

网络优先数字出版时间: 2016-01-27

网络优先数字出版地址: <http://www.cnki.net/kcms/detail/45.1075.N.20160127.1616.018.html>

自适应实时压缩感知跟踪算法*

Adaptive Real-time Compressive Sensing Tracking Algorithm

梁剑平, 朱晓姝

LIANG Jianping, ZHU Xiaoshu

(玉林师范学院计算机科学与工程学院, 广西玉林 537000)

(School of Computer Science and Engineering, Yulin Normal University, Yulin, Guangxi, 537000, China)

摘要:【目的】解决实时压缩感知跟踪算法分类器无法适应目标外观变化及过更新的问题。【方法】根据当前跟踪结果目标模型的哈希指纹与上一帧目标模型的哈希指纹之间的汉明距离(Hamming distance),在线实时调整分类器,以提高实时压缩感知目标跟踪算法的自适应能力。【结果】自适应实时压缩感知跟踪算法的跟踪成功率比实时压缩感知跟踪算法提高13%,在目标大小为40 pixel×43 pixel时,跟踪速率为37 fps,满足实时性要求。【结论】本研究建立的方法在背景中存在与目标有一定相似性的物体,且目标姿态、纹理变化和光照变化较大等情况下,能快速获取跟踪目标,并且具有较强的鲁棒性和准确性。

关键词: 压缩感知 目标跟踪 哈希指纹 汉明距离 自适应 实时压缩

中图分类号: TP391 **文献标识码:** A **文章编号:** 1002-7378(2016)01-0042-07

Abstract: 【Objective】To solve the problem of real-time compressed sensing tracking algorithm, which is the inadaptability of classifier to the changes of the target appearance and the over update. 【Methods】Base on Hamming distance between the Hash fingerprints of current target and original one, classifier is adjusted in real time, which improved the adaptive capacity of the real-time compressed sensing tracking. 【Results】As compared with the real-time compressive sensing tracking algorithm, the proposed algorithm improves the success rate by 13%, and average computing frame rate is 37 frames when the target scale is 40 pixel×43 pixel, which satisfies the requirements of real-time tracking. 【Conclusion】The proposed algorithm is tested with variant video sequences. The results show that the proposed algorithm is capable of speedily and accurately capturing the tracking target by target gestures, textures, or significant light change.

Key words: compressive sensing, target tracking, Hash fingerprint, Hamming distance, adaptive, real-time compressive

0 引言

【研究意义】目标跟踪技术在计算机视觉研究领

域中有着非常重要的地位,已被广泛应用在车辆导航、视频监控以及人机交互等方面^[1]。近几十年来,学者们提出了许多有效的目标跟踪算法,并且取得很大的突破^[2-9],但是目标跟踪仍然是一个具有挑战性的课题,制约跟踪效果的因素包括目标姿态和形状等内因变化,以及相机视角、纹理、光照和遮挡等外因变化。理想的目标跟踪算法是指该算法在鲁棒性和实时性两个方面达到平衡。【前人研究进展】早期的目标跟踪算法有均值滤波、卡尔曼滤波、粒子滤

收稿日期: 2015-10-14

作者简介: 梁剑平(1987-),女,助教,主要从事数字图像处理研究。

* 广西高校科学技术研究项目(201204LX339和YB2014321)资助。

波、光流法和模板匹配法等。目前实时跟踪的热点方法是将跟踪问题看成是在线的二值分类问题,一类属于背景,一类属于目标,其任务是将目标从背景中分离出来。在跟踪的过程中通过结合背景和前景(目标)信息来实时更新目标模型的分分类器,从而提高跟踪性能。Aviclan^[10]提出的集成跟踪算法(Ensembl Tracker),是利用 Adboost 算法将在线训练得到的多个弱分类器合成一个强分类器,然后用这个强分类器标记下一帧的像素,用以对像素进行目标和背景分类。Grabner 等^[11]提出的在线提升(Online Boosting)算法,是将当前跟踪结果作为正样本,并在跟踪结果周围采集负样本,与上面方法的思想类似,利用 Adboost 算法进行在线特征选择。该方法能有效地解决跟踪过程中遇到的目标姿态、纹理及光照变化等问题。后来 Grabner 等^[5]又在上述理论的基础上进一步提出了一种半监督学习方法,克服了模型更新带来的漂移问题,但由于该方法是用带有误差的次优样本更新训练分类器,因此,随着时间的推移误差逐渐积累,最终将会导致跟踪失败。为了解决上述问题,Babenko 等^[12-13]提出在线多实例学习 MIL(Multiple Instance Learning)算法,该方法将多个样本放到一个集合里面,并给它们赋一个标签。当这个集合里面的样本存在一个或者一个以上的正样本时,这个标签就为正,否则为负。在多实例学习算法中,通过寻找集合中最正确的示例确定跟踪目标,解决了样本可能使得分类器产生迷惑的问题。Zhang 等^[14-15]提出一种简单高效的实时压缩感知跟踪算法,该算法通过一个利用符合压缩感知约束等距性(RIP)条件的、特别稀疏的投影矩阵去提取目标和背景的广义 harr-like 特征,分别作为在线学习更新分类器的正、负样本,并用朴素贝叶斯分类器进行背景和目标的分类,该方法的实时性很高。**【本研究切入点】**但是该方法由于分类器更新的过程中引入了更新程度参数 λ ,且在实际中 λ 需要提前给定,若给定的值过大会使得分类器更新过程无法适应目标外观的变化,过小则容易使分类器“过学习”。因此,必须解决实时压缩感知跟踪算法分类器无法适应目标外观变化及过更新的问题。**【拟解决的关键问题】**提出一种自适应实时压缩感知跟踪算法,根据当前跟踪结果目标模型的哈希指纹与上一帧目标模型的哈希指纹间的汉明距离(Hamming distance) d ,在线实时调整分类器更新程度参数 λ 。实验结果表明,在背景中存在与前景目标有一定相似性的物体,且目标姿态、纹理变化和

光照变化较大等情况下,自适应实时压缩感知跟踪算法仍然能够对目标进行实时准确地跟踪。

1 实时压缩感知跟踪

压缩感知理论^[16]指出:只要信号是可压缩的或者在某个变换域是稀疏的,那么就可以用一个与变换基不相关的投影矩阵,将原始的高维信号投影到一个低维空间上,然后通过求解一个优化问题就可以从这些少量的投影中以高概率重构出原始信号。在该理论框架下,采样速率很大程度上取决于两个基本准则:稀疏性和等距约束性,或者稀疏性和非相关性。文献[15]中的实时压缩感知跟踪算法提出了一个满足 RIP 条件非常稀疏的随机投影矩阵 R ,用这个投影矩阵将高维原始信号投影到一个低维压缩子空间上,该子空间可以很好地保留原始图像的特征信息,利用基于压缩感知提取特征,该算法的数学表达式如下:

$$V = RX, \quad (1)$$

其中, $X \in R^{m \times 1}$ 为原始高维信号, $V \in R^m$ 为压缩后特征, $R \in R^{n \times m}$ ($n \ll m$) 为稀疏随机投影矩阵,满足 RIP 条件,其矩阵元素定义如下:

$$r_{ij} = \sqrt{s} \times \begin{cases} 1, & \text{概率为 } \frac{1}{s}, \\ 0, & \text{概率为 } 1 - \frac{1}{s}, \\ -1, & \text{概率为 } \frac{1}{2s}, \end{cases} \quad (2)$$

其中, s 随机取 2 或者 3,新的特征 v_i 由原始图像特征通过 r_{ij} 加权求和得到,新特征压缩提取如图 1 所示。

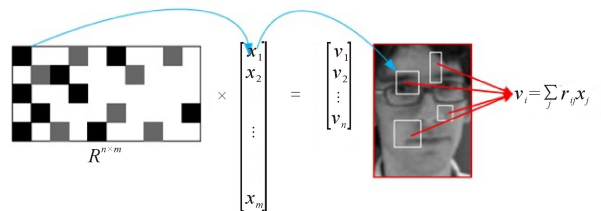


图 1 特征压缩提取示意图

Fig. 1 Diagram of feature extraction by compression

文献[15]的跟踪过程是在上一帧目标区域的周围采样 n 个区域作为候选区域,然后利用(1)式对这 n 个区域进行特征压缩提取,得到 n 个低维特征向量 $v = (v_1, v_2, v_3, \dots, v_n)^T$,最后通过朴素贝叶斯分类器 $H(v)$ 进行前景目标和背景目标分离,选取 n 个区域中 $H(v)$ 值最大的区域作为当前帧跟踪到的目标,朴素贝叶斯分类器 $H(v)$ 公式如下:

$$H(v) = \log\left(\frac{\prod_{i=1}^n p(v_i|y=1)p(y=1)}{\prod_{i=1}^n p(v_i|y=0)p(y=0)}\right) = \sum_{i=1}^n \log\left(\frac{p(v_i|y=1)}{p(v_i|y=0)}\right), \quad (3)$$

其中, $y=0$ 表示负样本, $y=1$ 表示正样本, 两个类先验概率相等 $p(y=1)=p(y=0)=0.5$, 分类器 $H(v)$ 中的条件概率 $p(v_i|y=1)$ 和 $p(v_i|y=0)$ 服从高斯分布, 其均值和标准差分别为 μ^1, σ^1 和 μ^0, σ^0 。得到当前帧目标后, 对这 4 个参数进行增量更新:

$$\begin{aligned} \mu_i^1 &\leftarrow \lambda \mu_i^1 + (1-\lambda)\mu^1, \\ \sigma_i^1 &\leftarrow \sqrt{\lambda(\sigma_i^1)^2 + (1-\lambda)(\sigma^1)^2 + \lambda(1-\lambda)(\mu_i^1 - \mu^1)^2}, \end{aligned} \quad (4)$$

式中, λ 为更新程度参数, 取值范围是 $0 < \lambda < 1$, 其取值影响更新速度的学习率, λ 值越小更新速度越快, 反之越慢。

该算法在背景中存在与目标有一定相似性的物体, 目标姿态、纹理变化和光照变化较大等情况下容易发生漂移或者跟丢目标, 如图 2 中蓝色虚线矩形框所示。

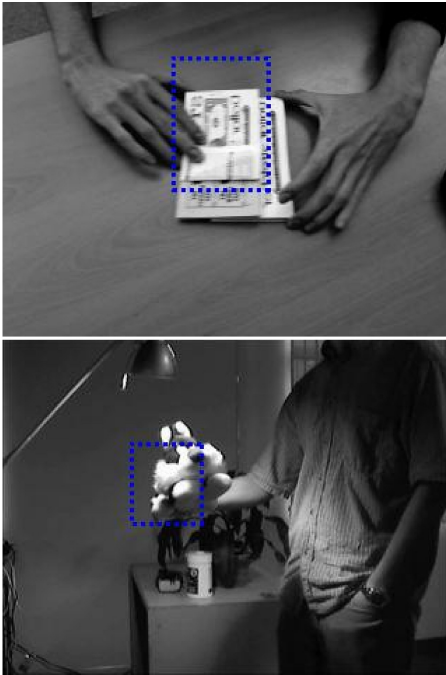


图 2 实时压缩感知跟踪结果

Fig. 2 Tracking result of real-time compressive tracking

2 感知哈希算法

感知哈希算法^[17-18]是哈希算法的一类, 主要用来做相似图片的搜索工作。它的原理是对每张图片生成一个“指纹”(fingerprint)字符串, 然后比较不

同图片的指纹。结果越接近说明图片越相似。下面用最简单的步骤来说明感知哈希算法的原理:

1) 缩小尺寸: 快速去除高频和细节, 只保留结构、明暗等基本信息的方法。将图片缩小到 8×8 的尺寸, 总共 64 个像素。摒弃不同尺寸、比例带来的图片差异。

2) 简化色彩: 将 8×8 的小图片转为 64 级灰度, 即所有像素点总共只有 64 种颜色。

3) 计算平均值: 计算所有 64 个像素的灰度平均值。

4) 比较像素的灰度: 将每个像素的灰度, 与平均值进行比较, 大于或等于平均值, 记为 1; 小于平均值, 记为 0。

5) 计算哈希值: 将上一步比较结果组合在一起构成一个 64 位的整数, 即为该图片的指纹。组合的次序不重要, 只要保证所有图片都采用同样次序即可。

若要比较两个图片的相似性, 首先要计算这两张图片的哈希指纹, 即 64 位 1 或 0 值, 然后计算出不同位的个数, 理论上, 这等同于计算“汉明距离”。汉明距离等于 0, 说明这两张图片非常相似; 汉明距离小于 5, 说明图片略有不同, 但比较相似; 汉明距离大于 10, 说明两张图片完全不同。

3 自适应实时压缩感知跟踪算法

本文在实时压缩感知跟踪的基础上, 利用压缩感知哈希算法计算当前帧跟踪结果目标模型的哈希指纹与上一帧目标模型的哈希指纹之间的汉明距离, 在线实时调整朴素贝叶斯分类器, 使得分类器能够根据目标外观变化快慢, 自适应调整更新程度参数 λ 。具体思想: 首先设置高阈值和低阈值, $H = 10$ 为高阈值, $L = 5$ 为低阈值, 计算当前一帧跟踪结果目标模型的哈希指纹与上一帧目标模型的哈希指纹间的汉明距离 d 。当 $d > H$ 时, 说明背景中存在与前景目标有一定相似性的物体或者目标姿态、纹理变化、光照变化较大等, 发生目标跟踪漂移或者跟丢, 这时候, 贝叶斯分类器的更新主要依赖于上一帧目标模型, 若更新程度参数 λ 太小, 则容易引起分类器过更新, 这时候要适当调大 λ 值; 当 $d \leq L$ 时, 说明目标外观变化缓慢, 这时候, 贝叶斯分类器的更新主要依赖于当前帧的目标模型, 若更新程度参数 λ 太大, 会造成分类器更新过程中无法适应目标外观的变化, 这时候要适当调小 λ 值; 当 $L < d \leq H$ 时, 贝叶斯分类器的更新依赖于当前帧的目标模型和上一

帧目标模型。根据上述思想,公式(4)中的 λ 更新程度参数按照下面的公式进行调整:

$$\lambda = \begin{cases} \lambda_1, d > H, \\ \lambda_2, L < d \leq H, \\ \lambda_3, d \leq L. \end{cases} \quad (5)$$

本文在文献[15]实时压缩感知跟踪算法的基础上,提出自适应实时压缩感知跟踪算法,其流程如下:

输入:第 t 帧图像

1)以 $t-1$ 帧目标位置的 l_{t-1} 为中心, γ 为半径采集候选目标样本集 $D^\gamma = \{z \mid \|l(z) - l_t\| < \gamma\}$,然后利用公式(1)对目标样本集进行特征提取,得到集合的特征向量 v ,计算 $t-1$ 帧跟踪目标的哈希指纹 Ha_{t-1} 并保存;

2)使用公式(3)中的 $H(v)$ 贝叶斯分类器对上述集合的特征向量 v 进行分类,将目标从背景中分离出来,具有最大 $H(v)$ 值的候选区即为当前帧跟踪的目标,其位置记为 l_t ,计算当前帧跟踪目标的哈希指纹 Ha_t 并保存;

3)计算当前帧跟踪目标的哈希指纹 Ha_t 与上一帧跟踪目标的哈希指纹 Ha_{t-1} 间的汉明距离 d ,将 d 与高低阈值进行比较,自适应调整更新程度参数 λ ;

4)采集目标样本和背景样本:以当前目标位置 l_t 为中心, α 为半径采集目标样本集 $D^\alpha = \{z \mid \|l(z) - l_t\| < \alpha\}$;以半径大于 ζ 且小于 β 的圆环采集背景样本, $D^{\zeta\beta} = \{z \mid \zeta < \|l(z) - l_t\| < \beta\}$,其中, $\alpha < \zeta < \beta$;

5)利用公式(1)对目标样本和背景样本进行特征提取,代入公式(4)来更新分类器参数 μ^1, σ^1 和 μ^0, σ^0 。

输出:第 t 帧跟踪到的目标位置 l_t 、自适应更新程度参数 λ 及更新后的分类器参数。

4 算法验证

为了验证本文提出的自适应实时压缩感知跟踪算法(简称 ACT 算法)在视频目标跟踪中的鲁棒性和有效性,使用 David、Coupon、Toy、Sylvester、Lemming 和 Jumping 这 6 个标准视频序列进行测试,并且与文献[15]的 Compressive Tracking 算法(简称 CT 算法)、文献[13]的 Multiple Instance Learning 算法(简称 MIL 算法)进行比较。为保证公平设置参数时每个算法处于最优跟踪性能,在 CT 算法中正样本采集半径 $\alpha=4$,产生 45 个正样本,

负样本采集内半径 $\zeta=8$,外半径 $\beta=30$,在圆环中随机选择 50 负样本,目标候选区域 $\gamma=20$,产生约 1100 个候选区域样本,更新程度参数 $\lambda=0.85$ 。在 ACT 算法中正样本采集半径 $\alpha=4$,产生 45 个正样本,负样本采集内半径 $\zeta=8$,外半径 $\beta=30$,在圆环中随机选择 50 负样本,目标候选区域 $\gamma=20$,产生约 1100 个候选区域样本,哈希指纹间的汉明距离 d 在高低阈值 H, L 不同范围内的自适应更新程度参数为 $\lambda_1=0.85, \lambda_2=0.35, \lambda_3=0.15$ 。在 MIL 算法中,正样本采集半径 $r=25$,产生一个 45 个正样本的正包,负样本采集半径 $\beta=50$,在负样本集中随机选择 65 个负样本组成负包,候选弱分类器的个数 $M=250$,从中挑选出响应最大的弱分类器个数为 $K=50$,弱分类器更新学习率 $\rho=0.85$ 。

4.1 定性分析

定性分析,即通过视觉对比,能够对算法进行直观地比较。图 3 给出 3 个算法在 David、Coupon、Toy、Sylvester、Lemming 和 Jumping 这 6 个标准视频序列中的跟踪结果。

(a)David 视频序列中,跟踪的目标是 David 的脸部,视频包含光照变化、目标姿势、尺度变化等挑战。由于 MIL 算法的外观更新公式较粗糙,(a)视频的第 290 帧光照和目标姿势剧烈变化时, MIL 算法跟踪目标失败。CT 算法在跟踪过程中出现跟踪漂移,而 ACT 算法却能准确地跟踪目标,在该视频中取得最佳表现。

(b)Coupon 视频序列中,跟踪的目标是一张赠券,背景干扰和遮挡增加了跟踪目标的难度,在该视频中由于没有光照、目标姿态的较大变化,在 3 个算法中 MIL 算法能够最优地跟踪目标,由于具有自适应调整更新程度参数的能力,ACT 算法比 CT 算法要好一些。

(c)Toy 视频序列中,跟踪的目标是玩偶的正脸,玩偶在跟踪过程中快速运动、多尺度变化和旋转,在(c)视频的第 3 帧中 MIL 算法跟丢目标,此时 ACT 和 CT 算法还可以准确跟踪目标,到了第 82 帧 CT 算法跟踪出现漂移,ACT 算法仍能很好地跟踪目标,在本视频序列中 ACT 算法最佳,CT 算法次佳。

(d)Sylvester 视频序列中,跟踪的目标是一个玩偶,视频包含光照变化和目标旋转的干扰。ACT 算法表现最佳,能准确地跟踪目标,CT 在跟踪过程中发生漂移, MIL 算法在第 959 帧中完全跟丢目标。

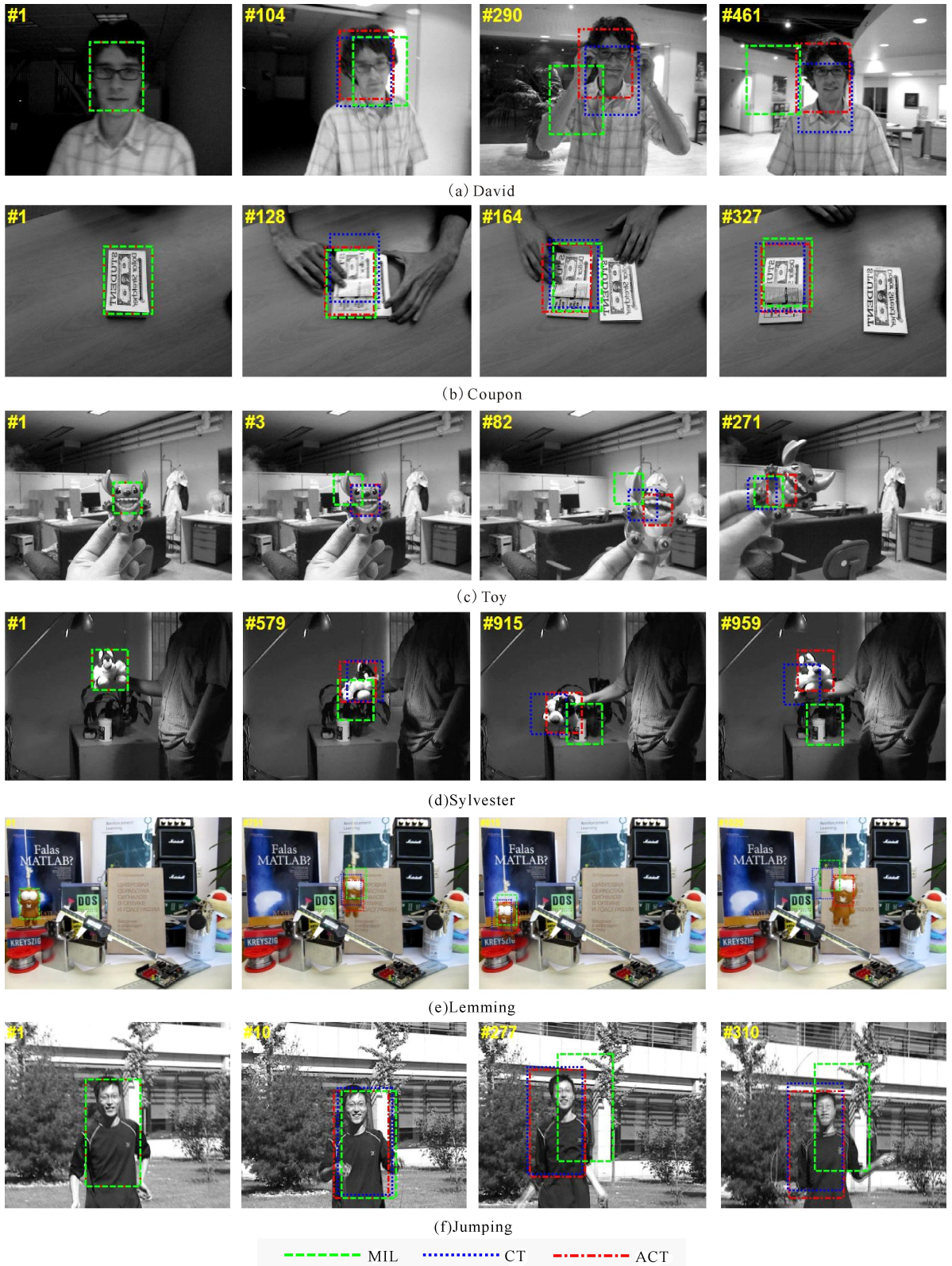


图3 6个测试序列的抽样跟踪结果

Fig. 3 Sampled tracking results of 6 tested sequences

(e) Lemming 视频序列中,跟踪的目标是玩偶旅鼠,视频包含背景杂乱、遮挡、旋转、序列较长的因素,在目标发生遮挡时,3个算法的表现并不理想。

跟踪过程中 ACT 算法由于目标被遮挡,导致目标跟踪丢,但快速调整后能重新准确跟踪目标,而 CT 算法和 MIL 算法在目标遮挡时,发生目标跟踪丢不能

很好地调整,导致跟踪失败。在该视频中 ACT 算法表现最佳。

(f) Jumping 视频序列中,跟踪的目标是跳绳男子上半身,视频包含模糊运动和快速运动因素,CT 算法和 ACT 算法表现较好,而 MIL 算法在跟踪目标的过程中出现漂移。

4.2 定量分析

定量分析,即使用平均中心误差和成功率这两个指标对跟踪结果进行评价。平均中心误差是指跟踪结果位置中心与真实目标位置中心之间距离的平均值,成功率定义为若 $\frac{ROI_r \cap ROI_G}{ROI_r \cup ROI_G} > 0.5$ 时,则认为跟踪成功,其中 ROI_r 表示算法的跟踪结果区域, ROI_G 表示目标的实际区域。表 1 给出 3 个算法在 6 个标准视频序列中的平均中心误差,从总体的总平均值看,ACT 算法的平均中心误差最小,MIL 算法的平均中心误差最大,CT 算法次佳。逐个视频评价,ACT 算法在 6 个标准视频中有 4 个表现最佳,2 个次佳;CT 算法有 1 表现最佳,4 个表现次佳;MIL 算法有 1 个表现最佳。表 2 给出 3 个算法在 6 个标准视频序列中的跟踪成功率,从总体的成功率来看,ACT 算法跟踪成功率高达 87%,远高于其它两个算法。单个视频序列来看,ACT 算法在 6 个标准视频中有 4 个表现最佳,2 个次佳;CT 算法有 1 表现最佳,3 个表现次佳;MIL 算法有 1 个表现最佳,1 个表现次佳。综上所述,定量分析与定性分析(平均中心误差和跟踪成功率)两个评价方法基本吻合。

表 1 平均中心误差

Table 1 Average center location errors

序列 Sequence	算法 Methods(pixels)		
	ACT	CT	MIL
David	12	17	23
Coupon	7	8	5
Toy	28	32	36
Sylvester	4	9	13
Lemming	5	19	22
Jumping	7	6	10
Total Average	10	15	18

注:粗体数字表示最佳,斜体表示次佳

Note: Bold fonts indicate the best performance while the italic fonts indicate the second best ones

表 2 跟踪成功率

Table 2 Success rate

序列 Sequence	算法 Methods(%)		
	ACT	CT	MIL
David	92	87	52
Coupon	94	91	98
Toy	44	35	42
Sylvester	99	75	68
Lemming	98	61	54
Jumping	95	98	57
Total Average	87	74	61

注:粗体数字表示最佳,斜体表示次佳

Note: Bold fonts indicate the best performance while the italic fonts indicate the second best ones

5 结论

针对实时压缩感知目标跟踪算法在目标姿态、纹理或光线变化剧烈时,不能自适应地调整更新分类器,导致跟踪容易发生漂移、丢失目标的问题,提出自适应的实时压缩感知跟踪算法。实验结果表明,在 6 个具有挑战性的视频中自适应实时压缩感知跟踪算法表现较好,跟踪成功率比 CT 算法高出 13%,比 MIL 算法高 26%。当目标大小为 $40 \text{ pixel} \times 43 \text{ pixel}$ 时,处理帧速约为 37 fps,满足实时性要求。此外,ACT 算法在目标运动速度非常快的情况下,跟踪效果不是很理想,这是本文后续要研究和解决的重点问题。

参考文献:

- [1] YILMAZ A, JAVED O, SHAH M. Object tracking: A survey[J]. ACM Computing Surveys (CSUR), 2006, 38(4): 1-45.
- [2] AVIDAN S, TECHNOL M V, JERUSALEM I. Support vector tracking[C]//Computer Vision and Pattern Recognition. [S. l.]: IEEE, 2001: 184-191.
- [3] JEPSON A D, FLEET D J, EL-MARAGHI T F. Robust online appearance models for visual tracking, IEEE Trans[J]. Pattern Anal Mach Intell, 2003, 25(10): 1296-1311.
- [4] COLLINS R T, LIU Y, LEORDEANU M. Online selection of discriminative tracking features, IEEE Trans[J]. Pattern Anal Mach Intell, 2005, 27(10): 1631-1643.
- [5] GRABNER H, LEISTNER C, BISCHOF H. Semi-supervised on-line boosting for robust tracking [C]// FORSYTH D, TORR P, ZISSERMAN A, eds. Computer Vision - ECCV 2008. [S. l.]: Springer Berlin Heidelberg, 2008: 234-247.

- [6] MEI X, LING H. Robust visual tracking using l_1 minimization[C]//Proc IEEE Int Comput Vis Conf. Sep. [S. l. : s. n.], 2009: 1436-1443. DOI: 10. 1109/ICCV. 2009. 5459292.
- [7] KWON J, LEE K M. Visual tracking decomposition [C]//Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. San Francisco, CA: IEEE, 2010: 1269 - 1276. DOI: 10. 1109/CVPR. 2010. 5539821.
- [8] KALAL Z, MATAS J, MIKOLAJCZYK K. P-N learning: Bootstrapping binary classifiers by structural constraints[C]//Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. San Francisco, CA: IEEE, 2010: 49 - 56. DOI: 10. 1109/CVPR. 2010. 5540231.
- [9] WANG Q, CHEN F, XU W L, et al. Online discriminative object tracking with local sparse representation [C]// M in Applications of Computer Vision (WACV), 2012 IEEE Workshop on. [S. l. : s. n.], 2012: 425-432.
- [10] AVIDAN S. Ensemble tracking[C]//Computer Vision and Pattern Recognition, IEEE Computer Society Conference. [S. l. : s. n.], 2005: 261-271.
- [11] GRABNER H, BISCHOF H. On-line boosting and vision[C]//Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. [S. l. : s. n.], 2006: 260-267.
- [12] BABENKO B, YANG M H, BELONGIE S. Visual tracking with online multiple instance learning[C]//Computer Vision and Pattern Recognition, 2009. [S. l.]: IEEE, 2009: 983-990.
- [13] BABENKO B, YANG M H, BELONGIE S. Robust object tracking with online multiple instance learning [J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2011, 33: 1619-1632.
- [14] ZHANG K H, SONG H H. Real-time visual tracking via online weighted multiple instance learning [J]. Pattern Recognition, 2013, 46: 397-411.
- [15] ZHANG K H, ZHANG L, YANG M H. Real-time compressive tracking[C]//FITZGIBBON A, LAZEBNIK A PERONA P, et al, eds. Computer Vision-ECVCV 2012. [S. l.]: Springer Berlin Heidelberg, 2012: 864-877.
- [16] DONOHO D L. Compressed sensing[J]. IEEE Transactions on Information Theory, 2006, 52 (4): 1289-1306.
- [17] KRAWETA N. The Hacker factor blog[EB/OL]. [2015-09-22]. <http://www.hackerfactor.com/blog/index.php?/archives/432-Looks-Like-It.html>.
- [18] CHEN C. 感知哈希算法: 找出相似的图片[EB/OL]. [2015-09-12]. <http://www.cnblogs.com/technology/archive/2012/07/12/Perceptual-Hash-Algorithm.html>.
- CHEN C. Perception of hash algorithm: Find the similar images[EB/OL]. [2015-09-12]. <http://www.cnblogs.com/technology/archive/2012/07/12/Perceptual-Hash-Algorithm.html>.

(责任编辑: 陆雁)