

一种具有跟踪外观变化目标能力的均值漂移算法^{*}

左军毅 赵春晖 梁彦 潘泉 张洪才

(西北工业大学自动化学院 西安 710072)

摘要 视角变化往往会引起目标外观特征的变化,基于单一颜色直方图模型的均值漂移跟踪算法往往不能适应这种变化。本文在对连续自适应均值漂移算法深入分析的基础上,提出利用目标外观信息的先验知识,对其建立多个颜色分布模型。每帧跟踪结束后,算法都会根据当前的目标特征和周围环境从多个模型的凸组合中选出最有利于下一帧跟踪的参考模型。实验结果表明,该算法能很好地适应目标外观的变化,且计算代价不大。

关键词 目标跟踪,连续自适应均值漂移,多参考模型

A Mean Shift Tracker with the Ability to Track Object with Changing Appearance

ZUO Jun-Yi ZHAO Chun-Hui LIANG Yan PAN Quan ZHANG Hong-Cai

(College of Automation, Northwestern Polytechnical University, Xi'an 710072)

Abstract Changes in the viewpoint often cause appearance change of object. In this case, mean shift tracker, which is based on single color histogram model, often lost the target. In this paper, we research the CamShift tracker in detail and then proposed to build multiple color-distribution-model for the target according to prior knowledge of the object's appearance. After tracking in each frame, a model selecting step is carried out to find the best model from the convex group of the reference models for tracking in next frame according to the current object appearances and background appearance in its neighborhood. Experimental results show our tracker can deal with appearance change of object in an elegant manner with low computational cost.

Keywords Target tracking, CamShift, Multiple reference model

1 引言

出于在公共安全监控、人机交互、智能交通、辅助驾驶以及军事等领域应用的需要,针对视频对象的跟踪技术已成为计算机应用领域的一个研究热点。人们针对不同的应用背景和应用环境提出了许多跟踪算法。其中均值漂移算法^[1,2]是匹配搜索类 2D 跟踪算法的一个典型代表。这种算法计算量小,对目标变形、旋转变换、局部遮挡适应性强,自从提出以来受到普遍的重视和广泛的研究。其早期版本——CamShift^[1](Continuously Adaptive Mean Shift)算法通过连续迭代计算跟踪框内概率图的“质心”而获取目标的形心位置。CamShift 被 INTEL 公司作为一种标准算法写进了 OpenCv 的函数库。随后,Comaniciu D^[2]用距离加权直方图作为目标表示模型,通过最小化目标模型和候选目标模型之间的 Bhattacharyya 系数,而得到了一种跟踪算法。该算法数学机理清楚,被作为均值漂移跟踪算法的另一个版本。

以往的均值漂移算法都是基于单一颜色分布模型的,在遭遇由视角变化而引起外观变化的跟踪对象时鲁棒性不好,比如跟踪转动的头部。目前,解决这类跟踪问题的主要途径是采用自适应模型法^[3,4]和子空间投影法^[5~7]。前者利用当前观测来实时地修正目标模型以提高模型的适应能力,然而这种修正缺少必要的监督机制,潜在的模型漂移问题使算法总存在失跟的危险。后者通过对训练图像集应用 PCA 方法

获得子空间表示模型,然后在目标可能出现的区域搜索距离该子空间最近的图像块。和均值漂移算法相比,子空间跟踪算法搜索效率较低,且一般只用于灰度图像序列。

考虑到 CamShift 算法在目标表示上比标准 MeanShift^[2]算法更简单、直观,便于进行模型的组合,本文提出在该算法框架下引入多模型的思想,即使 CamShift 算法同时维持多个来自不同视角的颜色模型,并设计一个目标函数,通过对目标函数的优化来实时地为下一帧跟踪选取最佳的目标模型。和子空间方法一样,多模型表示法充分利用了目标外观的先验知识,不存在模型漂移问题。

2 CamShift 目标跟踪算法

CamShift 跟踪算法的计算流程如图 1。跟踪起始时,先初始化跟踪窗,初始化通常由目标检测模块或用手标定式的方式来完成。得到初始跟踪窗 R_m 后,跟踪过程按以下步骤进行:

1) 计算目标模型的颜色分布直方图

$$q = \{q(u)\}_{u=1, \dots, b}, \sum_{u=1}^b q(u) = 1 \quad (1)$$

其中 b 为直方图箱格的数目。

$$q(u) = \frac{1}{N} \sum_{(x,y) \in R_m} \delta[c(f(x,y)) - u] \quad (2)$$

这里 $f(x,y): R^2 \rightarrow R^3$ 为图像函数; $c(\cdot): R^3 \rightarrow \{1, 2, \dots, b\}$ 为

^{*} 国家自然科学基金重点项目(60634030)、国家自然科学基金(60372085)资助。左军毅 博士研究生,研究方向:图像与视频处理;赵春晖 讲师,博士研究生,研究领域:图像处理,视频分析与处理;梁彦 副教授,博士,研究方向:多目标跟踪,视频分析与处理;潘泉 教授,博士生导师,研究方向:智能信息处理,多目标跟踪理论及应用;张洪才 教授,博士生导师,研究方向:估计理论,信息融合理论及应用,多目标跟踪,智能信息处理。

颜色空间量化函数; $\delta(\cdot)$ 是 Kronecker 函数; N 为 R_m 中的像素个数, 其作用是将直方图 q 归一化。

2) 设置感兴趣区域(ROI)

以跟踪窗形心为中心, 设置一个比跟踪窗尺寸稍大的区域作为感兴趣区域(Region of Interesting), 其大小由目标在图像上的单帧最大移动距离决定。

3) 利用直方图后投影计算概率图

计算概率图只需要在 ROI 内进行, 若 ROI 的图像函数为 $f'(x, y)$, 检查 $f'(x, y)$ 落在直方图 q 的哪个箱格中, 该箱格对应的概率值即为像素 (x, y) 在概率图中的亮度值, 即:

$$I(x, y) = \sum_{u=1}^b q(u) \delta[c(f'(x, y)) - u] \quad (3)$$

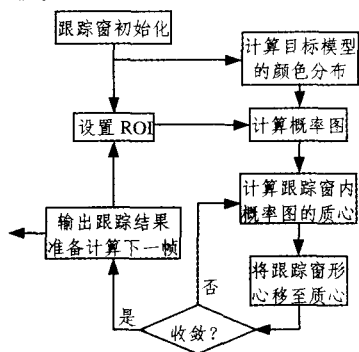


图 1 CamShift 跟踪算法流程

4) 计算跟踪窗内概率图的“质心”

计算零阶矩

$$M_{00} = \sum_x \sum_y I(x, y) \quad (4)$$

计算一阶矩

$$M_{10} = \sum_x \sum_y x I(x, y) \quad (5)$$

$$M_{01} = \sum_x \sum_y y I(x, y) \quad (6)$$

计算跟踪窗“质心”

$$x_c = M_{10} / M_{00} \quad (7)$$

$$y_c = M_{01} / M_{00} \quad (8)$$

若将概率图上的像素看作有质量的点, 亮度越高表示质量越大, 则 (x_c, y_c) 便可看作跟踪窗的“质心”。

5) 判断是否满足收敛条件

设跟踪窗形心为 (\bar{x}, \bar{y}) , 若 $\|x_c - \bar{x}\| < \epsilon$ 且 $\|y_c - \bar{y}\| < \epsilon$, 则收敛, 同时将跟踪窗形心移至“质心”, 即令 $\bar{x} \leftarrow x_c, \bar{y} \leftarrow y_c$, 输出目标形心位置 (\bar{x}, \bar{y}) , 转到第二步处理下一帧; 否则, $\bar{x} \leftarrow x_c, \bar{y} \leftarrow y_c$, 转到第 3 步。 ϵ 为给定阈值。

完整的 CamShift 算法还通过计算概率图的二阶矩来获得目标的尺度和转角, 为了简化问题, 本文将主要精力放在对目标外观变化的适应性上, 在研究时只考虑对目标形心的连续定位。

3 多模型 CamShift 跟踪方案

单模型 CamShift 算法跟踪外观变化目标时鲁棒性不好, 其主要原因是单一颜色模型没有抓住目标外观特征的全部, 因此对外观特征的描述是不完备的。如何提升模型描述能力以克服算法的这一缺陷呢? 一种可能方法是在原算法中引入多模型思想。

我们给出图 2 所示的多模型 CamShift 跟踪方案。其中模型 1, ..., 模型 m 是根据目标外观的先验知识建立的 m 个参考模型。每帧跟踪结束时, 模型优化组合模块都会针对当

前跟踪环境并依据一定的准则从 m 个模型的凸组合中选取最佳的模型组合, 作为下一帧跟踪的参考模型。一个好的目标模型应该和当前目标间有大的相似性, 同时也要考虑和周围背景模式间有足够的差异性^[8], 因此在框图中有四个模块用于提取并计算目标区域和背景区域的颜色分布。当每帧跟踪完毕, 目标形心被确定后, 提取目标区域就变得很简单(假设目标尺度不变); 而背景区域的提取可参考文[8]的方法。方案中 CamShift 跟踪模块以及 ROI 的设置方法和第 2 节中介绍的完全相同, 因此下节重点介绍多模型组合算法。

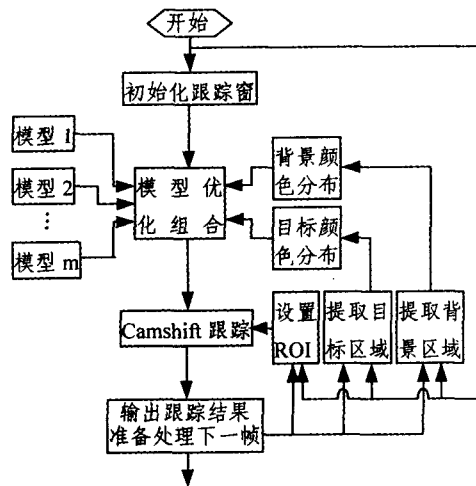


图 2 多模型 CamShift 跟踪算法框图

4 模型优化组合算法

上节的跟踪方案中, 最核心的问题是以什么为原则来选取最优模型。通常的作法是设计一个指标函数, 通过指标函数将模型选取问题转化为数学上的优化问题。设 m 个目标模型分别为 q_1, q_2, \dots, q_m , 其中 $q_i = [q_{i,1}, q_{i,2}, \dots, q_{i,b}]^T, i \in [1, \dots, m]$ 是关于目标的 b 维颜色分布直方图。将这组模型用矩阵 $Q \in R^{b \times m}$ 来表示, 即 $Q = [q_1, q_2, \dots, q_m]$ 。为了保证最优模型本身是一个分布, 假设最优目标模型都来自这 m 个模型的凸组合, 即:

$$q_{opt} = \alpha_1 q_1 + \alpha_2 q_2 + \dots + \alpha_m q_m = Q\alpha \quad (9)$$

其中, 向量 $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_m]^T$ 为最优组合系数, 满足:

$$\alpha_i \geq 0, i = 1, 2, \dots, m \text{ 且 } \sum_{i=1}^m \alpha_i = 1. \quad (10)$$

若用 $p_o \in R^{b \times 1}, p_b \in R^{b \times 1}$ 来表示某时刻目标区域和背景区域的颜色分布, 则 q_{opt} 和 p_o 以及 p_b 之间欧氏距离的平方为:

$$d_o^2 = \|p_o - q_{opt}\|_2^2 \quad (11)$$

$$d_b^2 = \|p_b - q_{opt}\|_2^2 \quad (12)$$

最优目标模型应使 d_o^2 尽可能地小, 同时使 d_b^2 尽可能地大。这是一个典型的多目标规划问题, 用线性加权法将其转化为单目标规划问题, 即设计模型评价指标函数

$$d^2 = \lambda_1 d_o^2 - \lambda_2 d_b^2 \quad (13)$$

λ_1, λ_2 为选定的权系数, 满足:

$$\lambda_1 \geq 0, \lambda_2 \geq 0, \lambda_2 + \lambda_1 = 1 \quad (14)$$

λ_1, λ_2 的大小反映了两个目标函数的重要程度。将(9)(11)(12)代入(13), 并将目标函数看作 α 的函数, 即:

$$d^2(\alpha) = (\lambda_1 - \lambda_2) \alpha^T Q^T Q \alpha - 2(\lambda_1 p_o - \lambda_2 p_b)^T Q \alpha + \lambda_1 p_o^T p_o - \lambda_2 p_b^T p_b \quad (15)$$

略去与 α 无关的项, 剩余部分用 $d^2(\alpha)$ 表示, 问题转化为求

$$\alpha = \underset{\alpha}{\text{mind}}^2(\alpha) \quad (16)$$

其中:

$$d^2_{\alpha}(\alpha) = (\lambda_1 - \lambda_2)\alpha^T Q^T Q \alpha - 2(\lambda_1 p_o - \lambda_2 p_b)^T Q \alpha \quad (17)$$

α 的约束条件由(10)给出。又因为

$$\frac{\partial^2 d^2_{\alpha}(\alpha)}{\partial \alpha \partial \alpha^T} = 2(\lambda_1 - \lambda_2) Q^T Q \quad (18)$$

一般认为,模型和目标的相似程度比模型和背景的差异程度更重一些,即 $\lambda_1 > \lambda_2$ 。若 Q 阵的 m 个列向量线性无关,则 $Q^T Q$ 是 $m \times m$ 正定对称阵,模型优化组合问题实质就是凸二次规划问题。二次规划问题的求解有很多标准算法,比如内点法、外点法、罚函数法、可行方向法等等,可以直接利用。求出 α 后,通过(9)可计算出最优模型。

5 实验结果

我们在 P IV 1.8GHZ 256M 内存的实验平台上编制了

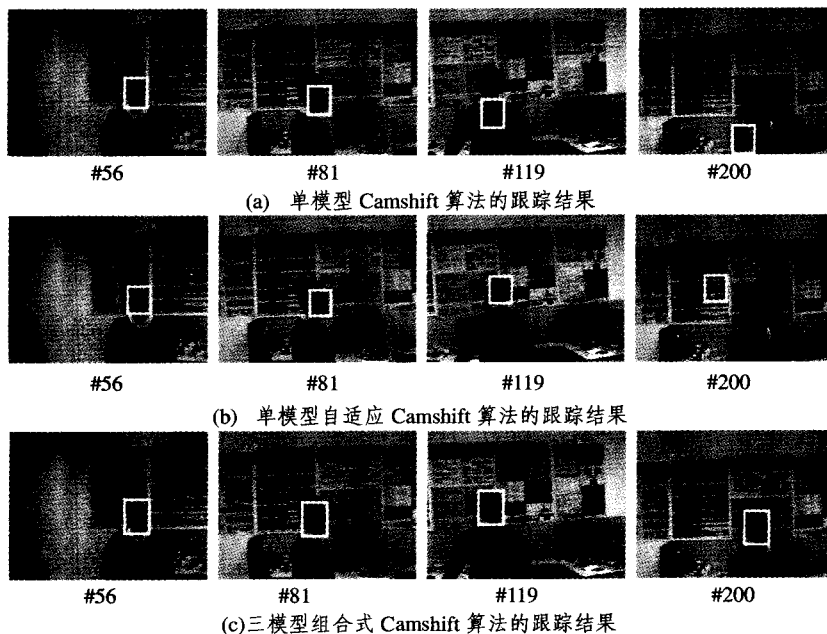


图 4 多模型 CamShift 跟踪算法的实验结果

序列中头部影像在做无规律的平移、旋转运动,表现为目标时而正脸,时而侧脸,时而背脸。图 4(a)是仅用模型 I 时 CamShift 算法的跟踪结果。第 56 帧人处于正脸状态,算法能精确锁定目标;第 119 帧人背脸时,头部颜色模式变为以黑色为主,算法立即失跟,即使第 200 帧人恢复正脸,算法亦不能从错误中恢复。图 4(b)是单模型自适应算法的跟踪结果,其中模型学习速率为 0.05。可以看出,和非自适应模型相比,自适应模型的鲁棒性有所增强,但当跟踪进行到第 180 帧左右时,由于目标运动突然加速,用不精确的当前观测更新模型使模型受到“污染”,并最终导致目标丢失。换用其它学习参数重复实验,结果发现很难找到一个合适的参数能顺利完成跟踪(序列长 1300 帧)。这是因为为了适应目标外观的快速变化就必须选择大的学习参数,而大的学习参数反过来又使模型漂移问题更加严重。图 4(c)是模型组合 CamShift 算法的跟踪结果。人脸转动时,算法能从三个模型的凸组合中选取最优模型用于跟踪,结果非常满意。该序列前 400 帧的最优模型组合系数变化情况如图 5 所示。

实验中我们还测得单模型、三模型组合算法的单帧平均执行时间分别为 6.0、18.2 毫秒。模型组合所带来的附加计

C++ 代码,针对一段头部视频序列执行本文提出的算法,并将结果与单模型跟踪算法进行比较。为了方便起见,求解二次规划问题是通过调用 Matlab 中的库函数实现的。目标及背景分布直方图都是在 RGB 颜色空间中构建的,整个 RGB 颜色空间被分成 $16 \times 16 \times 16$ 个量化级。模型个数为 3,编号为 I、II、III,分别用图 3(a)、(b)、(c)的三个图像块计算得到;(13)式中的 $\lambda_1 = 0.7$ 。

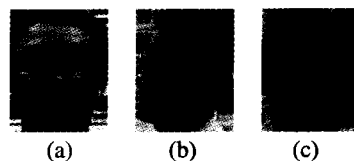


图 3 用于计算目标模型三个图像块

算量较大,其单帧执行时间比单模型算法增大了两倍,即便如此,18.2 毫秒/帧(相当于 55 帧/秒)的处理速度完全能够满足实际系统的需要。实验中我们还充分利用了最优模型组合系数具有连续性这一特点(图 5)。组合系数之所以表现出一定的连续性,是因为视频序列中头部的外观在临帧之间不会有很大的变化。因此,每次求解二次规划问题时都将前一帧最优模型系数作为初值进行迭代,这样可以减少迭代次数,降低计算量。

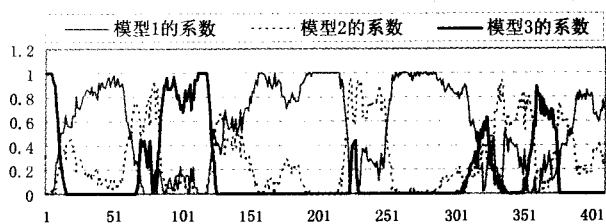


图 5 最优模型的组合系数

结论与展望 本文将多模型的思想与 CamShift 算法相 (下转第 253 页)

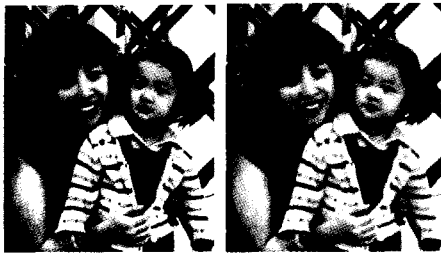


图6 视频帧 suzie 加入水印前后对比图

通过设定 P 的值为 128, 创建一个代表位 0 且含有 64 个码字的子集, 及另外一个代表位 1 且含有 64 个码字的子集。这样, 足够多的码字可以确保嵌入了水印的视频有较高的鲁棒性能, 同时也使得在编解码阶段的系统承担合理的计算量。其中的码字被规范化后的长度 L 为 16, 且设定 0.4 为间距 d_0 的值(图 7)。

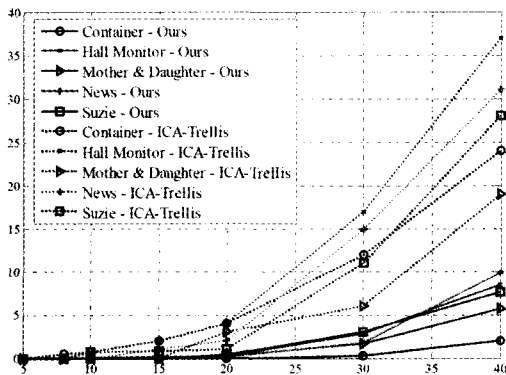


图7 实施 MPEG-II 压缩后鲁棒性比较

与 ICA-Trellis 方案比较所有五个视频数据帧实施不同的压缩率后体现出来的性能, 发现在压缩比为 20:1, 亦即相对应的码率达到 758kbps 的时候, 我们的方法数据位误码率仅有 0.2%, 而 ICA-Trellis 方案的数据位误码率却达到了 2%~4% 的程度, 这意味着它的平均数据位误码率为 3%, 相当于平均下来就有 15 倍之多的数据位错误。而当压缩比率达到 30:1, 亦即相对应的码率达到 505kbps 的时候, 采用我们的方法平均仅会有少于 2% 的嵌入数据信息被丢失, 而相比之下的 ICA-Trellis 方案的数据位误码率却达到了 6%~

(上接第 246 页)

结合, 成功地实现了目标模型根据当前跟踪环境的自动组合, 克服了 CamShift 跟踪算法不能适应目标外观变化的固有缺陷, 证明了用多模型思想解决变外观目标跟踪问题是可行的。然而, 本文仅考虑了目标做平移运动时的情形, 对更复杂运动形式还暂未考虑; 另外, 多模型的引入等于放宽了模型的约束条件, 这在一定程度上减小了模型对目标的识别能力, 如何通过引入目标边缘、形状等信息来提高算法的抗干扰能力? 这些都是今后需进一步研究的内容。

参考文献

- 1 Bradski G, Clara S. Computer vision face tracking for use in a perceptual user interface[J]. International Technology Journal, 1998, 2:1~15
- 2 Comaniciu D, Ramesh V, Meer P. Kernel-based object tracking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(5): 564~577

17% 的程度, 这意味着它的平均数据位误码率为 12.2%, 相当于平均就有 6 倍之多的数据位错误。最后, 当压缩比率达到 40:1, 亦即相对应的码率达到 379kbps 的时候, 采用我们方法而产生的数据位误码率范围在 2%~9.8%, 而相比之下的 ICA-Trellis 方案的数据位误码率却达到了 19%~37% 的程度, 这意味着它的平均数据位误码率为 27.8%, 平均就有 4 倍之多的数据位错误。

结论 本文提出了一个与内容关联的视频水印方案。一个每帧 44 位的二进制数据信息被嵌入视频图像中。经过水印嵌入之后的视频序列仍具备值为 42dB PSNR 的保真度。通过对 MPEG-II 压缩比率改变的测试, 不难发现我们的方法所具备的鲁棒性能优于近期被认为是最优视频水印插入算法的 ICA-Trellis 方案。分别在压缩比率为 20:1, 30:1 及 40:1 的时候, 我们提出的方法与 ICA-Trellis 相比, 体现出对应的 15 倍、6 倍及 4 倍的性能优化, 这对于数字水印技术的发展与完善有着极其重要的应用价值。

参考文献

- 1 Toyota Y, Mizuki T, Sone H. A study on security assessment of digital watermarking[J]. In: Record of Electrical and Communication Engineering, 2006. 547~548
- 2 Porter J, Rajan P. Image adaptive watermarking techniques using models of the human visual system[J]. In: IEEE Proceedings of the 38th Southeastern Symposium on System Theory, 2005. 354~357
- 3 Mayer J, Silva R. Efficient informed embedding of multi-bit watermark[J]. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. 389~392
- 4 Silva R, Mayer J. Informed embedding for multibit watermarking [A]. In: Proceedings 16th Brazilian Symposium on Computer Graphics and Image Processing[C]. Sao Carlos, Brazil: IEEE Comput. Soc, 2003; 214~21
- 5 Gwenaél D, Ingemar J, Miller L. An efficient algorithm for informed embedding of dirty-paper trellis codes for watermarking [J]. In: IEEE International Conference on Image Processing, 2005. 697~700
- 6 Hussein J, Franck D. An ICA Based Algorithm for Video Watermarking[J]. In: IEEE International Conference on Acoustics, Speech and Signal Processing, 2005. 805~808
- 7 Mayer J, Bermudez J. Multi-bit informed embedding watermarking with constant robustness[A]. In: International Conference on Image Processing[C]. Piscataway, NJ, USA, IEEE, 2005. 669~72
- 8 Yamashita T, Sato A, Yamashita K. Digital watermarking using DCT fractal coding without original image information[J]. Electrical Engineering, 2006. 48~55

gence, 2003, 25(5): 564~577

- 3 Stern H, Efron B. Adaptive color space switching for tracking under varying illumination[J]. Image and Vision Computing, 2005, 23(3): 353~364
- 4 Peng N S, Yang J, Liu Z. Mean shift blob tracking with kernel histogram filtering and hypothesis testing[J]. Pattern Recognition Letters, 2005, 26(5): 605~614
- 5 Nguyen H T, Ji Q, Arnold W M. Spatio-temporal context for robust multitarget tracking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(1): 52~64
- 6 Black M J, Jepson A D. Eigen Tracking: Robust matching and tracking of articulated objects using a view-based representation [J]. International Journal of Computer vision, 1998, 26(1): 63~84
- 7 Ho J, Lee K C, Yang M H, et al. Visual tracking using learned linear subspace[C]. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2004
- 8 Collins R, Liu Y X. Online Selection of Discriminative Tracking Features[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(10): 1631~1643