

一种改进的 Mean Shift 跟踪算法

李培华¹

摘要 本文主要针对经典的 Mean Shift 跟踪算法均匀剖分整个颜色空间造成许多空的直方图区间以及不能准确表达目标颜色分布的缺点, 提出了一种改进算法. 该改进算法首先对目标的颜色进行聚类分析, 根据聚类结果通过矩阵分解和正交变换自适应地剖分目标的颜色空间从而确定对应于每一聚类的子空间. 在此基础上定义了一种新的颜色模型, 该模型统计落入每一颜色子空间的像素的加权个数并用高斯分布建模每一个子空间的颜色分布, 并推导了一种相似性度量来比较目标和候选目标的颜色模型之间的相似程度. 最后基于该颜色模型提出了改进算法. 实验表明, 基于该颜色模型的改进算法比经典的 Mean Shift 算法具有更好的性能, 而跟踪时间与经典算法大致相同.

关键词 目标跟踪, Mean Shift, 颜色模型, 相似性度量
中图分类号 TP391

An Improved Mean Shift Algorithm for Object Tracking

LI Pei-Hua¹

Abstract The traditional Mean Shift tracking algorithm partitions uniformly the whole color space, leading to a great number of void histogram bins, and is unable to represent accurately the color distribution of the object. To address the problems, we present an improved algorithm. Firstly the object color is analyzed using a clustering algorithm, and according to the clustering result the color space of the object is partitioned into subspaces by matrix factorization and orthonormal transformation. Then a new color model is defined by considering the weighted number of pixels as well as within-cluster distribution with Gaussian, and a novel measure is derived to evaluate the similarity between the reference color model and the candidate model. Finally an improved algorithm is proposed based on the color model. Experiments show that the improved algorithm has better performance than and is computationally comparable to the conventional mean shift algorithm.

Key words Object tracking, Mean Shift, color model, similarity measure

1 引言

颜色信息由于对目标的平移、旋转和尺度变化具有不变性以及对于遮挡和姿态变化不敏感的特点而成为图像处理中一种重要的特征. 因此基于颜色信息的目标跟踪方法引起了许多研究者的关注^[1~8], 在视频监控、智能人机交互、图像和视频编码以及智能机器人等许多研究领域得到了广泛的应用.

传统的颜色直方图是建模目标颜色信息的基本方法, 该方法统计落入每一个直方图区间的像素的数目. 然而由于传统直方图用于目标跟踪时不存在快速算法, 需要穷尽地对搜索范围内的每一个可能位置进行计算, 从而在计算效率上不可行. Comaniciu 等^[4] 提出了一种加权直方图 (Weighted histogram) 建模目标的颜色概率分布 (又称为基于核的概率密度表达), 该方法统计落入每一个直方图区间的像素的加权个数, 其中权重是落入该区间的像素二维空间坐标和目标中心的距离的函数, 距离越小权重越大. 这种方法用 Bhattacharyya 距离来度量颜色模型之间的相似度, 通过一阶梯度下降算法将该距离测度极小化, 从而在搜索范围内得到目标的位置. 这种方法被称之为 Mean Shift (MS) 算法, 能够通过较少的几次迭代收敛到目标的真实位置.

Comaniciu 等提出的经典 Mean Shift (MS) 算法由于计算速度快、跟踪效果好而吸引了众多研究者的关注, 在过去的几年中改进算法不断涌现. Collins 基于 Lindeberg 的尺度空间理论, 提出了一种用来解决 Mean Shift 算法不能很好地适应目标尺度变化的缺点的方法^[5]. Birchfield 等引入一种

收稿日期 2006-7-25 收修改稿日期 2006-9-12
Received July 25, 2006; in revised form September 12, 2006
国家自然科学基金 (60505006, 60673110) 资助, 黑龙江省自然科学基金 (F200512) 资助; 同时得到下列科研基金的部分资助: 教育部归国留学人员基金, 黑龙江省普通高等学校青年学术骨干支持计划项目 (1151G033), 黑龙江省博士后科研启动金 (LHK-04093), 黑龙江大学杰出青年基金 (JC200406)
Supported by National Natural Science Foundation of P. R. China (60505006, 60673110), Natural Science Foundation of Heilongjiang Province (F200512), and supported in part by the Scientific Research Foundation for the Returned Overseas Chinese Scholars of State Education Ministry, Excellent Young Scholars of Higher University of Heilongjiang Province (1151G033), Postdoctoral Fund for Scientific Research of Heilongjiang Province (LHK-04093) and Science Fund of Heilongjiang University for Distinguished Young Scholars (JC200406)
1. 黑龙江大学计算机科学与技术学院 哈尔滨 150080
1. School of Computer Science and Technology, Heilongjiang University, Harbin 150080
DOI: 10.1360/aas-007-0347

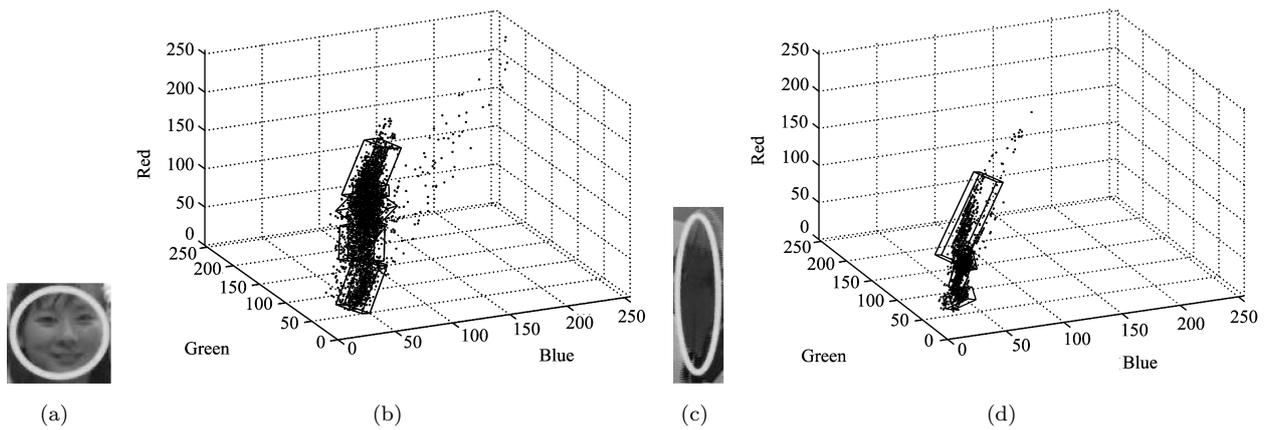


图 1 两种典型目标人脸 (a) 和行人 (c) 的参考图像以及对应的颜色空间自适应剖分 (b) 和 (d)
Fig. 1 The reference images of two typical targets of human face (a) and pedestrian (c), and the corresponding automatic partitions of color space (b) and (d)

空间直方图模型, 该模型不仅考虑目标的颜色信息而且考虑像素之间的空间位置关系^[6]. Zhao 等对 Correlogram 直方图进行简化, 同时将目标的三维信息考虑进来, 结合 Mean Shift 算法对目标进行跟踪^[7]. Yang 等人提出了基于一种具有更强分辨能力的相似性测度的 Mean Shift 算法, 应用快速高斯变换减少计算复杂性^[8].

目标的颜色通常紧凑地分布在颜色空间的一部分子空间中, 图 1 所示的是两种典型目标人脸和行人的颜色在 RGB 空间分布情况. 经典的 Mean Shift 算法及改进算法通常将整个颜色空间 ($256 \times 256 \times 256$) 均匀剖分成 $16 \times 16 \times 16$ (或 $32 \times 32 \times 32$) 的直方图区间^[4~8]. 这种均匀剖分的方法会造成许多空的直方图区间, 引起内存空间的浪费; 同时也不能准确地表达目标的颜色分布情况. 采用更多的直方图区间能更准确地表达目标的颜色分布, 可能会使跟踪算法更加准确, 但同时将使算法对光照或表观的变化更加敏感并增加算法的计算量.

针对上述问题, 我们提出了一种基于聚类分析的改进 Mean Shift 算法. 在第 2 节中, 应用基于核概率密度的聚类算法将目标的颜色空间自适应地剖分, 从而自动地确定对应每一聚类的子空间; 定义了一种新的颜色模型并推导了度量颜色模型相似程度的测度; 基于该颜色模型提出了改进的 Mean Shift 算法. 在第 3 节中比较了经典的 Mean Shift 算法和改进算法的性能.

2 改进的 Mean Shift 算法

在聚类分析的基础上自适应地划分区间的直方图能够使用数量较少的区间更准确地表达目标的颜

色概率密度.

2.1 自适应地剖分目标的颜色空间

我们应用 Comaniciu 等提出的算法^[9] 对目标的颜色分布进行聚类分析. 该算法依赖于核概率密度表达, 能够自动地确定聚类数 d . 在聚类分析的基础上, 根据如下的方法对目标的颜色空间进行剖分.

对每一聚类 C_u , $u = 1, \dots, d$, 令 $\mathbf{z}_u(k)$, $k = 1, \dots, n_u$ 是属于聚类 C_u 的第 k 个像素的颜色向量, 令 $\mathbf{z}_u(k)$ 的均值为 $\bar{\mathbf{z}}_u$, 对协方差矩阵 R_u 进行如下形式的分解^[10]

$$R_u = \frac{1}{n_u} \sum_{k=1}^{n_u} (\mathbf{z}_u(k) - \bar{\mathbf{z}}_u)(\mathbf{z}_u(k) - \bar{\mathbf{z}}_u)^T = V_u \Sigma_u V_u^T$$

其中 Σ_u 是按降序排列的 R_u 特征值 $\lambda_{u,i}$, $i = 1, 2, 3$, 组成的对角矩阵, 即 $\Sigma_u = \text{diag}\{\lambda_{u,1}, \lambda_{u,2}, \lambda_{u,3}\}$, V_u 是相应的归一化的正交向量组成的矩阵. 对颜色空间进行标准正交变换

$$\mathbf{w}_u(k) = V_u^T (\mathbf{z}_u(k) - \bar{\mathbf{z}}_u) \quad k = 1, \dots, n_u \quad (1)$$

这样 $\mathbf{z}_u(k)$, $k = 1, \dots, n_u$, 的不同分量 $z_{u,i}(k)$ 彼此正交, 其标准差是 $\sqrt{\lambda_{u,i}}$. 因此在变换空间中聚类 C_u 可以用一个中心在坐标原点三条边分别平行于三个坐标轴的三维矩形来表示, 该三维矩形的边长分别是 $4\sqrt{\lambda_{u,i}}$, $i = 1, 2, 3$. 根据等式 (1) 进行如下的反变换

$$\mathbf{z}_u(k) = V_u \mathbf{w}_u(k) + \bar{\mathbf{z}}_u$$

能够在原来的颜色空间中确定表征聚类 C_u 的三维矩形. 表征两个相邻聚类的三维矩形可能相交, 相交区域可以通过计算机图形学的知识来确定. 该相交区域的像素属于哪一个聚类通过最近邻法则确定.

图 1 显示了两种典型目标颜色的分布和颜色空间的自适应剖分. 图 1(a) 和图 1(c) 分别是人脸和行人的参考图像, 图 1(b) 和图 1(d) 分别是对应的颜色分布及颜色空间的剖分, 其中每一个聚类用一个三维矩形表示, 属于同一聚类的像素用同一颜色表示. 从图中可以看到这两种目标的颜色紧凑地分布在颜色空间的很小的一部分子空间中, 根据聚类结果自适应地剖分颜色空间能够更有效、更准确地表达目标的颜色分布.

2.2 颜色模型及相似性度量

基于上述自适应剖分的颜色子空间, 给定由 N 个像素组成的目标的参考图像 $\mathbf{z}(\mathbf{x}_i) = [z_1(\mathbf{x}_i) \ z_2(\mathbf{x}_i) \ z_3(\mathbf{x}_i)]^T, i = 1, \dots, N$, 其中 \mathbf{x}_i 是第 i 个像素的空间坐标, 目标的模型表达为 $\mathbf{p} = \{p_u\}, u = 1, \dots, d$, 其中 p_u 按如下方式定义

$$p_u(\mathbf{z}(\mathbf{x}); b_u, \boldsymbol{\mu}_u, R_u) = b_u G(\boldsymbol{\mu}_u, R_u) \quad (2)$$

在 (2) 式中, $G(\boldsymbol{\mu}_u, \Sigma_u)$ 是均值向量为 $\boldsymbol{\mu}_u$ 、协方差矩阵为 R_u 的高斯分布, $b_u, \boldsymbol{\mu}_u, R_u$ 具有如下形式

$$\begin{aligned} b_u &= \frac{\sum_{i=1}^N k(\|\mathbf{x}_i\|^2) \delta_{iu}}{\sum_{i=1}^N k(\|\mathbf{x}_i\|^2)} \\ \boldsymbol{\mu}_u &= \frac{1}{n_u} \sum_{i=1}^N \mathbf{z}(\mathbf{x}_i) \delta_{iu} \\ R_u &= \frac{1}{n_u} \sum_{i=1}^N (\mathbf{z}(\mathbf{x}_i) - \boldsymbol{\mu}_u)(\mathbf{z}(\mathbf{x}_i) - \boldsymbol{\mu}_u)^T \delta_{iu} \end{aligned} \quad (3)$$

其中 δ_{iu} 是 Kronecker 函数, 即如果 $\mathbf{z}(\mathbf{x}_i)$ 属于第 u 个区间则 δ_{iu} 等于 1, 否则为 0; $n_u = \sum_{i=1}^N \delta_{iu}$ 是第 u 个直方图区间的像素总数.

考虑一个由 N' 个像素组成的中心坐标为 \mathbf{y} 的候选图像区域的颜色分布 $\mathbf{q}(\mathbf{y}) = \{q_u(\mathbf{y})\}, u = 1, \dots, d$, 其中分量 $q_u(\mathbf{y})$ 具有如下形式

$$q_u(\mathbf{z}(\mathbf{x}); b'_u(\mathbf{y}), \boldsymbol{\mu}'_u, R'_u) = b'_u(\mathbf{y}) G(\boldsymbol{\mu}'_u, R'_u) \quad (4)$$

在 (4) 式中, $\boldsymbol{\mu}'_u$ 和 R'_u 具有与式 (3) 所示的相似的形式, 而 $b'_u(\mathbf{y})$ 为

$$b'_u(\mathbf{y}) = \frac{\sum_{i=1}^{N_h} k(\|\frac{\mathbf{x}_i - \mathbf{y}}{h}\|^2) \delta_{iu}}{\sum_{i=1}^{N_h} k(\|\frac{\mathbf{x}_i - \mathbf{y}}{h}\|^2)}$$

两个分量分布 $p_u(\mathbf{z}(\mathbf{x}); n_u, \boldsymbol{\mu}_u, R_u)$ 和 $q_u(\mathbf{y}) \triangleq q_u(\mathbf{z}(\mathbf{x}); n'_u(\mathbf{y}), \boldsymbol{\mu}'_u, R'_u)$ 之间的相似性根据

Bhattacharrya 距离来度量 $\rho(p_u, q_u(\mathbf{y})) = \int p_u^{\frac{1}{2}} q_u(\mathbf{y})^{\frac{1}{2}} d\mathbf{z}(\mathbf{x})$. 通过积分得到

$$\rho(p_u, q_u(\mathbf{y})) = c_u (b_u b'_u)^{\frac{1}{2}} \quad (5)$$

其中 c_u 为

$$\begin{aligned} c_u &= \left(\frac{2|R_u|^{\frac{1}{2}}|R'_u|^{\frac{1}{2}}}{|R_u + R'_u|} \right)^{\frac{1}{2}} \\ &\exp\left(-\frac{1}{4}(\boldsymbol{\mu}_u - \boldsymbol{\mu}'_u)^T (R_u + R'_u)^{-1} (\boldsymbol{\mu}_u - \boldsymbol{\mu}'_u)\right) \end{aligned} \quad (6)$$

因此 $\mathbf{p} = \{p_u\}$ 和 $\mathbf{q}(\mathbf{y}) = \{q_u(\mathbf{y})\}$ 之间的相似性定义如下

$$\rho(\mathbf{p}, \mathbf{q}(\mathbf{y})) = \sum_{u=1}^d \rho(p_u, q_u(\mathbf{y})) \quad (7)$$

直方图建模了图像某一像素 $p(\mathbf{z}(\mathbf{x}))$ 属于第 u 个区间的概率 p_u . 对不同的直方图定义的比较有助于理解本文提出的颜色模型

$$p_u = \begin{cases} n_u/N & \text{传统直方图} \\ b_u & \text{加权直方图} \\ b_u G(\boldsymbol{\mu}_u, \Sigma_u) & \text{本文提出的直方图} \end{cases}$$

传统直方图仅仅统计落入直方图区间的像素的个数, 而加权直方图进一步考虑了像素与目标中心的距离, 远离目标中心的像素对直方图的贡献较小. 这两种直方图都未考虑每一个直方图区间内的颜色分布情况. 在本文中, 直方图的每一个区间对应一个聚类, 在加权直方图的基础上进一步用高斯分布建模每一个聚类内的颜色分布.

2.3 改进的 Mean Shift 跟踪算法

设在前一帧中目标的中心位于 \mathbf{y}_0 , 在当前帧中, Mean Shift 算法基于梯度下降方法寻找式 (7) 的极值. 将等式 (7) 在 $\mathbf{q}(\mathbf{y}_0)$ 进行一阶泰勒展开, 相对于 \mathbf{y} 求偏导数并令其为零, 得到如下等式

$$\mathbf{y}_1 = \frac{\sum_{i=1}^{N_h} \omega_i g(\|\frac{\mathbf{y}_0 - \mathbf{x}_i}{h}\|^2) \mathbf{x}_i}{\sum_{i=1}^{N_h} \omega_i g(\|\frac{\mathbf{y}_0 - \mathbf{x}_i}{h}\|^2)} \quad (8)$$

其中 $g(\cdot)$ 是加权函数 $k(\cdot)$ 的导数, ω_i 由下式给出

$$\omega_i = \sum_{u=1}^d c_u \sqrt{b_u/b'_u(\mathbf{y}_0)} \delta_{iu} \quad (9)$$

令 $\mathbf{y}_0 = \mathbf{y}_1$, 用式 (8) 进行迭代计算, 直到 $\|\mathbf{y}_1 - \mathbf{y}_0\|$ 小于某一阈值或达到了最大迭代次数. 为了适应目

标的尺度变化, 加权函数的半径 h 以 $\pm 10\%$ 变化, 在新的尺度上执行跟踪算法直到收敛, 最后选择使 (7) 式取极大值的尺度.

当 $k(\cdot)$ 取 Epanechnikov 函数时^[4], 式 (8) 变为

$$\mathbf{y}_1 = \frac{\sum_{i=1}^{N_h} \omega_i \mathbf{x}_i}{\sum_{i=1}^{N_h} \omega_i} \quad (10)$$

注意到在经典 MS 算法中没有 c_u 这一项.

2.4 算法的复杂性分析

经典的 MS 算法未考虑目标颜色在颜色空间的实际分布情况 (即目标颜色通常紧凑地分布在颜色空间的一部分区域中), 将整个颜色空间均匀分割为 $2^{12}=16 \times 16 \times 16$ (或 $2^{15}=32 \times 32 \times 32$) 个子区间. 这种均匀分割造成大量空直方图区间 (即函数值为 0 的直方图区间). 尽管这些空直方图区间对颜色概率密度的估计没有贡献, 经典 MS 算法仍然不得不为它们分配内存空间, 这样会引起存储空间的浪费. 在人脸跟踪和行人跟踪的例子中, 直方图区间的函数值为 0 的区间个数分别为 4000 和 4060, 占直方图区间总数的百分比分别为 $4000/2^{12} \approx 97.7\%$ 和 $4060/2^{12} \approx 99.1\%$. 如果函数值用双精度的浮点数 (8 字节) 表示, 则函数值为 0 的直方图区间占用的内存空间分别为 32,000 Byte 和 32,480 Byte. 同时在利用 Bhattacharry 系数计算相似性时也要为这些函数值为 0 的直方图区间的计算付出代价.

而本文提出的模型自适应地剖分目标的颜色空间, 能够用区间数目很少的直方图准确地表达目标的颜色分布 (每个直方图区间对应一个聚类), 不存在这种由于函数值为 0 的直方图区间而引起的资源消耗. 尽管如此, 与经典的 MS 算法相比, 改进方法由于引入了多维高斯模型, 需要额外地计算均值向量、协方差矩阵和计算与多维高斯函数相关的量 (包括矩阵行列式计算、向量和矩阵乘法运算以及指数运算). 由于这些额外的计算量, 改进算法的平均跟踪时间要高于经典的 MS 算法. 我们对两种算法的计算复杂性分析如下.

在经典的 MS 算法中, 令 N 是为了适应目标尺寸变化而搜索的尺度数目, N_r 为梯度下降算法的平均迭代次数, N_h 为目标所包含的像素数目, N_b 为经典 MS 算法的直方图数目, c_h 为加权直方图的计算代价, c_w 是在迭代过程中为每一像素付出的计算代价, c_s 为计算相似性的代价, 则经典算法的总计算代价为

$$C_m = N(N_r(c_h + N_h c_w) + N_b c_s) \quad (11)$$

在改进算法中, 由于引入了多维高斯分布建模每一聚类的颜色分布从而相应地增加了计算量. 令 c_{g1} 为计算均值向量和协方差矩阵的代价, c_{g2} 是由于引入高斯模型在迭代过程中为每一像素付出的额外计算代价 (见式 (10) 和 (11)), c_{g3} 是由于引入高斯模型在计算相似性时需要额外付出的代价 (见式 (7)), 则改进算法的总的计算代价为

$$C_i = N(N_r(c_h + c_{g1} + d(c_w + c_{g2})) + d(c_s + c_{g3})) \quad (12)$$

经典 MS 算法和改进算法计算代价的差为

$$(C_m - C_i)/N = N_r(N_h - d)c_w + (N_b - d)c_s - (N_r c_{g1} + d N_r c_{g2} + d c_{g3}) \quad (13)$$

注意到一般地 $d < 20$, 则 $N_b = 2^{12} \gg d$, 目标所包含的像素数 (即目标的面积) ($N_h > 20 * 20 = 400$) $\gg d$, 则上式变为

$$(C_m - C_i)/N \approx N_r N_h c_w + N_b c_s - (N_r c_{g1} + d N_r c_{g2} + d c_{g3}) \quad (14)$$

由于引入多维高斯模型而增加的计算代价 ($N_r c_{g1} + d N_r c_{g2} + d c_{g3}$) 一般要大于 $N_r N_h c_w + N_b c_s$, 因此改进算法的计算代价要大于经典 MS 算法.

3 实验

程序在 Visual C++ 6.0 编程环境下, 主频为 1.8GHz、内存为 1.5GB 的笔记本电脑 IBM T42 2373KHC 上调试通过. 在经典的 Mean Shift 算法的实现中采用了 $16 \times 16 \times 16$ 的直方图. 程序使用 RGB 颜色空间, 初始化是在第一帧中用手工完成的, 真值 (Ground truth) 也是用手工标注得到的.

3.1 人脸跟踪

人脸图像序列是在典型的办公室环境中采集得到的^[11], 包括 752 帧, 其中每一幅图像的尺寸是 256×192 像素. 从第 150 到 165 帧测试者转过身去背对摄像机, 在接下来的 100 帧中光照变化比较大. 从第 270 帧到 360 帧被跟踪的人脸又变得不可见. 当人脸不可见或光照变化较大时, 两种跟踪算法的跟踪误差都变得比较大但并没有失锁. 从第 630 帧到第 710 帧, 另外一位测试者逐渐遮挡了被跟踪的人脸.

图 2 显示了经典 Mean Shift 算法和改进算法的跟踪误差, 图 2(a)、2(b) 和 2(c) 分别显示在每一帧中 X 坐标误差曲线、 Y 坐标误差曲线和尺度误差曲线, 其中实线表示改进算法的误差曲线, 虚线表示经典的 Mean Shift 误差曲线. 从图中可见几乎在每一帧中改进算法的误差都要小. 平均跟踪误差和跟

踪时间如表 1 所示, 改进算法的跟踪时间是 8.5ms, 略多于经典算法 7.9ms.

表 1 人脸序列中跟踪误差 (均值 \pm 标准差) 和平均跟踪时间

Table 1 Tracking errors (mean \pm std) and average tracking time in face sequence

	X 误差 (pixels)	Y 误差 (pixels)	尺度误差 (%)	跟踪时间 (ms)
经典 MS 算法	7.0 \pm 8.4	9.5 \pm 11.7	0.17 \pm 0.19	7.9
改进算法	6.0 \pm 6.1	9.3 \pm 10.2	0.11 \pm 0.14	8.5

两种算法的典型跟踪结果分别如图 3 和图 4 所示, 其中从左至右、从上到下依次为第 1, 260, 350, 375, 500, 685, 700 和 750 帧图像.

3.2 行人跟踪

行人图像序列的采集环境是候车大厅^[12], 共 330 帧图像, 图像尺寸为 720×576 像素. 在图像序列中一位女士向候车大厅的出口走去, 从第 218 帧到 245 帧发生了遮挡, 从第 250 帧开始光照条件发生了变化.

经典 Mean Shift 算法和改进算法在每一帧中的 X 坐标、Y 坐标和尺度误差分别如图 5(a)、5(b) 和 5(c) 所示, 平均跟踪误差和跟踪时间如表 2 所示. 可以看到改进算法的性能明显优于经典 Mean Shift 的算法. 改进算法的跟踪时间略高于经典算法.

表 2 行人序列中跟踪误差 (均值 \pm 标准差) 和平均跟踪时间

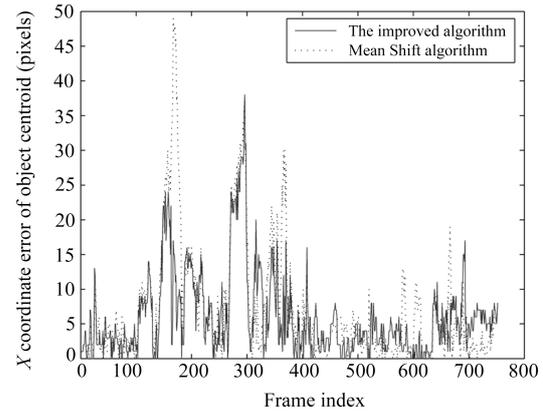
Table 2 Tracking errors (mean \pm std) and average tracking time in pedestrian sequence

	X 误差 (pixels)	Y 误差 (pixels)	尺度误差 (%)	跟踪时间 (ms)
经典 MS 算法	8.3 \pm 3.1	10.2 \pm 6.0	0.07 \pm 0.05	4.8
改进算法	5.9 \pm 3.4	7.0 \pm 5.0	0.04 \pm 0.04	5.4

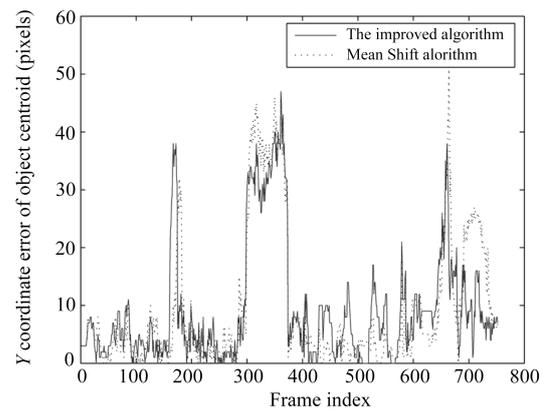
图 6 和图 7 (见 353 页) 分别给出了两种算法的一些典型跟踪结果, 其中从左至右、从上至下依次为第 100, 215, 240 和 290 帧图像.

3.3 光照变化序列中的人脸跟踪

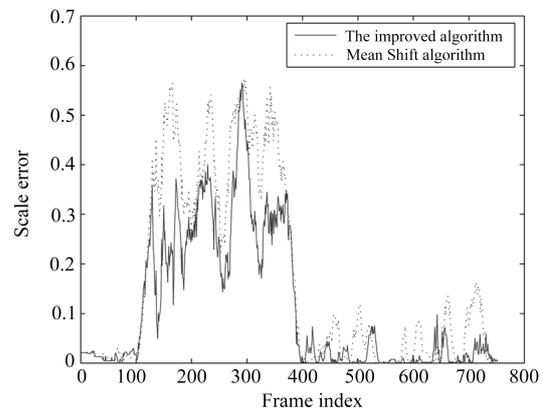
这一图像序列也是在办公室环境中采集的^[11], 包括 70 帧, 其中每一幅图像的尺寸是 256×192 像素. 从大约 20 帧起, 由于灯光的影响和摄像机的运动的影响, 测试者的人脸表观颜色发生了较大的变化. 从图 8 (见 353 页) 和表 3 所示的跟踪误差可见, 当光照变化较大时改进算法优于经典的 Mean Shift 算法. 两种算法的典型跟踪结果分别如图 9 和图 10 (见 353 页) 所示, 其中从左至右依次为第 10, 30, 50 和 70 帧图像.



(a)



(b)



(c)

图 2 人脸序列中经典 MS 算法 (虚线) 和改进算法 (实线) 跟踪误差比较

Fig. 2 Comparison of tracking errors in face sequence between using the classical MS algorithm (dotted) and the improved algorithm (solid)



图 3 改进算法对人脸的跟踪结果

Fig. 3 Some of the typical tracking results in face sequence using the proposed algorithm



图 4 经典 Mean Shift 算法对人脸的跟踪结果

Fig. 4 Some of the typical tracking results in face sequence using the classical MS algorithm

表 3 光照变化的序列中人脸跟踪误差 (均值 ± 标准差) 和平均跟踪时间

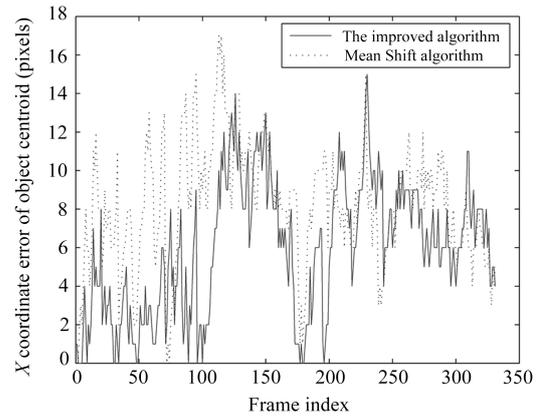
Table 3 Tracking errors (mean±std) and average tracking time in sequence in which lighting changes

	X 误差 (pixels)	Y 误差 (pixels)	尺度误差 (%)	跟踪时间 (ms)
经典 MS 算法	7.4±6.2	20.9±13.5	0.07±0.1	13
改进算法	5.2±3.3	9.7±6.4	0.04±0.03	18

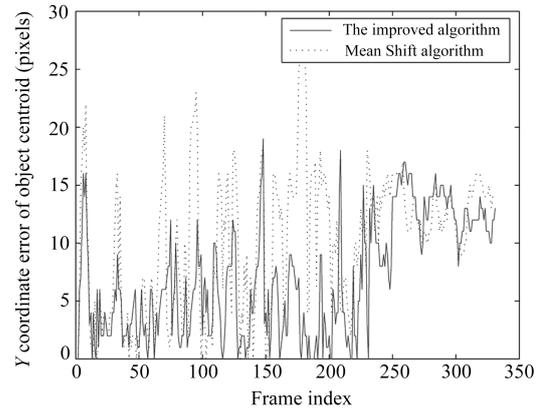
4 结论

本文提出了一种改进的 Mean Shift 算法用于目标跟踪. 本文的贡献在于: 在聚类分析的基础上自适应地划分目标的颜色空间, 根据矩阵分解和正交变换获得了对应不同聚类的子空间; 定义了一种新的颜色模型并推导了颜色模型之间的相似性测度; 最后将新的颜色模型应用于 Mean Shift 算法中. 实验表明, 改进算法具有更好的跟踪性能, 同时与经典算法的跟踪时间大致相同.

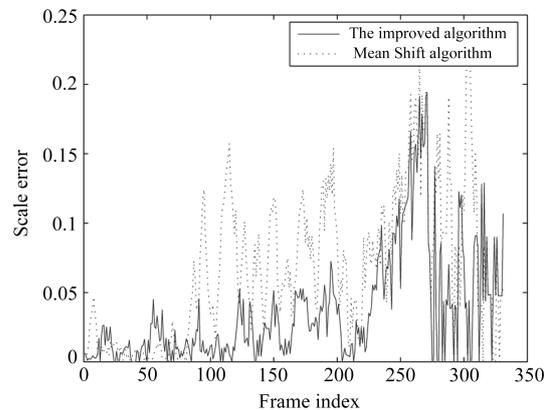
与 Comaniciu 等的经典论文^[4]及众多的改进算法^[5~8]相同, 我们假定在跟踪过程中目标的颜色模型保持不变. 因此我们的算法对目标颜色空间的自适应剖分只在首帧即初始化时进行, 聚类数是固定的. 尽管这种假设适合于多数应用, 然而在复杂的条件下, 环境的变化可能导致目标的表现颜色发生较大变化, 此时如果目标的颜色模型不进行更新可



(a)



(b)



(c)

图 5 行人序列中经典 Mean Shift 算法 (虚线) 和改进算法 (实线) 跟踪误差比较

Fig. 5 Comparison of tracking errors in pedestrian sequence between using the classical MS algorithm (dotted) and the improved algorithm (solid)

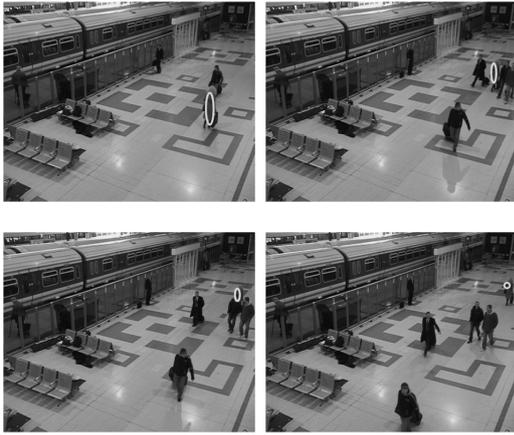


图 6 改进算法对行人的跟踪结果

Fig. 6 Some of the typical tracking results in pedestrian sequence using the proposed algorithm

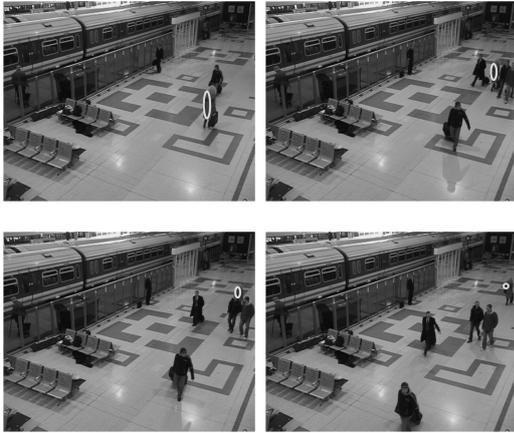
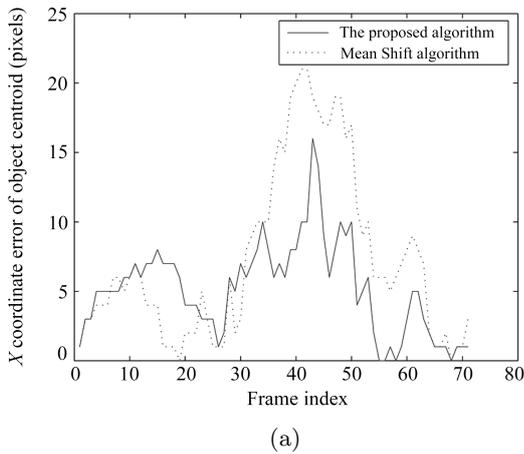
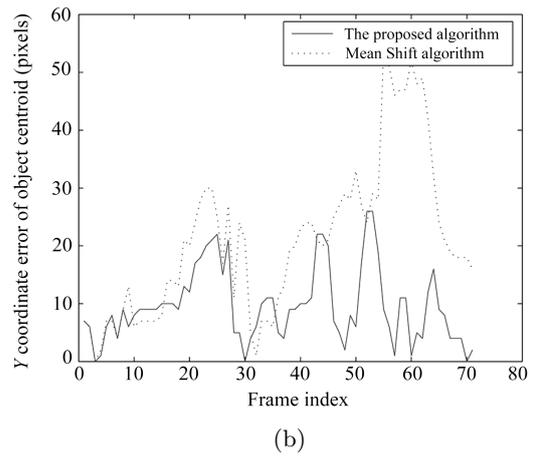


图 7 经典 Mean Shift 算法对行人的跟踪结果

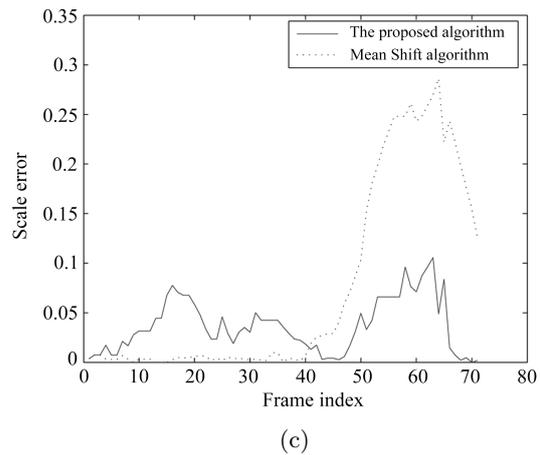
Fig. 7 Some of the typical tracking results in pedestrian sequence using the classical MS algorithm



(a)



(b)



(c)

图 8 光照变化的序列中经典 Mean Shift 算法 (虚线) 和改进算法 (实线) 跟踪误差比较

Fig. 8 Comparison of tracking errors in sequence in which lighting changes between using the classical MS algorithm (dotted) and the improved algorithm (solid)



图 9 改进算法对光照变化序列中人脸的跟踪结果

Fig. 9 Some of the typical tracking results in sequence where lighting changes using the improved algorithm



图 10 经典 Mean Shift 算法对光照变化序列中人脸的跟踪结果

Fig. 10 Some of the typical tracking results in sequence where lighting changes using the classical MS algorithm

能会使跟踪误差增大甚至使跟踪失败. 因此我们将来的工作是着重解决如何在线更新目标的颜色模型的问题. 将该颜色模型与粒子滤波器^[13]结合以获得更为鲁棒的跟踪器也是我们感兴趣的研究内容.

致谢

诚挚地感谢本文审稿专家中肯的审稿意见.

References

- 1 Wren C R, Azarbayejani A, Darrell T, Pentland A P. Pfunder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997, **19**(7): 780~785
- 2 Birchfield S T. Elliptical head tracking using intensity gradients and color histograms. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 1998. 232~237
- 3 Stauffer C, Grimson W E. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, **22**(8): 747~757
- 4 Comaniciu D, Ramesh V, Meer P. Real-time tracking of non-rigid objects using mean shift. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2000. 142~149
- 5 Collins R T. Mean-shift blob tracking through scale space. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2003. 234~241
- 6 Birchfield S T, Rangarajan S. Spatiograms versus histograms for region-based tracking. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2005. 1158~1163
- 7 Zhao Q, Tao H. Object tracking using color correlogram. In: Proceedings of IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. IEEE, 2005. 263~270
- 8 Yang C, Duraiswami R, Davis L. Efficient mean-shift tracking via a new similarity measure. In: Proceedings of IEEE

Conference on Computer Vision and Pattern Recognition. IEEE, 2005. 176~183

- 9 Comaniciu D, Meer P. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, **24**: 603~619
- 10 Zhang Xian-Da. *Matrix Analysis and Applications*. Tsinghua University Press: 2004. 505~513
(张贤达著. 矩阵分析与应用. 北京: 清华大学出版社, 2004. 505~513)
- 11 Test Image Sequences for Face Tracking by Stan Birchfield [Online], available: http://vision.stanford.edu/birch/head_tracker/seq
- 12 Iscacs C. Pets2006 Dataset S7 Camera 4 [Online], available: <http://ftp.cs.rdg.ac.uk/PETS2006/S3-T70-A.ZIP>, June 1, 2006
- 13 Li P, Zhang T, Pece A E. Visual contour tracking based on particle filters. *Image and Vision Computing*, 2003, **21**(1): 111~123



李培华 2003年1月获哈尔滨工业大学博士学位, 2003年8月至2004年9月在法国 INRIA 从事博士后研究, 现为黑龙江大学计算机学院副教授. 博士论文获2005年度全国百篇优秀博士论文提名. 主要研究方向为基于概率和统计及机器学习的目标检测, 跟踪和识别. E-mail: peihualijia@vip.sina.com

(**LI Pei-Hua** Received his Ph.D. degree from Department of Computer Science and Engineering, Harbin Institute of Technology in January 2003. From August 2003 to September 2004, he served as a postdoctoral fellow at INRIA, France. He is currently an associate professor of Heilongjiang University. He received Honorable Nomination of Hundred National Excellent Dissertations award in P. R. China, 2005. His main research interest covers object detection, tracking and recognition based on probability, statistics and machine learning.)