

DOI: 10.7652/xjtuxb201606008

周界视频监控中人员翻越行为检测算法

张泰, 张为, 刘艳艳

(天津大学电子信息工程学院, 300072, 天津)

摘要: 针对周界视频监控应用环境特殊的问题,提出了一种人员翻越行为的检测方法。该方法采用“目标检测-人员跟踪-轨迹分析”的流程。在算法的人员跟踪过程中,将混合高斯模型得到的前景区域与KLT光流法得到的特征点运动信息结合起来,提出了一种新型跟踪算法。该算法仅使用图像的灰度信息作为输入,一定程度上能够适应目标形变及遮挡,并具有很强的鲁棒性和实时性;在算法的轨迹分析过程中,结合墙体位置信息与先验知识设计了一种新的轨迹分析的方法,不需要通过在线学习可直接对轨迹进行分析。实验结果表明,该算法在测试视频集上检测准确率超过93%,与现有方法相比,能更好地适应实际应用中复杂的环境条件。

关键词: 视频监控;异常检测;跟踪;轨迹分析

中图分类号: TP391.4 **文献标志码:** A **文章编号:** 0253-987X(2016)06-0047-07

Detection Algorithm of Fence Climbing for Perimeter Video Surveillances

ZHANG Tai, ZHANG Wei, LIU Yanyan

(School of Electronic Information Engineering, Tianjin University, Tianjin 300072, China)

Abstract: An algorithm to detect fence climbing is proposed to meet the specific requirements of perimeter video surveillance. The algorithm bases on a process like “object detection-tracking-trajectory analysis”. The main scheme of the algorithm combines foreground areas obtained from Gaussian mixture model with displacements of feature points obtained from KLT algorithm in personnel tracking stage. In this way, the tracking algorithm just takes advantage of the image grey scale information, can adapt deformation and shade of the objects in a certain extent and has a strong robustness as well as real-time effect. A new method for the analysis of trajectory is proposed in trajectory analysis stage by using the information of wall location and other prior knowledge instead of traditional online learning algorithms. Experimental results show that the proposed algorithm is more suitable for noisy environments than other existing algorithms. A recognition rate above 93% is obtained on a test video set.

Keywords: video surveillance; anomaly detection; tracking; trajectory analysis

智能视频监控是计算机视觉领域中的重点研究课题之一,它以物体检测、识别以及跟踪为基础,对视频中对象的行为进行分析和描述^[1-2]。目前,在学校、银行、商店、车站、港口等一些重要的公共场所普遍架设了大量监控摄像机,为智能视频监控的实现

提供了硬件条件。

对于检测人员翻越围墙或护栏的行为,传统的周界安防有很多种解决方案,例如红外对射方案、微波对射方案、电子围栏、电网等,但是它们都存在很多不足:防护等级较低,容易跨越或规避;不适合特

收稿日期: 2016-03-03。 作者简介: 张泰(1992—),男,硕士生;张为(通信作者),男,教授。 基金项目: 天津市科技支撑计划重点资助项目(14ZCZDSF00020)。

网络出版时间: 2016-04-15

网络出版地址: <http://www.cnki.net/kcms/detail/61.1069.T.20160415.1612.014.html>

殊天气,受温度、光照、天气等影响,误报率高;功耗很大;有危险性;无法判断报警是否为误报,需派遣人员实地查看。如果能利用现有的监控平台获取视频并采用算法对人体行为进行分析来实现监控和报警,就可以较好地解决这些问题。

目前,国际上针对人员翻越围墙的行为检测算法的研究还处于起步阶段。现有的一系列研究^[3-5]都采用行为分析的思路,先根据人体轮廓剪影计算出人体星形骨架特征,然后利用这个特征训练一个隐马尔可夫模型(hidden markov model, HMM),将视频中人体的动作分为行走、攀爬、跨越、下降4种状态。当攀爬、跨越、下降3种状态连续出现的时候,就可以认为发生了人员翻越围墙的行为,但是这3篇文章考虑的模型比较理想,也只在仅有一个人的理想环境中做了测试,在实际应用环境中还有待改善。文献[6-7]将该问题归入行为分析的领域,认为可以提取相应特征,然后用HMM或贝叶斯网络对其进行建模分析,但是这种方式同样会面临目标遮挡严重、特征难以提取的问题。文献[8]提出对运动物体计算光流场,然后对光流场进行建模并用分类器进行分析,进而检测一些异常行为,这种全局性的特征适合分析遮挡严重的情况,可以进行借鉴,但是光流场计算复杂度高,难以做到实时处理。

实际中的人员翻越行为一般有如下特点:全天候发生;场景较复杂,有其他人员或车辆影响;目标人员受墙体或围栏的遮挡严重,自身形变非常大,运动没有明显的规律性;持续时间短,速度快。由于这些原因,设计算法时需考虑以下方面:受夜间红外成像的限制,处理视频的过程中只能使用图像灰度信息;跟踪算法需要具有鲁棒性,能够适应目标外观的变化以及部分遮挡,还需要具有较高的实时性。

根据以上分析,本文提出了一种新颖的人员翻越行为检测算法,采用“目标检测-人员跟踪-轨迹分析”的流程,针对人员由外至内的翻越行为的检测达到了令人满意的效果。

1 算法概述

算法分为目标检测、人员跟踪、轨迹分析3个部分。首先手工画线标注墙体的位置;接着,根据墙体位置,对获取的视频图片进行裁剪和缩放以减小算法的计算量;然后,对缩放之后的图片序列使用混合高斯法进行背景建模^[9]、形态学滤波、阴影去除,这就得到了图像中的运动前景区域。

随后使用前景块外接矩形长与宽、面积、形状复

杂度等信息进行了粗略的分类,淘汰掉一些过大、过小或形状特殊的前景块,然后计算剩余团块的外接矩形。对这些矩形进行位置判断,当其与标注出的墙体位置满足一定关系的时候进行头部检测,将检测出头部的目标交给跟踪部分,使用跟踪算法对其进行跟踪并记录他的运动轨迹,接着对跟踪轨迹进行分析处理,判断这一轨迹是否近似于翻越人员的轨迹。如果是,就可以输出结果并进行报警。

2 算法的具体设计

本算法的主要工作集中在3个部分:①训练了专用的分类器,用于检测翻越人员的头部;②设计了一种新型跟踪算法,将混合高斯模型法得到的前景区域与KLT光流法(Kanade-Lucas-Tomasi, KLT)得到的特征点的运动信息结合起来,得到了仅使用灰度图像作为输入、适应目标形变及遮挡、鲁棒性和实时性强的跟踪算法;③针对本问题设计了一种新的轨迹分析的方法,不需要通过在线学习,直接对轨迹进行分析。

2.1 头部检测分类器的训练

Adaboost算法^[10]检测速度较快,本算法中使用这种方法,能够满足视频处理的实时性。

人员在翻越围墙时头部姿态与正常站立行走时有一定的差别,因此笔者构造了专用的训练集。正样本共1300张图片,来自之前录制的一些人员翻越围墙的视频,以及人员做出翻越护栏、弯腰、鞠躬等动作时头部的图片。负样本是以无人的街道、汽车、树木、围栏、标志牌以及人物阴影为主题的图片。对这些图片提取方向梯度直方图特征,训练一个20阶的Adaboost分类器。该分类器能较好地检测正在翻越围墙的人员的头部,并且具有较高的效率。

2.2 基于混合高斯背景法和KLT算法的跟踪算法

近年来,学界提出了大量优秀的跟踪算法^[11-14],但是这些算法中,有些需要物体的颜色信息,有些对物体的形变遮挡缺乏适应能力,还有的实时性不足,因而本文选择了一种比较经典的实时跟踪算法作为基础,结合应用环境对其进行改进。

KLT算法^[15]是一种实时性非常优秀的特征点跟踪算法。根据算法的具体应用环境,本文对它提出了两个方面的改进:一是提出了一种重新计算特征点的策略;二是得到跟踪目标对应的前景区域,并将这个前景区域与KLT算法的跟踪结果融合。

2.2.1 基于KLT算法的特征点跟踪策略 本算法通过在一定时机重新计算特征点来保证特征点在

跟踪对象图像中梯度较大的部分均匀分布,进而确保随后计算的位移矢量能够代表跟踪对象整体的位移,同时尽可能地减小计算特征点带来的计算量。

在检测到头部之后,就认为头部所在的前景区域外接矩形中包含了一个有翻越倾向的人,将这个矩形区域放入跟踪队列进行跟踪。首先使用 KLT 算子检测这个矩形中的特征点,接下来使用运动前景对这些特征点做筛选,保留落在运动前景之上的特征点,这就组成了需要进行跟踪的点列

$$P_{in} = \{p_{in1}, p_{in2}, p_{in3}, p_{in4}, \dots\} \quad (1)$$

使用 LK 光流法进行跟踪,得到跟踪成功的点列

$$P_{out} = \{p_{out1}, p_{out2}, p_{out3}, p_{out4}, \dots\} \quad (2)$$

不需要在每一帧图像中都使用 KLT 算子进行特征点检测,跟踪得到的点列经过后面步骤的选择可以直接作为下一次跟踪的输入,但是随着跟踪的进行,人体本来被遮挡住的部分可能会逐渐显露出来,这些部分没有特征点覆盖。同时,在每次跟踪中也会丢失一部分特征点,这会造成特征点过少或分布不均匀,这时需要重新计算特征点。

本算法构造了两个 40×40 像素大小的直方图,通过它们之间的差来描述正在跟踪的矩形区域中特征点分布的均匀程度。其中图像梯度分布直方图 G 是由正在跟踪的矩形区域的图像的梯度图缩放至 40×40 像素得到的,特征点覆盖直方图 H 按以下步骤计算得到。

步骤 1 将原图中的特征点按坐标对应关系放置在 40×40 像素直方图中,使该像素点的灰度值为最大值,然后使用 $\Sigma=2$ 的高斯核的高斯滤波器对该直方图进行滤波,得到直方图 H^{temp} 。

步骤 2 通过 H^{temp} 计算 H ,即通过比例系数 r 使得 G, H 两个直方图所有像素之和相等

$$r = \sum G_{ij} / \sum H_{ij}^{\text{temp}} \quad (3)$$

$$H_{ij} = rH_{ij}^{\text{temp}} \quad (4)$$

由于进行了归一化,这时

$$\sum G_{ij} = \sum H_{ij} = \sum_{40 \times 40} \quad (5)$$

步骤 3 得到了直方图 G 与 H 之后,计算直方图 G 和 H 之间的差,这个运算定义如下

$$\|G - H\| = \sum \|G_{ij} - H_{ij}\| \quad (6)$$

$$\|G_{ij} - H_{ij}\| = \begin{cases} 0, & H_{ij} > G_{ij} \\ G_{ij} - H_{ij}, & \text{其他} \end{cases} \quad (7)$$

最后对结果进行归一化

$$R_{\text{diff}} = \|G - H\| / \sum_{40 \times 40} \quad (8)$$

该结果可以描述跟踪得到的矩形区域中的特征点分布的均匀情况。当 $R_{\text{diff}} > T_{\text{th}}$ 时,意味着图片中的特征点数过少或者分布很不均匀,这时重新计算特征点,得到分布更加均匀的结果。

执行改进的跟踪策略,得到跟踪成功的点列 P_{out} 后,可以计算跟踪目标的位移矢量。对于点列 P_{out} 以及它所对应的 P_{in} 中的点,可以计算每一特征点的位移,得到位移矢量序列 D ,但是有一些特征点在跟踪过程中会出现漂移,同时人员在翻越围墙的过程中身体每一部分的位移并不一致,因此 D 中存在大量噪声。将这些位移矢量通过 MeanShift 算法^[16]进行聚类,假设 P_{out} 中共有 M 个点,选择聚类结果中点数最多的一类,这一类中有 N 个点,将该类的聚类中心 v 作为人物的位移矢量,此方法可以有效地去除噪声。

2.2.2 跟踪目标对应的前景区域的选择 图 1 给出了前景检测的一些结果。可以看出,有时前景提取的结果并不准确,这使前景信息的使用变得比较困难。下面将阐述如何得到可以使用的前景信息。

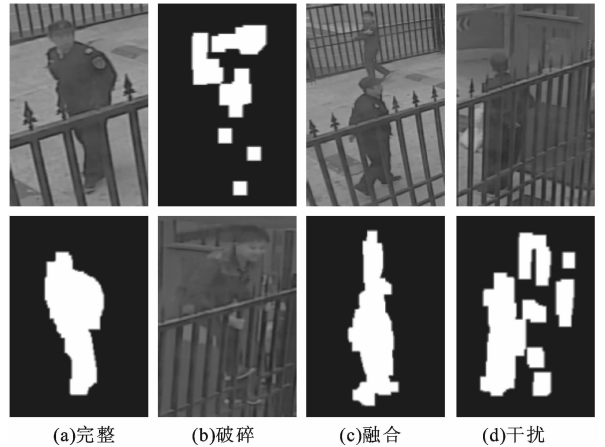


图 1 前景提取中常出现的 4 种情况

本节从前景检测的结果中计算出跟踪目标对应的前景区域外接矩形,这一工作可以分为 3 个步骤。**步骤 1** 使用 Kalman 滤波器来对跟踪目标的位置进行预测,得到预测矩形 r_k 。

Kalman 滤波可以用于对系统状态的预测。跟踪结果用矩形框 r_{result} 表示。每获得一张新的图像 I ,就将上一帧图像中跟踪的最终结果 r_{result}^p 的中心 $c_{\text{result}}^p = (c_x, c_y)^p$ 作为测量值输入 Kalman 滤波器,并使用 Kalman 滤波器输出这一帧的位置预测 c_k 。以这个位置为中心,保持 r_{result}^p 的长与宽不变,可以得到一个预测矩形 r_k 。

步骤 2 在所有前景团块的外接矩形中,找出与步

骤1得到的预测矩形有重叠的矩形,并计算这些矩形所有“合理的”合并的结果。

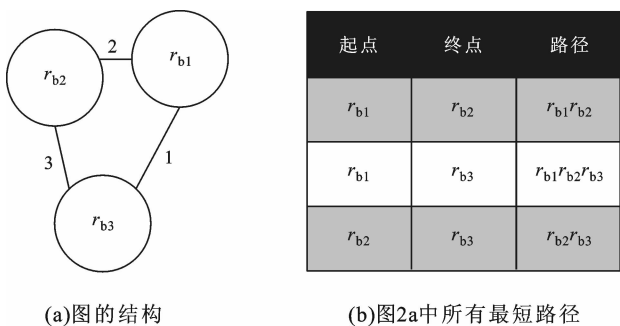
前面已经得到前景区域的外接矩形。步骤1得到的预测矩形 r_k 会与其中的一些外接矩形相交,如图2所示。把与 r_k 相交的矩形记为

$$R_b = \{r_{b1}, r_{b2}, r_{b3}, r_{b4}, \dots\} \quad (9)$$

它们表示在这一帧中可能属于跟踪目标的前景矩形。本节最终要得到的跟踪目标对应的前景区域的外接矩形,就是 R_b 中的某一个矩形或由 R_b 中的矩形合并得到(这里两个矩形的合并指计算这两个矩形的外接矩形)。将 R_b 中的矩形任意组合,可以得到

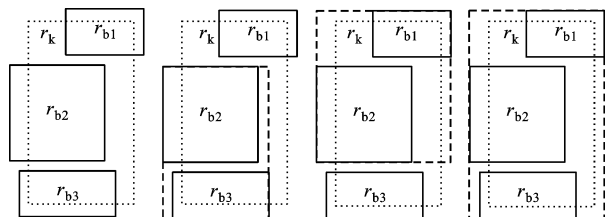
$$R_{cb} = \{r_{b1}, r_{b2}, r_{b3}, \dots, r_{b1}r_{b2}, \dots, r_{b1}r_{b2}r_{b3}, \dots\} \quad (10)$$

但是,有些矩形组合合并之后的区域会包含本没有出现在该组合中的矩形,或者会将两个距离很远的矩形合并起来而没有包括距离它们各自都很近的一个矩形。算法认为这些组合是不合理的组合,它会干扰算法后续的工作,例如图2c中 r_{b1} 与 r_{b2} 的组合就属于这种情况,使用图算法可以计算得到合理的组合^[17]。



(a)图的结构

(b)图2a中所有最短路径



(c)前景矩形以及前景矩形的几种合理组合

图2 前景区域合理组合的计算

首先构建一个无向图 $G = \{V, E, \omega\}$ 。顶点 $V = \{V_1, V_2, \dots, V_n\}$ 表示 R_b 中的前景矩形,边 $E = \{E_1, E_2, \dots, E_m\}$ 表示这些矩形之间的距离。对每一条边,都有一个权重系数, ω_{ij} 表示前景矩形 V_i 和 V_j 之间距离的大小。从一个顶点出发,到另一个顶点的最短路径可以通过 Dijkstra 算法求得。每一条最短路径所经过的顶点的集合,都是一种合理的矩形

组合,如图2b所示。

将式(9) R_b 中所有的矩形与由 Dijkstra 算法组合得到的矩形组合作为一个整体,记为

$$R_{rcb} = \{r_{b1}, r_{b2}, r_{b3}, \dots, r_{b1}r_{b2}, \dots\} \quad (11)$$

它其中的元素是 R_b 中的矩形,以及由 R_b 中的多个矩形经过合理的合并形成的矩形。

步骤3 从步骤2得到的结果 R_{rcb} 中选出最能代表跟踪目标前景区域的元素,设 R_{rcb}^i 中特征点的个数为 N_{rcb}^i ,上标 i 表示它是 R_{rcb} 中的第 i 个元素。可以定义一个公式来计算这些元素与跟踪目标对应区域的吻合程度

$$s^i = \left(\left(\alpha_S (s_S^i)^2 + \alpha_N (s_N^i)^2 + \alpha_E \sum_j (s_E^i)^2 \right) / (\alpha_S + \alpha_N + \alpha_E) \right)^{1/2} \quad (12)$$

其中

$$s_S^i = \text{area}(R_{rcb}^i \cap r_k) / \text{area}(R_{rcb}^i) \quad (13)$$

$$s_N^i = N_{rcb}^i / M \quad (14)$$

$$s_E^i = 1 - \text{area}(R_b^i - r_k) / \text{area}(R_b^i) \quad (15)$$

式(12)中的 α_S 、 α_N 和 α_E 为3个权重系数,表示公式中3部分的权重;式(13)描述矩形 R_{rcb}^i 与预测矩形 r_k 的重合程度;式(14)描述 P_{out} 中落在矩形 R_{rcb}^i 内的点的个数, P_{out} 中共有 M 个点;式(15)表示组成 R_{rcb}^i 的 j 个矩形没有被 r_k 覆盖的面积与其自身面积之比。

对于 R_{rcb} 中的每个元素,按照式(12)计算 s^i 的值,并选择其中 s^i 值最大的元素作为跟踪目标的前景区域,将这个元素记为 r_B 。这就将分成几部分的前景融合起来,解决了前景破碎的问题,并对前景干扰问题有一定的效果。

2.2.3 基于权重计算的最终结果确定 经过前面的计算已经得到了4个可以使用的信息:2.2.1节得到的位移矢量 $\bar{v}$;2.2.2节得到的 Kalman 滤波器对跟踪目标位置的预测 c_k 和跟踪目标对应的前景矩形区域 r_B ;上一帧跟踪的最终结果 r_{result}^p (包括位置和大小两部分信息)。在这一步中算法综合这些信息,得到最终的跟踪结果 r_{result} ,即得到 r_{result} 的中心位置 c_{result} 和大小 S_{result} 2个量。

算法借鉴了民主融合策略^[18]的合并方式,但与民主决策制算法的不同之处在于,每一项信息的权重并不是在线更新的,而是直接由当前帧的一些信息以及一些先验知识计算得到的。由于这些信息在每一帧中变化很大且缺乏规律,此方法比在线更新的策略更适合这个问题。

定义权重函数的表示形式为

$$\omega_z = \exp(-u_z \|Q\|) \quad (16)$$

式中: z 表示参与融合的各个信息; $\|Q\|$ 表示对信息质量的一种评价; u_z 是一个经验系数。

首先计算 r_{result} 的中心位置 c_{result} ,它由跟踪目标对应的前景区域的中心位置 c_B 、位移矢量 $\mathbf{v}$ 以及由 Kalman 滤波器预测的位置 c_k 三者融合得到。

对于前景矩形中心位置 c_B ,权重系数的形式为 $\omega_B^c = \exp\{-u_B [((w-\bar{w})/\bar{w})^2 + ((h-\bar{h})/\bar{h})^2]\}$ (17)

式中: w, h 为 r_B 的宽与高; $\bar{w}$ 和 $\bar{h}$ 分别是按照每帧 20% 和 30% 的固定速率更新的平均宽与高。得到的前景矩形的长与宽越接近平均的长与宽,前景矩形的信息质量就越高,权重也就越大,本算法中 $u_B = 15$ 。

对于位移矢量 $\mathbf{v}$,权重系数的形式为

$$\omega_d = \exp[-u_d(1 - N/M)] \quad (18)$$

式中: M 和 N 为 P_{out} 中点的总数和聚类结果中包含的点的个数。聚类中得到的点占总数的比例越大,位移矢量的信息质量就越高,权重也就越大,本算法中 $u_d = 2$ 。

对于由 Kalman 滤波器预测的位置 c_k ,权重系数的形式为

$$\omega_k^c = \exp[-u_k d(\mathbf{v}, \mathbf{v}_k)](1 - \omega_d^c) = \exp\left[-u_k \frac{(x_d - x_k)^2 + (y_d - y_k)^2}{x_d^2 + x_k^2 + y_d^2 + y_k^2}\right] \cdot \{1 - \exp[-u_d(1 - N/M)]\} \quad (19)$$

式中: $\mathbf{v}_k$ 为原位置中心点到 Kalman 滤波器预测位置中心点的位移。当位移矢量权重较高时,由于可以比较明确地判断物体的位移,Kalman 滤波器预测结果的权重较低;否则,提高 Kalman 滤波器预测结果的权重,对物体的位置进行预测,本算法中 $u_k = 2$ 。

最后对式(17)~式(19)计算得到的3个权重系数进行归一化,得到归一化权重系数

$$(\omega_z^c)^* = \omega_z^c / (\omega_B^c + \omega_d^c + \omega_k^c) \quad (20)$$

其中 z 在 B, d, k 中选择,最终得到 r_{result} 中心点为

$$c_{\text{result}} = (\omega_B^c)^* c_B + (\omega_d^c)^* c_d + (\omega_k^c)^* c_k \quad (21)$$

式中: c_B 为跟踪目标对应的前景矩形区域的中心; c_d 为 r_{result} 的中心经过位移矢量 $\mathbf{v}$ 位移得到的位置; c_k 为 Kalman 滤波器对这一帧跟踪目标位置的预测。

接下来计算 r_{result} 的大小 S_{result} ,它是由跟踪目标对应的前景区域 r_B 的大小和上一帧最终结果 r_{result}^p 的大小两者融合得到,计算思路与前文一致。

对于跟踪目标对应的前景区域 r_B 的大小,权重系数的形式与进行位置预测时相同

$$\omega_B^S = \omega_B^c \quad (22)$$

对于 r_{result}^p 的大小 S_p ,权重系数为一个固定值

$$\omega_p^S = \Omega \quad (23)$$

本算法中取 $\Omega = 0.7$ 。对这两个权重进行归一化

$$(\omega_z^S)^* = \omega_z^S / (\omega_B^S + \omega_p^S) \quad (24)$$

z 在 B, p 中选择,最终得到的 r_{result} 大小 S_{result} 为

$$\tau S_{\text{result}} = (\omega_B^S)^* S_B + (\omega_p^S)^* S_p \quad (25)$$

式中: τ 是一个时间常数,用来表示 r_{result} 的更新速率。每隔 τ 时间对结果进行更新,因为更新速度太快容易受到前景提取的噪声影响。本算法中,1 s 更新 2.5 次, S_{result} 计算完成。

至此得到了跟踪矩形 r_{result} 的位置和大小信息,也就得到了最终的跟踪结果。这个结果综合了各部分的信息,解决了前景提取中融合与干扰的问题。融合过程在处理具体帧图片时的结果如图3所示。

使用 r_{result} 对 P_{out} 中的特征点进行筛选,去除落在 r_{result} 以外的点,剩余点还能应用在下一帧的跟踪中。

2.2.4 退出跟踪的策略 本算法给出了2个退出跟踪的条件:一是跟踪结果 r_{result} 中包含的前景面积过小的情况;二是一次性跟踪丢失大量点的情况,它们表示人物消失、严重遮挡或者跟踪出现错误。

2.3 基于先验知识的轨迹分析

现有的轨迹分析方法大多需要通过离线或在线学习来获取轨迹的规律,而本算法受其检测行为及应用环境的限制,无法获取代表全部环境和足够多的学习样本,所以算法使用墙体的位置信息以及一些先验知识来对轨迹进行分析。

在跟踪过程中,每处理一帧视频,得到一段轨迹。在轨迹分析中,首先去除掉太短或太长的轨迹,然后对轨迹进行平滑,接着判断轨迹的后几段中是否与标注出的墙体位置的线段有交点。如果存在交点,说明有可能是人员翻越围墙形成的轨迹,计算下列特征进行进一步判断。

特征1 轨迹最高点在轨迹中间,即轨迹中间有一点在图像中高于初始点和结束点的位置。

特征2 对轨迹最高点和最低点取平均,轨迹中有50%以上的点高度高于这个平均值。

特征3 轨迹的高度与宽度应满足

$$\left. \begin{aligned} l/h &< 5, \\ l/h &< (5 - 0.6 \times (10 - |\theta|)/2), \text{其他} \end{aligned} \right\} \quad |\theta| > 5^\circ \quad (26)$$

式中: l 为轨迹宽度,即最左侧点与最右侧点水平坐标之差的绝对值。 h 为轨迹高度,定义与 l 类似。 θ 为标注出的墙体位置的线段与水平方向所成的夹角。

特征 4 轨迹初始点与结束点之间的距离 D 应满足

$$\left. \begin{aligned} h \times (0.4 - 0.05(|45 - |\theta||/15)) < D, \quad |\theta| > 5^\circ \\ h \times (0.3 - 0.12(|10 - |\theta||/2)) < D, \quad \text{其他} \end{aligned} \right\} \quad (27)$$
$$\left. \begin{aligned} h \times 4.5 > D, \quad |\theta| > 5^\circ \\ h \times (0.4 - 0.05(|45 - |\theta||/15)) < D, \quad \text{其他} \end{aligned} \right\} \quad (28)$$

其中 h 和 θ 的定义与上一步相同。

如果这些特征全部满足,就将轨迹判定为人员翻越围墙形成的轨迹,这时停止对该目标的跟踪并保存当前帧图片,发出警报信号;否则,继续跟踪并记录目标轨迹。

3 实验结果与分析

为了验证算法的有效性,在天津市内选择了 3 处合适的场景,架设海康网络摄像头及硬盘录像机,录制了 8 段共含有 30 次翻越实验的视频(其中 8 次为夜间拍摄),以及 12 段共计 1 h 左右的没有出现翻越行为的视频对算法进行实测。其中 A 场景中除了翻越人员,有 2~3 人在四周活动;B 场景中,有大量人员及车辆在附近过往,夜间场景还伴有车灯等光源干扰;场景 C 中,大量人员、车辆在栅栏附近经过,场景中还有树枝、线缆的抖动。实验视频中不包含用来训练头部检测分类器的图片。

实验中使用的摄像头帧率为 25 帧/s,场景图片和其中标出的墙体位置如图 4 所示。算法以标准 C++ 程序实现,并使用了 OpenCV 计算机视觉库,机器配置为 Intel 处理器 i5-2400@3.10 GHz,6 GB 内存台式机,未用显卡做加速。实验中算法专门检测翻入行为,即检测从墙的外侧翻入到有摄像头的一侧。

前景区域的选择与各个信息融合的效果如图 3 所示。即使出现前景破碎、融合和干扰的情况,本文算法也能合理地利用前景信息,结合特征点的运动信息正确地跟踪人体的位置,而文献[3-5]中的星形骨骼表示算法在这些情况下完全无法准确地计算人体星形骨架特征,后面的步骤也无法进行。实验结果见表 1,对比实验见表 2。

总计 30 次翻越行为,算法共检出 28 次,检出率在 93.3% 以上,并且全部 20 段视频中无任何误报。

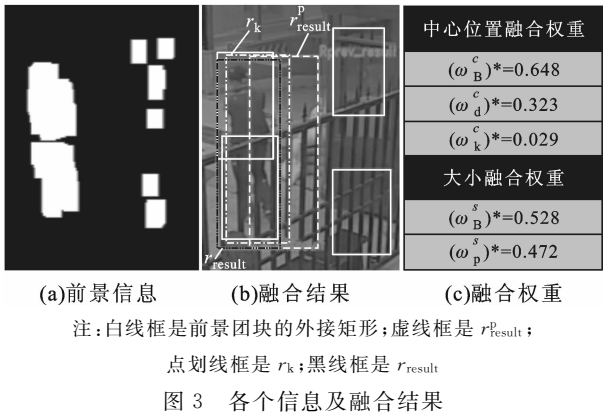
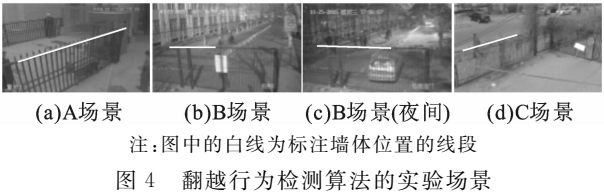


表 1 翻越行为检测算法的实验结果

视频编号	原始尺寸/ 像素	缩放尺寸/ 像素	检出率	平均耗时/ ms · 帧 ⁻¹
A	960×576	451×288	2/2	8.97
B1	1 920×1 080	492×540	2/2	13.54
B2	1 920×1 080	492×540	2/2	14.20
B3	1 280×720	428×360	4/4	7.57
B4	1 280×720	428×360	4/4	7.70
B5	1 280×720	428×360	7/8	7.45
B6(夜)	1 280×720	428×360	4/4	7.57
B7(夜)	1 280×720	428×360	3/4	10.58
C1-C12	1 920×1 080	540×517	0/0	14.92

表 2 翻越行为检测算法与其他方法的实验对比

检测方法	总帧数	正确率/%
2SS+HMM ^[3]	约 10 000	25/(18 行走+10 攀爬)=89.3%
VSS+HMM ^[5]	12 652	137/(90 行走+50 攀爬)=97.9%
本文方法	178 575	28/(N 行走+30 攀爬)=93.3%



由此可见,本文算法对翻越行为的检测效果良好,并具有很强的鲁棒性,对夜间场景也能正常处理。

由表 1 可以看出,算法平均处理一帧图片的时间小于 15 ms,每秒可以处理 65 帧以上,完全满足实时性要求。

表 2 中本文算法的检出率在现有的两种方法之间,其他两种算法的数据来自于文献[3,5],但是实际上,其他两种算法统计出的正确率来自于它对行走和翻越两种行为识别结果的统计,而本文方法只

统计了检测翻越行为的结果。在测试视频中,有数百个行人以及各种样式的车辆从围栏附近经过,均未发生误报。算法将其视为非翻越行为,但未计入最后结果。本文算法测试环境非常复杂,而另两种方法只在一个人活动的理想环境中做了简单测试,墙体的角度非常单一,测试视频长度也远远小于本文算法。实际上,在实际应用场景中本文算法的应用效果要远远好于另两种算法。

总的来说,对于人员翻入行为,算法具有很好的检测效果和很强的鲁棒性,并且算法整体的实时性非常良好,有着重要的实用价值。

参考文献:

- [1] HU W, TAN T, WANG L, et al. A survey on visual surveillance of object motion and behaviors [J]. IEEE Transactions on Systems, Man, and Cybernetics: Part C Applications and Reviews, 2004, 34(3): 334-352.
- [2] KIM I S, CHOI H S, YI K M, et al. Intelligent visual surveillance—a survey [J]. International Journal of Control, Automation and Systems, 2010, 8(5): 926-939.
- [3] YU E, AGGARWAL J K. Detection of fence climbing from monocular video [C]//Proceedings of the 18th 2006 International Conference on Pattern Recognition. Piscataway, NJ, USA: IEEE, 2006: 375-378.
- [4] YU E, AGGARWAL J K. Recognizing persons climbing fences [J]. International Journal of Pattern Recognition and Artificial Intelligence, 2009, 23(7): 1309-1332.
- [5] YU E, AGGARWAL J K. Human action recognition with extremities as semantic posture representation [C]//Proceedings of the IEEE Computer Vision and Pattern Recognition Workshops. Piscataway, NJ, USA: IEEE, 2009: 1-8.
- [6] CHENG Guangchun, WAN Yiwen, SAVDAGAR A N, et al. Advances in human action recognition: a survey [J]. ArXiv Preprint ArXiv, 2015: 1501.05964.
- [7] VISHWAKARMA S, AGRAWAL A. A survey on activity recognition and behavior understanding in video surveillance [J]. The Visual Computer, 2013, 29(10): 983-1009.
- [8] ZHANG T, YANG Z, JIA W, et al. A new method for violence detection in surveillance scenes [J]. Multi-media Tools and Applications, 2015: 1-23.

- [9] ZIVKOVIC Z. Improved adaptive Gaussian mixture model for background subtraction [C]//Proceedings of the 17th 2004 International Conference on Pattern Recognition. Piscataway, NJ, USA: IEEE, 2004: 28-31.
- [10] VIOLA P, JONES M. Fast and robust classification using asymmetric AdaBoost and a detector cascade [J]. Advances in Neural Information Processing System, 2002, 14: 1311-1318.
- [11] KALAL Z, MIKOLAJCZYK K, MATAS J. Tracking-learning-detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(7): 1409-1422.
- [12] HARE S, SAFFARI A, TORR P H S. Struck: structured output tracking with kernels [C]//Proceedings of the 2011 IEEE International Conference on Computer Vision. Piscataway, NJ, USA: IEEE, 2011: 263-270.
- [13] 查宇飞, 杨源, 王锦江, 等. 利用密度描述符对应的视觉跟踪算法 [J]. 西安交通大学学报, 2014, 48(9): 13-18.
ZHA Yufei, YANG Yuan, WANG Jinjiang, et al. A visual object tracking algorithm using dense descriptors [J]. Journal of the Xi'an Jiaotong University, 2014, 48(9): 13-18.
- [14] CEHOVIN L, KRISTAN M, LEONARDIS A. Robust visual tracking using an adaptive coupled-layer visual model [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(4): 941-953.
- [15] SHI J, TOMASI C. Good features to track [C]//Proceedings of the 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE, 1994: 593-600.
- [16] CHENG Y. Mean shift, mode seeking, and clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 17(8): 790-799.
- [17] FOGGIA P, PERCANNELLA G, SAGGESE A, et al. Real-time tracking of single people and groups simultaneously by contextual graph-based reasoning dealing complex occlusions [C]//Proceedings of the 2013 IEEE International Workshop on Performance Evaluation of Tracking and Surveillance. Piscataway, NJ, USA: IEEE, 2013: 29-36.
- [18] TRIESCH J, VON DER MALSBURG C. Democratic integration: self-organized integration of adaptive cues [J]. Neural computation, 2001, 13(9): 2049-2074.

(编辑 武红江)