

## 一种用于运动物体检测的自适应更新背景模型

张彦 周忠 吴威

(虚拟现实技术与系统国家重点实验室 北京 100083)

(北京航空航天大学计算机学院 北京 100083)

(zy@vrlab.buaa.edu.cn)

**摘要** 提出一种能自适应更新的背景模型,使得运动物体检测中场景变化时提取的背景能进行相应的更新.该模型利用特征点信息将场景中的像素点分类,并针对不同类别像素点计算其更新速率;然后对像素点采用相应速率进行更新,从而能根据场景不同变化进行自适应更新.实验结果表明,该模型能较好地处理混合高斯模型因采用同一更新速率导致的背景模型更新错误问题.

**关键词** 背景模型;背景差;特征点;视频监控;运动物体检测

中图分类号 TP391

## Adaptive Update Background Model for Detecting Moving Objects

Zhang Yan Zhou Zhong Wu Wei

(State Key Laboratory of Virtual Reality Technology and Systems, Beijing 100083)

(School of Computer Science and Engineering, Beihang University, Beijing 100083)

**Abstract** An adaptive background updating model is proposed to tackle possible background changes in moving object detection. The model uses feature points to classify the pixels in the scene, then compute their updating rates according to their classification information. Finally, background is updated adaptively with varying update rate. The experimental results show that the proposed model can effectively cope with the inaccurate updating problem in the Gaussian mixture model due to the fixed update rate.

**Key words** background model; background subtraction; feature points; video surveillance; moving object detection

背景差是利用背景模型进行前景物体检测时经常使用的方法,背景模型是对视频中背景的表达.背景差通过当前帧和背景模型比较区分前景物体和背景物体.通过比较得到背景模型中发生显著变化的区域,然后利用连通区域分析方法对前景物体区域进行标记.得到的前景物体区域信息可用于更高层的视觉系统任务,如运动预测、跟踪等.而背景差方法中重要的一步就是对背景采用合适的模型进行

表达,并且能够根据背景的变化情况以及前景物体的运动情况对背景模型进行相应更新.因此,背景模型在背景差方法中起着关键作用.

背景模型的建模方法按照特征采样的基本单位可分为 2 类:1) 基于区域的建模方法;2) 基于像素点的背景建模方法.基于区域的建模方法通常将图像分成多个区域,然后对这些区域分别进行计算,这样做是为了避免大量的计算,降低计算复杂度.例如,

文献[1]在采用在线自回归模型的方法时,为了避免自回归模型采用连续多帧原始图像序列作为输入引起的高维问题,将图像分为固定大小的图像块;然后对图像块进行主成分分析以得到基本特征向量,并以此作为自回归模型的输入特征.类似地,文献[2]将这种随时间变化的动态区域采用自回归移动平均模型建模;然后采用 Kalman 滤波器对动态区域的外观进行估计,并通过调节权重函数的阈值得到前景物体的区域.

由于第 1 类方法以区域为单位进行计算,在降低计算复杂度的同时也带来了计算精度的下降,因此很多研究人员采用基于像素点的背景建模方法.其中,最直观的检测运动物体的方法是比较当前帧图像和背景图像相对应的像素点的灰度值变化,当变化值超过一定阈值时,该像素被认为属于运动物体.文献[3]对图像中像素点在 $(Y, U, V)$ 颜色空间中以高斯模型进行建模,但在很多情况下,一个高斯模型无法适应较复杂的背景的颜色变化情况.因此,文献[4]对单个像素点采用高斯混合模型进行建模.但是对于背景中的动态纹理区域,如水面的波纹、树叶的摇摆等,该方法存在难以确定高斯模型数目和选择模型更新速率的问题.文献[5]采用了计算背景像素点在特定状态时的最大值和最小值,并将其用于对后续帧进行前景背景判别.文献[6]采用对像素灰度值进行归类的方法进行背景的重建,但它的前提条件是背景像素在图像序列中出现频率最高,然而在很多情况下该条件并不成立.

也有研究人员从像素点变化的历史信息计算像素点在场景中的状态.文献[7-8]提出采用隐马尔可夫模型(hidden Markov model, HMM)对背景进行建模.文献[9]采用非参数核密度估计方法对像素点进行建模.文献[10]采用了类似的方法,不同之处在于它强调使用更多不同带宽的核进行自适应密度估计,并结合颜色值信息和光流信息作为输入特征.

由于只从像素值变化信息难以区分环境的光线变化、动态纹理区域的变化和前景物体运动等场景,而上述方法在背景更新时不区分不同场景变化,因此无法针对不同情况进行背景更新,存在背景模型逐渐偏离真实背景情况的问题.如果结合特征点信息则能对场景变化做出相应判断,并根据不同情况对背景模型更新速率进行计算.虽然文献[11]通过计算 Harris 特征点<sup>[12]</sup>,并采用类尺度不变性特征变换(scale invariant feature transform, SIFT)<sup>[13]</sup>描述子描述,减少了背景差方法的计算量,但存在难以将前景特征点与前景物体对应的问题.由于特征点分布的稀疏性质,在很多情况下,尤其是场景中有很多同步运动物体的情况下,难以将特征点和运动物体进行正确对应,从而错误地估计运动物体的数量和大小.如图 1 所示(图 1 b 的特征点是由图 1 a 提取得到的),中间区域的点为检测出的 Harris 点,并且都是向右运动,因此我们无论是从其运动轨迹还是特征点的分布状况,都难以辨认出这些点是来自于一个物体还是多个物体.

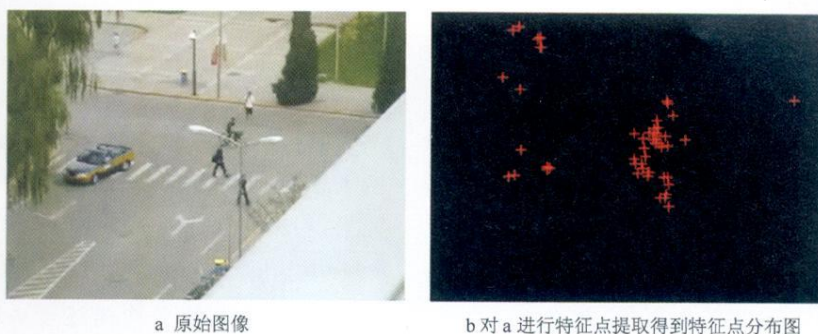


图 1 原始图像和提取特征点后图像

## 1 本文方法

采用基于像素点的背景模型的优点是可以利用像素点之间的密集分布特性,通过连通区域分析方法获得物体所占区域信息.因此,如果能将基于特征

点的方法和基于像素点的背景建模方法结合起来,就可以提高检测效率和前景物体分割的准确性.如图 2 所示,一方面,通过特征点分布的计算得到运动前景物体的特征点.在场景像素点分类时,根据运动前景特征点将场景中的像素点分为无前景物体经过像素点、有前景物体经过像素点以及原背景迅速变化

区域像素点;然后分别计算它们的更新速率并进行模型更新.另一方面,通过像素点模型得到前景物体对应的像素点,可以将前景物体的特征点与前景物体进行对应.

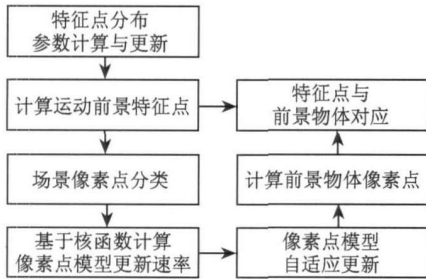


图2 基于特征点的自适应更新背景模型

我们将图像区域分为动态纹理区域和静态区域,在这2个区域都计算其特征点分布,并且在静态区域还采用像素点的高斯模型.由于静态区域可以使用单高斯模型对其建模,避免了混合高斯模型所需的大量计算,并且在一般情况下能较为真实地反映背景信息.特征点和像素点信息的结合,使得双方能互相利用其信息,起到互补效果;而这种效果是单独采用其中一方所无法达到的.若只采用高斯模型,则无法很好地对动态背景区域进行建模;而单独计算特征点信息,则无法获得前景特征点之间的关联信息,在判断特征点与真实物体的对应关系时存在较大困难.

本文利用特征点运动信息对高斯模型进行自适应更新.在静态区域中,特征点的运动情况反映出物体的运动情况,而根据运动信息,能够区分区域在图像序列当中所在的不同状态,如是处于被前景物体暂时遮挡状态还是原来的背景物体刚刚离开状态等.如果是前者,可以降低该区域的更新速率,以避免背景高斯模型匹配上前景物体;如果是后者,则通过提高该区域的更新速率,使真实背景能得到迅速更新.在进行特征点检测时,我们采用 Harris 特征点检测器,其特征描述子在文献[11]中的 SIFT-like 特征点描述子基础上增加了移动方向数据项.

## 2 背景特征点计算与更新

首先对视频中的图像进行区域划分,以确定动态纹理区域和静态区域;然后根据不同区域在帧序列中的历史信息,采用不同的方法进行建模,计算模型参数,并随着视频中场景的变化进行相应的参数更新.

### 2.1 图像区域划分

假设图像序列  $I$  中第  $t$  帧的图像表示为  $I^t$ , 其中位于  $(x, y)$  处的像素点表示为  $I^t(x, y)$ . 图像中分为静态背景区域  $R_s$  和动态纹理背景区域  $R_D$ , 并且满足以下条件

$$R_s \cup R_D = I^t, \quad R_s \cap R_D = \emptyset$$

其中,  $I^t(x_0, y_0)$  的区域属性用  $R^t(x_0, y_0)$  表示, 即当  $I^t(x_0, y_0) \in R_s$  时,  $R^t(x_0, y_0) = 0$ ; 而  $I^t(x_0, y_0) \in R_D$  时,  $R^t(x_0, y_0) = 1$ . 假设图像序列  $I$  中动态纹理背景区域和静态背景区域是固定的, 即对于某个像素点  $(x_0, y_0)$ , 其在任一帧中的背景区域属性都是相同的(在一般情况下, 该假设符合实际情况). 因此可以省略上标  $t$  以  $R(x_0, y_0)$  表示, 那么对于位于  $(x_0, y_0)$  处的像素点, 满足条件  $R(x_0, y_0) = \{0, 1\}$ .

$R_s$  区域和  $R_D$  区域的划分采用对连续  $p$  帧进行图像差分的方法. 在差分图像中变化次数较少且变化幅度较小的像素点属于  $R_s$  区域, 而变化频繁且变化幅度较大的像素点属于  $R_D$  区域. 对于每个像素点, 计算出每次图像差分时的值  $X_t (1 \leq t \leq p-1)$ , 形成向量  $X = \{X_1, \dots, X_{p-1}\}$ ; 然后找出其中  $X_t$  小于变化阈值  $\varepsilon$  的个数  $n_s$ . 如果  $n_s > p \times \alpha_s$ , 则判定该像素点处于  $R_s$  区域; 反之, 则判定该像素处于  $R_D$  区域, 其中  $\alpha_s$  为变动比例阈值.

### 2.2 特征点表示

在所有图像区域中均需计算特征点信息. 由于特征点的稀疏性, 因此虽然是在整个图像区域中计算特征点, 但在进行逐帧匹配时实际匹配次数少, 大大降低了计算复杂度.

背景特征点矩阵 (background feature points matrix, BFM) 用于对背景特征点进行表示. 由于特征点的稀疏特性, 该矩阵是一个稀疏矩阵, 即矩阵中的大多数元素都为 0. 而对于每个特征点所在位置的元素, 均采用 1 个 9 元向量  $\{\text{descriptor}, \text{frequency}, \text{correlation}, \text{var}, s_0, s_1, s_2, s_3, s_4\}$  表示, 其中,

1) 特征点的平均描述子 descriptor. 对于在同一位置反复出现的特征点, 记录它们的描述子的平均值.

2) 特征点出现的频率 frequency. 用于记录在一组连续帧图像中该特征点出现的次数. 在动态纹理区域, 由于特征点出现的位置会随时间变化, 2 个连续帧不同位置的特征点可能对应同一个特征点, 因此只要二者能匹配, 位置的不同不影响增加该特征点的 frequency.

3) 最近连续  $n$  帧图像序列中对应的特征点的相关性  $correlation$  的均值  $\overline{correlation}$  和方差  $var$ . 2 个特征点  $P_1$  和  $P_2$  之间的  $correlation$  按

$$correlation = \frac{\sum_i^N \min(U_i, V_i)}{\sum_i^N \max(U_i, V_i)} \text{ 计算. } U \text{ 和 } V \text{ 分别是}$$

$P_1$  和  $P_2$  的描述子,  $N$  表示描述子的维数.

$$\overline{correlation} = \frac{1}{n} \sum_{i=1}^n correlation_i \quad (1)$$

$$var = \frac{1}{n} \sum_{i=1}^n (correlation_i - \overline{correlation})^2 \quad (2)$$

4) 特征点在 4 个方向的位移为  $s_1, s_2, s_3$  和  $s_4$ , 特征点位置保持不动的次数为  $s_0$ . 4 个方向如图 3 所示.

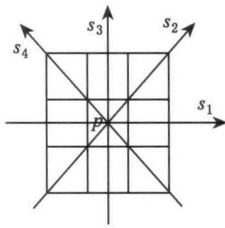


图 3 特征点移动方向示意图

其中,  $s_1$  表示特征点  $p$  沿水平移动,  $s_2$  表示  $p$  沿右上方向或左下方向移动,  $s_3$  表示  $p$  沿垂直方向移动,  $s_4$  表示  $p$  沿左上方向或右下方向移动, 箭头方向为移动正方向. 若  $p$  向正方向移动  $n$  个像素, 则相应方向位移值增加  $n$ , 若向负方向移动, 则相应方向位移减少  $n$ . 若  $p$  的位置保持不变, 则  $s_0 + 1$ . 这些位移数据用于判断特征点的运动方向连续性, 以便对前景特征和背景特征进行分类(这在后面的特征点分类算法中会用到).

### 2.3 背景特征点参数计算和更新

如果在训练模型时出现移动的前景物体, 那么不仅相同的背景特征点之间有强关联性, 而且相同的前景物体特征点之间也有强关联性, 这就会导致将前景特征点加入到背景模型中. 虽然随着前景物体移动出图像画面和模型的更新, 该特征点会从背景模型中删除, 但为了能在前景物体依然存在于场景中时就得到比较真实的背景模型, 背景特征点初始算法增加了根据特征点移动方向作为判断依据.

由输入训练图像得到第一阶段的特征点模型的过程如图 4 所示. 该模型用特征点矩阵 (feature

points matrix, FM) 表示, FM 的表示方法与背景特征点 BFM 表示方法相同.

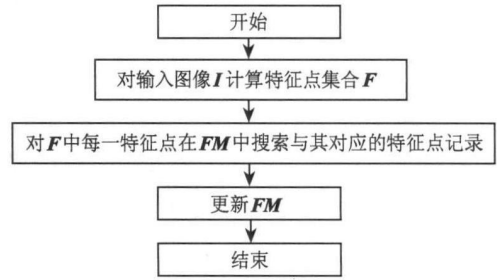


图 4 FM 计算流程图

#### 算法 1. FM 计算

输入. 连续  $p$  帧图像  $I^1, \dots, I^p$ .

输出. 稳定特征点矩阵 FM.

Step1. 对第 1 帧图像提取特征点, 得到特征点集合  $F^1$ , 即为初始 FM. 对于后续的每帧图像, 从 Step2 开始执行.

Step2. 对第  $t$  帧图像提取特征点, 得到特征点集合  $F^t$ .

Step3. 对于  $F^t$  中的每一元素  $I^t(x_0, y_0)$ , 在 FM 中, 采用一个  $5 \times 5$  大小的窗口 (该窗口以  $(x_0, y_0)$  为中心) 进行特征点匹配. 即将  $I^t(x_0, y_0)$  与该窗口内的  $FM(x, y)$  中的平均特征描述子进行匹配. 匹配原则是计算最大的相关性  $correlation$  值, 记为  $correlation_{max}$ , 并将该值对应的特征点作为最终匹配结果. 如果能找到匹配点  $(x_1, y_1)$ , 执行下一步; 否则, 转 Step5.

Step4. 如果  $correlation_{max} > \overline{correlation} - 3 \times var$  ( $\overline{correlation}$  和  $var$  由存储在矩阵 FM 中位于  $(x_1, y_1)$  处的元素值取得), 则判别为二者为真实对应, 转 Step6; 否则, 二者非真实对应, 并不匹配, 执行下一步.

Step5. 在  $(x_0, y_0)$  处生成一个新特征, 得到其对应的 9 元向量 (其中 frequency 初始值设为 2, 其余值设为 0), 并更新 FM. 转 Step7.

Step6. 更新  $FM(x_1, y_1)$ . 其中  $frequency + 2$ , 根据式 (1), (2) 重新计算  $\overline{correlation}$  和  $var$ ; 并根据特征点的移动方向更新相应的  $s_i (0 \leq i \leq 4)$ .

Step7. 所有 FM 元素的  $frequency - 1$ . 如果 FM 中有元素 frequency 为 0, 则将其从中删除.

处理完所有  $p$  帧之后, 对 FM 进行分析, 判断其中的元素为前景特征点记录还是背景特征点记录, 得到 BFM. 由于背景特征点包括静态区域和动态纹理区域 2 部分, 静态区域特征点的特点是在图像中的位置基本固定; 而动态纹理区域的特征点会在某个区域来回运动. 相对而言, 前景特征点的运动方向更为稳定, 因此可从特征点的运动区分背景特征点和前景特征点.

图 5 所示为由 *FM* 计算 *BFM* 的过程. 对于 *FM* 中每一非零元素, 判断其  $s_0$  的值, 如果  $s_0 > p \times q$ , 则该特征点为静态背景特征点, *BFM* 中对应位置添加该元素, 判断结束; 否则, 在  $s_1, s_2, s_3$  和  $s_4$  中选取数值最大者  $s_{max}$  和最小者  $s_{min}$ , 取大于  $s_{max}$  和  $s_{min}$  的最大正整数  $s'_{max}$  和  $s'_{min}$ , 如果  $s'_{max}/s'_{min} > p \times \gamma$ , 则该特征点为前景特征点; 否则, 该特征点为动态纹理背景特征点, 在 *BFM* 中对应位置添加该元素. 其中,  $p$  为图像帧数,  $\sigma$  为特征点静止比例阈值,  $\gamma$  为持续运动特征点比例阈值.

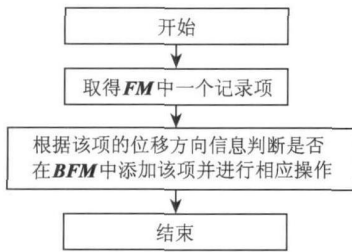


图 5 由 *FM* 计算 *BFM* 流程图

背景特征点的更新方法同算法 1, 只是输入由 *FM* 变成 *BFM*, 得到的输出即为更新后的 *BFM*. 在更新过程中若未能找到对应的背景特征点和非真实对应的特征点, 即判断为前景特征点. 对于前景特征点所在区域, 利用高斯模型得到密集像素点分布, 通过连通区域分析方法得到前景物体对应的连通区域, 然后将特征点与前景物体进行对应.

### 3 背景像素点参数计算和更新

像素点参数的计算和更新只对  $R_s$  区域进行, 因此利用  $R_s$  区域的静态特点, 对每个像素点采用一个高斯模型来表示.

#### 3.1 背景像素点参数计算

在单高斯模型中, 第  $t$  帧图像中位于  $(x_0, y_0)$  的像素点, 其像素值为  $X$  的概率为  $P(X) = \eta(X, \mu, \Sigma)$ . 其中,  $\mu$  和  $\Sigma$  分别表示第  $t$  帧时该像素点的高斯分布的均值和协方差,  $\eta$  是高斯概率密度函数,

$$\eta(X, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{p}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X-\mu)^T \Sigma^{-1}(X-\mu)}$$

模型中参数  $\mu$  和  $\Sigma$  由最近连续  $p$  帧图像中位于相应位置的像素点的颜色值  $\{X_1, \dots, X_p\}$  计算得出,

$$\mu = \frac{1}{p} \sum_{i=1}^p X_i, \quad \Sigma = \frac{1}{p} \sum_{i=1}^p (X_i - \mu)^T (X_i - \mu)$$

#### 3.2 参数自适应更新

为了使背景模型实时适应场景变化, 如光线的变化、背景物体的运动等, 模型需随着当前帧图像的数据而更新; 并且, 像素点高斯模型的更新同样只对  $R_s$  区域内的像素点进行. 为了使像素点高斯模型能进行自适应更新, 运动特征点信息被用于动态调节像素点的更新速率. 根据当前图像进行像素点参数更新的过程如图 6 所示.



图 6 根据当前图像进行模型更新流程图

##### 3.2.1 更新速率计算

对于第  $t$  帧,  $R_s$  区域的像素点可以分为 3 类:

1) 无前景物体经过区域. 它们按初始更新速率  $\alpha_0$  进行更新.

2) 运动前景物体所在区域. 由于该区域被前景物体遮挡, 不应该进行背景模型更新, 所以更新速率为 0.

3) 原先静止的背景物体离开后形成的背景区域和原先运动的前景物体静止后成为背景的区域. 由于新背景的产生, 需要及时更新该背景区域, 因此该区域更新速率为  $\alpha_{max}$ .

由于并不知道前景物体所在区域, 也就无法确切地对上述区域进行分割. 但可以根据特征点信息计算上述区域像素点的更新速率. 第 1 类区域像素点的特征是在第  $t-1$  帧和第  $t$  帧时, 自身及邻近区域都不存在运动前景特征点的像素点. 第 2 类区域像素点即在第  $t$  帧时存在运动特征点的像素点及邻近区域的像素点. 第 3 类区域像素点的特征是在  $t-1$  帧时特征点静止, 在  $t$  帧时变为运动, 其  $t-1$  帧时该特征点邻近区域; 以及在  $t-1$  帧时为运动, 在  $t$  帧时为静止的特征点, 其  $t$  帧时该特征点的邻近区域. 图 7 所示为某帧原始图像以及采用特征点模型计算出来的该帧图像的特征点信息, 其中, 绿色十字交叉点表示背景特征点, 红色十字交叉点表示前景特征点.

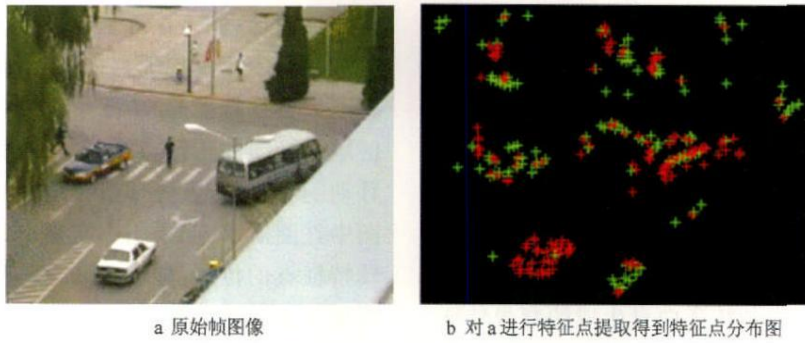


图7 背景特征点和前景特征点分布

由于无法知道物体轮廓信息,就无法确切地知道运动特征点周围像素点是否属于运动前景物体.本文采用核函数  $f(x)$  计算第 2, 3 类区域中特征点邻近像素点的更新速率  $\alpha$ .核函数的定义如下: $X$  代表一个  $d$  维的欧氏空间,  $\mathbf{x}$  是该空间中的一个点,用一列向量表示.  $\mathbf{x}$  的模  $\|\mathbf{x}\|^2 = \mathbf{x}^T \mathbf{x}$ .如果一个函数  $K: X \rightarrow \mathbb{R}$  存在一个函数  $k: [0, \infty] \rightarrow \mathbb{R}$ , 即  $K(\mathbf{x}) = k(\|\mathbf{x}\|^2)$ , 并且满足:

- 1)  $k$  是非负函数;
- 2)  $k$  是非增的, 即如果  $a < b$ , 那么  $k(a) \geq k(b)$ ;
- 3)  $k$  是分段连续的, 并且  $\int_0^{\infty} k(r) dr < \infty$ ;

那么函数  $K(\mathbf{x})$  就被称为核函数.

采用截尾核函数

$$f(\mathbf{x}) = \begin{cases} e^{-\beta \|\mathbf{x}\|^2}, & \text{if } \|\mathbf{x}\| < \lambda \\ 0, & \text{else} \end{cases}$$

计算特征点周围像素点的更新速率. 其中,  $\lambda$  和  $\beta$  是函数参数,  $\lambda$  表示核函数的作用范围,  $\beta$  决定空间距离对函数值变化的影响大小.

对于位于  $(x_0, y_0)$  的第 2 类特征点, 其周围像素点  $(x', y')$  的更新速率

$$\alpha(x', y') = 1 - e^{-\beta[(x' - x_0)^2 + (y' - y_0)^2]} \quad (3)$$

对于位于  $(x_0, y_0)$  的第 3 类特征点, 其周围像素点  $(x', y')$  的更新速率

$$\alpha(x', y') = \alpha_{\max} e^{-\beta[(x' - x_0)^2 + (y' - y_0)^2]} \quad (4)$$

其中  $\alpha_{\max}$  为最大更新速率.

当像素点  $(x', y')$  周围有多个特征点时, 其更新速率取分别计算时的均值, 即

$$\alpha(x', y') = \frac{1}{n} \sum_{i=1}^n \alpha_i(x', y') \quad (5)$$

其中  $\alpha_i(x', y')$  表示依据第  $i$  个特征点计算得到的更新速率.

对于不在运动特征点周围的像素点, 按照初始更新速率  $\alpha_0$  进行更新.

### 3.2.2 参数更新

一般而言, 在实际场景中, 静态场景变化速度缓慢, 所以对于没有运动前景物体经过的区域采取每隔  $\theta$  帧进行更新的策略.  $\theta$  的取值采用类似滑动窗口协议中的取值策略, 当像素点第  $t$  帧的颜色值  $X_t$  与第  $t-1$  帧的高斯模型均值  $\mu_{t-1}$  的差值小于阈值  $\varepsilon$  时, 按速率  $\rho_t$  ( $\rho_t > 1$ ) 增大  $\theta$ , 直到等于最大限制值  $\theta_{\max}$ . 当  $X_t$  与  $\mu_{t-1}$  的差值大于阈值  $\varepsilon$  时, 则按速率  $\rho_t$  ( $\rho_t < 1$ ) 减小  $\theta$ , 直到  $\theta = 1$ .

单个像素点的更新步骤如下:

Step1. 对于第  $t$  帧图像  $I^t$  中每一像素点  $I(x, y)$ , 如果  $R(x, y)$  为 1, 则转至下一像素点; 否则, 执行下一步.

Step2. 如果  $I(x, y)$  属于第 1 类区域像素点, 执行下一步; 否则, 转 Step4.

Step3. 当时  $\text{mod}(t, \theta) \neq 0$  时,  $\mu_t = \mu_{t-1}$ ,  $\sigma_t = \sigma_{t-1}$ , 转至下一像素点; 否则, 执行下一步.

Step4. 由式(3)~(5)计算更新速率  $\alpha$ . 根据  $\alpha$  更新高斯模型参数  $\mu_t = (1 - \alpha)\mu_{t-1} + \alpha X_t$ ,  $\sigma_t^2 = (1 - \alpha)\sigma_{t-1}^2 + \alpha(X_t - \mu_t)^T(X_t - \mu_t)$ . 如果该像素点属于第 2, 3 类区域, 则结束处理; 否则, 执行下一步.

Step5. 如果  $\|X_t - \mu_{t-1}\| > \varepsilon$ , 则  $\theta = \max(1, \text{floor}(\theta \times \rho_t))$ ; 否则,  $\theta = \min(\theta_{\max}, \text{floor}(\theta \times \rho_t))$ . 其中  $\text{floor}(x)$  表示取不大于  $x$  的整数.

## 4 实验与结果分析

我们将本文提出的背景模型分别与混合高斯背景模型、稀疏特征点背景模型进行了实验比较. 实验硬件环境为 P IV 2.4 GHz CPU, 768 MB 内存的 PC 机; 软件环境为 Matlab 7.0; 处理的图像序列的图像尺寸为  $320 \times 240$  像素, 彩色图像. 为了验证本文背景模型自适应调节更新速率的能力, 我们对比了混合高斯模型在低更新速率和高更新速率情况下的结果.

在实验中, 设训练图像帧数  $p = 50$ , 变化阈值  $\epsilon_s = 20$ ,  $\alpha_s = 0.75$ , 静止比例阈值  $\sigma = 0.6$ ,  $\gamma = 0.2$ ;  $\lambda = 5$ ,  $\beta = 0.01$ ,  $\alpha_{max} = 0.2$ . 在进行参数更新时, 初始值  $\theta = 1$ ,  $\epsilon$  取值为 3 倍的对应像素点颜色值方差,  $\theta_{max} = 64$ , 表示在帧速为 30 帧/s 的视频中, 最大更新间隔约为 2 s.  $\theta$  的增大速率  $\rho_u = 2$ , 减小速率  $\rho_l = 0.5$ .

我们比较了采用文献[11]中的特征向量和本文的增加特征点运动方向的特征向量生成的背景特征点矩阵的结果, 如图 8 所示. 第 1~50 帧图像为训练

图像, 用于计算背景特征点模型. 图 8a 所示为第 50 帧训练图像, 图 8b 所示为采用文献[11]中的特征向量得到的背景特征点模型的显示结果, 绿色十字交点为背景特征点. 从图中可以看出, 虽然动态纹理区域(树叶)的特征点被正确计算为背景特征点, 但移动的车辆、行人等被错误地计算为背景特征点, 如图中红圈所示. 而采用本文提出的特征向量, 通过比较特征点的位移, 能够区分出移动物体的特征点, 得到更为正确的结果, 如图 8c 所示.

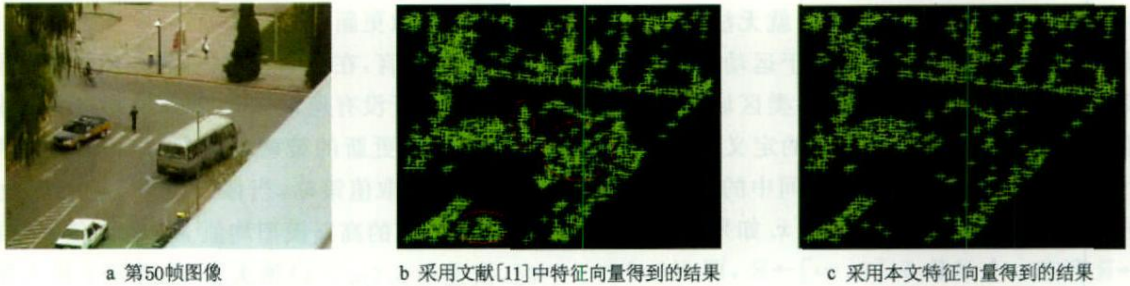


图 8 采用文献[11]中特征向量和本文特征向量得到的背景特征点结果比较

图 9 所示为本文方法与混合高斯模型的结果对比. 图 9a 所示为原始输入帧序列中的第 70, 148,

250, 286, 342 帧, 红框所框住的物体表示我们关注的运动中的物体(运动中的小车). 第 70 帧时, 小车

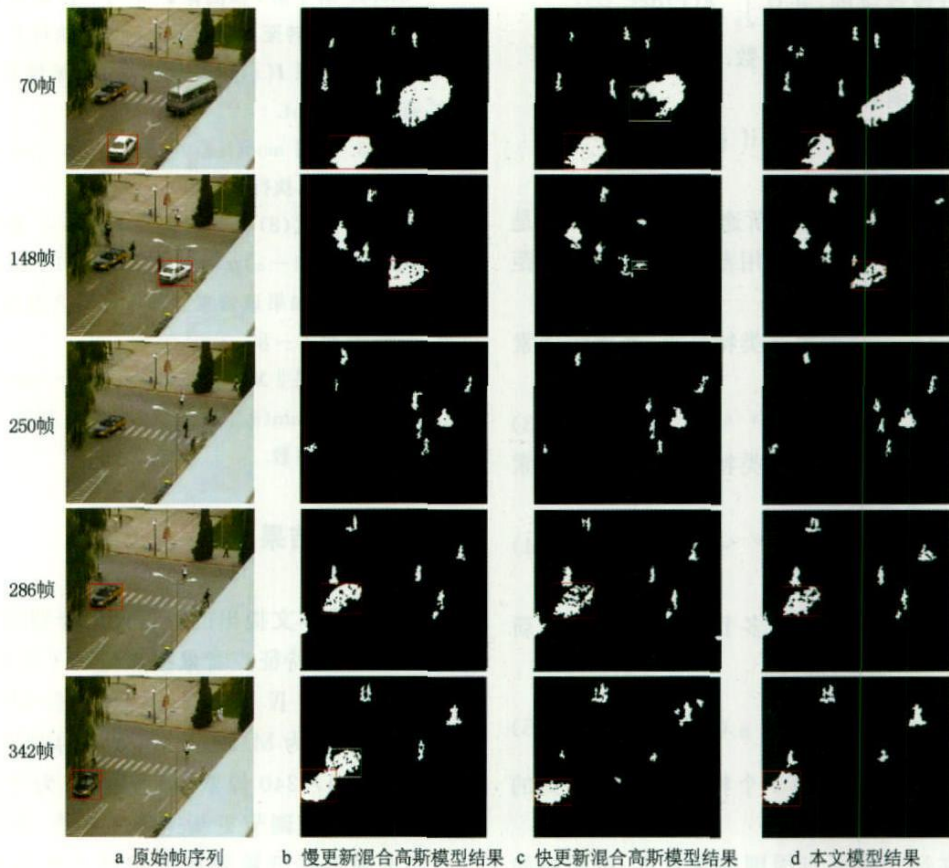


图 9 混合高斯模型不同更新速率下检测结果和本文背景模型检测结果比较

驶入场景;第 148 帧时,小车继续行驶;第 250 帧时,出租车尚未启动,作为背景;第 286 帧时,出租车启动;第 342 帧时,出租车即将驶出场景.图 9 b 所示为采用慢更新(更新速率  $\alpha=0.01$ )的混合高斯模型检测结果,红框为关注物体的检测结果.图 9 c 所示为采用快更新(更新速率  $\alpha=0.1$ )的混合高斯模型检测结果.图 9 d 所示为采用本文基于特征点自适应背景模型的检测结果.

图 9 b 第 342 帧时,原来出租车的位置依然检测到一个出租车物体,显示该区域模型未得到及时更新.实际上,在我们的实验中,该区域直到 450 帧以后也未完全正确更新.

图 9 c 第 70 帧时,白框部位显示由于背景更新速率太快,使得背景模型拟合了汽车的表面颜色,使得汽车很大一部分区域无法被正确检测;在第 148 帧时,由于背景在汽车经过时过快更新,产生了不正确的模型参数,因此使得小汽车基本上已经无法检测出来.但是由于其快速更新的特点,在第 342 帧时,原来出租车的区域已经得到及时更新.

图 9 d 第 70 帧时,小车进入场景即被正确检

测;在第 148 帧时,正确检测出小车;在第 250 帧时,出租车作为背景;第 286 帧时,出租车启动,被正确检测出;第 342 帧时,出租车被正确检测,而且由于原来出租车停放区域的特征点移动离开,于是该区域背景模型的更新速率提高,使背景得到了及时更新.实验表明:该模型能自适应调节不同区域的更新速率.

我们还对本文模型与稀疏特征点背景模型进行了比较,实验结果如图 10 所示.由于稀疏特征点模型中前景物体检测结果只能以特征点的形式表现出来,所以图中的结果只显示出特征点形式.图 10 a 所示为原始帧序列,可以看到从第 50~100 帧,红框中的 2 位行人始终保持同一速度沿同一方向行进;图 10 b 所示为稀疏特征点模型检测的结果,在这种情况下,我们难以判断出这些特征点是属于同一个物体还是不同的物体,由于其运动一致性,往往会将其判别为同一物体.图 10 c 所示为本文模型的结果,可以看出,由于采用的是基于像素的方法,很容易通过连通区域分析的方法,将行人区分开来,使特征点能够正确地对对应到各自物体.

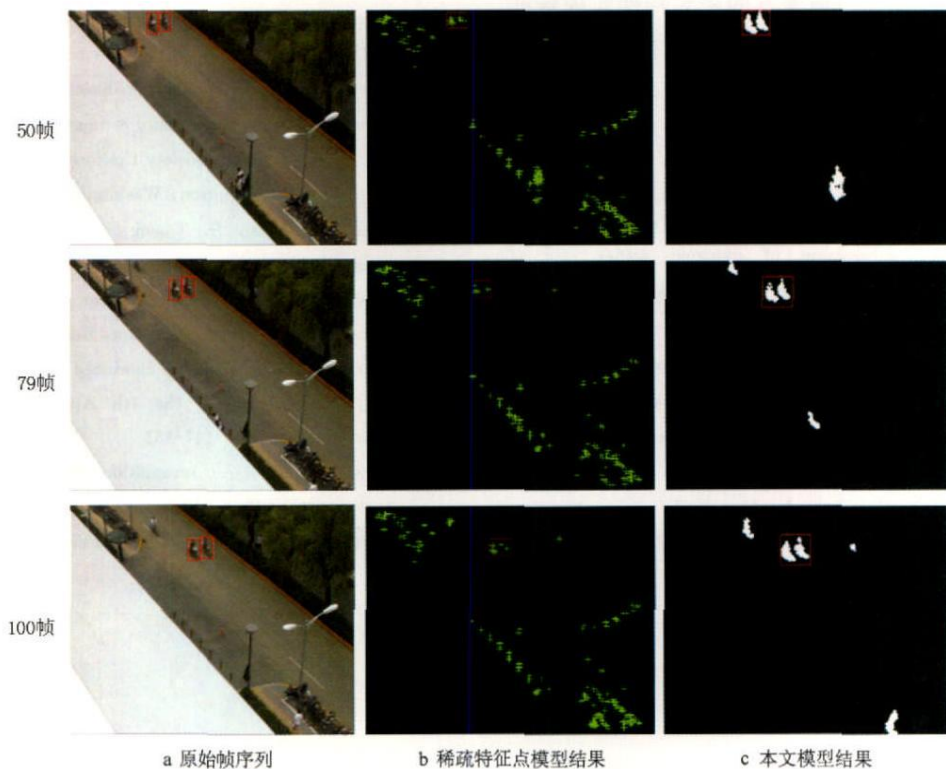


图 10 稀疏特征点背景模型与本文背景模型检测结果比较



## 5 总 结

本文提出了基于特征点的自适应更新背景模型,该模型对于图像中全部区域采用特征点表示,并且对静态区域采用基于像素点的单高斯模型表示.这种结合使得在处理动态纹理区域时,利用了特征点信息能较好表现动态纹理的特点;同时,在静态区域又利用基于像素点模型的密集特性,避免了特征点的稀疏性导致的有时难以区分前景物体的问题.我们采用单高斯模型并利用静态区域的性质,避免了混合高斯模型的复杂计算.另外,利用特征点反映出的物体运动信息,采用基于核函数的方法计算像素点更新速率,实现了高斯模型的自适应调节更新速率,使其能及时、准确地更新背景模型.

通过与不同更新速率下的混合高斯模型实验表明,本文模型能够在原背景物体离开时迅速更新原来背景区域,同时又能使前景物体频繁运动的区域确保背景模型参数的正确性,避免了混合高斯模型无法有效地调节更新速率以适应场景物体变化的情况.与稀疏特征点背景模型比较的实验结果表明,本文模型能够有效地利用像素点的密集特性实现特征点与前景物体的对应.

## 参 考 文 献

- [1] Monnet A, Mittal A, Paragios N, *et al.* Background modeling and subtraction of dynamic scenes [C] // Proceedings of the 9th IEEE International Conference on Computer Vision, Nice, 2003: 1305-1313
- [2] Zhong J, Sclaroff S. Segmenting foreground objects from a dynamic textured background via a robust Kalman filter [C] // Proceedings of the 9th IEEE International Conference on Computer Vision, Nice, 2003: 44-50
- [3] Wren C R, Azarbayejani A, Darrell T, *et al.* Pfister: real time tracking of the human body [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(7): 780-785
- [4] Stauffer C, Grimson W E L. Adaptive background mixture models for real-time tracking [C] // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, 1999: 246-252
- [5] Haritaoglu I, Harwood D, Davis L S. W<sub>4</sub>: real-time surveillance of people and their activities [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8): 809-830
- [6] Hou Zhiqiang, Han Chongzhao. A background reconstruction algorithm based on pixel intensity classification [J]. Journal of Software, 2005, 16(9): 1568-1576 (in Chinese)  
(侯志强, 韩崇昭. 基于像素灰度归类的背景重构算法[J]. 软件学报, 2005, 16(9): 1568-1576)
- [7] Rittscher J, Kato J, Joga S, *et al.* A probabilistic background model for tracking [C] // Proceedings of the European Conference on Computer Vision, Dublin, 2000: 336-350
- [8] Stenger B, Ramesh V, Paragios N, *et al.* Topology free hidden Markov models: application to background modeling [C] // Proceedings of IEEE International Conference on Computer Vision, Vancouver, 2001: 294-301
- [9] Elgammal A, Harwood D, Davis L. Nonr parametric model for background subtraction [C] // Proceedings of the 6th European Conference on Computer Vision, Dublin, 2000: 751-767
- [10] Mittal A, Paragios N. Motion based background subtraction using adaptive kernel density estimation [C] // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington D C, 2004: 302-309
- [11] Zhu Q, Avidan S, Cheng K T. Learning a sparse, corner based representation for time-varying background modelling [C] // Proceedings of the 10th IEEE International Conference on Computer Vision, Beijing, 2005: 678-685
- [12] Harris C, Stephens M. A combined corner and edge detector [C] // Proceedings of the 4th Alvey Vision Conference, Manchester, 1988: 147-151
- [13] Lowe D G. Object recognition from local scale-invariant features [C] // Proceedings of the International Conference on Computer Vision, Corfu, 1999: 1150-1157