

Fragments-Based Tracking with Multiple Kernels Fusion

Yajun Zhang, Hongli Xu

School of Computer and Information Technology, Beijing Jiaotong University, Beijing
Email: yjzhang12@gmail.com

Received: Sep. 17th, 2014; revised: Sep. 30th, 2014; accepted: Oct. 6th, 2014

Copyright © 2014 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Objective: Accuracy and robustness are the main challenges of the visual tracking, especially in the case of occlusion and target deformation. Target tracking based on fragments will be able to keep the spatial information of the target, and based on this, a fragments-based tracking algorithm with multiple kernels fusion is proposed in this paper. **Method:** Vertical projection method is used to get proper fragments in the algorithm, and for each corresponding fragment, it selects a plurality of different locations within the target area to build several kernel function weighted histograms, taking the Bhattacharyya coefficient as the similarity measurement between the target template and the candidate template, and making use of the mean shift iteration to determine the final position of the target. In the process of tracking, it takes advantage of the back-projection of the components to distinguish deformation or occlusion, and makes a real-time updates for the target template and fragments weight. **Result:** According to the results, which are obtained from several testing of video sequence, the method is almost not influenced by illumination, and can still achieve good tracking even in a large area of occlusion. **Conclusion:** The proposed algorithm, by combining fragments and multiple kernels to tracking, is not only insensitive to illumination, but also has a good performance in dealing with a large area of occlusion, which is beneficial to the research of next stage.

Keywords

Target Tracking, Target Occlusion, Multiple Kernels, Bhattacharyya Coefficient, Similarity Measurement

融合多核的目标分块跟踪

张亚军, 许宏丽

北京交通大学, 计算机与信息技术学院, 北京
Email: yjzhang12@gmail.com

收稿日期: 2014年9月17日; 修回日期: 2014年9月30日; 录用日期: 2014年10月6日

摘要

目的: 准确性和鲁棒性是视觉跟踪的主要挑战, 尤其是在遮挡和目标形变的情况下。对目标进行分块跟踪能够保持目标的空间信息, 基于此, 本文提出一种融合多核的目标分块跟踪方法。**方法:** 算法采用垂直投影法将目标分成若干合适的子块, 针对相应子块, 选择目标区域内的多个不同位置构建多个核函数加权直方图, 以Bhattacharyya系数表征目标模版和候选目标模版之间的相似性度量, 并进行均值迭代确定目标最终位置。在跟踪过程中, 利用分量的反投影运算区分目标发生形变或遮挡, 并对目标模版和子块权重进行实时更新。**结果:** 对多个视频序列进行了实验测试, 实验表明, 该方法几乎不受光照影响, 并且在目标大面积遮挡的情况下仍能实现很好的跟踪。**结论:** 通过将分块和多核融合结合起来进行跟踪, 不仅对光照不敏感, 而且在处理目标大面积遮挡方面鲁棒性很强, 利于后续论题的研究。

关键词

目标跟踪, 目标遮挡, 多核, Bhattacharyya系数, 相似性度量

1. 引言

运动目标跟踪是计算机视觉领域的一项重要研究课题, 其中对形变、光照和遮挡等问题的有效处理一直是影响跟踪效果的关键。近年来研究者已经提出了许多有效的目标跟踪算法, 而 Mean Shift 跟踪算法 [1] 以其低复杂性, 鲁棒性和不变性的优点被广泛应用于目标跟踪领域。但是, 由于其使用颜色和直方图特征进行目标建模, 导致目标空间信息大量丢失, 且受到遮挡的限制使得跟踪准确性降低。

为此, 研究者提出了许多方法。针对空间信息缺失问题, 文献 [2] [3] 使用一种新的相似性度量来代替传统的巴氏度量, 通过将空间均值和像素位置的协方差等加入到相似性度量中来融合空间信息。文献 [4] 引入了多内核协作机制, 使用预先学习子空间模型来复现复杂运动。在处理遮挡问题方面, 基于分块的方法 [5]-[10] 被广泛使用。文献 [5] 提出了分块跟踪方法, 通过将目标分成多个片段来解释遮挡和目标姿势的变化, 但由于使用穷举搜索来反复选择分块, 增加了计算复杂度。文献 [6] [7] 使用多个独立的均值漂移跟踪器进行随机选择或预先指定分块, 并跟踪每一个跟踪器的输出组合来获得最终跟踪结果。

但是, 算法没有考虑权重更新问题, 且仅利用相似度最大的子块位置来定位目标, 容易发生跟踪漂移。文献 [8] 采用一种基于 GrabCut 算法的分块方法, 并通过将多个子块进行融合机制来实现鲁棒跟踪。但是, 算法未考虑在遮挡情况下子块权重引起的跟漂移问题。文献 [9] 将每一个子块作为核跟踪器, 使用均值漂移对各块相似性的加权和进行推导, 且在跟踪过程中, 实时更新目标模版和子块权重, 很好的解决了漂移问题, 但该方法不适宜在目标与背景颜色相近的情况下使用。文献 [10] 选择目标的颜色特征, 通过将分块、均值漂移和卡尔曼滤波融合来实现遮挡情况下的多目标跟踪, 但是无法自适应的处理目标尺度和方向问题, 而且也没有考虑光照的变化。

本文提出一种融合多核的目标分块跟踪方法, 该方法采用基于颜色形似度的方法进行前景提取, 以垂直投影法来进行目标分块。对相应子块, 选择目标区域内的多个不同位置构造多个非对称核函数, 分别进行均值漂移跟踪。利用分量的反投影运算来检测遮挡是否发生, 并在跟踪过程实时更新目标模版和

子块权重。通过对多个视频序列进行实验，验证了本文算法的有效性。

2. 融合多核的目标分块跟踪框架

本文提出的融合多核的目标分块跟踪算法大致可以分为两部分：基于分块的多核跟踪以及目标模型和子块权重的实时更新。

目标跟踪部分：该部分包括目标模型及候选目标模型的背景建模，相似性度量及目标定位。首先，在初始选择跟踪目标后，采用垂直投影法将目标分成若干不重叠的子块；然后，利用每个子模型集合来描述目标模型及候选目标模型，通过线性加权将目标与候选目标之间对应的子模型的相似度组合起来表示目标与候选目标之间总的相似性度量。最后，利用改进的 **Mean Shift** 算法进行目标定位，得到最终跟踪结果。

目标模型及权重更新部分：对数似然函数可以有效区分前景像素和背景像素，因此可以根据似然值的大小更新子块权重。利用分量的反投影运算判断遮挡，根据目标的形变度及目标模型和候选模型断的相似度来判断是否更新模版。

目标跟踪要考虑实时性，而目标在跟踪过程中，可能只是部分或者一段时间内处于被遮挡状态，而不是整个过程都被遮挡，因此可以通过检测目标是否发生遮挡来选择跟踪算法。首先，输入图像帧，初始化感兴趣区域(*ROI*)，定位我们要跟踪的目标；然后，利用 4.1 节遮挡检测方法来进行遮挡检测，若没有发生遮挡，则采用基于分块的 **Mean Shift** 跟踪，否则，采用本文提出的融合多核的分块跟踪。

该算法的优势在于：一方面，采用了多个核函数，这样目标中的各个像素在不同的直方图中的权重是不同的，当目标被部分遮挡时，总有某个或某些直方图受到的影响较小；另一方面，融合了分块思想，在初始选择了跟踪目标后，将目标分成若干不重叠的子块，对每一子块都采用多个核函数来构造加权直方图，极大的提高了算法的鲁棒性。

3. 融合多核的目标跟踪算法

3.1. 基于分块的核跟踪

给定一组采样点，**Mean Shift** 算法就是找到其概率密度函数的局部最大值，也就是说，使得目标模型和候选模型之间的巴氏距离最小。在核跟踪中，首先要提取图像的某种特征，将目标模型和候选区域以核函数加权直方图的形式表示，定义目标模型及候选目标模型[11]如下：

$$\text{目标模型: } Q = \{\hat{q}^{(l)}\}_{l=1, \dots, L}$$

$$\text{候选模型: } P = \{\hat{p}^{(l)}(y)\}_{l=1, \dots, L}$$

其中， L 是分块的数目， $\hat{q}^{(l)}$ 和 $\hat{p}^{(l)}(y)$ 分别表示第 l 个子块的目标模型和位于 y 处的第 l 个子块的目标候选模型。

假设目标包含 N 个像素 $\{x_i\}_{i=1, \dots, N}$ ，特征值量化等级为 m ，在位置 $\bar{z}^{(l)}$ 处，目标模版 T 的第 l 个分块的

模型表示如下：

$$\hat{q}_u^{(l)} = C_q^{(l)} \sum_{i=1}^N k \left(\left\| \frac{x_i^{(l)} - \bar{z}^{(l)}}{h^{(l)}} \right\|^2 \right) \delta \left[b(x_i^{(l)}) - u \right] \quad (1)$$

同样的，在位置 y 处，候选目标模型表示如下：

$$\hat{p}_u^{(l)}(y) = C_p^{(l)} \sum_{i=1}^{N^{(l)}} k \left(\left\| \frac{x_i^{(l)} - y}{h^{(l)}} \right\|^2 \right) \delta [b(x_i^{(l)}) - u] \quad (2)$$

其中, $h^{(l)}$ 表示第 l 个子块核函数的带宽, δ 是 Kronecher delta 函数, 用于判断像素 x_i 的特征值是否属于第 u 个 bin, $k(\cdot)$ 是核函数, $b(\cdot)$ 是像素到相应 bin 值的映射, $C_q^{(l)}$ 和 $C_p^{(l)}$ 是归一化常数, 使得 $\sum_{u=1}^m \hat{q}_u^{(l)} = 1$, $\sum_{u=1}^m p_u^{(l)} = 1$ 。

在获得目标和候选目标的表示模型后, 需要度量目标与候选目标之间的相似性。目标模型和候选目标模型之间总的相似性度量 $\hat{\rho}(y)$ 是各子模型之间相似性 $\hat{\rho}^{(l)}$ 的加权总和, 如下:

$$\hat{\rho}(y) = \lambda^{(1)} \hat{\rho}^{(1)} + \lambda^{(2)} \hat{\rho}^{(2)} + \dots + \lambda^{(L)} \hat{\rho}^{(L)} = \sum_{l=1}^L \lambda^{(l)} \hat{\rho}^{(l)} \quad (3)$$

其中, $\lambda^{(l)}$ 是权重, 满足 $\sum_{l=1}^L \lambda^{(l)} = 1$ 。

子块的相似性函数 $\hat{\rho}^{(l)}$ 有:

$$\hat{\rho}^{(l)} \equiv \rho(\hat{p}^{(l)}(y), \hat{q}^{(l)}) = \sum_{u=1}^m \sqrt{\hat{q}_u^{(l)}(y) \hat{p}_u^{(l)}(y)} \quad (4)$$

跟踪的目的就是在每帧中查找使得式(4)最大的候选区域位置。假设目标的初始位置为 \hat{y}_0 , 将式(4)泰勒展开, 可得:

$$\hat{\rho}^{(l)} \approx \frac{1}{2} \rho(\hat{p}^{(l)}(y_0), \hat{q}^{(l)}) + \frac{1}{2} \sum_{u=1}^m \sqrt{\frac{\hat{q}_u^{(l)}}{\hat{p}_u^{(l)}(y_0)}} \hat{p}_u^{(l)}(y) \quad (5)$$

结合式(2)、(3)、(5)得到:

$$\hat{\rho}(y) \approx \frac{1}{2} \sum_{l=1}^L \lambda^{(l)} \rho(\hat{p}^{(l)}(y_0), \hat{q}^{(l)}) + \frac{1}{2} \sum_{l=1}^L C_p^{(l)} \lambda^{(l)} \omega_i^{(l)} k \left(\left\| \frac{x_i^{(l)} - y}{h^{(l)}} \right\|^2 \right) \quad (6)$$

其中,

$$\omega_i^{(l)} = \sum_{u=1}^m \sqrt{\frac{\hat{q}_u^{(l)}}{\hat{p}_u^{(l)}(y_0)}} \delta [b(x_i^{(l)}) - u] \quad (7)$$

式(6)中, 第一项为常项, 可通过最大化第二项使其最大化, 目标从初始位置 \hat{y}_0 到新位置 \hat{y}_1 的位移:

$$\hat{y}_1 = \frac{\sum_{l=1}^L \sum_{i=1}^{N^{(l)}} \lambda^{(l)} \omega_i^{(l)} (x_i^{(l)} - y_0) g \left(\left\| \frac{x_i^{(l)} - y_0}{h^{(l)}} \right\|^2 \right)}{\sum_{l=1}^L \sum_{i=1}^{N^{(l)}} \lambda^{(l)} \omega_i^{(l)} g \left(\left\| \frac{x_i^{(l)} - y_0}{h^{(l)}} \right\|^2 \right)} + \hat{y}_0 \quad (8)$$

采用 Mean Shift [12] 算法, 得到以初始位置 \hat{y}_0 为起始的位移矢量 $\Delta \hat{y}_0 (\hat{y}_1 - \hat{y}_0)$:

$$\Delta \hat{y}_0 = \frac{\sum_{i=1}^{N^{(l)}} x_i^{(l)} \omega_i^{(l)} g \left(\left\| \frac{x_i^{(l)} - y_0}{h^{(l)}} \right\|^2 \right)}{\sum_{i=1}^{N^{(l)}} \omega_i^{(l)} g \left(\left\| \frac{x_i^{(l)} - y_0}{h^{(l)}} \right\|^2 \right)} \quad (9)$$

其中, $g(x) = -k(x)$ 。Epanechnikov [1]核函数的剖面函数为:

$$k_g(x) = \begin{cases} 1-x, & 0 \leq x \leq 1 \\ 0, & x > 1 \end{cases} \quad (10)$$

由式(10)可知, Epanechnikov 核函数的剖面函数的导数为常数, 式(9)可简化为:

$$\Delta \hat{y}_0 = \frac{\sum_{i=1}^{N^{(l)}} x_i^{(l)} \omega_i^{(l)}}{\sum_{i=1}^{N^{(l)}} \omega_i^{(l)}} \quad (11)$$

3.2. 前景提取

当跟踪目标发生部分遮挡时, 如果能够将前景区域提取出来, 则可以更加有效地进行跟踪。本文采用基于颜色相似度的方法, 如图 1, 选择感兴趣区域(ROI)作为我们跟踪的重点, 而背景区域就在感兴趣区域内定义。此处设置目标矩形和背景矩形间的边缘值为 $0.2\max(w, h)$, 其中, w 和 h 分别表示目标矩形的宽和高。定义每个像素点的似然函数 $L(x_i)$

如下:

$$L(x_i) = \log \frac{\max\{h_{fg}[b(x_i)], \varepsilon\}}{\max\{h_{bg}[b(x_i)], \varepsilon\}} \quad (12)$$

其中, h_{fg} 和 h_{bg} 分别表示目标区域和背景区域的颜色直方图, ε 是一个用来避免数据不稳定的很小的非零常数。定义每个像素的加权因子:

$$T(x_i) = \begin{cases} 1, & L(x_i) > th_0 \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

其中, th_0 是用来减少目标抖动的阈值。

3.3. 分块实现

如何分块是跟踪算法鲁棒性的一个重要因素, 考虑到实际问题, 分块既不能太大, 也不能太小。本文分块过程如下:

1) 采用基于颜色相似度的方法(3.2 节)进行前景提取,

2) 垂直和水平投影:

计算所提取目标区域的每一行的平均强度, 得到垂直方向和水平方向的向量函数:

垂直方向: $V_m = (w_1, w_2, \dots, w_m)$, 其中, m 是所提取目标区域的宽度。

水平方向: $V_n = (w_1, w_2, \dots, w_n)$, 其中, n 是所提取目标区域的高度。

3) 分割点检测:

使用基于 Sobel 算子的边缘检测算法来取得分割点, 实现过程如下:

a) 计算每一个向量的最大值。

b) 定义条件(k 是子向量的个数):

$$sv_i = |v_{i-1} + v_{i+1} - 2v_i| \quad T = \max(v_1, v_2, \dots, v_k)/4$$

$$\Delta H_i = |H_i - H_{i-1}| \geq k/10 \quad \Delta V_i = |V_i - V_{i-1}| \geq k/10$$

c) 使用垂直分割点和水平分割点将目标分成若干子块, 结果如图 2 所示。

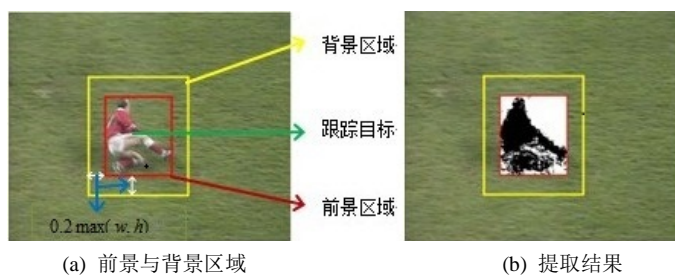


Figure 1. Foreground and background separation

图 1. 前景与背景分离

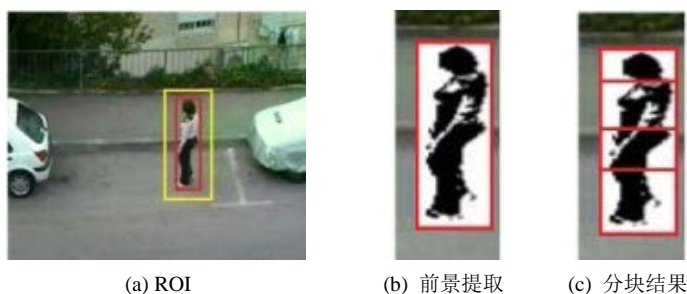


Figure 2. Fragments for target

图 2. 目标分块

3.4. 基于分块的核跟踪算法的改进

由式(10)可知,核函数的剖面函数具有单调递减的性质,在核函数加权直方图中,边缘区域对直方图的贡献较小,边沿小部分的遮挡对直方图影响不大,因此核跟踪算法对部分遮挡有一定的鲁棒性。但是,如果目标被大面积遮挡,尤其是被遮挡位置接近目标中心时,则会严重影响特征直方图的分布。

针对这一问题,我们可以通过偏移核函数中心在目标区域位置来解决。如图 3 所示为以灰度等级表示的两个核函数,图 3(a)中核函数中心位于目标中心,如果将核函数中心偏移到目标的右下角,如图 3(b)所示,那么左下角的函数值就会降低,当该部分被遮挡时,其对直方图的影响将会减小。

为降低计算复杂度,文献[13]提供了一种合适的方法来选择核函数中心,如图 4(a)所示,以目标中心为坐标原点,以 w , h 为长短轴建立椭圆,核函数中心位于该椭圆上。以 θ 角为起点,每隔 Π/n 得到一个核函数中心,核函数带宽保持不变,建立核函数加权直方图。其中, n 是直方图的个数。图 4(b)为初始目标和后续帧中被遮挡的目标。表 1 为 $\theta = 45^\circ$, $n = 4$ 时得到的一组核函数加权直方图巴氏系数的相似性测度。

可见,当目标的下半部分被遮挡时,传统的以目标中心为核函数中心的直方图相似性降低为 0.6,而以左上和右上为核函数中心的直方图的相似度保持在 0.9,以左下和右下为核函数中心的直方图相似度下降为 0.3。

但是,如果偏移核函数中心,就会使得目标区域内的样本点分布不对称,必须对 Mean Shift 算法进行改进。Comaniciu [14]等人证明了 Mean Shift 算法的收敛性,即在迭代过程中,有

$$\Delta \hat{y}_0 \rightarrow 0 \quad (14)$$

随着目标模型与候选模型的相似度的递增,有 $\omega_i^{(l)} \rightarrow 1$, 因此,式(11)可写为:

$$\Delta \hat{y}_0 \rightarrow \frac{1}{N^{(l)}} \sum x_i^{(l)} \quad (15)$$

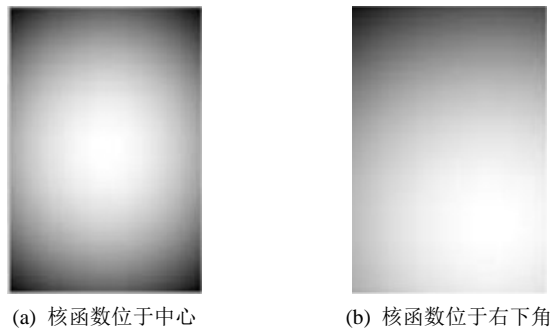


Figure 3. The kernel function of different position
图 3. 不同位置的核函数

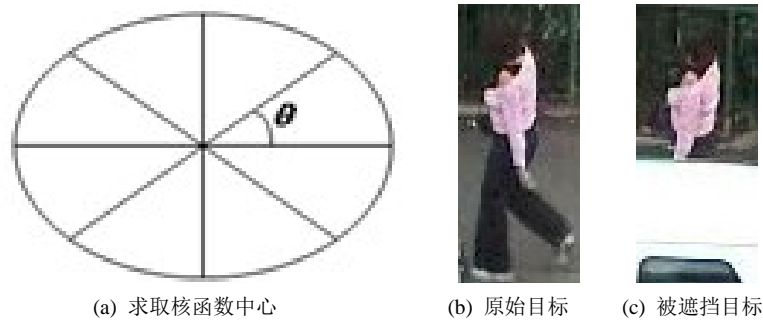


Figure 4. Method for getting kernel function center
图 4. 核函数中心获取方法

Table 1. The influence of kernel function on histogram Bhattacharyya
表 1. 核函数中心对直方图 Bhattacharyya 系数的影响

核函数位置	中心	左上	右上	左下	右下
相似度	0.6492	0.9109	0.9244	0.3024	0.3266

当样本点关于核函数中心对称时， $\sum x_i^{(l)} = 0$ ；不对称时，则 $\sum x_i^{(l)} \neq 0$ ，不满足收敛性。为此，定义

$$\Delta \hat{y}_1 = \frac{x_i^{(l)}}{\sum_{i=1}^{N^{(l)}} \omega_i^{(l)}} - \frac{1}{N^{(l)}} x_i^{(l)} \quad (16)$$

则有：

$$\sum_{i=1}^{N^{(l)}} \Delta \hat{y}_1 = \frac{\sum_{i=1}^{N^{(l)}} x_i^{(l)} \omega_i^{(l)}}{\sum_{i=1}^{N^{(l)}} \omega_i^{(l)}} - \frac{1}{N^{(l)}} \sum_{i=1}^{N^{(l)}} x_i^{(l)} \quad (17)$$

由式(11)和式(15)得：

$$\sum_{i=1}^{N^{(l)}} \Delta \hat{y}_1 \rightarrow 0 \quad (18)$$

因此，当样本点不对称时，改进迭代公式如下即可满足收敛性。

$$\Delta \hat{y}_0 = \frac{\sum_{i=1}^{N^{(l)}} x_i^{(l)} \omega_i^{(l)}}{\sum_{i=1}^{N^{(l)}} \omega_i^{(l)}} - \frac{1}{N^{(l)}} \sum_{i=1}^{N^{(l)}} x_i^{(l)} \quad (19)$$

4. 目标模型和权值的自适应更新

4.1. 遮挡检测

定义 $R = B(p_e, q_e) / B(p_k, q_k)$ 。其中 $B(p, q)$ 表示 Bhattacharya 系数； p_k, q_k 分别表示目标模型和候选区域以中心核函数建立的加权直方图； p_e, q_e 分别表示目标模型和候选区域以边缘核函数建立的加权直方图。正常情况下， R 值应接近于 1，但当目标发生遮挡时，跟踪窗口内包含大量的背景像素点， R 值将显著偏离于 1，因此可以设置 $|R-1| > T$ 作为判断遮挡的依据，其中 T 为预设的阈值。实验中取 $T = 0.2$ 。由于目标尺度变化等其他因素也可能导致 R 值改变，但尺度变化过程一般比较缓慢，而算法中又采用了模板更新机制，故当 $B(p_k, q_k) \leq 0.85$ 且 $T > 0.2$ 时使用多核融合跟踪。

4.2. 子块权重分配

由 2.2 可知，对数似然函数 $L(x_i)$ 可以用来区分前景像素和背景像素，似然值越大，与前景像素越接近，而式(3)中权重 $\lambda^{(l)}$ 则说明了每一个子块的重要程度， $\lambda^{(l)}$ 越大，表明该子块与目标模版相似。

以 $\lambda_{fg}^{(l)}$ ， $\lambda_{bg}^{(l)}$ 分别表示前景和背景的权重，定义 $\lambda^{(l)}$ ：

$$\lambda^{(l)} = \gamma \lambda_{fg}^{(l)} + (1 - \gamma) \lambda_{bg}^{(l)} \quad (20)$$

$$\lambda_{fg}^{(l)} = \frac{N_{fg}^{(l)}}{\sum_{i=1}^L N_{fg}^{(l)}} \quad (21)$$

$$\lambda_{bg}^{(l)} = 1 - \frac{N_{fg}^{(l)}}{\sum_{i=1}^L N_{fg}^{(l)}} \quad (22)$$

其中， $N_{fg}^{(l)}$ 表示第 l 个分块的前景区中的像素数。

4.3. 模版更新

在实际的目标跟踪过程中，目标会受到光照、自身形变等因素的影响，为了提高跟踪的准确性，需要实时更新目标模版。根据目标从背景中的分割可以判断某一像素对应的是目标还是背景，根据目标的变化程度以及目标模型与候选模型的相似度来判断是否更新模版。如式(23)所示：

$$\hat{q}^{(l)} = \begin{cases} \hat{p}^{(l)}, & \rho[\hat{p}^{(l)}, \hat{q}^{(l)}] \geq th_0 \\ \hat{q}^{(l-1)}, & \text{otherwise} \end{cases} \quad (23)$$

5. 实验结果与分析

5.1. 实验环境

笔记本电脑：ThinkPad T420；CPU：Intel(R) Core(TM)i5 2.30GHz；内存：2GB；操作系统：Windows 7 旗舰版 SP1；开发平台：Visual Studio 2010；开发库：OpenCV 2.4.3。目标初始位置以手工方式输入，跟踪结果在视频画面上以矩形框显示。

采用 Epanechnikov 核作为空间加权的核函数, 目标特征量化等级为 8, 模版更新阈值设置为 0.8, 设定 $\theta = 45^\circ$, $n = 45^\circ$, 以目标中心为坐标原点, 得到分别位于 $(0, 0)$ 、 $(w/2, h/2)$ 、 $(-w/2, h/2)$ 、 $(-w/2, -h/2)$ 、 $(w/2, -h/2)$ 处的 5 个核函数。本文提出的融合多核的分块跟踪算法称为 MKF (Fragments-based Tracking Algorithm with Multiple Kernel Fusion), 而传统的采用 Mean Shift 算法称为 BMS (Basic Mean Shift Tracking)。

5.2. 实验结果

实验 1: 采用文献[5]中 Adam 等人使用的图像序列, 图 5 中, 红色矩形框和蓝色矩形框分别表示了本文算法(MKF)和 BMS 算法的跟踪结果, 可以看出, 本文算法几乎不受光照因素的影响, 而 BMS 算法受光照影响十分严重。在目标从图 5(b)移动到图 5(d)过程中, 目标所处的光照环境发生了很大变化, 且目标有很长一段时间被汽车部分遮挡, 但是本文算法依然能够进行很好的跟踪, 而 BMS 算法在跟踪过程中发生了很大的偏移。

实验 2: 采用一段长 462 帧的图像序列作实验序列, 在序列起始帧中, 目标位于一个光照非常暗的环境下, 在后续帧中, 目标逐渐走到光照条件比较复杂的环境下, 而在目标移动过程中发生了轻微形变(自身姿势的变化), 并且有简单的手势动作(摘眼睛和戴眼镜)。实验中, 手工设定跟踪目标的初始窗口, 并将本文算法与文献[5]中算法进行了对比, 跟踪效果如图 6 所示。在第 146 帧, 目标移动到挂有一副图片的墙壁处, 文献[5]跟踪窗向图片偏移, 之后目标移动到第 172 帧处, 此时目标处于强光下且自身姿势发生了变化, 可以看出本文算法跟踪效果良好, 而文献[5]跟踪目标发生了变化。在第 399 帧处理, 目标从光线暗处移动到光线亮处, 而且有戴眼镜的动作, 可以看出文献[5]跟踪窗偏移到手部, 而本文算法依然能够实现很好的跟踪。

实验 3: 采用数据库[15]中的一段图像序列, 手工设定“身穿红色皮上衣的男子”为跟踪目标。目标在经过一位“身穿灰色上衣的男子”过程中被其严重遮挡, 而在目标完全越过该男子后, 一位“身穿白色衬衫的男子”从商店走出, 并且对目标发生了小部分的遮挡。图 7 中, 红色矩形框和蓝色矩形框分别显示了本文算法和文献[7]中算法的跟踪效果, 可以看出, 在目标移动过程中, 本文算法都能实现很好的跟踪, 而在第 48 帧处, 目标被严重遮挡, 文献[7]中算法跟踪框偏移到了“身穿灰色上衣的男子”身上, 跟踪对象出现了严重错误。

实验 4: 采用数据库[16]中的两端图像序列来验证本文算法。(a)中以“背包的女子”为跟踪目标, 在该视角下, 目标从左向右移动, 且在其经过人群时被多人遮挡。(b)以“挎包的男子”为跟踪目标, 在该视角下, 目标从右向左移动, 且在其经过人群时被多人遮挡。本文取了目标被严重遮挡的图像帧进行对比, 如(a)中 61 帧和 78 帧, (b)中 59 帧和 72 帧, 图 8 显示了本文算法和文献[8]中算法的跟踪效果, 其中红色框表示本文算法, 蓝色框表示文献[8]中算法。从跟踪结果可以看出, 尽管两种算法都能实现很好的跟踪, 但是当发生大面积的遮挡时, 本文算法的跟踪效果更理想, 而文献[8]中算法发生了明显的偏差。

6. 总结

针对目标被大面积遮挡时的鲁棒跟踪问题, 本文提出一种基于分块的融合多核的目标跟踪算法, 该算法采用分块的思想, 构造多个以不同位置为中心的核函数, 分别进行均值漂移跟踪; 另外, 在目标跟踪过程中, 为处理目标形变和遮挡等情况, 采用了自适应权重和目标模版更新机制。实验表明本算法对存在大面积遮挡情况下的目标有良好的跟踪性能。

为了进一步提高鲁棒性, 可以将本文算法与卡尔曼滤波结合使用来处理全遮挡问题; 当前景与背景颜色相近时, 可以采用更多的目标特征来进行前景提取; 此外, 由于采用多核融合将增加计算复杂度, 可

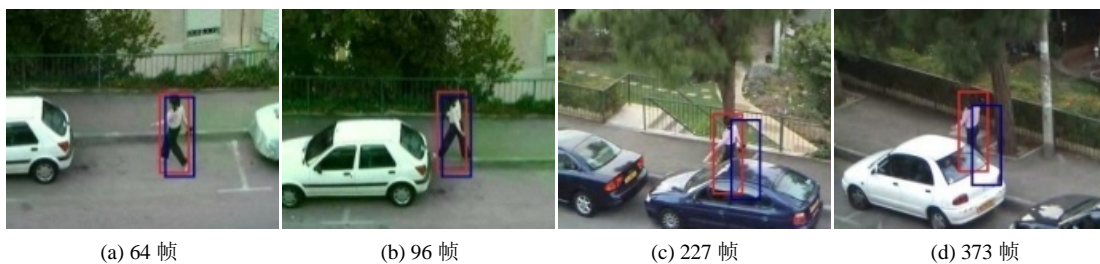


Figure 5. Tracking results of the proposed algorithm and BMS

图 5. 本文算法和 BMS 算法的跟踪效果

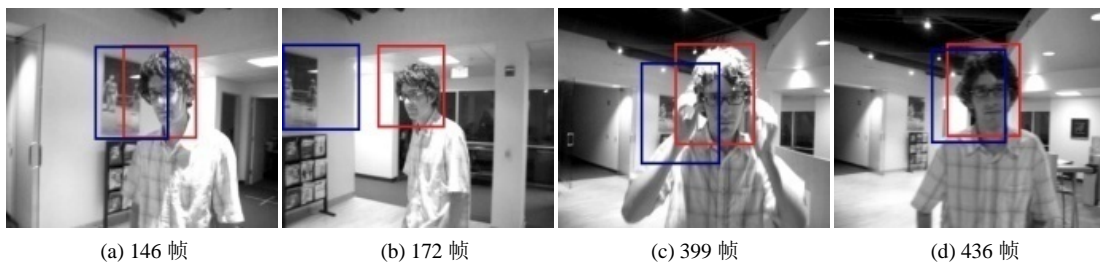


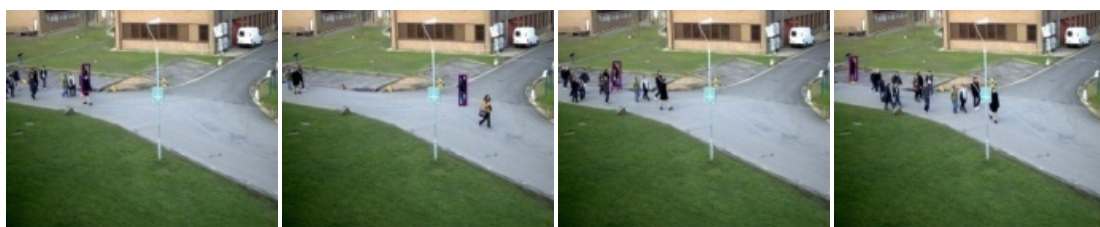
Figure 6. Robustness of the proposed algorithm for illumination

图 6. 本文算法对光照变化的鲁棒性



Figure 7. Tracking results of the proposed algorithm and Ref. [7]

图 7. 本文算法和文献[7]算法的跟踪效果



(a) 目标从右边移动到左边依次帧数为: 20 帧、59 帧、72 帧、95 帧



(b) 目标从左边移动到右边依次帧数为: 34 帧、61 帧、78 帧、92 帧

Figure 8. Tracking results of proposed algorithm and Ref. [8]

图 8. 本文算法与文献[8]算法的跟踪效果

以考虑一种自适应预测下一帧中可能有效的核函数机制来降低复杂度。

参考文献 (References)

- [1] Comaniciu, D., Ramesh, V. and Meer, P. (2003) Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**, 564-577.
- [2] Yang, C., Duraiswami, R. and Davis, L. (2005) Efficient mean-shift tracking via a new similarity measure. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, 20-25 June 2005, Vol. 2, 176-183.
- [3] Birchfield, S.T. and Rangarajan, S. (2005) Spatiograms versus histograms for region-based tracking. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, 20-25 June 2005, Vol. 1, 1158-1163.
- [4] Fan, Z., Yang, M. and Wu, Y. (2007) Multiple collaborative kernel tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29**, 1268-1273.
- [5] Adam, A., Rivlin, E. and Shimshoni, I. (2006) Robust fragments-based tracking using the integral histogram. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, 17-22 June 2006, 798-805.
- [6] Babu, R., Perez, P. and Bouthemy, P. (2007) Robust tracking with motion estimation and local kernel-based color modeling. *Image and Vision Computing*, **25**, 1205-1216.
- [7] Jeyakar, J., Babu, R. and Ramakrishnan, K.R. (2008) Robust object tracking with background-weighted local kernels. *Computer Vision and Image Understanding*, **112**, 296-309.
- [8] Wang, F.L., Yu, S.Y. and Yang, J. (2010) Robust and efficient fragments-based tracking using mean shift. *International Journal of Electronics and Communications*, **64**, 614-623.
- [9] Fang, J.X., Yang, J. and Liu, H.X. (2011) Efficient and robust fragments-based multiple kernels tracking. *International Journal of Electronics and Communications*, **65**, 915-923.
- [10] Phadke, G. (2011) Robust multiple target tracking under occlusion using fragmented mean shift and kalman filter. 2011 *International Conference on Communications and Signal Processing (ICCSP)*, Kerala, 10-12 February 2011, 517-521.
- [11] Wang, Y.Z., Liang, Y., Zhao, C.H., et al. (2008) Kernel-based tracking based on adaptive fusion of multiple cues. *Acta Automatica Sinica*, **34**, 393-399.
- [12] Comaniciu, D. and Meer, P. (2002) Mean shift: A Robust Approach toward Feature Space Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**, 603-619.
- [13] Liu, Q., Tang, L.B., Zhao, B.J., et al. (2013) Improved mean shift target tracking algorithm. *Systems Engineering and Electronics*, **35**, 1318-1323.
- [14] Comaniciu, D. and Meer, P. (1997) Robust analysis of feature spaces: Color image segmentation. 1997 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Juan, 17-19 Jun 1997, Vol. 6, 750-755.
- [15] Datasets Available at: <http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/>
- [16] Datasets Available at Website: http://www-prima.inrialpes.fr/PETS04/caviar_data.html