

序列图像中彩色目标跟踪的加权颜色分布方法

姚红革 郝重阳 雷松则 齐华 齐敏

(西北工业大学电子信息学院, 西安 710072)

摘要 提出一种基于加权颜色概率分布模型的彩色目标跟踪方法。对目标的颜色直方图进行改造形成目标的颜色概率分布图,同时考虑到了颜色模型的大小及其中像素点的位置对颜色分布的影响,再对颜色分布图进行加权处理成为加权颜色分布图,以此作为目标的颜色模型。与加权直方图相比,加权颜色分布图对区域特征描述更加合理。实验结果表明,本文方法与典型的 mean-shift 算法相比,提高了目标的定位精度的同时,降低了计算复杂度,而且在快速运动和被大比例遮挡的情况下对彩色目标的实时跟踪有良好的鲁棒性。

关键词 彩色目标实时跟踪 加权颜色概率分布 mean-shift 卡尔曼滤波

中图法分类号: TP391.4 文献标识码: A 文章编号: 1006-8961(2009)01-0099-07

Visual Target Real-time Tracking Based on the Weighted Color Probability

YAO Hong-ge, HAO Chong-yang, LEI Song-ze, QI Hua, QI Min

(School of Electronic and Information Northwestern Polytechnical University, Xi'an 710072)

Abstract This paper presents a real-time color image tracking method. In order to implement an effective and robust tracking task, a novel approach of weighted color target model based on color histogram is discussed. It takes into account the target's shape and position of pixels as necessary factors in target model. This makes the description of target's feature more reasonable. Furthermore, the starting position of target is found with Kalman filter in every frame, and then mean shift is utilized to track the target position. Experiment result shows, in Contrast with mean shift, the proposed method can locate object more accurately and have lower complication of calculation. The proposed method is more robust, especially for some complex tracking conditions such as occultation and object which is moving fast.

Keywords colored object real-time tracking, weighted color probability contribution, mean shift, Kalman filter

1 引言

运动目标的检测与跟踪是机器视觉应用领域的研究热点。其中,复杂情况或环境下的图像处理环节一直是个难点问题。在实际应用中,复杂跟踪(目标有遮挡或运行速度快)的适应能力以及跟踪的实时性,对方法都提出了更高的要求。

已有的目标跟踪方法分为两类:

(1)基于运动的方法 这类方法的特点是把一

段时间内的具有运动一致性的点归为一类,如光流法和特征点法,这类方法的优点是适用面广,像光流法,任何估计算法都可以获得光流^[1]。缺陷是计算量较大。

(2)基于模型的方法 根据利用目标信息部分的不同又可分为:基于目标边界和基于目标区域两种方法。它们都是根据目标本身的信息,如颜色和纹理等^[2]来排除背景的干扰进行跟踪。由于目标本身的信息较多,如不加简化将不可避免地带来信息匹配时的大量运算。因此对于实时性要求很高的

基金项目:国家博士点基金项目(20040699015)

收稿日期:2007-02-20;改回日期:2007-07-02

第一作者简介:姚红革(1968~),男,讲师。西北工业大学电子信息学院电子与信息工程研究所信号与信息处理专业博士研究生。主要研究方向为机器视觉和计算机仿真。E-mail: yaohongge@xatu.edu.cn

运动目标跟踪而言,如何选取目标的特征信息,并在可靠的前提下简化运算是目标跟踪的关键^[3]。

颜色直方图^[4]作为组成目标像素概率分布的一种表达方式,可认为是一种较为实用的区域颜色建模方法。该方法对于目标被部分遮挡、旋转以及形变等情况都有较强的适应性,且易于实现^[5]。

基于颜色直方图建立目标模板的颜色概率分布图并将其与 mean-shift 和 Kalman 滤波相结合,提出了一种基于模型的方法,称为基于加权颜色概率分布模型的彩色物体跟踪方法。首先分别建立目标模板与候选目标的颜色分布图作为目标的颜色分布模型,同时引入巴特查里亚距离,比较两者概率分布图的相似程度作为对目标物体定位的依据。考虑到目标区域的大小及其像素点所在位置的不同对颜色分布的影响,对颜色概率分布图进行加权处理,使得两者的匹配更加合理有效。同时,利用 Kalman 预测预估下一帧目标所在区域,再在该区域采用 mean_shift 迭代得到确切位置,这样用局部计算代替对图像的全局计算,将大大降低计算量,提高跟踪的实时性。

2 以颜色概率分布为特征的目标搜索

2.1 目标的颜色概率分布

在复杂跟踪情况下,实时地连续长距离跟踪目标是一件很困难的事情。跟踪某些非刚体目标比如人就更加困难。因为在跟踪过程中目标图像会随着人的姿势的改变而变化,因而用一般的方式简单地以模板匹配的方法进行跟踪往往会丢失目标。目标图像颜色直方图不受目标形状变化的影响。所以,用颜色直方图作为目标的概率分布模式,依据颜色分布进行匹配,具有较好的稳定性。本文利用颜色直方图的这一优点对目标图进行改造,形成对应的颜色概率分布图,以此作为目标概率分布模型在匹配有效性上达到了较好的效果。目标颜色概率分布图的生成步骤如下:

(1) 建立 H 分量直方图

首先将目标本图的每个像素从 RGB 空间转换到 HSV 空间,然后统计其 H 分量,建立 H 分量直方图。

(2) 计算每个像素点为目标概率

设 H 分量直方图为 $H(\mathbf{u})$, \mathbf{u} 为 H 分量的某个区域,则 $H(\mathbf{u})$ 为落在该区域的像素个数。若将 $H(\mathbf{u})$ 的最大值表示为 $Max_Val[H(\mathbf{u})]$,则落在 H

分量区间为 u 上的像素 μ 是目标的概率为

$$p(\mu) = H(\mu) / Max_Val[H(\mathbf{u})]$$

这样就建立了 H 分量值 u 与目标像素 μ 之间的概率对应关系。

(3) 颜色概率的灰度表达

通过步骤 2 所建立的对应关系,可以很方便得到目标像素的概率值。若将此概率值乘以 255,就可以将目标像素的概率表达以灰度图的形式反映出来。本文将此称为目标像素的颜色概率分布图。

以上形成的分布图虽只反映了目标像素的 H 分量。也可加入 S, V 分量组成一个 1 维矢量

$$L = HQ_S Q_V + SQ_V + V$$

式中, Q_S 和 Q_V 分别是 S 和 V 的量化级数,一般取 $Q_S = Q_V = 4$ 。对于求出的 L ,它所反映的目标颜色特征较之单纯 H 分量更全面,同样可通过以上 3 步求出基于 L 的颜色概率分布图。

同理以上方法也可用在其他颜色空间中,比如 YUV、RGB、CIE 的 $L^* a^* b^*$ 、Munsell 空间等等。出于运算量、实时性方面的考虑,采用基于 HSV 的 H 分量的分布图。

2.2 对分布图的加权改造及模板更新

目标颜色概率分布可表示为

$$\hat{q}_u = C_q \sum_{i=1}^n \delta[b(x_i) - u] \quad (1)$$

式中, n 表示目标区域内的像素数,函数 $b: \mathbf{R}^2 \rightarrow \{1, \dots, m\}$ 是位于 x_i 处像素颜色索引的索引函数, δ 是狄拉克函数, C_q 为归一化系数, u 为目标图像的特征值 $u = 1, \dots, m$ 。由于目标的外围像素可能被遮挡或者受到背景的影响,是相对不可靠的。所以应对目标内不同位置的像素赋予不同的权重,即给那些位置与目标中心的距离越近的像素赋予较大的权值,而给那些距离目标区域中心较远的像素赋予较小权值。

基于此,对颜色概率分布图进行加权改造。对于一个长半轴为 w ,短半轴为 h 的特定目标区域,其权值函数可设为

$$k(r) = \begin{cases} 1 - r^2 & r < 1 \\ 0 & r > 1 \end{cases} \quad (2)$$

式中, r 为某点到区域中心的距离。那么基于此权值函数的中心点在 y 处的目标颜色分布变为

$$\hat{q}_u = C_q \sum_{i=1}^n k\left(\left\|\frac{y - x_i}{d}\right\|^2\right) \delta[b(x_i) - u] \quad (3)$$

令归一化系数 $C_q = \left[\sum_{i=1}^n k \left(\left\| \frac{y - x_i}{d} \right\|^2 \right) \right]^{-1}$ 使得 $\sum_{n=1}^m \hat{q}_u = 1$; 参数 $d = (w, h)$ 在此描述目标区域的大小。

\hat{q}_u 即可作为目标模板,用于与当前帧图像中的可能目标或称候选目标进行匹配。同样在第 k 帧图像中的候选目标也可用加权概率分布的形式表示,记为 $\hat{p}_{u,k}(y)$ 的形式, y 为候选目标中心点。

为适应目标变形、旋转、翻滚以及颜色变化等状态变化,可对模板 \hat{q}_u 在跟踪过程中进行及时更新,第 k 帧模板 $\hat{q}_{u,k}$ 可由第 $k-1$ 帧的目标模板 $\hat{q}_{u,k-1}$ 与候选目标分布 $\hat{p}_{u,k-1}(y)$ 混合计算得到:

$$\hat{q}_{u,k} = (1 - \alpha) \hat{q}_{u,k-1} + \alpha \hat{p}_{u,k-1}(y) \quad (4)$$

α 为两者的混合因子,取值可在 $0.5 \sim 0.8$ 之间。

2.3 用 mean-shift 迭代进行空间定位

这里采用 Bhattacharyya 系数作为衡量候选目标分布 p 和目标模板分布 q 之间相似性的判据。BH 系数越大则表明两帧图像越相似。BH 系数为^[6]

$$\hat{\rho}(y) \equiv \rho[\hat{p}(y), \hat{q}] = \sum_{u=1}^m \sqrt{\hat{p}_u(y) \hat{q}_u} \quad (5)$$

当前帧目标位置估计为 \hat{y}_0 , 则该处候选目标 \hat{y}_0 的颜色分布为 $\hat{p}_u(\hat{y}_0)$, 若对式(5)在 $\hat{p}_u(\hat{y}_0)$ 处运用 Taylor 扩展,可以近似为

$$\rho[\hat{p}(y), \hat{q}] \approx \frac{1}{2} \sum_{u=1}^m \sqrt{\hat{p}_u(\hat{y}_0) \hat{q}_u} + \frac{1}{2} \sum_{u=1}^m \hat{p}_u(y) \sqrt{\hat{q}_u / \hat{p}_u(\hat{y}_0)} \quad (6)$$

将式(3)代入式(6)有

$$\rho[\hat{p}(y), \hat{q}] \approx \frac{1}{2} \sum_{u=1}^m \sqrt{\hat{p}_u(\hat{y}_0) \hat{q}_u} + \frac{C_{q,k}}{2} \sum_{u=1}^m w_i k \left(\left\| \frac{y_{c,k} - x_{i,k}}{d} \right\|^2 \right) \quad (7)$$

式中,

$$w_i = \sum_{i=1}^n \sqrt{\hat{q}_u / \hat{p}_u(\hat{y}_0)} \delta[b(x_{i,k}) - u] \quad (8)$$

这样要使 BH 系数最大就要使式(7)第2项最大,因为第1项与 y 无关。若以 $k-1$ 帧的目标中心位置 $y_{c,k-1}$ 为中心,计算出 $p_{u,k}(y_{c,k-1})$, 视其为 $\hat{p}_u(\hat{y}_0)$ 。然后使用 mean-shift 算法寻找邻域内的极大值^[7], 则计算出的第 k 帧的目标中心位置为^[8]

$$y_{c,k} = \frac{\sum_{i=1}^{n_k} x_{i,k} \cdot w_i}{\sum_{i=1}^{n_k} w_i} \quad (9)$$

对于所得到的 $y_{c,k}$, 计算 $\|y_{c,k} - y_{c,k-1}\|$, 如果其值大于 ε (ε 为一个小的正常数), 则令 $y_{c,k-1} = y_{c,k}$, 重新计算 $\hat{p}_u(\hat{y}_0)$ 和 w_i , 从而得到新的 $y_{c,k}$ 。循环迭代, 直至 $\|y_{c,k} - y_{c,k-1}\| < \varepsilon$, 此时 $y_{c,k}$ 就是第 k 帧的目标中心位置。应注意较低的 ε 虽使搜索结果的精度提高但耗时, 所以为了保证跟踪过程的实时性, ε 不能选得太小。实验中取 ε 小于等于一个像素距离, 此时迭代次数平均不超过 7 次, 实时性和精度都较好。

3 用 Kalman 滤波器估计搜索起始点

利用所建立的目标模型, 可将运动目标的跟踪问题转化为在后一帧的图像中寻找模型的最优匹配的问题。由于模型匹配是一个计算量相对较大的任务, 若对整幅图像全局搜索, 将无法满足实时要求。因此, 对目标的运动进行必要的估计可以缩小模型的搜索区域。

Kalman 滤波器是一个对动态系统的状态序列进行线性最小方差估计的算法, 具有计算量小、可实时计算的特点。用 Kalman 滤波器估计新一帧中目标定位的起始点, 以缩小模型的搜索区域。

在图像序列中, 假设跟踪目标 t 时刻的坐标为 (x_t, y_t) , 采用 Kalman 滤波进行运动预测, 可得到 $t+1$ 时刻目标的运动状态, 包括位置信息和速度信息以及协方差 $\mathcal{R}(k+1)$ 。在跟踪过程中, 可将相邻两帧中目标的运动看作是匀速的, 目标的运动特征可用位置和速度来描述, 表示为向量的形式为 $X_t = (x_t, y_t, x_{vt}, y_{vt})^T$, x_t, y_t, x_{vt}, y_{vt} 分别表示目标在 x 轴和 y 轴方向上的位置和速度。系统的状态模型可表示为

$$x_{t+1} = \Phi x_t + w_t \quad (10)$$

式中, w_t 为系统噪声。 Φ 为状态转移矩阵, 可以设定为

$$\Phi = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Δt 为两帧图像之间的时间间隔。

系统的观测模型为 t 时刻目标的位置, 它反映理想跟踪结果与实际跟踪结果之间的关系, 因此系统的观测模型形式可定为

$$\mathbf{z}_i = \mathbf{H}\mathbf{x}_i + \mathbf{v}_i \quad (11)$$

式中, \mathbf{v}_i 为零均值白噪声。由于在图像上只能观测到目标的位置即 \mathbf{z}_i 仅与位置有关, 因此可设观测矩阵为

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

如果给出目标的初始位置和速度, 根据系统的状态方程和观测方程就可以估计出下一帧中目标的状态向量 \mathbf{X}_i 即可估计出目标在下一帧的位置和速度。其位置可作为在下一帧中进行 mean-shift 迭代的起始点, 速度信息可作为目标被遮挡时与本帧位置结合对下一帧目标位置的估计。

4 本文方法的完整描述

进行目标跟踪时首先要初始帧中检测和定位出目标的位置, 并构造出目标模板, 然后利用图像运行信息进行 Kalman 预测, 预估目标在下一帧的位置。以此位置为起始点确定一个子区域, 用 mean-shift 迭代在该区域内对目标进行进一步搜索定位, 直到取得与目标模板颜色分布最相似分布后停止, 所得位置即为被跟踪目标的位置。由于进行了运动估算, 从而大大减少了搜索范围和 mean-shift 迭代次数, 所以与单纯的 mean-shift 算法相比本文方法明显减少了运算量, 提高了跟踪的实时性和有效性。本文基于加权颜色概率分布跟踪方法具体实现步骤如下:

(1) 计算目标模板加权概率分布 \hat{q}_u , 并在初始帧中(如前几帧中)检测和定位出目标位置 \hat{y}_0 。

(2) 提取当前帧中在 \hat{y}_0 处运动目标加权概率分布 $\{\hat{p}_u(y_0)\}_{u=1, \dots, m}$ 计算

$$\rho[\hat{p}(\hat{y}), \hat{q}] = \sum_{u=1}^m \sqrt{\hat{p}_u(\hat{y})\hat{q}_u}$$

(3) 设置系统噪声方差

$$\mathbf{Q} = \begin{bmatrix} \sigma_{gx}^2 & 0 & 0 & 0 \\ 0 & \sigma_{gy}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \text{ 和量测噪声方差 } \mathbf{R} =$$

$$\begin{bmatrix} \sigma_{rx}^2 & 0 \\ 0 & \sigma_{ry}^2 \end{bmatrix}, \text{ 并计算预测协方差 } P(n) = E\{[x(n) -$$

$\hat{x}(n)][x(n) - \hat{x}(n)]^T\}$, $P(n)$ 可由前 n 个初始帧来计算, 实验中取 $n=3$ 。进入 Kalman 预测阶段。

(4) 已知第 k 帧目标中心点坐标值, 计算 $\mathbf{x}_{k+1} = \Phi\mathbf{x}_k + \mathbf{w}_k$ 得 $k+1$ 帧目标中心点的预测坐标值。

(5) 在该预测坐标值处计算 $\{w_i\}_{i=1, \dots, m}$ 参见 2.3 节中式(7), 然后计算目标中心位置 $\hat{y}_1 = \frac{\sum_{i=1}^{n_k} x_{i,k} \cdot w_i}{\sum_{i=1}^{n_k} w_i}$, 更新 $\{\hat{p}_u(y_0)\}_{u=1, \dots, m}$, 计算 $\rho[\hat{p}(\hat{y}), \hat{q}] = \sum_{u=1}^m \sqrt{\hat{p}_u(\hat{y})\hat{q}_u}$ 。

(6) 若 $\rho[\hat{p}(y_1), \hat{q}] \leq \rho[\hat{p}(y_0), \hat{q}]$, 则 $y_1 \leftarrow \frac{1}{2}(\hat{y}_0 + \hat{y}_1)$, 并计算 $\rho[\hat{p}(y_1), \hat{q}]$ 。

(7) 判定如果 $\|\hat{y}_1 - \hat{y}_0\| < \varepsilon$ 停止, 即认为找到了最优匹配点 \hat{y}_1 , 否则 $\hat{y}_0 < -\hat{y}_1$ 。重复步骤 5、步骤 6, 更新 \hat{y}_1 位置。

(8) 以此最优匹配点作为坐标修正值代入预测方程组中求取预测均方误差 $\mathcal{R}(k+1)$, 预测修正系数 $G(k+1)$, 求下一帧预测目标状态: $\hat{\mathbf{x}}_{k+2} = \Phi\mathbf{x}_{k+1} + G(k+1)[\hat{\mathbf{Y}}_1 - \mathbf{H}\hat{\mathbf{x}}_{k+1}]$ 。返回步骤 5 继续跟踪后续帧。

重复进行上述过程, 可实现对目标的连续跟踪。

5 跟踪实例与分析

图像的采样速率为 30 帧/秒, 把 RGB 图像转换为 HSV 图像, 取 H 值直方图, 量化级数为 $m=64$, 机器的配置为 PIII 866MHz, 256M 内存的 PC 机, WindowsXP 系统下 Visual C++ 编程实现。

第 1 个实例, 目标被完全遮挡如图 1 序列所示, 是一个单人穿过迎面而来的两个并排行走人的一段视频场景。整个场景较暗且目标的被跟踪部分(头和上肢)颜色与背景也很接近, 从而使跟踪的难度增加。图 2 为该序列目标模板及其 H 分量直方图和概率分布图。图 1(a)为 mean-shift 算法的跟踪结果, 被跟踪目标由左向右运动, 图第 26 帧时目标被迎面而来的两人遮挡, 图第 32、39 帧时目标已经丢失, 表明 mean-shift 算法跟踪失败。之所以失败是因为目标被遮挡时当前帧的搜索窗口中融入了很多新背景信息, 使得窗中像素概率分布与目标模型分布的相似值(BH 系数)很小, 则匹配失败。对此类目标被严重遮挡的情况, 本文方法是先用 BH 值对遮挡程度进行标量, 若 BH 系数小于 0.4 则可认



图 1 mean-shift 算法与本文方法对有遮挡情况下的跟踪检测结果
 Fig. 1 Tracking results of mean-shift and proposed method in occlusion

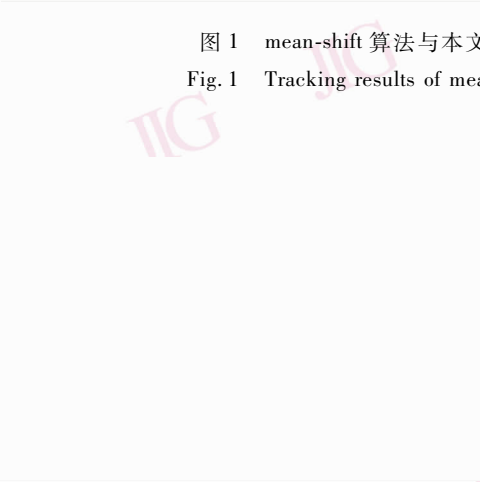


图 2 图 1 序列目标模板及其直方图和概率分布图
 Fig. 2 Target templet, its histogram and probability distribution drawing

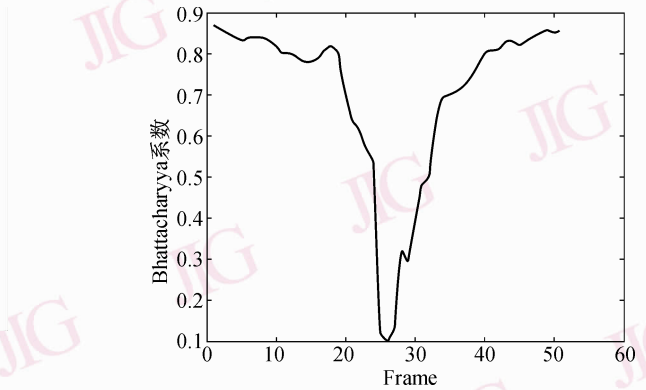


图 3 图 1 跟踪过程中 Bhattacharyya 系数变化
 Fig. 3 Change of Bhattacharyya coefficient in tracking of Fig. 1

为目标被严重遮挡,此时停止 mean-shift 运算,而直接用 Kalman 预测预估下一帧目标的位置直到目标重新出现,即 BH 系数重新大于 0.4,再重新启用 mean-shift 算法来进行精确定位。图 1(b)所示为本文方法对同一场景进行跟踪的效果,可以明显的看到穿过遮挡后的目标依然能被很好地跟踪。

图 3 为用本文算法跟踪过程中 Bhattacharyya 系数变化情况,在第 26 帧前后 BH 系数跌落于 0.4 以下,此时已停止 mean-shift 运算,而直接用 Kalman 预测的位置速度估计值来对目标进行跟踪。

第 2 个例子是对快速移动目标的跟踪。如图 4 序列所示。图 5 为该序列目标模板及其 H 分量直方图和概率分布图。图 4(a)为 mean-shift 算法的跟踪结果。其中第 20、24、28 帧为起飞的初始阶段,速度较慢,帧间目标位移小,mean-shift 能较好跟踪。在即将离地的第 167、174、181 帧阶段,速度加快,帧间目标位移大,可以看到目标跑飞,mean-shift 跟踪失败。图 4(b)是本文方法对第 167、174、181 帧的跟踪结果,可以看出,跟踪效果较好。mean-shift 算

图 4 本文方法与 mean-shift 算法对快速运动目标的跟踪检测结果

Fig. 4 Tracking results of mean-shift and proposed method in target's speediness

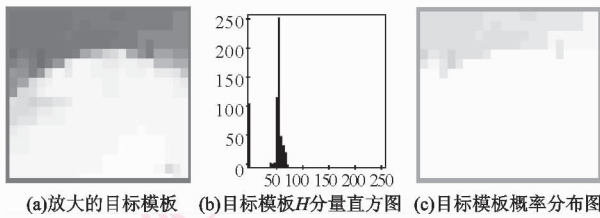


图 5 图 4 序列机头处目标模板及其直方图和概率分布图

Fig. 5 Target templet, its histogram and probability distribution

法之所以失败,是因为该算法在推导过程中(参见式(6))引入了近似计算,要求满足目标位移在连续两帧间不能发生剧烈变化,也就是说前后两帧的目标窗口要有重叠,如果目标运动过快,使得前后两帧窗口没有重叠,mean-shift 算法将失效,会出现丢失目标的情况。而本文方法结合 Kalman 预测来预估下一帧目标的位置,在所预估的位置中再进行 mean-shift 搜索定位,避免了对窗口重叠的依赖,消除了此类情况的可能。尽管增加了 Kalman 滤波运算,但减少了 mean-shift 在跟踪窗内的迭代次数,且 Kalman 预估本身的运算量很小,所以本方法总体每帧的运算时间减少,提高了跟踪的实时性。两种方

法运算量对比如图 6 和图 7 所示。

如图 6 所示,从 51 帧的统计中可看出总体迭代次数,mean-shift 算法平均为 8.73 次/帧,本算法为 6.85 次/帧,平均降低 27%。平均运行时间如图 7 所示,mean-shift 算法为 0.035 2 s 而本算法为 0.028 4 s,平均降低 23%。

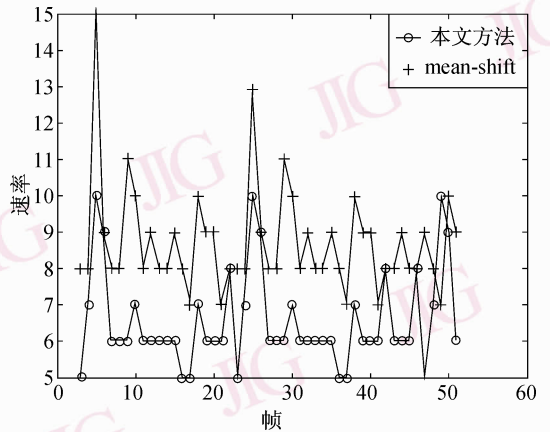


图 6 本文方法与 mean-shift 算法迭代次数对比

Fig. 6 Proposed method in contrast to mean-shift in terative times

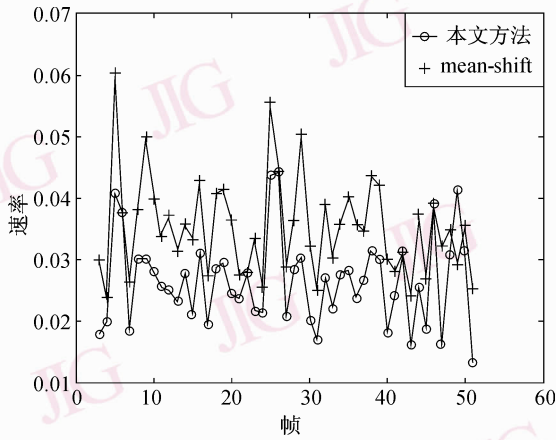


图7 本文方法与 mean-shift 算法每帧耗时对比

Fig.7 Proposed method in contrast to mean-shift in time per frame

6 结论

本文提出的基于加权颜色概率模型的彩色运动目标跟踪方法,在对噪声环境下的彩色物体跟踪时,定位目标区域更快更精确,避免了 mean-shift 算法不能适应目标复杂运动的缺点,很好地解决了快速运动目标的跟踪问题和目标被严重遮挡时的跟踪问题,并提高了跟踪的实时性。本文方法已经成功地应用于机场实时跟踪监控系统中。下一步研究的重

点主要集中在对伴随有大机动性的运动目标和多目标的跟踪方面。

参考文献 (References)

- 1 David Beymer, Philip McLauch. A real-time computer vision system for measuring traffic parameters[J]. IEEE Proceedings on Computer Vision and Pattern Recognition, 1997, 2(5): 238-242.
- 2 Jang Dae-sik, Kim Gye-young. Model-based tracking of moving object [J]. Pattern Recognition, 1997, 30(11):999-1008.
- 3 Hu Ming-hao, Ren Ming-wu, Yang Jing-yu. A real-time moving object tracking algorithm based on histogram model[J]. Computer Engineering and Applications, 2004, 3(8):71-74
- 4 John R Smith, Chang Shih-Fu. Tools and Techniques for Color Image Retrieval [R] . TR2670, Columbia, NY, US: Columbia University, 1996.
- 5 Nummiaro K, Koller M E, Van G L. Object tracking with an adaptive color-based particle filter [A] . In: Proceedings of the Symposium for Pattern Recognition of the DAGM [C], Zurich, Switzerland, 2002, 2449: 353-360.
- 6 Kailath T. The divergence Bhattacharyya distance measures in signal selection [J] . IEEE Transactions on Communication Technology, 1999, 15(2):253-259.
- 7 Comaniciu D, Meer P. Mean shift: a robust application toward feature space analysis [J] . IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(5):603-619.
- 8 Comaniciu D, Meer P. Kernel-based object tracking [J] . IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(5): 564 - 577.