

文章编号: 100120920(2007)012005206

基于自适应表面模型的概率视频跟踪算法

李安平^a, 敬忠良^{a, b}, 胡士强^{a, b}

(上海交通大学 a. 航空航天信息与控制研究所, b. 空天科学技术研究院, 上海 200030)

摘要: 提出一种鲁棒自适应表面模型, 该模型中每个像素值的变化过程由一混合高斯分布描述. 为了适应目标表面的变化, 这些高斯参数在跟踪期间通过在线的 EM 算法自适应更新; 在估计目标状态时, 采用了粒子滤波算法, 设计了基于自适应表面模型的观测模型; 在处理遮挡时, 采用了一种鲁棒估计技术. 多组试验结果表明, 该算法对光照变化、姿态变化、部分或完全遮挡下的跟踪具有较强的鲁棒性.

关键词: 混合高斯模型; 自适应表面模型; 在线 EM 算法; 鲁棒估计技术; 粒子滤波

中图分类号: TP391.4

文献标识码: A

Probabilistic visual tracking based on adaptive appearance model

LI Anping^a, JING Zhongliang^{a, b}, H U Shi qiang^{a, b}

(a. Institute of Aerospace Information and Control, b. Institute of Aerospace Science and Technology, Shanghai Jiaotong University, Shanghai 200030, China. Correspondent: LI Anping, E2mail: lapjt@sjtu.edu.cn)

Abstract: A robust and adaptive appearance model is proposed, in which the value of each pixel over time is modeled by a mixture of Gaussians. To adapt to changes in object appearance, an online expectation maximization (EM) algorithm is developed to update the Gaussian parameters. When estimating the target state, particle filter is adopted, and the observation model is designed based on the adaptive appearance model. Occlusion is handled using a robust estimation technique. Numerous experimental results show that the proposed algorithm can track targets well under illumination changes, large pose variations, and partial or full occlusions.

Key words: Gaussian mixture model; Adaptive appearance model; Online EM algorithm; Robust estimation technique; Particle filter

1 引言

视频跟踪中一个重要的研究课题是如何设计一个鲁棒的表面模型, 用于处理跟踪期间目标表面发生的变化. 目前, 多数跟踪算法所用的表面模型或固定不变, 或被快速更新. 在第 1 种情况下, 表面模型从第 1 帧图像中提取后始终保持不变, 如文献[1, 2]中的颜色表面模型和文献[3]中的灰度表面模型; 在第 2 种情况下, 表面模型的每帧或每隔几帧都要利用前一帧的跟踪结果来更新, 如文献[4, 5]中的灰度表面模型和文献[6]中的颜色表面模型. 这两种方法在很多场合下不适用. 如果表面模型固定不变, 则跟踪器在目标出现光照变化、姿态变化或发生遮挡时容易跟踪失败; 但如果表面模型更新太快, 则跟踪器会由于遮挡或前些帧的累积跟踪误差而利用了错误

的表面信息. 因此, 对于一个鲁棒的跟踪器, 具有合适的更新策略非常重要.

目前, 人们提出了灰度表面模型^[3]、颜色表面模型^[11]、特征基表面模型^[7]以及纹理表面模型^[8]. 尽管人们研究了多种表面模型, 但对自适应表面模型的研究还较少^[7, 8]. 文献[7]提出了一种特征基表面模型, 并利用在线奇异值分解方法(SVD)更新特征基. 试验结果表明, 文献[7]算法对光照变化、目标姿态变化以及大小变化均具有较好的跟踪效果, 但没有讨论遮挡问题. 如果不进行遮挡处理, 则特征基表面会吸收遮挡部分的信息而变得不可靠. 文献[8]提出了一种自适应 WSL 表面模型, 用 3 个混合分量(W, S 和 L 分量)描述目标表面的变化. 该算法在处理光照变化、目标姿态变化以及面部表情变化时鲁棒性

收稿日期: 20051022; 修回日期: 20051219.

基金项目: 国家自然科学基金项目(60375008); 国家科技攻关计划世博科技专项(2004BA908B07); 高校博士点基金项目(20020248029); 航空科学基金项目(02D57003); 航天支撑技术基金项目(20031.3.02).

作者简介: 李安平(1977), 男, 安徽肥县人, 博士生, 从事视频跟踪、处理及数据融合的研究; 敬忠良(1960-), 男, 四川南部人, 教授, 博士生导师, 从事随机控制、目标跟踪、智能信息融合等研究.

较强,但该表面模型是基于/相位0信息的,即利用小波对目标表面灰度信息进行处理,这种处理一般运算量较大^[9].另外,该算法在目标发生完全遮挡时,会出现跟踪失败,其原因是目标发生完全遮挡时,文献[8]的跟踪器类似于基于两帧差的跟踪.

本文提出了一种自适应表面模型,该模型能适应目标表面的缓慢或快速变化.在估计目标状态时,设计了基于自适应表面模型的观测模型,并结合到粒子滤波算法中.在处理遮挡时,采用了鲁棒估计技术.实验结果表明了所提出算法的鲁棒性和有效性.

2 目标模型

2.1 运动模型

目标在图像中的运动可认为是对目标区域参数的映射变化.用 $W_t = (u_t, v_t, s_t, r_t)$ 表示 t 时刻的目标区域,其中 u_t, v_t, s_t, r_t 分别对应目标区域的平移、比例和旋转.这些参数的变化由下述运动方程描述:

$$X_t = \begin{bmatrix} 1 & \$T & 0 & , & 0 \\ 0 & 1 & & & \\ & & 1 & \$T & s \\ s & & 0 & 1 & \\ 0 & , & & 1 & 0 \\ & & & 0 & 1 \end{bmatrix} X_{t-1} + \begin{bmatrix} \$T^2/2 & 0 & , & 0 \\ \$T & & & \\ 0 & \$T^2/2 & s & \\ s & \$T & & \\ & & 1 & 0 \\ 0 & , & 0 & 1 \end{bmatrix} U_t. \quad (1)$$

式中: $X_t = (u_t, \hat{u}_t, v_t, \hat{v}_t, s_t, r_t)^T$; \hat{u}_t 和 \hat{v}_t 为 u_t 和 v_t 的速度; U_t 为零均值的高斯噪声,其协方差为 2 ; 2 为对角矩阵,对角线上的元素对应 W_t 中 4 个参数的方差,即 $\hat{R}_t, \hat{R}_t, \hat{R}_t, \hat{R}_t$.

2.2 自适应观测模型

目标的观测模型是基于自适应表面模型设计的.下面重点讨论自适应表面模型的设计.

2.2.1 混合表面模型

在跟踪期间,目标表面会由于姿态变化、光照影响以及遮挡而发生明显变化.单个高斯分布很难描述目标表面像素值的变化过程,因而本文采用混合高斯分布来描述目标表面像素的变化过程.用 $A_{t-1} = \{P_{m,t-1}(i), L_{m,t-1}(i), R_{m,t-1}(i); i = 1, \dots, d\}_{m=1}^G$ 表示 $t-1$ 时刻的表面模型.其中: i 为 A_{t-1} 中的第 i 个像素; d 为 A_{t-1} 中像素的总个数; $P_{m,t-1}(i), L_{m,t-1}(i)$ 和 $R_{m,t-1}(i)$ 分别为第 m 个高斯分布的权重、均值和标准方差.假设表面模型中每个像素相互独立,则目

标的观测似然函数设计为

$$p(Y_t | X_t) = p(Z_t | X_t) = \prod_{i=1}^d \prod_{m=1}^G P_{m,t-1}(i) N(Z_t(i); L_{m,t-1}(i), R_{m,t-1}(i)). \quad (2)$$

式中: $\prod_{m=1}^G P_{m,t-1}(i) = 1$; Z_t 为对应于状态 X_t 时的观测值; G 为高斯分布的个数,一般情况下 $G = 3 \sim 5$, 本文选取 $G = 3$; $N(\#; \#, \#)$ 为正态密度函数.

2.2.2 模型更新

为了适应目标表面的变化,式(2)中的参数 $\{P_{m,t-1}(i), L_{m,t-1}(i), R_{m,t-1}(i)\}_{m=1}^G$ 应在线更新.下面讨论在已知当前观测值 Z_t 时,如何更新式(2)中的参数.假设过去观测值对当前模型参数估计的贡献程度服从 A^k ($k \in [t, \infty]$) 衰减分布,则可得到以下一组递推公式:

$$O_{m,t}(i) = \frac{P_{m,t-1}(i) N(Z_t(i); L_{m,t-1}(i), R_{m,t-1}(i))}{\sum_{m=1}^G P_{m,t-1}(i) N(Z_t(i); L_{m,t-1}(i), R_{m,t-1}(i))}, \quad (3)$$

$$P_{m,t}(i) = A P_{m,t-1}(i) + (1 - A) O_{m,t}(i); \quad (4)$$

$$L_{m,t}(i) = [1 - B_{m,t}(i)] L_{m,t-1}(i) + B_{m,t}(i) Z_t(i); \quad (5)$$

$$R_{m,t}(i) = [1 - B_{m,t}(i)] R_{m,t-1}(i) + B_{m,t}(i) [Z_t(i) - L_{m,t}(i)]^2. \quad (6)$$

其中 $B_{m,t}(i) = (1 - A) O_{m,t}(i) / P_{m,t}(i)$. 由于表面模型中的像素是相互独立的,上述公式对所有像素均适用.下面给出这些公式的推导.

当过去观测值对当前模型参数估计的贡献程度服从 A^k 衰减分布时,第 i 个像素历史观测值的对数似然函数可表示为

$$L((i) | Z_{1:t}(i)) = \prod_{k=1}^t A^k \log \left[\prod_{m=1}^G P_{m,t}(i) N(Z_k(i); L_{m,t}(i), R_{m,t}(i)) \right], \quad (7)$$

式中 $(i) = \{P_{m,t}(i), L_{m,t}(i), R_{m,t}(i)\}_{m=1}^G$. 本文的目的是求出 (i) 使式(7)的值最大.与标准 EM 算法思想相似^[10],如果给定 (i) 一个初始猜测值,那么 (i) 可通过迭代下面两步求出:

Step1: E2step

$$O_{m,k}(i) = \frac{P_{m,t-1}(i) N(Z_k(i); L_{m,t-1}(i), R_{m,t-1}(i))}{\sum_{m=1}^G P_{m,t-1}(i) N(Z_k(i); L_{m,t-1}(i), R_{m,t-1}(i))}; \quad (8)$$

Step2: M2step

$$P_{m,t}(i) = \frac{\prod_{k=1}^t A^{k \cdot O_{m,k}(i)}}{G \prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}}, \quad (9)$$

$$L_{m,t}(i) = \frac{\prod_{k=1}^t A^{k \cdot O_{m,k}(i)} Z_k(i)}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}}, \quad (10)$$

$$R_{m,t}(i) = \frac{\prod_{k=1}^t A^{k \cdot O_{m,k}(i)} [Z_k(i) - L_{m,t}(i)]^2}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}}. \quad (11)$$

这里, 前一时刻的参数 $(t-1)(i)$ 作为最初的猜测参数. 上述两步在计算 $o_{m,k}(i)$ 时, 需要存储以前所有时刻的观测值, 这在实际中很难做到. 为此, 本文通过一些假设推导出一组近似公式来代替式(8) ~ (11).

考虑到实际中前后时刻的高斯参数变化不会太大, 因而可假设 $o_{m,t}(i) \sim U(o_{m,t-1}(i))$. 在这种假设下, 可得到以下推导:

对于 $P_{m,t}(i)$, 得

$$P_{m,t}(i) = \frac{\prod_{k=1}^t A^{k \cdot O_{m,k}(i)}}{G \prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}} \sim \frac{\prod_{k=1}^{t-1} A^{k \cdot O_{m,k}(i)} + o_{m,t}(i)}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}} = \frac{A \prod_{k=1}^{t-1} A^{k \cdot O_{m,k}(i)} P_{m,t-1}(i) + \frac{1}{G} o_{m,t}(i)}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}} \quad (12)$$

对于 $L_{m,t}(i)$, 得

$$L_{m,t}(i) = \frac{\prod_{k=1}^t A^{k \cdot O_{m,k}(i)} Z_k(i)}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}} \sim \frac{A \prod_{k=1}^{t-1} A^{k \cdot O_{m,k}(i)} L_{m,t-1}(i) + \frac{o_{m,t}(i)}{G} Z_t(i)}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}} = [1 - B_{m,t}(i)] L_{m,t-1}(i) + B_{m,t}(i) Z_t(i). \quad (13)$$

其中

$$B_{m,t}(i) = \frac{o_{m,t}(i)}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}} = \frac{o_{m,t}(i)}{\prod_{k=1}^t \sum_{m=1}^G \prod_{k=1}^t A^{k \cdot O_{m,k}(i)}} U(1 - \frac{o_{m,t}(i)}{P_{m,t}(i)}). \quad (14)$$

对于 $R_{m,t}(i)$, 其推导与 $L_{m,t}(i)$ 相同.

总之, 通过近似推导, 可得到式(3) ~ (6). 式(3) ~ (6) 表明, 在计算 $(t)(i)$ 时只需存储前一时刻的参数值 $(t-1)(i)$. 这就大大降低了计算机的存储容量.

当获得当前观测值 Z_t 时, 可根据式(3) ~ (6) 更新前一时刻的模型参数. 这种更新保证了混合模型能及时反映目标表面的变化. 在更新一开始, 表面模型中像素值的分布先由第 1 个高斯分量加以描述, 其余高斯分量在后续更新过程中逐渐被初始化. 在更新过程中, 一些像素偶尔需要一个复位步骤. 复位的准则是, 当没有高斯分量与当前像素值匹配时进行复位. 匹配的定义是像素值在某一分布的 T_R 个方差以内. 复位的原因主要是: 1) 当前更新过程对于这些像素变得不可靠; 2) 这些像素相对于以前时刻发生较快变化. 在第 1 种情况下, 将权重最小的高斯分量的均值复位为最初模板的像素值, 方差复位为初始方差, 权重复位为较小值, 这种复位使表面模型在更新失败时至少包含最初模板信息; 第 2 种情况下, 均值复位到当前观测的像素值, 使混合模型直接包含快速变化量. 本文设计了一个计算器 C 来计算连续复位次数. 如果计算器的值超过了 T_c , 则认为当前像素发生了快速变化. 复位后, 归一化权重. 本文以第 i 个像素为例来说明更新过程:

Step1: 初始化

在 $t = 0$ 时, 有

$$P_{m,0}(i) = \begin{cases} 1, & m = 1; \\ 0, & m = 2, \dots, G. \end{cases}$$

$$L_{m,0}(i) = \begin{cases} T_0(i), & m = 1; \\ \inf, & m = 2, \dots, G; \\ P_{m=1}, \dots, G, R_{m,0}(i) = R_0^2. \end{cases}$$

设定: $C(i) = 0$.

Step2:

For $t = 1, \dots$, 有:

1) $P_{m=1}, \dots, G$, 有

$$p_m(i) = \begin{cases} N(Z_t(i); L_{m,t-1}(i), R_{m,t-1}(i)), \\ \quad | Z_t(i) - L_{m,t-1}(i) | \leq T_R R_{m,t-1}(i); \\ 0, \text{ Otherwise.} \end{cases}$$

$$2) p(i) = \prod_{m=1}^G P_{m,t-1}(i) p_m(i).$$

3) If $p(i) > 0$, 有

$C(i) = 0$ // 一旦更新条件满足, 计算器清零

For $m = 1, \dots, G$ // 更新步骤

$$O_{m,t}(i) = P_{m,t-1}(i) p_m(i) / p(i)$$

$$P_{m,t}(i) = A P_{m,t-1}(i) + (1 - A) O_{m,t}(i)$$

If $O_{m,t}(i) > 0$

$$B_{n,t}(i) = (1 - A) O_{m,t}(i) / P_{m,t}(i)$$

$$L_{n,t}(i) = [1 - B_{n,t}(i)] L_{n,t-1}(i) + B_{n,t}(i) Z_t(i)$$

$$R_{n,t}^2(i) = [1 - B_{n,t}(i)] R_{n,t-1}^2(i) + B_{n,t}(i) [Z_t(i) - L_{n,t}(i)]^2$$

End If

End For

Else // 复位步骤

$$P_m = 1, \dots, G; P_{m,t}(i) = A P_{m,t-1}(i);$$

$$k = \arg \min_m \{P_{m,t}(i)\}$$

If $C(i) \leq T_c$

$$P_{k,t}(i) = P_0, L_{k,t}(i) = T_0(i),$$

$$R_{k,t}^2(i) = R^2$$

$$C(i) = C(i) + 1 // 计算器加1$$

Else

$$P_{k,t}(i) = P_0, L_{k,t}(i) = Z_t(i), R_{k,t}^2(i) = R^2$$

$$C(i) = 0 // 计算器清零$$

End If

$P_m = 1, \dots, G;$

$$P_{m,t}(i) = P_{m,t}(i) / \prod_{m=1}^G P_{m,t}(i)$$

End If

End For

3 遮挡处理

3.1 鲁棒估计技术

一般情况下, 位于遮挡区域内的像素会产生较大的图像误差, 称这些像素为/ 异常像素0(Outlier pixels). / 异常像素0对状态估计的影响应该减小, 鲁棒估计技术提供了这种功能^[9].

本文采用鲁棒的 Huber 函数 Q 针对文中的问题, 函数 Q 定义为

$$Q(E_n(i)) = \begin{cases} T_R \left| E_n(i) \right| - \frac{T_R}{2}, & P_m = 1, \dots, G, \left| E_n(i) \right| > T_R; \\ \frac{1}{2} [E_n(i)]^2, & \text{Otherwise.} \end{cases} \quad (15)$$

式中: $E_n(i) = [Z_t(i) - L_{n,t-1}(i)] / R_{n,t-1}(i), i = 1,$

, d . 在试验中, 如果没有高斯分量与 $Z_t(i)$ 匹配, 则认为第 i 个像素为/ 异常像素0, 此时式(2)中的正态密度函数由下式替代:

$$N(Z_t(i); L_{n,t-1}(i), R_{n,t-1}^2(i)) = \frac{1}{\sqrt{2PR_{n,t-1}(i)}} \exp \left\{ -T_R \left| \frac{Z_t(i) - L_{n,t-1}(i)}{R_{n,t-1}(i)} \right| - \frac{T_R}{2} \right\}. \quad (16)$$

这种替代使得/ 异常像素0对似然函数计算的影响变小, 从而对状态估计的影响也变小.

3.2 遮挡判断

如果当前观测值 Z_t 中的/ 异常像素0个数 d_{out} 超过某一比例 C , 即 $d_{out}/d > C$, 则认为目标发生遮挡. 试验中选取 $C = 0.2$. 一旦发生遮挡, 则当前的更新过程停止, 即 $A_t = A_{t-1}$. 这种做法防止了表面模型在更新时被遮挡部分的像素所破坏, 保证了表面模型的可靠性.

4 概率跟踪算法

对于视频跟踪, 目标的状态与观测之间经常为非线性, 且状态和观测噪声经常为非高斯. 在这种情况下, 传统的滤波算法很难处理. 粒子滤波是一种基于蒙特卡罗方法的算法, 它能有效解决非线性、非高斯情况下的状态估计问题^[11]. 粒子滤波算法通过一组带权重的粒子来逼近目标状态的后验分布, 当粒子的数目趋于无穷大时, 这些粒子在理论上可以逼近状态的真实后验分布. 本文将自适应表面模型与粒子滤波算法相结合, 实现了一种鲁棒的视频跟踪. 跟踪算法步骤如下:

Step1: 重要性采样. $t-1$ 时刻的粒子 $\{X_t^{(j)}\}_{j=1}^N$ 由运动模型 $p(X_t^{(j)} | X_{t-1}^{(j)})$ 传递到下一时刻, 得到新的粒子 $\{X_t^{(j)}\}_{j=1}^N$.

Step2: 权值计算. 在给定的观测值 Y_t , 每个粒子 $X_t^{(j)}$ 被赋予权值 $w_t^{(j)} = W_p(Y_t | X_t^{(j)})$, 其中 $p(Y_t | X_t^{(j)})$ 由式(2)定义. 通过这一步, 自适应表面模型被结合到了粒子滤波中. 最后归一化这些权值.

Step3: 重采样. 每个粒子 $X_t^{(j)}$ 将被重新选择, 被选择次数与它的权值大小成正比. 重采样结果为一组权值相同的粒子 $\{x_t^{(j)}, 1/N\}_{j=1}^N$, 它代表 t 时刻的状态后验分布.

Step4: 状态估计. 在估计出状态的后验分布后, 采用最小均方根误差估计(MMSE)来估计目标的状态, 即

$$\hat{X}_t = E[X_t | Y_{1:t}] \approx \frac{1}{N} \sum_{j=1}^N X_t^{(j)}. \quad (17)$$

5 试验结果及分析

下面通过实际视频数据对文中算法进行验证^[12]. 这组数据包含光照变化、姿态变化和遮挡等

情况. 目标在第 1 帧的位置通过手工初始化得到. 对于自适应表面模型, 一些参数的设定如下: $A = 0.85$, $R = 20$, $P_0 = 0.1$, $T_R = 3$, $T_c = 5$. 在 PIII 1.2 G 的计算机上用 VC 实现算法, 在粒子数 $N = 50$ 时, 可达到大约 20~25 帧/s 的处理速度. 跟踪结果由一白色矩形框表示. 下面给出一些具有代表性的视频跟踪结果.

5.1 不同光照下的人脸跟踪结果

图 1 为不同光照条件下的人脸跟踪结果. 在这组视频中, 目标的表面由于光照变化而发生明显变化. 为了比较, 采用了两种不同的粒子滤波算法, 第 1 种采用固定表面模型, 第 2 种采用自适应表面模型, 分别称为非自适应跟踪算法和自适应跟踪算法. 跟踪结果表明, 自适应跟踪算法能较准确地跟踪目标, 而非自适应跟踪算法在光照发生变化时跟踪效果差, 甚至跟踪失败. 图 1(b) 左上角的 5 个小图像从左到右分别对应于高斯分量的均值、期望图像和被跟踪的图像, 其中期望图像为

$$I_t(i) = \sum_{m=1}^G P_{n,t}(i) L_{n,t}(i).$$



(a) 非自适应跟踪算法



(b) 自适应跟踪算法

图 1 不同光照下的跟踪结果

5.2 不同姿态下的人脸跟踪结果

在图 2 中, 一男士以任意姿态运动头部, 其困难是在头部姿态发生变化时, 人脸的表面也会发生较大变化. 试验结果表明, 当头部姿态发生变化不久,



(a) 非自适应跟踪算法



(b) 自适应跟踪算法

图 2 不同姿态下的跟踪结果

非自适应跟踪算法就会跟踪失败; 而自适应跟踪算法能较稳定地跟上面部变化.

5.3 部分遮挡及表情变化人脸跟踪结果

这组视频包含了较多难以处理的跟踪情况, 如人脸姿态变化、表情变化、部分遮挡、表面模糊(由运动造成)以及复杂背景. 图 3 给出了部分遮挡以及面部表情变化情况下的跟踪结果. 跟踪结果表明, 本文算法对上述情况均具有较好的跟踪效果.



(a) 部分遮挡下的跟踪结果



(b) 表情变化下的跟踪结果

图 3 部分遮挡及表情变化下的跟踪结果

5.4 完全遮挡下的人头跟踪结果

图 4 为完全遮挡时的跟踪结果. 在这种情况下, 必须解决好以下问题: 1) 遮挡发生时如何处理表面模型的更新; 2) 如何防止跟踪器陷入遮挡部分. 尽管文献[7, 8]也提出了自适应表面模型, 但没能很好地处理这两个问题. 本文在处理遮挡时, 利用鲁棒估计技术来降低遮挡部分像素对状态估计的影响, 在检测到遮挡发生时, 停止更新表面模型, 从而保证了表面模型的可靠性. 为了提高跟踪器对遮挡部分的鲁棒性, 采用了粒子滤波算法, 其优点是, 在跟踪期间能维持目标状态的多种假设, 不易陷入局部最优. 因此, 本文算法在目标出现完全遮挡时表现出了较好的跟踪效果.



图 4 完全遮挡下的跟踪结果

6 结 论

本文提出了一种基于自适应表面模型的跟踪算法, 该算法能适应目标表面的缓慢或快速变化; 在设

计跟踪算法时,将自适应表面模型结合到粒子滤波中;处理遮挡时,采用 Huber 函数来减少遮挡区域内像素对状态估计的影响.实验结果表明,所提出的算法对光照变化、姿态变化、表情变化以及部分或完全遮挡下的跟踪均具有较好的跟踪效果.

本文算法尚存在以下不足:对于特别小或特别大的目标不太适用.如果目标特别小,目标的表面信息将变得很少,此时不能仅利用灰度表面信息来跟踪目标;若目标特别大,则表面模型的尺寸将变大,算法的实时性受影响.因此,如何提高该算法对特别大的目标跟踪的实时性还需进一步研究.

参考文献(References)

- [1] Comaniciu D, Ramesh V, Meer P. Realtime tracking of nonrigid objects using mean shift[C]. Proc IEEE Conf on Computer Vision and Pattern Recognition. South Carolina: Hilton Head Island, 2004, 2: 142-149.
- [2] Liu T L, Chen H T. Realtime tracking using trust region methods[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2004, 26(3): 397-402.
- [3] Hager G D, Belhumeur P N. Efficient region tracking with parametric models of geometry and illumination[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1998, 20(10): 1025-1039.
- [4] Nguyen H T, Worring M, Van Den Boomgaard R. Occlusion robust adaptive template tracking[C]. Proc Int Conf on Computer Vision. Vancouver, 2001, 1: 672-683.
- [5] Sidenbladh H, Black M J, Fleet D J. Stochastic tracking of 3D human figures using 2D image motion[C]. Proc European Conf on Computer Vision. Dublin, 2000, 2: 702-718.
- [6] Wu Y, Huang T S. Color tracking by transductive learning[C]. Proc IEEE Conf on Computer Vision and Pattern Recognition. South Carolina: Hilton Head Island, 2000, 1: 132-138.
- [7] Ross D, Lim J, Yang M H. Adaptive probabilistic visual tracking with incremental subspace update[C]. Proc European Conf on Computer Vision. Prague, 2004, 2: 472-482.
- [8] Jepson A D, Fleet D J, ElMaraghi T F. Robust online appearance models for visual tracking[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2003, 25(10): 1296-1311.
- [9] Zhou S K, Chellappa R, Moghaddam B. Visual tracking and recognition using appearance-adaptive models in particle filters[J]. IEEE Trans on Image Processing, 2004, 13(11): 1491-1506.
- [10] Dempster A P, Laird N M, Rubin D B. Maximum likelihood from incomplete data via the EM algorithm[J]. J of the Royal Statistical Society B, 1977, 39(1): 1-38.
- [11] Arulampalam M S, Maskell S, Gordon N, et al. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking[J]. IEEE Trans on Signal Processing, 2002, 50(2): 174-188.
- [12] David Ross. Some test sequences are available at the websites[DB/OL]. (2002-08-24). <http://www.cs.toronto.edu/~dross/ivt>.
- [8] Miguel Sánchez, Pietro Manzoni. ANEJOS: A Java based simulator for Ad Hoc networks[J]. Future Generation System, 2001, 17(5): 573-583.
- [9] Nedo Celandron, Erina Ferro, Francesco Potort. A traffic generator for testing communication system: Presentation, implementation and performance[J]. Real Time System, 1997, 13(1): 5-24.
- [10] 王建新. 基于非精确网络状态信息 QoS 路由优化算法研究[D]. 长沙: 中南大学, 2001: 5-19.
- (Wang J X. QoS routing algorithms with inaccurate network state information[D]. Changsha: Central South University, 2001: 5-19.)
- [11] Paul K, Bandyopadhyay S, Mukherjee A, et al. A stability-based distributed routing mechanism to support unicast and multicast routing in Ad Hoc wireless network[J]. Computer Communication, 2001, 24(18): 1822-1845.

(上接第 52 页)