

复杂情况下的多目标跟踪统计技术

金 鑫 梁雪春 袁晓龙

(南京工业大学自动化与电气工程学院 南京 211816)

摘 要 根据基于视频监控客流量统计的应用要求,得到了一种改进的背景检测和跟踪计数方法,实现了多人准确跟踪计数。传统的高斯背景建模是对一帧图像的每个像素点进行更新且分配固定的高斯分布个数,使得资源消耗量增大;这里通过先判断待更新区域,然后对更新区域采用动态调节高斯分布的方法对像素点进行更新,同时考虑到均值与方差的特点,分别设置了各自的更新速率。跟踪部分利用连通域分析创建人体结点并得到目标的形心,采用向前优先搜索像素点的原则搜索下一帧图像的所有像素点,通过搜索到的像素点来确定目标的新中心位置,再根据目标中心与计数线的关系进行计数。实验证明该算法简单可行,实现了多目标的准确跟踪,统计数据具有较高的正确率。

关键词 视频监控,客流量统计,背景建模,混合高斯模型,连通域分析,多目标跟踪

中图分类号 TP391.4 文献标识码 A

Multi-target Tracking Statistical Techniques in Complex Case

JIN Xin LIANG Xue-chun YUAN Xiao-long

(College of Automation and Electrical Engineering, Nanjing University of Technology, Nanjing 211816, China)

Abstract According to application requirement of video-based traffic statistics in video surveillance, the paper proposed improved background detection and tracking count method. Each pixel of a frame is to be updated in the traditional Gaussian background modeling and the number of Gaussian distribution is fixed which makes resources consumption increase. This article proposed that the update area is need to be fund firstly and then update the area. The region is updated by using dynamic adjusting Gaussian distribution method. At the same time, considering the characteristics of the mean and variance, and seting respectively their update rate. Connected domain analysis is applied to create the human node and get the node's centroid. Finally according to the nodes' centroid which has been created in the list, first search forward is used. All target pixels are searched in the next frame which are used to determine the new position of the object in the video. Experiments show that the algorithm is simple and feasible and implements accurate tracking of multiple targets. The system achieves statistical data with high accuracy.

Keywords Video surveillance, Traffic statistics, Background model, Mixture Gaussian model, Tracking of multiple targets, Connected domain analysis

1 引言

人体运动的视觉分析是计算机视觉领域的重要课题之一。视频中背景的更新与目标提取和运动目标跟踪是智能监控领域中比较关键的技术。

传统的混合高斯算法运行效率低,不能适应实时性,很多学者对其进行了改进。李等^[1]提出了图像参数更新模型,从而提高了算法速度。蒋等^[2]提出了两阶段视频处理方法设计,提高了检测质量。但两者有共同的缺点,即很难检测运动很慢的物体。本文在目标检测阶段结合了李等人的方法并通过指定待更新区,使用动态变动的高斯分布进行更新,最后在更新过程中根据相关性的高低,采用不同的均值与方差的更新速率,随着场景复杂度的变化,通过调节均值与方差的更新

速率,便能很好地检测到运动慢的物体。

目标跟踪算法包括基于视频运动对象区域灰度等特征的跟踪算法、基于轮廓特征的跟踪算法、基于运动估计的跟踪算法和基于视觉学习的跟踪算法等。其中 mean shift 算法^[3]、snake 算法^[4]和基于特征的跟踪是常用的跟踪算法。mean shift 算法是对目标的颜色进行建模,具有良好的实时性和跟踪性能。但 mean shift 根据颜色进行跟踪匹配,当跟踪物体颜色与背景颜色相差不大时跟踪就会失败。snake 活动轮廓模型算法(Active Contour Model, ACM)是一种较好的基于轮廓跟踪算法,但 snake 及其改进算法与初始轮廓给定的位置密切相关,若给定的初始形状和位置不佳,则不易收敛或形状演化不正确,且要取得理想的轮廓效果,活动轮廓算法演化时间较长,不适合实时操作。

到稿日期:2012-09-11 返修日期:2012-12-23 本文受江苏省博士后科研资助计划基金项目(1001027B),江苏省高校自然科学基金项目(09KJB510002)资助。

金 鑫(1989—),男,硕士生,主要研究领域为模式识别与图像处理,E-mail:superjx@foxmail.com;梁雪春(1969—),女,博士,教授,主要研究领域为计算机应用系统的研究与开发;袁晓龙 男,硕士生,主要研究领域为模式识别。

本文结合文献[5],采用基于特征的跟踪方法^[6,7],为了实现跟踪并不需要跟踪目标的每一点,只需跟踪目标上的某一特征点就可以实现对整个目标的跟踪。这个特征点可以是目标的质心,也可以是目标上的任意一点。但这种算法在处理多目标跟踪时经常会出现跟踪匹配错误,为了解决多目标精确跟踪的问题^[8],本文对其算法进行了改进,提出了根据方向向量向前优先搜索像素的方法来搜集目标像素,再利用计算特征点方法求出图像对象的位置,最后根据跟踪目标的方向矢量的变化寻找下一帧图像中跟踪目标的位置。该方法不但解决了实时性问题,而且解决了跟踪目标颜色与背景颜色相似的问题,最主要的是该方法很好地解决了在监视区域中有多人走动时也能够准确跟踪计数的问题。

2 人体运动的检测

2.1 混合高斯背景提取

Stauffer 等人用由 K 个高斯分布组成的混合高斯模型^[9]表示同一个像素在时间域上的概率分布,即图像中的第 n 个像素在 $1 \sim t$ 不同时刻的取值分别为 $\{X_1^n, \dots, X_t^n\}$,当前帧图像中的第 n 个像素值在 t 时刻的概率分布为:

$$P(X_t^n) = \sum_{i=1}^K w_{i,t}^n \cdot \eta(X_t^n, \mu_{i,t}^n, \sigma_{i,t}^n) \quad (1)$$

其中, $\eta(X_t^n, \mu_{i,t}^n, \sigma_{i,t}^n) = \frac{1}{(2\pi)^{\frac{d}{2}} |\sum_{i,t} \frac{1}{2}|^{\frac{1}{2}}} \times \exp[-\frac{1}{2} (X_t^n - \mu_{i,t}^n)$

$(\sum_{i,t} \frac{1}{2})^{-1} (X_t^n - \mu_{i,t}^n)]$ 为第 i 个高斯分布的概率密度; $\mu_{i,t}^n = (\sigma_{i,t}^n)^2 I(\sigma_{i,t}^n)$ 表示标准差, I 表示单位矩阵) 为分布对应的均值和协方差; $w_{i,t}^n (\sum_{i=1}^K w_{i,t}^n = 1)$ 为分布对应的权值。 K 个高斯分布总是按照优先级 $p_{i,t}^n = \frac{w_{i,t}^n}{\sigma_{i,t}^n}$ 从高到低的次序排列,然后将当前图像中的像素值 X_t^n 与 K 个高斯分布进行匹配,匹配的条件为 $|X_t^n - \mu_{i,t}^n| < 3 \cdot \sigma_{i,t}^n$ 。若都不匹配,则用新的高斯分布代替最小优先级的高斯分布。新的高斯分布以 X_t^n 作为均值,初始标准差及权重设为 σ_{init} 和 w_{init} ;若与第 m 个高斯分布匹配,则对各高斯分布的更新方法如下:

$$\begin{aligned} \mu_{m,t}^n &= (1-\alpha) \cdot \mu_{m,t-1}^n + \alpha \cdot X_t^n \\ (\sigma_{m,t}^n)^2 &= (1-\alpha) (\sigma_{m,t-1}^n)^2 + \alpha (X_t^n - \mu_{m,t-1}^n)^T (X_t^n - \mu_{m,t-1}^n) \quad (2) \\ w_{i,t}^n &= (1-\beta) \cdot w_{i,t-1}^n + \beta \end{aligned}$$

式中, α 为模型学习率, β 为权值更新率。

在每次更新完后,重新对混合高斯分布按优先级排序,选定 B_n 个高斯分布作为背景分布:

$$B_n = \arg \min_b (\sum_{i=1}^b w_{i,t}^n > T_B) \quad (3)$$

阈值 T_B 决定了高斯分布在背景选取所占的比例,取值较小时,背景通常用一个高斯分布表示;取值较大时,背景由多个分布混合表示。

2.2 改进的混合高斯背景提取

2.2.1 确定待更新区域

原始的混合高斯背景提取是针对一帧图像中的每个像素进行建模匹配,在速度上会增加系统的计算时间。为了能更快且更准确地进行背景建模,本文对混合高斯背景建模算法进行了改进。

在一帧图像中,有背景区域和前景区域。在进行处理的时候,可以直接提取当前帧的背景像素点作为更新点,含有前

景的区域周围的像素可以通过改进的混合高斯建模实现更新。

$$P(x, y) = \begin{cases} 1, & |I_n(x, y) - I_{n-1}(x, y)| \geq T \\ 0, & |I_n(x, y) - I_{n-1}(x, y)| < T \end{cases} \quad (4)$$

式中, $I_n(x, y)$ 表示第 n 帧图像的像素点, T 是变化阈值,值 0 与 1 表示像素点是否需要更新。根据 $P(x, y)$ 找到最左、最右、最上、最下的像素点,根据这 4 个像素点确定待更新矩形区域。

2.2.2 动态调节混合高斯分布个数

在确定了待更新区域后,采用动态混合高斯模型对待更新区域进行更新。具体步骤如下:

1) 初始化:在第一帧图像中对每个像素点只选取一个高斯分布;

2) 动态增加高斯分布:判断新像素是否与现有的高斯分布相匹配,并判断高斯分布个数是否达到最大值,若没有达到最大值就添加一个新的高斯分布,初始化标准差及权重设为 σ_{init} 和 w_{init} ,否则就用原来的方法替换优先级小的高斯分布。

3) 去除相似高斯分布:如果同一像素点中的 i, j 两个高斯分布满足 $|\mu_{i,t}^n - \mu_{j,t}^n| < \tau$,则认为两高斯分布相似,此时可将两高斯分布进行合并。

$$\begin{aligned} \sigma_{n,t} &= \sqrt{\sigma_{i,t}^2 + \sigma_{j,t}^2} \\ \mu_{n,t} &= (\mu_{i,t} + \mu_{j,t}) / 2 \end{aligned} \quad (5)$$

2.2.3 动态调节均值与方差的更新速率

在传统的混合高斯背景更新中均值和方差的更新率都是 β ,这没有考虑到均值和方差各自的特点。本文考虑到了每个像素点对背景的置信度和新获取的帧与背景模型的相关性。

一个点与背景差异值的二次方为

$$\epsilon_{i,t}^2 = (I_{i,t} - \mu_{i,t})^2 \quad (6)$$

差异值的标准差为

$$\delta_i = 1.4826(1 + \frac{5}{N-1}) \sqrt{l_i^2} \quad (7)$$

式中, $\sqrt{l_i^2}$ 是 $\epsilon_{i,t}^2$ 的中值,点的置信度为

$$\gamma_{i,t} = \exp(-\frac{\epsilon_{i,t}^2}{2\delta_i^2}) \quad (8)$$

点 i 对图像的相关系数为

$$\lambda_{i,t} = \frac{\gamma_{i,t}}{\sum \gamma_{i,t}} \quad (9)$$

图像与背景的相关性通过式(10)给出。

$$\rho_t = \frac{\sum \lambda_{i,t} (I_{i,t} - \bar{I}_t) (\mu_{i,t} - \bar{\mu}_t)}{\sqrt{\sum (\lambda_{i,t} (I_{i,t} - \bar{I}_t)^2)} \sqrt{\sum (\lambda_{i,t} (\mu_{i,t} - \bar{\mu}_t)^2)}} \quad (10)$$

式中, $\bar{I}_t = \sum \lambda_{i,t} I_{i,t}$, $\bar{\mu}_t = \sum \lambda_{i,t} \mu_{i,t}$ 。

本文考虑到实时性的要求,式(8)用式(11)代替。

$$\gamma_{i,t} = 1 + \theta(1 + \frac{\theta}{2} + \frac{\theta^2}{6}) \quad (11)$$

式中, $\theta = -\frac{\epsilon_{i,t}^2}{2\delta_i^2}$ 。

式(10)改成

$$\rho_t = \sum \lambda_{i,t} |I_{i,t} - \mu_{i,t}|$$

通过大量实验可知,高斯模型参数的更新率通常需要根据相关性和点的置信度来确定。通过函数 f_ρ 来表示在 a_t 时全局相关性的影响。为了能够更新复杂情况下的背景,在相关性升高(如高于某一阈值)时, f_ρ 应当迅速达到最小值。即

$$f_\rho = \begin{cases} f_{\max}, & \rho_t < t \\ f_{\min}, & \rho_t \geq t \end{cases}$$

式中, t 是阈值, 一般取 0 到 1 之间较大的数, 本文取 0.8。 $f_{\min}$, $f_{\max}$ 分别是 f_p 的下限值和上限值, $f_{\min}$ 一般在 0 到 0.1 之间取值, 本文取 0.01。 $f_{\max}$ 取值大于 0.9 小于 1, 本文取 0.95。相对具体某个点而言, 在置信度高时, 更新速率, 即应该高; 置信度低时, 更新速率应该低。由此确定每个点的均值更新速率, 即

$$\beta_t = \frac{f_p(1 + \gamma_{it})}{2}$$

$$\beta^2 = \frac{1}{t^2}$$

本文用 β^2 的原因是在刚开始时 t 较小, β^2 较大, 从而使模型的收敛性较好。经过一段时间后, β^2 变小使得模型趋于稳定。

3 跟踪与计数

3.1 人体创建

通过连通域分析^[9]的方法来收集人体所有前景像素, 通过连通域的分析可以去除干扰点, 并有效地分析出人的轮廓信息。其主要思想就是通过 9 宫格来判断周围 8 个点是否为有效点(黑点), 若是就保存下来, 再判断下个点, 依次迭代下去直到一帧全部扫描结束。

算法实现如下:

Step 1 用 10×10 的方格(根据视频悬挂高度决定)分割屏幕, 并用这个方格扫描屏幕, 这样做是为了减小系统开销。如果在这个方格中出现有效黑点, 就将这个方格记录下来并将其放到一个数组中, 其实是将方格左上角的点记录下来, 放到数组中去。这样循环下去, 屏幕上所有格子的有效点都将被装到数组里, 假设这个结构体数组为 $Point[i]$ 。

Step 2 接下来从这个数组 $Point$ 任取一个元素(也就是一个点), 将其放到另一个数组中去, 这个数组为 $zone[m][n]$, 二维数组 $zone$ 的目的就是存放一个连通域的所有有效点(格子)。 m 表示域个数, n 表示每个域最多能存放的有效点的个数。例如: 将刚才放到 $zone[i][0]$ 的点与 $Point$ 里的每个点作比较, 如果它们之间的距离小于 10 个像素点, 就认为它们之间连通。将中间的点与其周围的 8 个点作比较, 如果条件成立(它们之间的距离小于 10 个像素点)就将 $Point$ 里的这些点放到 $zone[i][j]$ 里, 依次循环下去, 人的连通域就可以找到了。

Step 3 排除干扰点, 有些干扰点就会被放到其他 $zone[i]$ 里, 最后判断 $zone[i]$ 里有效点的个数, 如果其小于一定数值就默认为不是一个连通域, 将其释放掉。最后留下来的就是一些人的连通域。

Step 4 根据得到的连通域的像素点计算 $zone[i]$ 连通域的宽度。本文通过测量, 得到一个标准人的宽度为 W (约 30) 个像素长度。根据这个数据就可以将人进行分割。假设 d 表示连通域的宽度, n 表示人数, u 表示人宽度的最小值。将人进行分割:

```
n=d/W;
if((n%W)>=u)
n=n+1;
```

Step 5 分割之后, 所有的连通域进行了一次新的组合, 现在每一个 $zone[i]$ 代表一个人, 根据 $zone[i]$ 里的像素可以算出整个连通域的 $zone[i]$ 的中心值, 即人的形心位置 (x, y) 。

将这些值都统一存放在 $person[i]$ 的结构数组里, 这个过程就成功创建了所有人的结点。

在最后 Step 4 分割这一部分其实就解决了并排多人行走跟踪的问题。

3.2 基于方向向量的跟踪

为了实现多人准确跟踪, 在文献[10]的基础上, 本文采用了新的方法, 即向前优先搜索前景像素的原则。通过采用方向偏移标志 $Direction(1, 2, 3, 4)$ 分别代表上下左右 4 个方向)确定物体的移动方向, 最后在移动的方向上找取人的下一帧像素点, 这样就能准确确定人的位置信息。通过第 $n-3$ 帧图像中的对象位置 (x_0, y_0) 与第 n 帧图像中相应的对象位置 (x_1, y_1) 的比较, 确定方向偏移标志 $Direction$ 值。

(1) 若 $|x_1 - x_0| - |y_1 - y_0| > 0, x_1 - x_0 > 0$, 则 $Direction = 4$;

(2) 若 $|x_1 - x_0| - |y_1 - y_0| > 0, x_1 - x_0 < 0$, 则 $Direction = 3$;

(3) 若 $|x_1 - x_0| - |y_1 - y_0| < 0, y_1 - y_0 > 0$, 则 $Direction = 2$;

(4) 若 $|x_1 - x_0| - |y_1 - y_0| < 0, x_1 - x_0 > 0$, 则 $Direction = 1$ 。

算法实现如下:

Step 1 根据创建的人体结点, 通过标志 $Direction$ 的值, 向该方向搜索像素; 通过半径为 R (约 15 个像素) 的圆依次向前移动, 而且每次向前移动 t (5) 个像素的距离, 共移动 k (3) 次, 每次移动后将在圆范围内的有效像素点都读取出来。

Step 2 从这些像素点中找到 x 的最大最小值 $\min x$, $\max x$ 和 y 的最大最小值 $\min y$, $\max y$ 。这样就可以根据 $\max y - \min y$ 的值的大小来判断前后是几个人。在通过测试后发现一个人的前后的宽度大约在 W 个像素左右。这样就可以通过前后的宽度来判断后面是否还有人。

Step 3 如果 $\max y - \min y > H$ (人的前后厚度, 约 40), 那么假设后面还有一个人。先把一个人的前后宽度里面的点全部找出来, 根据这些点来获得新的结点圆心坐标。

Step 4 那么多于一个人宽度的点还要释放出来, 这些点也许就是后面人的像素点。将释放出来的像素点置成未被读取状态, 以便后面的人创建信息。

在 Step 3 就很好地解决了前后多人一起走的情况, 将前后靠在一起的人分割开来。

3.3 计数

当被跟踪物体在创建区 $S1(S2)$ 创建时, 我们必须将这个结点 $person[i]$ 的 in_or_out 元素置 1(0)。当结点到达区域 $S2(S1)$ 时, 先判断是否已有结点, 若是就再根据 in_or_out 的值判断是否计数。若 $person[i].y >$ 监控区域上(下)边界值且 $person[i].in_or_out == 1(0)$, 则将 $person[i]$ 结点删除, $IN(OUT)$ 计数器加 1。其他出边界的情况则只删除结点不计数。

4 实验结果

4.1 实验 1 背景更新与提取

本文是在 Linux 环境下用标准 C 语言实现的, 视频的分辨率为 288×352 , 帧率为每秒 25 帧数据。图 1(a) 与图 1(b) 是分别使用李等方法与本文方法所得到的背景提取与目标提取效

果图。表 1 给出了本文方法与传统方法运行时间的对比。



图 1 背景提取更新

表 1 传统高斯背景建模与本文方法运行时间对比

算法名称	平均运行时间 ms/帧
传统高斯背景建模	33
李等人方法	26
本文方法	21

初始化各个变量,每个像素选取一个高斯分布, w_{init} 为 0.05,标准差 σ_{init} 为 20,权值更新率 β 为 0.1。

从图 1 中可以看出,本文的方法使背景提取与前景提取效果比李等人提出的方法更好,在运算时间上(见表 1)也有一定的提高。

4.2 实验 2 目标跟踪与统计计数

本文跟踪算法在跟踪时仅需要进行特征点计算,跟踪效果能够满足实时要求。

如图 2、图 3 中上下两个矩形框分别为创建区 S_1 (S_2)。只有人体进入创建区后才创建跟踪框,在离开跟踪区域后删除跟踪目标结点并计数。人身上的矩形框为跟踪框,其中心代表人体中心。图 2 对紧靠在一起的人进行了准确的分割与跟踪。图 3 对人体的合并与分离进行了准确跟踪,解决了复杂情况下的准确匹配跟踪。



图 2 两人并肩跟踪效果



图 3 复杂人群跟踪效果

通过视频播放,使用本文算法对某家商店从下午 1 点到 7 点之间的客流量进行数据统计所得到的数据记录如表 2 所列。

表 2 客流量统计精度

测试场景	实际进入人数	系统统计的进入人数	进入人数准确率(%)	实际出去人数	系统统计的出去人数	出去人数准确率(%)
场景 1	974	965	99	912	1789	93
场景 2	1311	1274	97	1215	1266	96

本文算法的优点在于解决了因客流拥挤进入视频监控区而造成的跟踪匹配错误,提高了跟踪和计数的精确度,同时满足了实时性的要求,但仍然存在误判,主要原因是人群中混有类似人大小(人的宽度、厚度)的移动物体,但一般的不同于标准人尺度大小的移动物体能够准确识别出来。

结束语 提出了一种改进的混合高斯模型在客流量统计研究中的应用。在高斯模型中主要是减少了计算量,区分开了均值与方差的更新速率。跟踪部分主要通过改进的特征跟踪,引入了预测方向向量,实现了在视频监控区内的单人来回走、多人并肩走、多人相向走、前后串走等多种复杂情况下的准确计数。通过实验表明,该设计方法的识别准确率在 96% 以上,具有较好的实用价值。

参考文献

- [1] 李娟,邵春福,杨励雅. 基于混合高斯模型的行人检测方法[J]. 吉林大学学报:工学版,2011,41(1):41-45
- [2] 蒋明,潘娇丽. 一种改进的混合高斯背景估计方法[J]. 微型机与应用,2011,30(11):31-34
- [3] 袁霄,王丽萍. 基于 MeanShift 算法的运动人体跟踪[J]. 计算机工程与科学,2008,30(4):46-49
- [4] 刘皓挺,姜国华,王丽. 一种基于 Snake 模型的多目标跟踪算法[J]. 计算机工程与应用,2006,23(12):76-79
- [5] 陈明,陈一民,黄诗华. 增强现实中视频对象跟踪算法[J]. 计算机工程,2010,36(12):229-231
- [6] Hager G D, Dewan M, Stewart C V. Multiple Kernel Tracking with SSD[J]. IEEE Conference on Computer Vision and Pattern Recognition, 2004, 23(4): 790-797
- [7] Dai Y P, Yu G H, Hirasawa K. New Development on Tracking Algorithm with Derivation Measurement[J]. IEEE International Conference on System, Man and Cybernetics, 2001, 25(4): 3181-3186
- [8] 魏娟丽,翟社平,王万诚. 视频序列中人体运动目标的检测与跟踪研究[J]. 计算机应用与软件, 2006, 23(4): 139-141
- [9] 赵宏伟,冯嘉. 一种实用的运动目标检测和跟踪算法[J]. 吉林大学学报:工学版, 2009, 39(2): 386-390
- [10] 张德才. 视频流中多人跟踪算法研究[D]. 长春:吉林大学, 2011

(上接第 255 页)

- [9] Stogbauer H, Kraskov A, Astakhov S A, et al. Least dependent component analysis based on mutual information [J]. Physical Review E, 2004, 70(6): 066123
- [10] Xing Hong-jie, Hu Bao-gang. Two-Phase Construction of Multi-layer Perceptions Using Information Theory[J]. IEEE Transactions on Neural Networks, 2009, 20(4): 542-550
- [11] Rossi F, Lendasse A, Francois D, et al. Mutual information for the selection of relevant variables in spectrometric nonlinear modeling[J]. Chemometrics and Intelligent Laboratory Systems, 2006, 80: 215-226

- [12] Huang Guang-bin, et al. A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation[J]. IEEE Transaction on Neural Networks, 2005, 16(1): 57-67
- [13] Xu Jin-hua, Ho D W C. A new training and pruning algorithm based on node dependence and Jacobian rank deficiency[J]. Neurocomputing, 2006, 70(1-3): 544-558
- [14] Ridella A, Rovetta S, Zunino R. Circular back propagation for classification[J]. IEEE Transaction on Neural Networks, 1997, 8(1): 84-97