

## 基于梯度方向直方图特征的多核跟踪

贾慧星<sup>1,2</sup> 章毓晋<sup>1,2</sup>

**摘要** 提出了基于梯度方向直方图特征的多核跟踪算法,对跟踪过程中的光线变化和部分遮挡具有较强的鲁棒性.该算法将目标分块,分别提取出每块的核函数加权的梯度方向直方图特征.目标模型和候选目标模型的相似度用所有块直方图间的 Bhattacharyya 系数之和进行度量,目标的跟踪通过 Mean shift 算法最大化两者的相似度实现.对车辆、人体等多个目标的跟踪验证了本文提出算法的有效性.

**关键词** Mean shift, 核跟踪, Bhattacharyya 系数, 梯度方向直方图  
**中图分类号** TP391.41

### Multiple Kernels Based Object Tracking Using Histograms of Oriented Gradients

JIA Hui-Xing<sup>1,2</sup> ZHANG Yu-Jin<sup>1,2</sup>

**Abstract** A novel multiple kernels based object tracking algorithm using histograms of oriented gradients is proposed in this paper, which is robust to illumination change and partial occlusion. The algorithm divides the object into blocks and extracts kernel weighted histograms of oriented gradients for each block. The similarity between target model and candidate model is measured by the sum of Bhattacharyya coefficients of all the corresponding histograms. The object is tracked by maximizing the similarity measure using the mean shift algorithm. Experiments on the tracking of vehicle and human demonstrate the effectiveness of the proposed algorithm.

**Key words** Mean shift, kernel based tracking (KBT), Bhattacharyya coefficient, histograms of oriented gradients

目标跟踪是计算机视觉领域的热点研究课题,它在智能视频监控、人机交互等领域有着广泛的应用.给定目标在第一帧的位置,跟踪的目的是确定目标在随后帧的位置.目标跟踪的主要难点包括目标颜色和形状的变化、目标所处环境光线的变化和目标交互引起的遮挡.现有的目标跟踪方法可以分为核跟踪和粒子滤波跟踪,核跟踪通过迭代的方法跟踪目标,速度较快,实现比较简单;粒子滤波跟踪通过蒙特卡洛采样的方法对目标进行跟踪,对目标的遮挡比较鲁棒,但速度较慢.本文主要研究核跟踪中的光线变化和部分遮挡问题.

核跟踪(Kernel based tracking, KBT)由 Comaniciu 等<sup>[1]</sup>提出,又称为 Mean shift 跟踪,跟踪过程通过 Mean shift 算法迭代实现,避免了穷尽搜索,简单易用,速度较快,自从其提出后就成为视觉跟踪领域的重要方法.针对原始的核跟踪算法存在的一

些缺点,很多作者提出了改进算法.原始的核跟踪算法只处理了平移运动,针对目标的尺度变化, Collins 等<sup>[2]</sup>提出了尺度空间上的 Mean shift 算法,彭宁嵩等<sup>[3]</sup>提出了基于形心匹配的尺度估计方法.针对目标的旋转运动, Yilmaz<sup>[4]</sup>提出了在平移、尺度和旋转四个维数空间上的 Mean shift 跟踪算法.针对目标的直方图表示丢失空间信息的缺点, Yang 等<sup>[5]</sup>提出了一种新的目标表示方法及其对应的 Mean shift 算法,以反映像素点的空间位置;李培华<sup>[6]</sup>提出将像素的位置信息加入到颜色直方图的表示之中.针对原始算法只采用一种特征的缺点,王永忠等<sup>[7]</sup>提出将颜色和纹理特征进行自适应融合的核跟踪算法.

基于颜色或纹理特征的核跟踪算法受光线变化的影响较大,为了提高核跟踪算法对光线变化的鲁棒性,本文提出了基于梯度方向直方图特征的核跟踪算法.其主要思想来源于梯度方向直方图特征在目标匹配<sup>[8]</sup>、目标检测<sup>[9]</sup>中的成功运用. Lowe<sup>[8]</sup>提出的尺度不变特征变换(Scale invariant feature transform, SIFT)描述子采用梯度方向直方图对特征点进行描述,达到了极好的匹配效果. Dalal<sup>[9]</sup>对行人目标利用梯度方向直方图特征进行描述,结合支持向量机(Support vector machine, SVM)分类器,在人体检测上达到了很好的性能.梯度方向直方图特征对光线变化和局部区域的微小形变比较鲁棒,所以本文将该特征引入到核跟踪领域.为了提高对

收稿日期 2008-08-13 收修改稿日期 2009-01-06  
Received August 13, 2008; in revised form January 6, 2009  
国家自然科学基金(60872084)资助  
Supported by National Natural Science Foundation of China (60872084)

1. 清华大学信息科学与技术国家实验室 北京 100084 2. 清华大学电子工程系 北京 100084

1. Tsinghua National Laboratory for Information Science and Technology, Tsinghua University Beijing 100084 2. Department of Electronic Engineering, Tsinghua University, Beijing 100084

DOI: 10.3724/SP.J.1004.2009.01283

局部遮挡的鲁棒性, 本文将目标分块, 针对每块提取核函数加权的梯度方向直方图特征, 提出了多核跟踪方法. 多核跟踪方法提高了模型的鉴别性, 同时对跟踪过程中的局部遮挡比较鲁棒.

本文其余部分安排如下: 第 1 节简介核跟踪算法; 第 2 节介绍目标的分块梯度方向直方图表示; 第 3 节介绍基于梯度方向直方图特征的多核跟踪算法; 第 4 节给出实验结果; 第 5 节总结全文.

## 1 核跟踪简介

在核跟踪中, 首先将原始像素灰度值转换到特征空间中, 目标模型  $\mathbf{q}$  和候选目标模型  $\mathbf{p}$  在特征空间中用直方图表示, 同时为了降低边缘像素点的权重, 提取直方图的时候采用了一个单调递减凸函数  $k$  作为核函数进行加权. 目标模型  $\mathbf{q} = [q_1, \dots, q_u, \dots, q_U]$  和以  $\mathbf{y}$  为中心的候选目标模型  $\mathbf{p}(\mathbf{y}) = [p_1(\mathbf{y}), \dots, p_u(\mathbf{y}), \dots, p_U(\mathbf{y})]$  计算如下:

$$q_u = Q \sum_{i=1}^{n_q} k \left( \left\| \frac{\mathbf{y}_i}{h_q} \right\|^2 \right) \delta [b(\mathbf{y}_i) - u] \quad (1)$$

$$Q = \left( \sum_{i=1}^{n_q} k \left( \left\| \frac{\mathbf{y}_i}{h_q} \right\|^2 \right) \right)^{-1} \quad (2)$$

$$p_u(\mathbf{y}) = P \sum_{i=1}^{n_p} k \left( \left\| \frac{\mathbf{y}_i - \mathbf{y}}{h_p} \right\|^2 \right) \delta [b(\mathbf{y}_i) - u] \quad (3)$$

$$P = \left( \sum_{i=1}^{n_p} k \left( \left\| \frac{\mathbf{y}_i - \mathbf{y}}{h_p} \right\|^2 \right) \right)^{-1} \quad (4)$$

其中,  $h_q$  和  $h_p$  分别是两个核函数的带宽, 表示目标的尺度;  $\mathbf{y}_i$  指像素的坐标位置;  $b$  是每个像素直方图的区间指示;  $U$  指直方图的区间数目;  $Q$  指目标模型直方图的归一化系数, 是一个依赖于  $h_q$  的常数;  $P$  指候选目标模型直方图的归一化系数, 是一个依赖于  $h_p$  的常数.

目标模型和候选目标模型间的相似度定义为这两个直方图的 Bhattacharyya 系数

$$\rho(\mathbf{y}) = \sum_{u=1}^U \sqrt{p_u(\mathbf{y}) q_u} \quad (5)$$

跟踪的目的就是在每帧中寻找使式 (5) 最大的候选目标的位置. 在目标的初始位置  $\hat{\mathbf{y}}_0$ , 利用 Taylor 展开, 式 (5) 可以近似为

$$\begin{aligned} \rho(\mathbf{y}) &\approx \frac{1}{2} \sum_{u=1}^U p_u(\mathbf{y}) \sqrt{\frac{q_u}{p_u(\hat{\mathbf{y}}_0)}} + \frac{1}{2} \sum_{u=1}^U \sqrt{p_u(\hat{\mathbf{y}}_0) q_u} \\ &= \frac{P}{2} \sum_{i=1}^{n_p} \omega_i k \left( \left\| \frac{\mathbf{y}_i - \mathbf{y}}{h_p} \right\|^2 \right) + C \end{aligned} \quad (6)$$

其中

$$\omega_i = \sum_{u=1}^U \delta [b(\mathbf{y}_i) - u] \sqrt{\frac{q_u}{p_u(\hat{\mathbf{y}}_0)}} \quad (7)$$

$$C = \frac{1}{2} \sum_{u=1}^U \sqrt{p_u(\hat{\mathbf{y}}_0) q_u} \quad (8)$$

最大化式 (6) 可以通过最大化其左半部分来实现, 利用 Mean shift 算法<sup>[10]</sup>, 以初始点  $\hat{\mathbf{y}}_0$  为起点的位移矢量可以计算为

$$\Delta \hat{\mathbf{y}}_0 = \frac{\sum_{i=1}^{n_p} \omega_i g \left( \left\| \frac{\mathbf{y}_i - \hat{\mathbf{y}}_0}{h_p} \right\|^2 \right) (\mathbf{y}_i - \hat{\mathbf{y}}_0)}{\sum_{i=1}^{n_p} \omega_i g \left( \left\| \frac{\mathbf{y}_i - \hat{\mathbf{y}}_0}{h_p} \right\|^2 \right)} \quad (9)$$

其中,  $g(r) = -k'(r)$ . 整个算法的流程如下:

**步骤 1.** 将目标在当前帧的位置初始化为  $\hat{\mathbf{y}}_0$ , 利用式 (3) 计算  $p_u(\hat{\mathbf{y}}_0)$ , 利用式 (5) 计算  $\rho(\hat{\mathbf{y}}_0)$ ;

**步骤 2.** 利用式 (7) 计算  $\{\omega_i\}_{i=1, \dots, n_p}$ ;

**步骤 3.** 利用式 (9) 计算当前点的位移矢量  $\Delta \hat{\mathbf{y}}_0$ ;

**步骤 4.** 利用式 (3) 计算  $p_u(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0)$ , 利用式 (5) 计算  $\rho(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0)$ ;

**步骤 5.** 当  $\rho(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0) < \rho(\hat{\mathbf{y}}_0)$  时, 执行  $\Delta \hat{\mathbf{y}}_0 \leftarrow 0.5 \Delta \hat{\mathbf{y}}_0$ , 重新计算  $\rho(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0)$ ;

**步骤 6.** 如果  $\Delta \hat{\mathbf{y}}_0 < \varepsilon$ , 则停止迭代; 否则令  $\hat{\mathbf{y}}_0 \leftarrow \hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0$ , 转步骤 2.

## 2 梯度方向直方图特征

假设目标的中心位置为  $\mathbf{0}$ , 首先利用梯度算子  $[-1 \ 0 \ 1]^T$  和  $[-1 \ 0 \ 1]$  与图像卷积, 得到任意像素点  $\mathbf{y}_i$  的梯度幅度  $m(\mathbf{y}_i)$  和梯度方向  $\theta(\mathbf{y}_i)$ , 对于彩色图像仅保留梯度幅度最大的颜色通道. 为了将梯度信息引入到核跟踪的框架中, 必须采用直方图的表示形式. 但若针对目标整体只提取一个直方图进行表示, 其局部的形状信息则不能得到保留. 所以如图 1 所示, 本文将目标分块, 针对每块提取核函数加权的梯度方向直方图. 每块的大小可以不同以获得不同尺度的信息; 块与块之间可以重叠以获得不同位置的信息. 每一个块在提取直方图时都采用了一个核函数进行加权, 以抑制边缘像素点的贡献, 同时增加每块的直方图表示对位移的平滑性.

假设目标用  $V$  个矩形块表示, 第  $v$  个矩形块的中心位置为  $\mathbf{l}^v$ , 尺度为  $h^v$ , 那么该块的梯度方向直方图  $\mathbf{q}^v = [q_1^v, \dots, q_u^v, \dots, q_U^v]$  可以计算为

$$q_u^v = Q^v \sum_{i=1}^{n^v} k^v \left( \left\| \frac{\mathbf{y}_i - \mathbf{l}^v}{h^v} \right\|^2 \right) \times$$

$$m(\mathbf{y}_i) \delta [b(\theta(\mathbf{y}_i)) - u] \quad (10)$$

$$Q^v = \left( \sum_{i=1}^{n^v} k^v \left( \left\| \frac{\mathbf{y}_i - \mathbf{l}^v}{h^v} \right\|^2 \right) m(\mathbf{y}_i) \right)^{-1} \quad (11)$$

其中,  $b$  表示每个像素梯度方向的量化函数;  $k^v$  表示第  $v$  个矩形块采用的核函数;  $n^v$  表示第  $v$  个矩形块像素个数;  $Q^v$  表示第  $v$  个矩形块直方图的归一化系数.

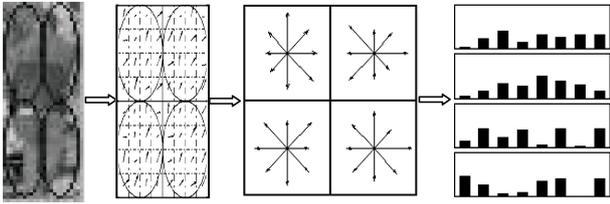


图 1 分块梯度方向直方图特征的提取

Fig. 1 Extraction of histogram of oriented gradients for each block

假设候选目标模型和目标模型的大小相同, 中心位置位于  $\mathbf{y}$  的候选目标模型的第  $v$  个矩形块的梯度方向直方图  $\mathbf{p}(\mathbf{y})^v = [p(\mathbf{y})_1^v, \dots, p(\mathbf{y})_u^v, \dots, p(\mathbf{y})_{U^v}^v]$  可以计算为

$$p(\mathbf{y})_u^v = P(\mathbf{y})^v \sum_{i=1}^{n^v} k^v \left( \left\| \frac{\mathbf{y}_i - \mathbf{l}^v - \mathbf{y}}{h^v} \right\|^2 \right) \times m(\mathbf{y}_i) \delta [b(\theta(\mathbf{y}_i)) - u] \quad (12)$$

$$P(\mathbf{y})^v = \left( \sum_{i=1}^{n^v} k^v \left( \left\| \frac{\mathbf{y}_i - \mathbf{l}^v - \mathbf{y}}{h^v} \right\|^2 \right) m(\mathbf{y}_i) \right)^{-1} \quad (13)$$

其中,  $P(\mathbf{y})^v$  表示直方图的归一化系数, 由于每一个像素点梯度幅度的存在, 使得归一化系数依赖于坐标  $\mathbf{y}$ .

### 3 基于梯度方向直方图的多核跟踪

目标模型基于分块梯度方向直方图的表示为  $\mathbf{q} = [q^1, \dots, q^v, \dots, q^V]$ , 中心位置位于  $\mathbf{y}$  的候选目标模型基于分块梯度方向直方图的表示为  $\mathbf{p}(\mathbf{y}) = [p^1(\mathbf{y}), \dots, p^v(\mathbf{y}), \dots, p^V(\mathbf{y})]$ . 目标模型和候选目标模型的相似度定义为

$$\rho(\mathbf{y}) = \sum_{v=1}^V \sum_{u=1}^U \sqrt{p_u^v(\mathbf{y}) q_u^v} \quad (14)$$

目标的跟踪可以通过最大化式 (14) 来实现. 在目标的初始位置  $\hat{\mathbf{y}}_0$ , 利用 Taylor 展开, 式 (14) 可以近似为

$$\begin{aligned} \rho(\mathbf{y}) &\approx \frac{1}{2} \sum_{v=1}^V \sum_{u=1}^U \left( \sqrt{p_u^v(\hat{\mathbf{y}}_0) q_u^v} + p_u^v(\mathbf{y}) \sqrt{\frac{q_u^v}{p_u^v(\hat{\mathbf{y}}_0)}} \right) \\ &= \frac{1}{2} \sum_{v=1}^V P(\mathbf{y})^v \sum_{i=1}^{n^v} k^v \left( \left\| \frac{\mathbf{y}_i - \mathbf{y} - \mathbf{l}^v}{h^v} \right\|^2 \right) \omega_i^v + C \end{aligned} \quad (15)$$

其中

$$\omega_i^v = \sum_{u=1}^U \left( m(\mathbf{y}_i) \delta [b(\theta(\mathbf{y}_i)) - u] \sqrt{\frac{q_u^v}{p_u^v(\hat{\mathbf{y}}_0)}} \right) \quad (16)$$

$$C = \frac{1}{2} \sum_{v=1}^V \sum_{u=1}^U \left( \sqrt{p_u^v(\hat{\mathbf{y}}_0) q_u^v} \right) \quad (17)$$

由于核函数  $k$  的平滑性, 在  $\hat{\mathbf{y}}_0$  附近,  $P(\mathbf{y})^v$  可以用  $P(\hat{\mathbf{y}}_0)^v$  近似, 于是式 (15) 可以近似为

$$\begin{aligned} \rho(\mathbf{y}) &\approx \\ &\frac{1}{2} \sum_{v=1}^V P(\hat{\mathbf{y}}_0)^v \sum_{i=1}^{n^v} k^v \left( \left\| \frac{\mathbf{y}_i - \mathbf{y} - \mathbf{l}^v}{h^v} \right\|^2 \right) \omega_i^v + C \end{aligned} \quad (18)$$

利用 Mean shift 算法<sup>[9]</sup>,  $\rho(\mathbf{y})$  的梯度可以计算为

$$\nabla \rho(\mathbf{y}) = \sum_{v=1}^V \frac{P(\hat{\mathbf{y}}_0)^v}{(h^v)^2} \sum_{i=1}^{n^v} g^v \left( \left\| \frac{\mathbf{y}_i - \hat{\mathbf{y}}_0 - \mathbf{l}^v}{h^v} \right\|^2 \right) \times (\mathbf{y}_i - \hat{\mathbf{y}}_0 - \mathbf{l}^v) \omega_i^v \quad (19)$$

于是, 为了最大化  $\rho(\mathbf{y})$ , 在  $\hat{\mathbf{y}}_0$  附近的位移矢量为

$$\begin{aligned} \Delta \hat{\mathbf{y}}_0 &= \\ &\frac{\sum_{v=1}^V \frac{P(\hat{\mathbf{y}}_0)^v}{(h^v)^2} \sum_{i=1}^{n^v} g^v \left( \left\| \frac{\mathbf{y}_i - \hat{\mathbf{y}}_0 - \mathbf{l}^v}{h^v} \right\|^2 \right) (\mathbf{y}_i - \hat{\mathbf{y}}_0 - \mathbf{l}^v) \omega_i^v}{\sum_{v=1}^V \frac{P(\hat{\mathbf{y}}_0)^v}{(h^v)^2} \sum_{i=1}^{n^v} g^v \left( \left\| \frac{\mathbf{y}_i - \hat{\mathbf{y}}_0 - \mathbf{l}^v}{h^v} \right\|^2 \right) \omega_i^v} \end{aligned} \quad (20)$$

以  $\hat{\mathbf{y}}_0$  为起点, 目标在当前帧位置可以通过式 (20) 迭代得到. 算法的流程如下:

**步骤 1.** 将目标在当前帧的位置初始化为  $\hat{\mathbf{y}}_0$ , 利用式 (12) 计算  $\mathbf{p}(\hat{\mathbf{y}}_0) = [p^1(\hat{\mathbf{y}}_0), \dots, p^v(\hat{\mathbf{y}}_0), \dots, p^V(\hat{\mathbf{y}}_0)]$ , 利用式 (14) 计算  $\rho(\hat{\mathbf{y}}_0)$ ;

**步骤 2.** 利用式 (16) 计算  $\{\omega_i^v\}$ ;

**步骤 3.** 利用式 (20) 计算当前点的位移矢量  $\Delta \hat{\mathbf{y}}_0$ ;

**步骤 4.** 利用式 (12) 计算  $p_u(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0)$ , 利用式 (14) 计算  $\rho(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0)$ ;

**步骤 5.** 当  $\rho(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0) < \rho(\hat{\mathbf{y}}_0)$  时, 执行  $\Delta \hat{\mathbf{y}}_0 \leftarrow 0.5 \Delta \hat{\mathbf{y}}_0$ , 重新计算  $\rho(\hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0)$ ;

**步骤 6.** 如果  $\Delta \hat{\mathbf{y}}_0 < \varepsilon$ , 则停止迭代; 否则令  $\hat{\mathbf{y}}_0 \leftarrow \hat{\mathbf{y}}_0 + \Delta \hat{\mathbf{y}}_0$ , 转步骤 2.

## 4 实验

本文提出了基于梯度方向直方图特征的多核跟踪算法, 由于梯度方向直方图特征反映了目标的形状信息, 本文称之为基于形状的多核跟踪算法 (Multiple kernel based object tracking using shape, S-MKBT), 当分块数目为 1 时, 称为基于形状的核跟踪算法 (Kernel based object tracking using shape, S-KBT). 同时第 3 节的多核跟踪算法的推导过程可以直接应用到颜色特征上得到基于颜色的多核跟踪算法 (Multiple kernel based object tracking using color, C-MKBT), 当分块数目为 1 时称为基于颜色的核跟踪算法 (Kernel based object tracking using color, C-KBT). 这几种算法的实现细节见第 4.1 节, 实验结果见第 4.2 节.

仿真的环境为: 笔记本电脑 HP-nc8430; CPU: Intel Core (TM) 2 2.0 GHz; 内存: 1G; 操作系统: Windows XP Professional, SP2; 软件: Matlab R2008a. 目标的初始位置通过手工的方式给出. 跟踪过程中尺度的处理不是本文重点论述的内容, 仅采用了 Comaniciu 等<sup>[1]</sup> 提出的最简单的处理方法, 即将上述式 (9) 和式 (20) 的跟踪过程, 在当前尺度和相邻的两个尺度上运行, 选取最终相似度最大的尺度为当前帧目标的尺度, 算法的流程图如图 2 所示.

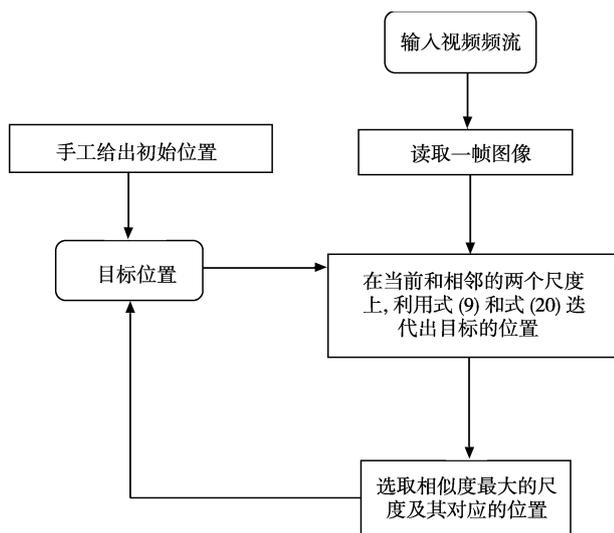


图 2 算法流程图

Fig. 2 Algorithm flow

### 4.1 算法实现

核函数采用 Epanechnikov 核, 即  $k(r) = \max(0, 1 - r)$ , 对应的  $g(r) = 1$ , 式 (9) 和式 (20) 都可以大大简化. 在 C-MKBT 和 C-KBT 算法中, 每个颜色通道的量化级数为 16, 所以每个直方图共

有  $16^3$  维. 在 S-MKBT 和 S-KBT 算法中, 角度的量化级数为 32, 即  $[0 \sim 2\pi]$  的角度区间被设为 32 份, 对应的直方图的维数为 32. 对于同一段序列, C-MKBT 和 S-MKBT 算法采用同样的分块策略, 本文采用三级分块策略: 最底层的长度和宽度为原始图像模板  $1/4$  的方块, 共 16 块; 中间层的长度和宽度为原始图像模板  $1/2$  的方块, 共 4 块; 最顶层的为与原始图像模板大小相同的 1 个方块. 所以对于 C-MKBT 和 S-MKBT 算法来说, 共有 21 个方块. 而对于 S-KBT 和 C-KBT 来说, 只有 1 个和原始图像模板大小相同的方块. 对于式 (9) 和式 (20) 表示的迭代过程, 当位移小于 1 个像素时停止迭代.

### 4.2 实验结果

本文用七段序列做实验, 对结果进行定性分析和定量分析. 定性分析通过观察跟踪结果获得, 定量分析用覆盖比率 (Overlap ratio)<sup>[11]</sup> 衡量. 覆盖比率的定义为

$$\Omega = 2 \frac{|R_{\text{object}} \cap R_{\text{tracked}}|}{|R_{\text{object}}| + |R_{\text{tracked}}|} \quad (21)$$

其中,  $R_{\text{object}}$  表示目标的真实位置 (通常由手工预先标定);  $R_{\text{tracked}}$  表示跟踪算法跟踪得到的目标区域位置;  $|\cdot|$  表示给定区域的面积. 显然  $\Omega \in [0, 1]$ ,  $\Omega$  越大表明跟踪结果越准确. 平均覆盖比率  $\bar{\Omega}$  表明一个算法在该序列上的平均性能.

**实验 1.** 用两段跟踪条件较好的序列比较 S-MKBT 算法和 C-MKBT 算法, 序列为 “Vehicle 1 ( $320 \times 240$ , 30 fps)” 和 “Human 1 ( $320 \times 240$ , 25 fps)”. 这两者的定量分析和定性分析结果分别如图 3 和图 4 所示, 其中实线框表示 S-MKBT 的结果, 虚线框表示 C-MKBT 的结果. 在跟踪条件较好、干扰较少时, 新提出的 S-MKBT 算法可以有效地跟踪车辆和行人目标. 针对 “Vehicle 1” 序列, S-MKBT 算法达到的平均覆盖率为 0.937, 而 C-MKBT 算法的平均覆盖率为 0.899, 所以 S-MKBT 算法的定位精度稍高于 C-MKBT 算法的定位精度. 在 “Human 1” 序列中, 两者的定位精度没有明显差别.

**实验 2.** 比较 S-MKBT 算法和 C-MKBT 算法在光线变化情况下的跟踪结果, 序列为 “Vehicle 2 ( $218 \times 145$ , 15 fps)”<sup>[12]</sup> 和 “Human 2 ( $320 \times 240$ , 15 fps)”<sup>[13]</sup>, 跟踪结果如图 5 和图 6 所示, 其中实线框表示 S-MKBT 的结果, 虚线框表示 C-MKBT 的结果. 如图 5 所示, 该车辆从 150 帧到 200 帧的过程中逐渐进入阴影区域, C-MKBT 算法在这一过程中跟踪失败, 而 S-MKBT 算法则没有受到阴影的影响. 如图 6 所示, 在第 65 帧左右, 行人的一小部分进入阴影后, C-MKBT 算法的跟踪误差相对较大, 而

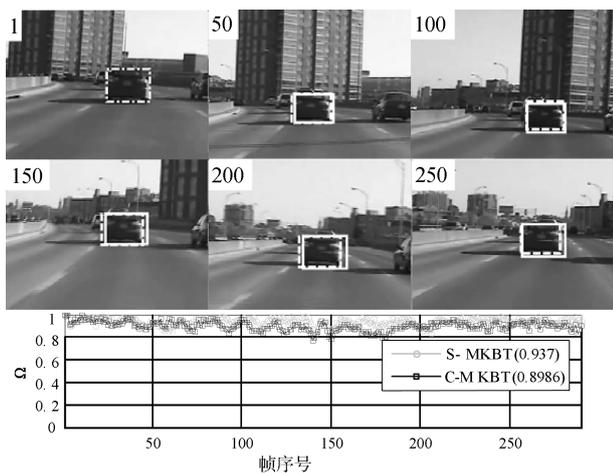


图 3 “Vehicle 1” 的实验结果  
Fig. 3 Results of “Vehicle 1”

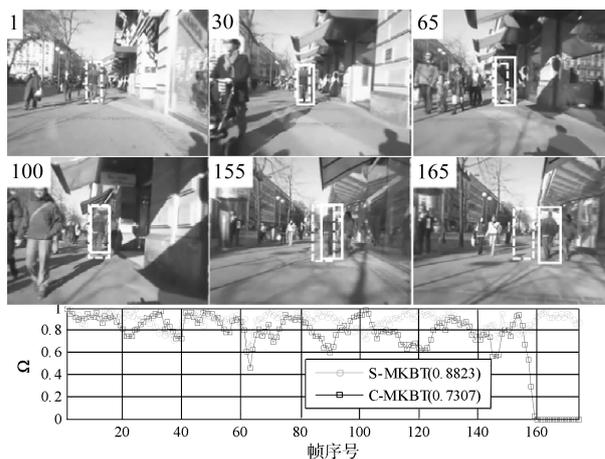


图 6 “Human 2” 的实验结果  
Fig. 6 Results of “Human 2”

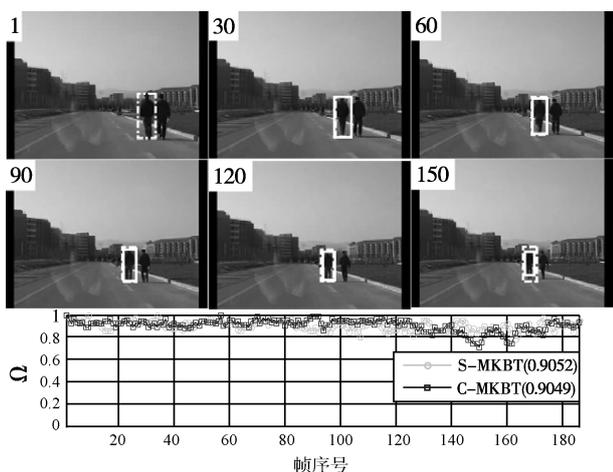


图 4 “Human 1” 的实验结果  
Fig. 4 Results of “Human 1”

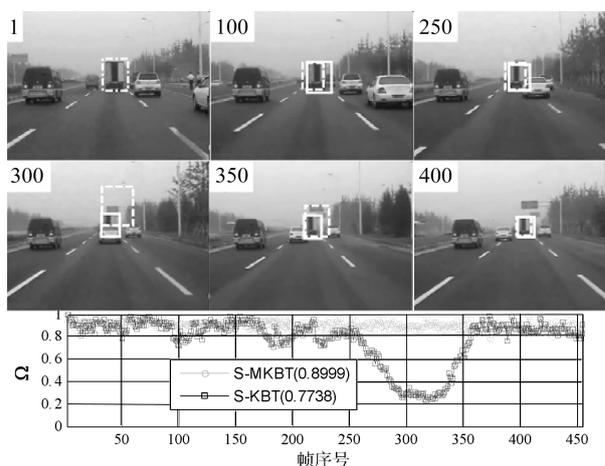


图 7 “Vehicle 3” 的实验结果  
Fig. 7 Results of “Vehicle 3”

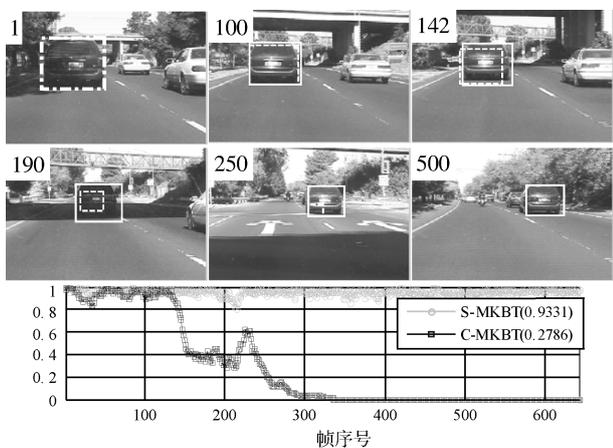


图 5 “Vehicle 2” 的实验结果  
Fig. 5 Results of “Vehicle 2”

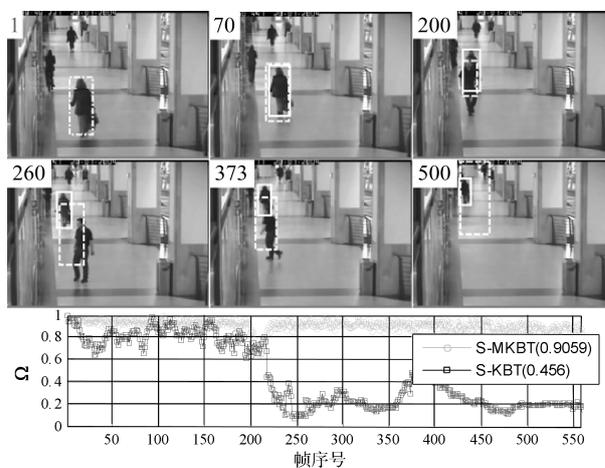


图 8 “Human 3” 的实验结果  
Fig. 8 Results of “Human 3”

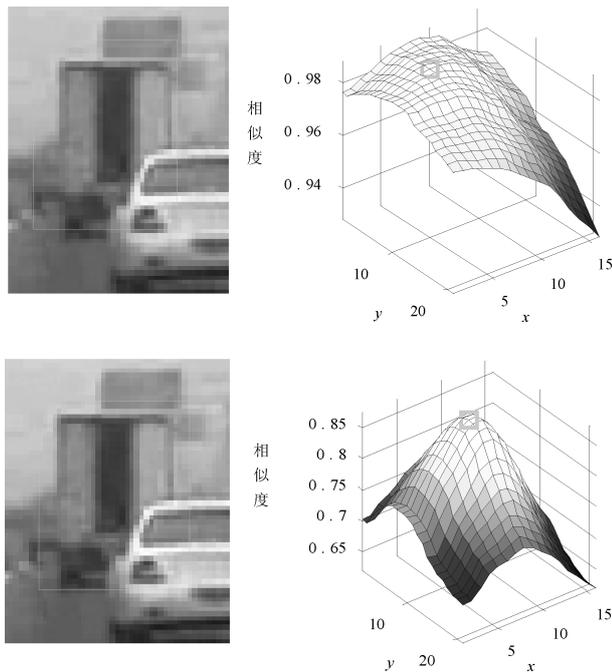


图 9 S-MKBT (下) 和 S-KBT (上) 的相似度曲面  
Fig. 9 Similarity surfaces of S-MKBT (bottom) and S-KBT (top)

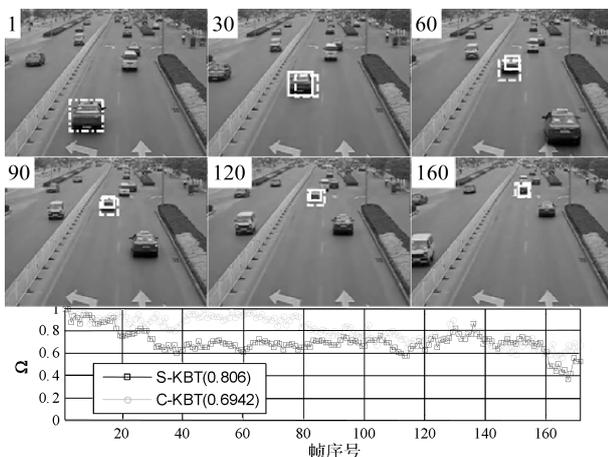


图 10 “Vehicle 4” 的实验结果  
Fig. 10 Results of “Vehicle 4”

S-MKBT 算法的误差则相对较小. 在这段序列的末尾处, C-MKBT 算法跟踪产生漂移, 而 S-MKBT 算法则可以成功地跟踪.

**实验 3.** 比较 S-MKBT 算法和 S-KBT 算法在部分遮挡情况下的跟踪结果, 序列为 “Vehicle 3 (320 × 240, 30 fps)” 和 “Human 3 (384 × 288, 25 fps)”<sup>[14]</sup>, 跟踪结果如图 7 和图 8 所示, 其中实线框表示 S-MKBT 的结果, 虚线框表示 S-KBT 的结果. 观察这两段序列, 相对于单核跟踪, 多核跟踪有两个明显的优点: 1) 定位精度较为准确, 这一点由

图 7 前 200 帧的定位精度可以看出; 2) 对于部分遮挡比较鲁棒, 图 7 中车辆受到一次遮挡, 图 8 中的行人受到两次遮挡, 在这三次遮挡过程中, S-MKBT 算法均能成功地跟踪到目标, 而 S-KBT 算法则失败. 图 9 演示了部分遮挡情况下 S-MKBT 算法和 S-KBT 算法的相似度曲面, 可以看出 S-MKBT 算法的多核表示有两个好处: 1) 鉴别性更好, 这可以由两者的曲面形状看出; 2) 受部分遮挡的影响较小, 单核表示的时候向没有遮挡的部分偏移, 而多核表示时相似度曲面的极值点依然正确地对应目标的位置.

**实验 4.** 比较了目标颜色变化时 S-KBT 算法和 C-KBT 算法, 序列为 “Vehicle 4 (320 × 240, 30 fps)”, 跟踪结果如图 10 所示, 其中实线框表示 S-KBT 的结果, 虚线框表示 C-KBT 的结果. 图中车辆的颜色在运动过程中发生了较大的变化. 在跟踪过程中, S-KBT 算法和 C-KBT 算法都产生了一定的偏差, 但从整体上来说, 基于梯度方向直方图特征的 S-KBT 算法仍然强于基于颜色特征的 C-KBT 算法.

## 5 结论

本文将梯度方向直方图特征引入到核跟踪算法中, 使得跟踪算法对光线的变化较鲁棒. 同时将物体分块的多核跟踪算法使得特征局部化, 对于部分遮挡具有一定的鲁棒性. 对车辆和行人的跟踪实验验证了我们的结论. 本文提出的多核跟踪算法本质上是梯度下降跟踪方法, 对目标的运动速度有较大的限制, 未来的研究包括采用嵌入核跟踪的粒子滤波以跟踪运动较快的物体. 此外, 本文的分块的描述方法中各个块的权重相等, 未来的研究可以为每个块设置不同的权重, 并在跟踪过程中动态地更新每个块的权重以反映相应块的目标和背景的鉴别性.

## References

- 1 Comaniciu D, Ramesh V, Meer P. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, **25**(5): 564–577
- 2 Collins R T. Mean-shift blob tracking through scale space. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. Madison Wisconsin, USA: IEEE, 2003. 234–240
- 3 Peng Ning-Song, Yang Jie, Liu Zhi, Zhang Feng-Chao. Automatic selection of kernel-bandwidth for mean-shift object tracking. *Journal of Software*, 2005, **16**(9): 1542–1550 (彭宁嵩, 杨杰, 刘志, 张凤超. Mean-Shift 跟踪算法中核函数窗宽的自动选取. *软件学报*, 2005, **16**(9): 1542–1550)
- 4 Yilmaz A. Object tracking by asymmetric kernel mean shift with automatic scale and orientation selection. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. Minneapolis, USA: IEEE, 2007. 1–6

- 5 Yang C J, Duraiswami R, Davis L S. Efficient mean-shift tracking via a new similarity measure. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. San Diego, USA: IEEE, 2005. 176–183
- 6 Li Pei-Hua. An improved mean shift algorithm for object tracking. *Acta Automatica Sinica*, 2007, **33**(4): 347–354 (李培华. 一种改进的 Mean Shift 跟踪算法. 自动化学报, 2007, **33**(4): 347–354)
- 7 Wang Yong-Zhong, Liang Yan, Zhao Chun-Hui, Pan Quan. Kernel-based tracking based on adaptive fusion of multiple cues. *Acta Automatica Sinica*, 2008, **34**(4): 393–399 (王永忠, 梁彦, 赵春晖, 潘泉. 基于多特征自适应融合的核跟踪方法. 自动化学报, 2008, **34**(4): 393–399)
- 8 Lowe D G. Distinctive image features from scale-invariant key-points. *International Journal of Computer Vision*, 2004, **60**(2): 91–110
- 9 Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. San Diego, USA: IEEE, 2005. 886–893
- 10 Comaniciu D, Meer P. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, **24**(5): 603–619
- 11 Nguyen H, Smeulders A W M. Robust tracking using foreground-background texture discrimination. *International Journal of Computer Vision*, 2006, **69**(3): 277–293
- 12 Ross D A, Lim J, Lin R S, Yang M H. Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 2008, **77**(1-3): 125–141
- 13 Ess A, Leibe B, Van Gool L. Depth and appearance for mobile scene analysis. In: Proceedings of the 11th International Conference on Computer Vision. Rio de Janeiro, Brazil: IEEE, 2007. 1–8

- 14 CAVIAR Test Case Scenarios. EC Funded CAVIAR Project/IST 2001 37540 Video Database [Online], available: <http://homepages.inf.ed.ac.uk/rbf/CAVIAR/>, December 20, 2008



**贾慧星** 清华大学电子工程系博士研究生. 2003 年获得北京交通大学电子信息工程学院学士学位. 主要研究方向为模式识别, 计算机视觉和智能车辆. 本文通信作者.

E-mail: [jiahx03@mails.tsinghua.edu.cn](mailto:jiahx03@mails.tsinghua.edu.cn)  
(**JIA Hui-Xing** Ph.D. candidate in the Department of Electronic Engineering,

Tsinghua University. He received his B. S. degree from Beijing Jiaotong University in 2003. His research interest covers pattern recognition, computer vision, and intelligent vehicle. Corresponding author of this paper.)



**章毓晋** 清华大学电子工程系教授. 主要研究方向为图像工程 (图像处理、图像分析、图像理解及其技术应用).

E-mail: [zhang-yj@tsinghua.edu.cn](mailto:zhang-yj@tsinghua.edu.cn)  
(**ZHANG Yu-Jin** Professor in the Department of Electronic Engineering, Tsinghua University. His research interest covers image engineering that includes image processing, image analysis and image understanding, as well as their applications.)

His research interest covers image engineering that includes image processing, image analysis and image understanding, as well as their applications.)