electronic Engineering, 2010, 37(1): 55-60.
- [7] 王兆光,王敬东,李鹏. 一种 Camshift 优化的粒子滤波跟踪算法 [J]. 光电子技术, 2010, 30(1): 58-63.  
WANG Zhaoguang, WANG Jingdong, LI Peng. A Camshift Optimized Particle Filter Tracking Algorithm [J]. Optoelectronic Technology, 2011, 30(1): 58-63.
- [8] 侯向丹,董永峰,坎启娇,等. 基于运动轨迹的视频目标跟踪算法 [J]. 计算机工程与设计, 2015, 36(4): 995-998.  
HOU Xiangdan, DONG Yongfeng, KAN Qijiao, et al. Video Target Tracking Based on Movement Trace [J]. Computer Engineering and Design, 2015, 36(4): 995-998.
- [9] 毛育佼,陈苒君,唐莉萍. 视频图像中运动目标检测算法的提高 [J]. 计算机系统应用, 2014, 23(4): 107-111.  
MAO Yujiao, CHEN Ranjun, TANG Liping. Optimization of the Moving Target Detection Algorithm in Video Images [J]. Computer Systems & Applications, 2014, 23(4): 107-111.
- [10] 王玲玲,裴东,王全州. 一种改进的 Camshift 视频目标跟踪算法 [J]. 激光与红外, 2015, 45(10): 1266-1271.  
WANG Lingling, PEI Dong, WANG Quanzhou. Video Target Tracking Algorithm Based on Improved Camshift [J]. Laser & Infrared, 2015, 45(10): 1266-1271.

(责任编辑: 刘东亮)