

一种健壮的目标多自由度 Mean Shift 序列图像跟踪算法

贾静平 柴艳妹 赵荣椿

(西北工业大学计算机学院, 西安 710072)

摘要 鉴于现有的 Mean Shift 跟踪方法都是使用单一半径参数来描述目标大小变化,且每个目标仅有位置和尺寸两个自由度,因而不能适应复杂的目标运动情况。针对该问题,首先提出了一种新的 Mean Shift 跟踪方法,由于该方法是通过引入带宽矩阵来描述目标尺寸,因此能够在水平和垂直两个方向上独立描述目标的大小变化,并通过加入目标倾角,使得目标旋转运动得以很好描述;然后借鉴了三步搜索的思想,提出了一种快速搜索策略,以解决目标遮挡问题。实验表明,该算法能够准确跟踪序列图像中的任意复杂运动,尤其对目标的缩放、旋转运动以及遮挡有良好的适应性。

关键词 中值平移 序列图像目标跟踪 目标倾角 带宽矩阵 目标自适应

中图分类号: TP391.41 文献标识码: A 文章编号: 1006-8961(2006)05-0707-07

Robust Tracking of Objects in Image Sequences Using Multiple Degrees of Freedom Mean Shift Algorithm

JIA Jing-ping, CHAI Yan-mei, ZHAO Rong-chun

(School of Computer, Northwestern Polytechnical Univ., Xi'an 710072)

Abstract Current Mean Shift tracking algorithm uses a single radius parameter to describe the scale of the target, and Each target has the position and size freeness only, which is not suitable for complex movements of the objects. In this paper we provide a new algorithm in which a bandwidth matrix is employed to describe the objects with two directions determined independently. Target-angle is also introduced to accommodate the rotation of objects. Furthermore we bring forward an efficient search strategy to cope with a temporal occlusion of the objects. Experimental results show that the new algorithm is able to adapt to any kind of object's movements and therefore has better tracking precision.

Keywords Mean Shift, tracking of objects in image sequences, target-angle, bandwidth matrix, adaptability

1 引言

众所周知,实时目标跟踪是视频监视、智能用户接口、基于对象的视频压缩、巡航导弹末端制导和辅助驾驶等许多计算机视觉应用中的关键技术。在这些系统中,目标的运动跟踪通常都十分困难,这是由于在序列图像中存在旋转、缩放和与背景或其他目

标发生部分遮挡等情况,因此跟踪算法既要计算量小,又要能适应目标和背景的复杂变化。

文献[1]通过使用 Mean Shift 方法对彩色图像中的目标进行跟踪,发展了一种在目标运动不很复杂的条件下,时间复杂度很小,而跟踪精度却较高的跟踪算法。该方法不仅能快速地在当前帧中搜索上一帧中目标的新位置,并能通过改变单个核半径参数来试探目标的新尺寸,仿真结果和理论推导表明,

基金项目:航空科学基金项目(02153073)

收稿日期:2005-03-17;改回日期:2005-09-12

第一作者简介:贾静平(1978~),男,2004年获西北工业大学工学硕士学位,现为西北工业大学博士研究生。研究方向为机器视觉和数字视频。E-mail:rocsovsky@163.com

这种策略在描述目标复杂运动方面还不够理想。文献[2]在使用 DOG-Epanichnikov 核的同时,在尺度空间和坐标空间中寻找极大值,虽有效地解决了目标的缩放问题,但对目标旋转仍然无能为力。

本文在 Mean Shift 算法的基础上,针对文献[1]和文献[2]算法的缺点,将窗半径 h 的概念推广为带宽矩阵,并通过引入目标倾角,推导得出了一种新的跟踪算法。理论分析和实验结果表明,本文的算法不仅能更好地适应目标复杂运动,还能提高跟踪的稳定性。

2 使用带宽矩阵的 Mean Shift 过程

现有的 Mean Shift 方法都是使用单一核半径来进行推导的,而文献[3]仅仅给出了一般意义的多变量核密度估计定义。为了说明本文的算法,首先必须给出使用带宽矩阵的 Mean Shift 方法。为简化起见,本文仅讨论 2 维图像空间中的情况。若给定 n 个点 $X_i \in \mathbf{R}^2, i=1, \dots, n$ 和一个对称正定的 2×2 矩阵 H ,则其使用的核函数 $K(X)$ 的多变量核密度估计^[3]为

$$\hat{f}_{H,K}(X) = \frac{1}{n|\mathbf{H}|^{1/2}} \sum_{i=1}^n K(\mathbf{H}^{-1/2}(X - X_i)) \quad (1)$$

$K(X)$ 为有界的紧支撑函数,并满足

$$\int_{\mathbf{R}^2} K(X) dX = 1 \quad \lim_{\|X\| \rightarrow \infty} \|X\|^2 K(X) = 0$$

$$\int_{\mathbf{R}^2} \mathbf{X}K(X) dX = 0 \quad \int_{\mathbf{R}^2} \mathbf{X}\mathbf{X}^T K(X) dX = c_k \mathbf{I}$$

c_k 为一个与 $K(X)$ 有关的常数。现在只考虑一类特殊情况, H 为对角阵 $\text{diag}[h_u^2, h_v^2]$, $K(X)$ 中心对称,且满足 $K(X) = c_k k(\|X\|^2)$, $k(x)$ 定义在 $x \geq 0$ 区间上。 $k(x)$ 称为 $K(X)$ 的轮廓(profile)函数, c_k 为与 $k(x)$ 有关的常数。这样核密度估计可改写为

$$\hat{f}_{H,k}(X) = \frac{c_k}{n|\mathbf{H}|^{1/2}} \sum_{i=1}^n k(\|\mathbf{H}^{-1/2}(X - X_i)\|^2) \quad (2)$$

如果 $k(x)$ 在 $[0, \infty)$ 上除有限个点外都可导,那么可定义 $g(x) = -k'(x)$ 。该 $g(x)$ 对应了一个新的核函数 $G(X) = c_g g(\|X\|^2)$, c_g 为与 $g(x)$ 有关的常数,使用该核函数的多变量核密度估计为

$$\hat{f}_{H,g}(X) = \frac{c_g}{n|\mathbf{H}|^{1/2}} \sum_{i=1}^n g(\|\mathbf{H}^{-1/2}(X - X_i)\|^2) \quad (3)$$

核密度梯度估计为

$$\hat{\nabla} f_{H,k}(X) \equiv \hat{\nabla} f_{H,g}(X) = \frac{c_k}{n|\mathbf{H}|^{1/2}} \sum_{i=1}^n \nabla k(\|\mathbf{H}^{-1/2}(X - X_i)\|^2)$$

$$= \frac{2c_k}{n|\mathbf{H}|^{1/2}} \mathbf{H}^{-1} \sum_{i=1}^n (X - X_i) k'(\|\mathbf{H}^{-1/2}(X - X_i)\|^2) \quad (4)$$

可记

$$\mathbf{m}_{H,c}(X) = \frac{\sum_{i=1}^n X_i g(\|\mathbf{H}^{-1/2}(X - X_i)\|^2)}{\sum_{i=1}^n g(\|\mathbf{H}^{-1/2}(X - X_i)\|^2)} - X \quad (5)$$

并定义它为一般意义下的 Mean Shift 向量。结合式(3)可得到

$$\hat{\nabla} f_{H,k}(X) = \frac{2c_k}{c_g} \mathbf{H}^{-1} \hat{f}_{H,g}(X) \mathbf{m}_{H,c}(X) \quad (6)$$

也即

$$\mathbf{m}_{H,c}(X) = \frac{c_g \mathbf{H} \hat{\nabla} f_{H,k}(X)}{2c_k \hat{f}_{H,g}(X)} \quad (7)$$

一个常用的核函数为下面的 Epanechnikov 核函

$$K_E(X) = \begin{cases} 2V(1 - \|X\|^2) & \|X\| \leq 1 \\ 0 & \text{其他} \end{cases}, \text{其中 } V \text{ 为单}$$

位球体积,其对应的轮廓函数为

$$k_E(x) = \begin{cases} 1-x & 0 \leq x \leq 1 \\ 0 & \text{其他} \end{cases}, g_E(x) = \begin{cases} 1 & 0 \leq x \leq 1 \\ 0 & \text{其他} \end{cases}$$

和文献[1]中的结论进行比较,笔者发现,使用带宽矩阵的 Mean Shift 向量虽不一定指向核密度梯度估计的方向,但这并不影响本文算法的收敛性。下面的定理证明了当 $K(X) = K_E(X)$ 时,不仅使用 $\mathbf{m}_{H,c}(X)$ 的 Mean Shift 过程收敛,而且收敛使得 $\hat{\nabla} f_{H,k}(X) = 0$,即能够使得式(1)取得极大值。

定理: 设 Mean Shift 过程中,核函数 $G(X)$ 位置序列为 $\{Y_j\}_{j=1,2,\dots}$,由式(7)可见,如果初始位置为 Y_1 ,则用核函数 $G(X)$ 计算的 Y_j 处的加权平均为

$$Y_{j+1} = \frac{\sum_{i=1}^n X_i g(\|\mathbf{H}^{-1/2}(X - X_i)\|^2)}{\sum_{i=1}^n g(\|\mathbf{H}^{-1/2}(X - X_i)\|^2)} \quad (8)$$

其对应的用核函数 $K(X)$ 计算的核密度估计序列为 $\{\hat{f}_{H,K}(j)\}_{j=1,2,\dots}$, $\hat{f}_{H,K}(j) = \hat{f}_{H,K}(Y_j), j=1, 2, \dots$, Mean Shift 向量序列为 $\{\mathbf{m}_{H,c}(j)\}_{j=1,2,\dots}$, $\mathbf{m}_{H,c}(j) = Y_{j+1} - Y_j$ 。如果 $K(X) = K_E(X)$ (其轮廓函数 $k_E(x)$ 为凸的,且单调递减),那么 $\{\mathbf{m}_{H,c}(j)\}_{j=1,2,\dots}$ 和 $\{\hat{f}_{H,K}(j)\}_{j=1,2,\dots}$ 收敛。证明过程见文献[4]。

文献[1]力图证明 $\{Y_j\}_{j=1,2,\dots}$ 也是收敛的。其证明过程认为,若 $\|Y_{j+1} - Y_j\|$ 收敛,则 $\{Y_j\}_{j=1,2,\dots}$ 为柯西数列,也是收敛数列。但这个说法是不严密的,存在反例。柯西数列的判决条件比较严格(见文献[5])。本文的定理严格证明了 $\{\mathbf{m}_{H,c}(j)\}_{j=1,2,\dots}$ 收

敛,已经为本文算法的收敛提供了足够依据。

3 目标特征

本文用目标矩形来描述目标,目标矩形定义如下:该矩形的两个对称轴分别为横、竖轴,其轴长度分别为带宽矩阵的 h_h 和 h_v 。从坐标平面的 x 轴正方向逆时针旋转到竖轴所形成的角 α 定义为目标的倾角。目标中心为 (x_c, y_c) ,其中任一点 (x_i, y_i) 到两轴的距离为其到两轴的垂线长度(如图 1 所示),而且目标矩形有一个邻接矩形,该邻接矩形的边平行于 x, y 轴。

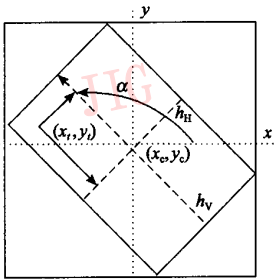


图 1 目标矩形

Fig. 1 Target rectangle

设 $\{X_i\}_{i=1, \dots, n}$ 为候选目标矩形所在区域的点集,该点集以点 (x_c, y_c) 为中心。若定义函数 $b: \mathbf{R}^2 \rightarrow \{1, \dots, m\}$, 则对于每个像素 $X_i, b(X_i)$ 是该像素的颜色值。若使用 Epanechnikov 核,则颜色 $u = 1, \dots, m$ 在目标模式中出现的概率为

$$p_u(Y) = C_H \sum_{i=1}^{n_h} k(\|H^{-1}(Y - X_i)\|^2) \delta[b(X_i) - u] \quad (9)$$

δ 是 Kronecker 函数, C_H 是矩阵 H 归一化因子,它使得 $\sum_{u=1}^m p_u = 1, C_H = \frac{1}{\sum_{i=1}^{n_h} k(\|H^{-1}(Y - X_i)\|^2)}$ (n_h 代

表当前矩形内的像素总数)。本文算法不同于原算法仅使用单独的 h , 而使用了对角阵 $H = \text{diag}[h_h^2, h_v^2]$, h_h, h_v 分别反映目标在横竖两个方向上的宽度。设点 X_i 到横竖两轴的距离分别为 d_h, d_v , 由于目标倾角 α 的存在,因此核函数的输入应由

$$\|H^{-1}(Y - X_i)\|^2 = 4 \left(\frac{(x_i - x_c)^2}{h_h^2} + \frac{(y_i - y_c)^2}{h_v^2} \right) \text{ 修改为 } 4 \left(\frac{d_h^2}{h_h^2} + \frac{d_v^2}{h_v^2} \right)。$$

初始目标特征沿用文献 [1] 中的方法。设 $\{X_i\}_{i=1, \dots, n}$ 为目标所在区域的点集,而且区域坐标已调整到以 0 为中心,并按照区域大小 (h_h, h_v) 归一化了。若使用 Epanechnikov 核,则颜色 $u = 1, \dots, m$ 在目标模式中出现的概率为

$$q_u = C \sum_{i=1}^n k(\|X_i\|^2) \delta[b(X_i) - u] \quad (10)$$

并可使用 Bhattacharyya 系数^[6]

$$\rho(Y) \equiv \rho[p(Y), q] = \sum_{u=1}^m \sqrt{p_u(Y)q_u} \quad (11)$$

来描述目标和候选目标的相似程度。它们之间的距离定义为

$$d(Y) = \sqrt{1 - \rho[p(Y), q]} \quad (12)$$

由该式可见,系数 ρ 越大,距离就越小,目标和候选目标也越相似,因此目标跟踪就变成了在当前帧中搜索对应于目标新位置的过程,该位置将使得以 Y 为自变量的距离函数取得最小值。搜索从前一帧中目标的位置开始,在其周围邻域内寻找。假设前一帧中目标的位置为 Y_0 , 若将 $\rho[p(Y), q]$ 在目标预测位置 Y_0 处展开,则可获得以下 $\rho[p(Y), q]$ 的线性近似:

$$\rho[p(Y), q] \approx \frac{1}{2} \sum_{u=1}^m \sqrt{p_u(Y_0)q_u} + \frac{1}{2} \sum_{u=1}^m p_u(Y) \sqrt{\frac{q_u}{p_u(Y_0)}} \quad (13)$$

将式(9)代入式(13)得

$$\rho[p(Y), q] \approx \frac{1}{2} \sum_{u=1}^m \sqrt{p_u(Y_0)q_u} + \frac{C_H}{2} \sum_{i=1}^{n_h} w_i k(\|H^{-1}(Y - X_i)\|^2) \quad (14)$$

其中

$$w_i = \sum_{u=1}^m \delta[b(X_i) - u] \sqrt{\frac{q_u}{p_u(Y_0)}} \quad (15)$$

这样,最小化式(12) $d(Y)$, 就变成最大化式(14)中的第 2 项,而该项则代表着当前帧中 Y 位置处的使用 $k(x)$ 计算的密度估计,只不过数据多了加权值 w_i 罢了。这样就可以使用 Mean-Shift 过程来寻找邻域内该密度估计的极大值。在这个过程中,核就从前位置 Y_0 移向新位置 Y_1 :

$$Y_1 = \frac{\sum_{i=1}^{n_h} X_i w_i g(\|H^{-1}(Y - X_i)\|^2)}{\sum_{i=1}^{n_h} w_i g(\|H^{-1}(Y - X_i)\|^2)} = \frac{\sum_{i=1}^{n_h} X_i w_i}{\sum_{i=1}^{n_h} w_i} \quad (16)$$

4 目标丢失时的搜索策略

当 Mean-Shift 过程的收敛区域为当前帧中的目标矩形, 即当第 $k+1$ 帧中的目标与第 k 帧中的目标矩形有相交部分时, 则 Mean-Shift 过程最终寻找到的与式(14)极大值对应的位置, 就是第 $k+1$ 帧中目标的位置。当不相交时, 则 Mean Shift 过程陷入局部极大值, 同时目标丢失, 这种情况在摄像机存在全局运动或目标被完全遮挡时, 经常发生。这里可借鉴文献[7]中的三步搜索思想, 在目标周围搜索, 以找回目标(如图 2 所示)。

2		2		2		2
	1		1		1	
2						2
	1	A	0		1	
2						2
	1		1		1	
2		2		2		2

图 2 搜索策略

Fig. 2 Search strategy

图 2 中的每一个方块都和目标的邻接矩形相同大小。假设第 k 帧时目标矩形位于块 0 内(见图 2), 目标正向左运动, 并在第 $k+1$ 帧时到达块 A, 同时被完全遮挡, 并使 Mean-Shift 过程丢失目标。这时需启动搜索过程, 首先搜索图中所有的块 1, 并计算它们的 Bhattacharyya 系数, 如果目标出现在某个块 1 中, 则通过更新目标矩形的位置来重新捕获目标, 否则根据目标前若干帧的平均速度和自丢失以来经过的时间来计算其离开块 0 可能的距离范围, 若范围超出块 1, 则进一步搜索块 2, 以进一步捕获目标。

另一种找回目标的策略是通过估计目标运动状态来估计目标消失期间目标的位置, 以便在估计位置附近搜索目标(如文献[8]中所述)。但这种策略在摄像机存在抖动时, 经常失效, 因为抖动使目标的运动模型不准确, 从而使运动状态的估计变得困难。而本文的策略则克服了这一缺点, 实验表明, 在多数情况下, 只需搜索 8 个块 1 就能重新捕获目标, 最多只需搜索 20 个块(全局搜索要 48 个块)。

5 跟踪算法

单个目标的一次跟踪过程(TrackOnce)如下: 如

果目标具有特征 $\{q_u\}_{u=1, \dots, m}$, 且在前一帧中位于 Y_0 , 那么重复以下步骤就可得到目标在当前帧中的新位置 Y_1 :

(1) 估计当前帧中 Y_0 处候选目标的特征

$$\{p_u(Y_0)\}_{u=1, \dots, m}, \text{ 并计算 } \rho[p(Y_0), q] = \sum_{u=1}^m \sqrt{p_u(Y_0)q_u};$$

(2) 计算权值集 $\{w_i\}_{i=1, \dots, m}$;

(3) 使用 Mean Shift 方法, 计算目标的新位置

$$Y_1 = \frac{\sum_{i=1}^{n_h} X_i w_i g(\|H^{-1}(Y - X_i)\|^2)}{\sum_{i=1}^{n_h} w_i g(\|H^{-1}(Y - X_i)\|^2)} = \frac{\sum_{i=1}^{n_h} X_i w_i}{\sum_{i=1}^{n_h} w_i}$$

(4) 更新 $\{p_u(Y_1)\}_{u=1, \dots, m}$, 并计算

$$\rho[p(Y_1), q] = \sum_{u=1}^m \sqrt{p_u(Y_1)q_u}$$

(5) 如果 $\rho[p(Y_1), q] > \rho[p(Y_0), q]$, 那么 $Y_1 \leftarrow \frac{1}{2}(Y_0 + Y_1)$, 直到

$\rho[p(Y_1), q] > \rho[p(Y_0), q]$;

(6) 如果 $\|Y_1 - Y_0\| < \varepsilon$, 则停止, 否则 $Y_0 \leftarrow Y_1$, 转步骤(2)。 ε 的选取应使 Y_0 和 Y_1 间距离小于一个像素。

当有了对单个目标的一次跟踪过程(TrackOnce), 则目标未丢失时的跟踪过程(TrackNormal)如下:

(1) 倾角自适应 先重复单个目标的一次跟踪过程 3 次, 其中两次使用不同的倾角 α 值, 分别为原倾角值加、减 10° , 然后比较 3 次收敛后的 Bhattacharyya 系数, 并取对应系数较大的 α 为新的倾角值;

(2) 横轴自适应 先重复单个目标的一次跟踪过程 3 次, 其中两次使用不同的横轴 h_h 值, 分别为原 h_h 值增减 10% , 然后比较 3 次收敛后的 Bhattacharyya 系数, 并取对应系数较大的 h_h 为新的横轴值;

(3) 竖轴自适应 先重复单个目标的一次跟踪过程 3 次, 其中两次使用不同的竖轴 h_v 值, 分别为原 h_v 值增减 10% , 然后比较 3 次收敛后的 Bhattacharyya 系数, 并取对应系数较大的 h_v 为新的竖轴值。

为减少计算量, 以上 3 步可交替进行, 即对某一帧图像只进行其中一步, 后一帧进行另一步。为减少噪音影响, 更新 α , h_h 和 h_v 可采用类似 IIR 滤波器的方法。

完整的跟踪算法如下:

(1) 对目标执行跟踪过程(TrackNormal), 若收敛后的 Bhattacharyya 系数大于事先设定的阈值 T_c (下角 G 代表目标确实重现), 则目标未丢失, 继续跟踪过程, 否则转步骤(2);

(2) 通过搜索块 1 寻找目标,若某块的 Bhattacharyya 系数大于事先设定的阈值 T_p (下角 P 代表目标可能重现),则在该块内执行跟踪过程 (TrackOnce),若收敛后的 Bhattacharyya 系数大于 T_c ,则目标被重新捕获,否则转步骤(3);

(3) 计算目标可能的位移范围,如目标在块 1 之外,则搜索块 2,若某块的 Bhattacharyya 系数大于事先设定的阈值 T_p ,则在该块内执行跟踪过程 (TrackOnce),若收敛后的 Bhattacharyya 系数大于 T_c ,则目标被重新捕获,否则认为目标未出现。

6 跟踪实例

为验证本文新算法的效果,用几个不同序列图像对新的跟踪算法进行了大量测试,图 3 是“卡车”序列的跟踪结果。该序列含 143 帧 320×320 大小的图像。跟踪时,取 RGB 空间中的 $16 \times 16 \times 16$ 个直方图特征,跟踪图中的方框显示跟踪的卡车,在 104 帧中新加入一个目标,证明本文算法可同时跟踪多个目标。由图 3 可以看出,在 105 帧以前,由于多自由度算法的目标矩形可以随目标旋转而旋转,且随目标放大而放大,因此对目标的描述明显更准确。在 143 帧中,由于背景和目標十分类似,并且连接,因而导致跟踪精度有所下降。图 4 是使用文献 [1] 方法跟踪的结果,用该方法跟踪时,目标方框只能以固定长宽比缩放,且不能旋转。

图 5 是“水杯”序列目标跟踪结果,该序列含 376 帧 320×240 大小的图像,图中蓝色水杯是目标,它逐渐靠近镜头(放大)并且发生顺时针旋转,目标方框也随之放大并且旋转,方框内的竖线代表竖轴正向。204 帧后的角度误差是初始选定目标时,由选中的一部分背景引入的,该误差可在跟踪过程中,通过更新目标特征来消除,其跟踪过程的参数如图 6 所示。不考虑目标倾角时,由于目标方框只能通过改变长宽比来适应目标的旋转,因此描述目标的能力差了很多。

图 7 对比了本文算法和文献 [1] 算法对“行人”序列的跟踪结果。由图 7 可见,从 154 帧到 162 帧,被跟踪的行人被电线杆完全遮挡,目标丢失,但在第 168 帧时又重新捕获目标,继续跟踪,而文献 [1] 中的 Mean Shift 过程则由于未能跳出局部极大值点,致使遮挡结束后无法继续跟踪。实验中取 $T_G = 0.5, T_p = 0.3$ 。

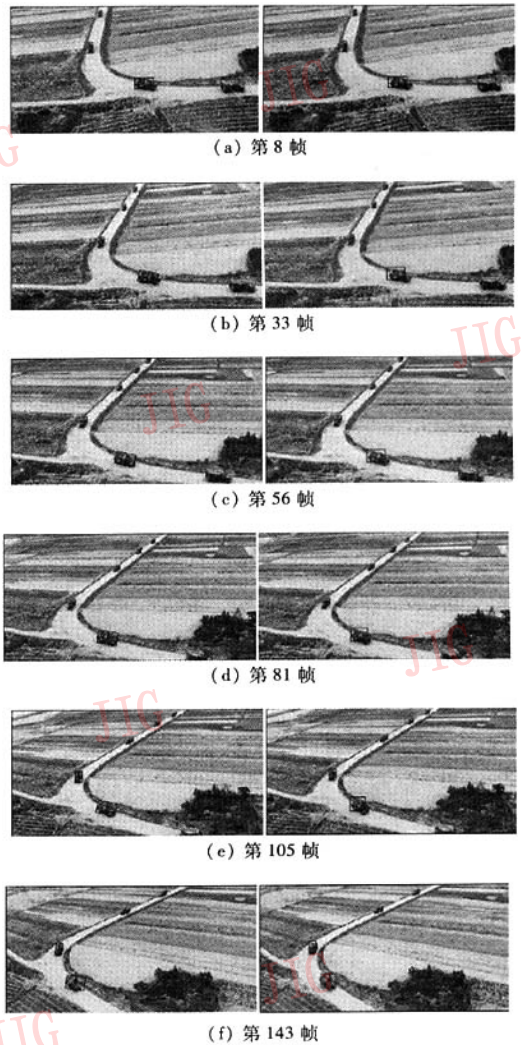


图 3 “卡车”序列:考虑目标倾角时多自由度算法的跟踪结果(左)与不考虑倾角时的跟踪结果(右)比较,

Fig. 3 The cup sequence, tracked with and without the target angle

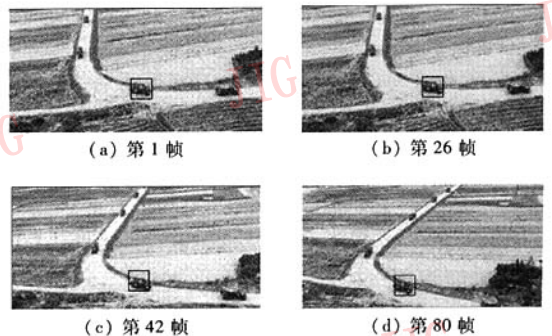


图 4 “卡车”序列:文献 [1] 方法跟踪的结果

Fig. 4 The “truck” sequence, tracked using the algorithm in [1]

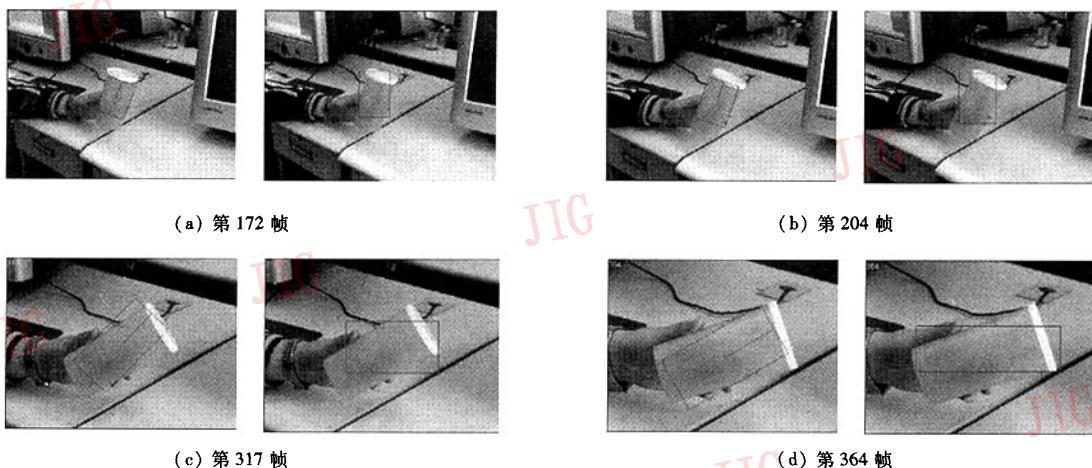


图 5 “水杯”序列:考虑目标倾角时多自由度算法跟踪结果与不考虑倾角时的跟踪结果比较
 Fig. 5 The cup sequence, tracked by the multi-freedom algorithm with and without the target angle

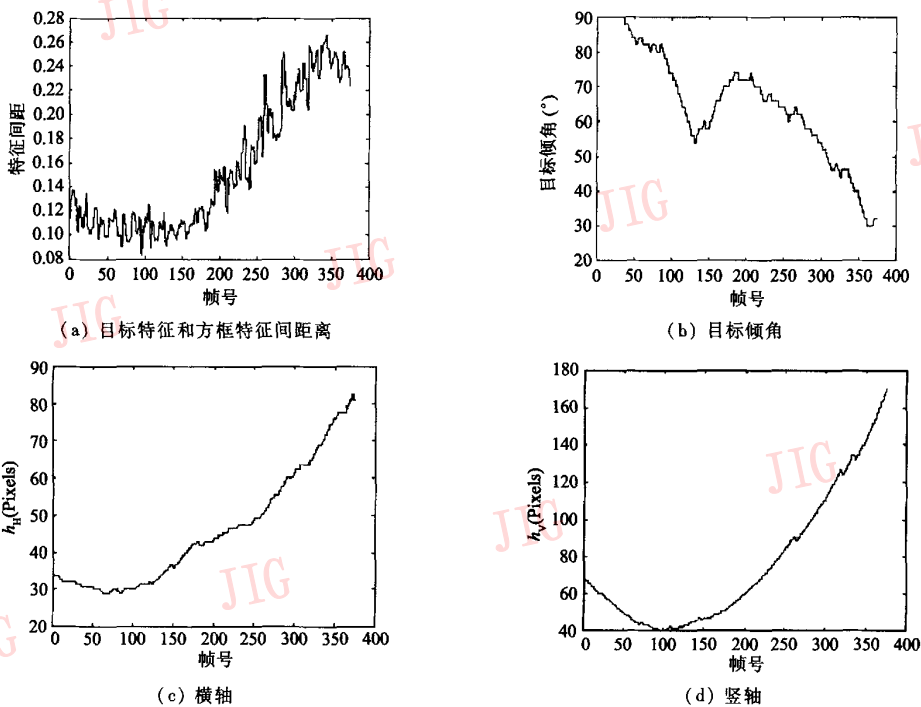


图 6 “水杯”序列:多自由度算法跟踪参数

Fig. 6 The tracking parameters of the cup sequence

7 计算量分析

本文提出的多自由度 Mean-shift 跟踪算法可满足实时应用的要求,在跟踪面积为 25×20 的单个目标时,平均处理速度为每帧每目标 100ms,运行环境为 Pentium III 866MHz, Windows 2000 Server, J2SE 1.4.1_02-b06。表 1 比较了文献[1]、[2]以及本文

算法的计算量,从中可以看出,本文算法在适当的计算量下,提高了跟踪的性能。

表 1 3 种算法的计算量比较

Tab. 1 The comparison of the three algorithms

25 × 20 目标算法	处理时间(ms)	跟踪性能
文献[1]的算法	40 ~ 80	无旋转,缩放不准确
文献[2]的算法	120 ~ 200	无旋转,固定比例缩放
本文的算法	50 ~ 180	可旋转,任意比例缩放,抗遮挡

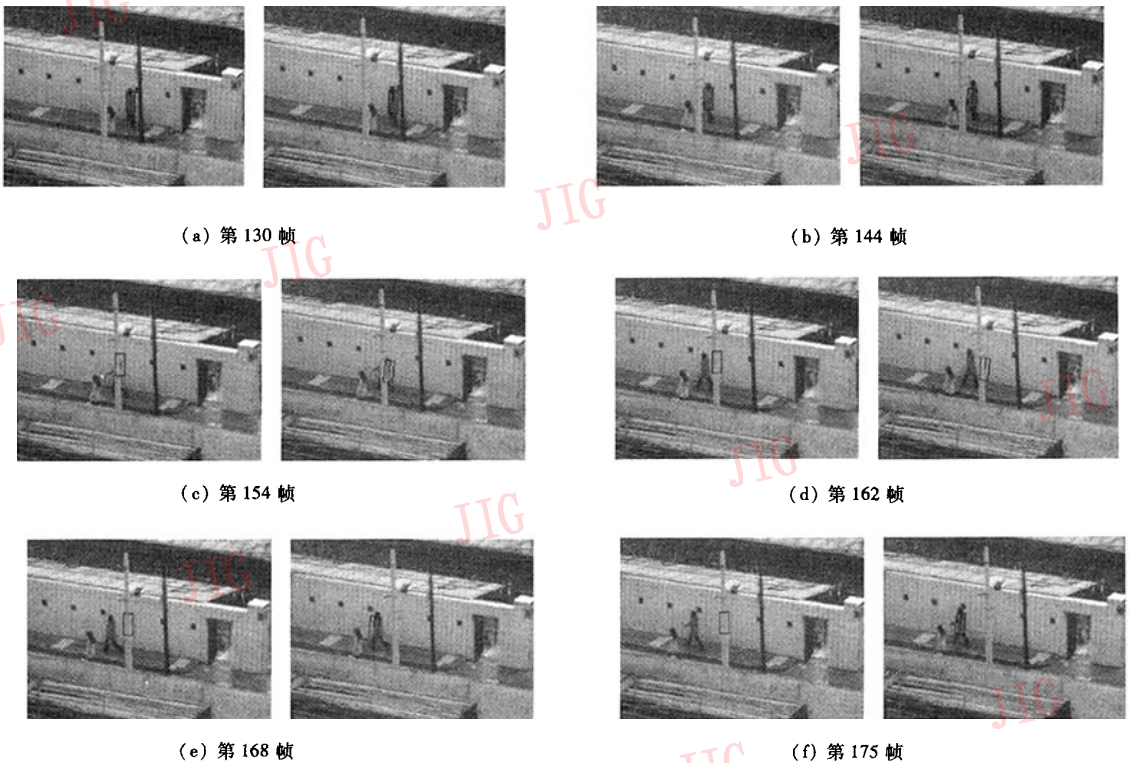


图7 行人跟踪,文献[1]中算法结果(左列)与本文算法的比较(右列)

Fig.7 The tracking of a pedestrian, left column: using the algorithm in[1], right column: using our algorithm

8 结 论

本文针对现有的 Mean Shift 方法不能适应复杂的目标运动的缺点,通过引入带宽矩阵和目标倾角,提出了一种使用带宽矩阵来描述目标尺寸和使用目标倾角来描述目标旋转的多自由度 Mean-shift 跟踪算法,同时借鉴了三步搜索的思想,提出了一种快速搜索策略。实验表明,与现有算法比较,本文的算法不仅描述目标区域更精确,而且更能适应目标在图像中的复杂运动。下一步研究的重点是跟踪过程中的目标特征更新,而更新的策略则要求既能适应目标自身的变化,又能减少背景的干扰。

参考文献 (References)

- Comanicu Dorin, Ramesh Visvanathan, Meer Peter. Real-time tracking of non-rigid objects using mean shift[A]. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition[C], Hilton Head Island, South Carolina, USA, 2000, 2: 142 ~ 149.
- Collins Robert T. Mean-shift blob tracking through scale space[A]. In: Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition[C], Madison, Wisconsin,

USA, 2003, 2: 234 ~ 240.

- Comanicu Dorin, Meer Peter. Mean Shift: A robust approach toward feature space analysis[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(5): 603 ~ 619.
- Jia Jing-ping, Zhao Rong-chun. Tracking of objects in image sequences using bandwidth matrix mean shift algorithm[A]. In: Proceedings of the 2004 International Conference on Signal Processing (ICSP'04)[C], Beijing, 2004, 2: 918 ~ 921.
- Wang Miansen. The fundamental of engineering analysis[M]. The Press of Xi'an Jiao Tong University, 1996: 25. [王棉森. 工科分析基础[M]. 西安:西安交通大学出版社,1996:25.]
- Kailath T. The divergence and bhattacharyya distance measures in signal selection [J]. IEEE Transactions Communications, Technology, 1967, COM-15(1): 52 ~ 60.
- Ren Jin-chang, Zhang Wen-zhe, Zhao Rong-chun, et al. An automatic target tracking method based on self-adaptive threshold under complex background[J]. Application Research of Computers, 2003, 20(4): 55 ~ 57. [任金昌,张文哲,赵荣椿等. 一种基于自适应阈值的复杂背景下自动目标跟踪算法[J]. 计算机应用研究, 2003, 20(4): 55 ~ 57.]
- Koichiro Deguchi, Oki Kawanaka, Takayuki Okatani. Object tracking by the mean-shift of regional color distribution combined with the particle-filter algorithm [A]. In: proceedings of the 17th International Conference on Pattern Recognition [C], Cambridge, UK, 2004, 3: 506 ~ 509.