

基于局部稀疏表示的模板匹配跟踪算法研究

包晓安,陈磊,万微祥,张俊为,桂江生

(浙江理工大学 信息学院,杭州 310018)

摘 要: 在目标跟踪过程中,局部稀疏表示跟踪算法(LSRTA)在目标被遮挡面积过大、运动过快或形变量过大等情况下,跟踪过程中会发生目标偏移现象。针对这个问题,在LSRTA算法基础上,融入surf、flann和knn相结合的模板匹配算法,提出了基于局部稀疏表示模板匹配算法(LSRTMTA)以解决目标跟踪偏移问题。在LSRTA算法跟踪过程中,通过不断地计算新模板与当前模板的匹配值来判断是否发生偏移。当目标发生偏移时,停止LSRTA算法的跟踪,通过模板与帧图像之间匹配来重新确定目标位置;当确定目标位置后,再次进行LSRTA算法的跟踪。实验结果表明:该算法不仅保留了LSRTA算法的优点,还具有自动修正目标偏移的功能,改善了跟踪效果,增强了目标跟踪过程的容错性。

关键词: 目标跟踪;模板匹配算法;目标跟踪偏移;自动修正

中图分类号: TP391.4

文献标志码: A

文章编号: 1673-3851(2018)01-0082-08

0 引言

在计算机视觉领域中,目标跟踪是研究的重点之一,应用非常广泛,在智能视频监控、人机交互、计算机视觉导航、虚拟现实和医学诊断等领域中都有涉及。目标跟踪是对连续视频图像进行特征提取,分析获得目标的位置、大小、选择角度和速度等信息,并且持续地显示目标在每帧图像中的位置^[1]。目前已有大量目标跟踪算法,这些算法可以分为以下5大类:a) 基于梯度场决策模型的跟踪算法,主流算法有基于Mean-shift跟踪算法^[2]和基于均值漂移的目标跟踪算法^[3]等;b) 基于概率决策的模型跟踪算法,主流算法有卡尔曼滤波跟踪算法^[4]和基于粒子滤波的跟踪算法^[5]等;c) 基于分类器决策模型的跟踪算法,主流算法有基于支持向量机跟踪算法^[6]和TLD(Tracking learning detection)算法^[7]等;d) 基于子空间表示模型跟踪算法,主流算法有增量视觉跟踪IVT(Incremental learning for robust visual tracking)算法^[8]和基于增量PCA(Principal

component analysis)跟踪算法^[9]等;e) 基于稀疏表示模型的跟踪算法,主要算法有 l_1 范数最小化视频跟踪算法^[10]和局部稀疏表示跟踪算法(Local sparse representation tracking algorithm,LSRTA)^[11]等。

基于稀疏表示模型的跟踪算法首先是由Mei等^[10]提出的 l_1 范数最小化的目标跟踪算法,该算法首次将稀疏表示的理论知识应用到目标跟踪领域,跟踪过程中需要人为设置初始模板以及正负最小模板,通过最小模板构造稀疏字典并进行粒子滤波产生粒子,最后用稀疏字典^[12]对粒子进行 l_1 范数最小化处理,选择构造误差最小的粒子得到目标的位置,该算法运行速度较快,能初步实现目标的跟踪。之后,许多学者开始加入到稀疏表示跟踪算法的研究中,提出了一系列算法,如:詹曙等^[13]提出了Gabor特征稀疏表示目标跟踪算法,该算法在稀疏表示模板字典的更新过程中加入主元分析理论,在目标发生形变时,该算法有较好的跟踪效果。胡昭华等^[14]提出了特征联合的稀疏表示跟踪算法,该算法用字典模板对候选粒子进行联合稀疏表示,改善了跟踪

收稿日期:2017-07-04 网络出版日期:2017-11-09

基金项目:国家自然科学基金项目(61379036,61502430)

作者简介:包晓安(1973-),男,浙江金华人,教授,主要从事软件工程、智能信息处理方面的研究。

通信作者:桂江生,E-mail:jsgui@zstu.edu.cn

效果。Wang 等^[11]提出了 LSRTA 算法,将模板进行了分块并独立构造稀疏字典,再获得每个小块在帧图像中的投票图,最后合并为全局投票图,从而估计目标的位置。LSRTA 算法能够在目标发生形变或者部分遮挡时具有较好的跟踪效果,但是当目标被遮挡面积过大、运动过快或形变量过大等情况存在时,跟踪过程中仍然会发生目标偏移现象。针对 LSRTA 算法的这个不足,本文提出了一种具有自动修正目标功能的稀疏表示跟踪算法。该算法在 LSRTA 算法的基础上,融入了模板匹配算法,通过不断计算新模板与当前模板的匹配值来判断跟踪是否发生偏移。当目标发生偏移时,通过模板与帧图像之间匹配来重新确定目标位置以修正偏移,改善跟踪效果。

1 模板匹配算法

本文中,模板匹配算法采用 surf、flann 和 knn 相结合的算法。surf 算法找到两幅图像中感兴趣点的位置;flann 算法对这些感兴趣点进行匹配,找出匹配成功的点对;knn 算法对匹配成功的点进行筛选,去除零散的点对。该算法可以有效找出两幅图像之间相似点的集合。

surf 算法通过计算一幅图像上所有像素点的 Hessian 矩阵的行列式与特征值,来判断整幅图像上哪些是感兴趣点^[15]。Hessian 矩阵可用式(1)表示为:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (1)$$

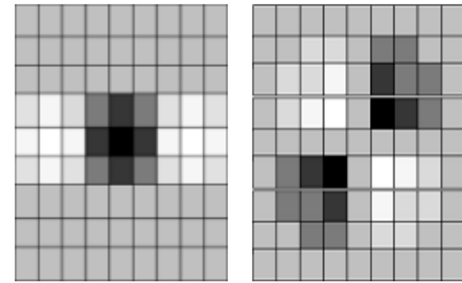
式(1)中, $L_{xx}(x, \sigma)$ 代表图像点 $f(x, y)$ 与二阶高斯滤波图像 $\frac{\partial^2 g(\sigma)}{\partial x^2}$ 的卷积,在得到该像素点的 Hessian 矩阵后,可以很容易通过计算得到该点像素的特征值。为了简化运算,Bay 提出了盒滤波器近似高斯滤波器,降低了计算复杂度,大大加快了计算时间。高斯滤波器与盒滤波器滤波后的示例图像如图 1 所示。

图 1(a)是高斯滤波器滤波后的图像,其在 x 、 y 和 xy 方向上的离散二阶导数用 L_{xx} 、 L_{yy} 和 L_{xy} 表示。图 1(b)是盒滤波器滤波后的图像,其在 x 、 y 和 xy 方向上的离散二阶导数用 D_{xx} 、 D_{yy} 和 D_{xy} 表示。图 1 中用简单权值表示黑色为 1,白色为-1。在计算 Hessian 矩阵的过程中,可以使用盒滤波器近似计算,得出了 Hessian 矩阵行列式公式:

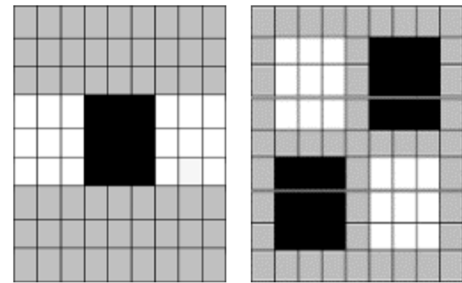
$$\det(H_{\text{approx}}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \quad (2)$$

计算图像中所有点的行列式与特征值,如果行列式的值为负,并且特征值异号,则该点不是感兴趣

点,如果行列式为正,并且特征值同号,则该点是感兴趣点。



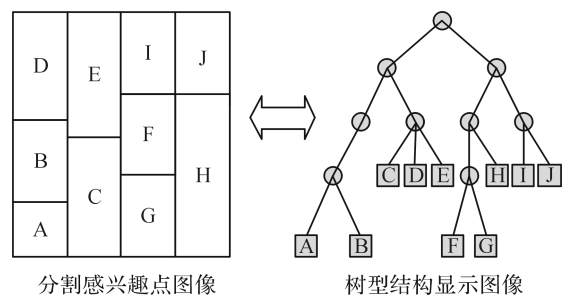
(a)高斯滤波器滤波后的图像



(b)盒滤波器滤波后的图像

图 1 高斯滤波器与盒滤波器滤波后的图像

flann 算法^[16]是对两幅图像中的感兴趣点进行匹配,找出匹配成功的点对。图 2 为 flann 算法示意图。图 2 左边是包含感兴趣点的图像,图像根据一定的分割规则分成多块,每一块包含一个感兴趣点。图 2 右边是将包含感兴趣点的图像以树型结构表示。先将两幅图像分别按以上方式分块树状结构表示,再从第一幅图像顶部根节点块开始,在第二副图像中搜索与第一幅图像顶部块最相似图像块,通过计算两幅图像块的欧氏距离来判断两个图像块是否相似,如果欧氏距离大于设定的阈值,则两幅图像块中感兴趣点匹配成功。否则进行下一个点的匹配,直至搜索到最底部;如果还没有找到匹配点,则第一幅图像中的感兴趣点没有匹配成功的点。再从第一幅图像中按自顶向下的原则寻找第二个块的匹配块,如此循环遍历第一幅图像中所有图像块,得到两幅图像中匹配的感兴趣点对集合。



分割感兴趣点图像

树型结构显示图像

图 2 flann 算法示意图

knn 算法^[17]对感兴趣点进行筛选,去除零散的感兴趣点对。先设定一个距离阈值,低于这个阈值的归为一类,否则归结到另一类;如此对图像中感兴趣点对进行分类;当某一类的数目小于数目阈值时,则该类可以被删除。图 3(a)是经过 surf 和 flann 算法的匹配图,图 3(b)是经过 surf、flann 和 knn 算法的匹配图。



(a)surf和flann算法匹配图



(b)surf、flann和knn算法匹配图

图3 模板与帧图像匹配示例图像

2 LSRTMTA 跟踪算法

本文在 LSRTA 算法的基础之上,融入模板匹配算法,提出了一种基于局部稀疏表示模板匹配跟踪算法(Local sparse representation template matching tracking algorithm, LSRTMTA),以实现在发生目标跟踪偏移时,自动修正重新定位功能。

图 4 为 LSRTMTA 跟踪算法整体流程图,先选定初始目标从而获得初始模板,再利用 LSRTA 跟踪算法对初始的目标进行跟踪。当得到新模板时,计算新模板与当前模板的匹配值。如果匹配值大于阈值,说明新模板没有发生太大变化,目标并没有发生偏移现象,则将新模板保存到模板集中。如果匹配值小于阈值,说明模板发生了大的变化,目标发生了偏移现象,则舍弃该模板,并且停止 LSRTA 算法,通过当前模板与帧图像进行模板匹配重新查找目标位置,如果目标在当前帧没有被找到时,该模板与下一帧图像匹配查找,直到找到目标;然后再次执行 LSRTA 算法,如此循环跟踪。

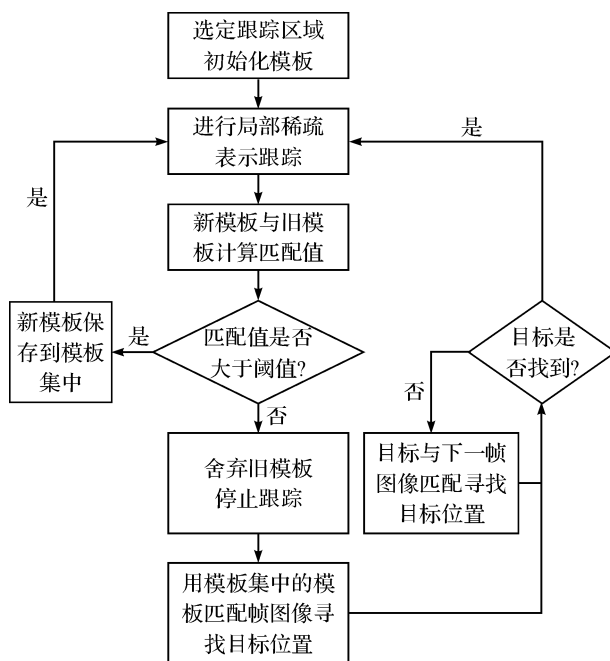


图4 LSRTMTA 跟踪算法流程

图 5 为 LSRTA 算法跟踪示意图,先将模板进行均匀分块处理,左侧模板图像中 $P_T = (d_x, d_y, h, \omega)$ 是模板分块后的一个矩形小块,模板中心滑动到该候选位置时,用 $P_{F_i(x,y)}$ 表示帧图像 F 中对应模板 P_T 的矩形块。

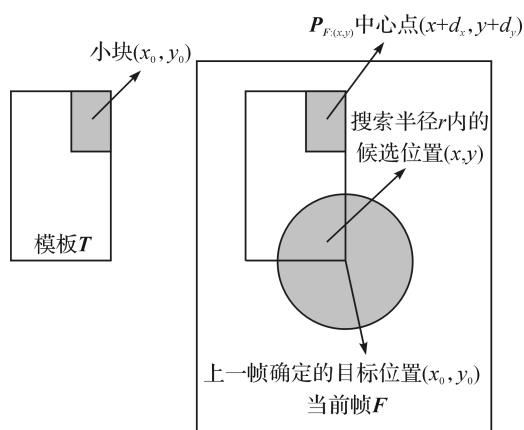


图5 LSRTA 算法跟踪示意图

再通过计算帧图像中的矩形块 $P_{F_i(x,y)}$ 与目标模板中的 P_T 矩形块两幅图像的欧氏距离,来衡量它们之间的相似性,定义了投票函数 $V_{P_T}(x,y)$ 来直观的反应他们之间的相似性,投票值越大,则相似程度越高。

$$V_{P_T}(x,y) = d(P_{F_i(x,y)}, P_T) \quad (3)$$

在式(3)的求解过程中,先对图像进行稀疏表示,其基本思想是用尽可能少的元素表示元素图像,用完备字典 $A = \{a_1, a_2, \dots, a_m\}$ 线性叠加表示图像 y ,图像 y 的表示模型为:

$$\begin{aligned} \min \quad & \|x\|_1 \\ \text{st} \quad & y = Ax + e \end{aligned} \quad (4)$$

式(4)中: x 为图像 y 在字典 A 上的稀疏表示, e 为逼近精度。可以将上述 l_1 范数最小化求解问题转化为 l_1 正则最小均方问题:

$$\min \|y - Ax\|_2^2 + \lambda \|x\|_1 \quad (5)$$

对于式(5)的求解,选择基追踪(BP)算法^[18]进行求解,得到稀疏值,再对稀疏值进行欧氏距离求解。为了能够求出每个小块 $P_{F_i(x,y)}$ 的稀疏表示,需要对目标模板中所有的矩形小块 P_T 构造各自独立的稀疏字典 $B_{F_i(x,y)}$,使模板在周围 8 个方向抖动来获得一个包含 9 个目标模板的稀疏字典,设 $P_{F_i(x,y)}$ 在当前 F 中的观测值是 $y_{F_i(x,y)}$,则有:

$$\min \|B_{F_i(x,y)}c - y_{F_i(x,y)}\|_2^2 + \lambda \|c\|_1 \quad (6)$$

在得到观测 $y_{F_i(x,y)}$ 的系数向量 c 之后,计算得到其他与对应模板的似然:

$$p_{F_i(x,y)} = \exp(-\|y_{F_i(x,y)} - T_{F_i(x,y)}a\|_2^2) = V_{P_T}(x,y) \quad (7)$$

通过相同的方法可以得到模板中其他矩形小块的投票图 $V_{P_T}(\cdot, \cdot)$ 。最后需要将所有小块投票图合并成一个整体投票图。图 6 为投票图的合并流程图,在所有的矩形小块的投票图中选取投票值最大的作为该点最终得分,得分最大的点作为下一帧目标的位置。

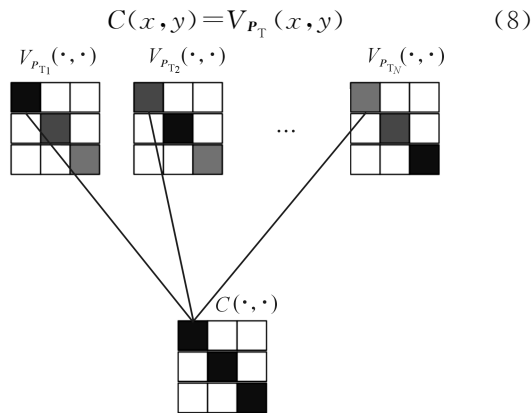


图 6 投票图的合并流程

图 7 为模板与模板匹配图,左边表示当前模板,右边表示新模板,通过模板匹配算法得到匹配图,计算匹配值,匹配值公式为:

$$p/\% = \frac{k}{\min(m,n)} \times 100 \quad (9)$$

式(9)中: m 、 n 分别表示两幅模板图经过 surf 算法后得到的感兴趣点数量, k 表示经过 flann 算法和 knn 滤波后感兴趣点数量。模板与模板匹配的示例见图 7。图 7 中 $k=14$, $m=18$, $n=22$,根据匹配值

公式可以得到匹配值 $p=0.78$,假设匹配阈值为 0.62,由于匹配值 p 大于阈值,说明目标并没有发生偏移,新模板存入模板集中。



图 7 模板与模板匹配图

图 8 为模板的存储过程,模板集中模板空间是固定的,新模板按照栈的形式依次保存在模板集中,在追踪过程中会产生大量模板,当模板存满时,最先存入的模板会被删除来保证不断的新模板被存入。

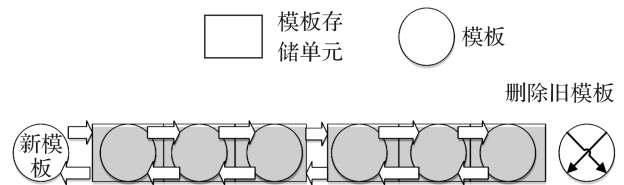


图 8 模板存储过程

3 实验结果

为了验证本文的算法,在 Matlab 软件环境下编程实现 LSRTMTA 跟踪算法,采用 tracker benchmark 数据库进行实验对比。tracker benchmark 数据库中有 50 组视频,其中:目标外观变形的有 19 组,目标遮挡的有 29 组,目标快速移动的有 17 组,这些视频已经基本考虑到实际运行过程中受到的各种干扰,在实际应用中有一定的代表性。选取 ORIA 算法^[19](Online robust image alignment)、CT 算法^[20](Real-time compressive tracking)、IVT (Incremental learning for robust visual tracking)算法、CPF 算法^[21](Color-based probabilistic tracking)、LSRTA 算法和本文提出的 LSRTMTA 算法进行对比。

在目标快速移动的视频中,其中 boy 视频 6 种算法实时跟踪结果如图 9 所示。在目标外观变形的视频中,其中 mountainBike 视频 6 种算法实时跟踪结果如图 10 所示。在目标遮挡的视频中,其中 faceoccl 视频 6 种算法实时跟踪结果如图 11 所示。



图9 boy 视频运行效果

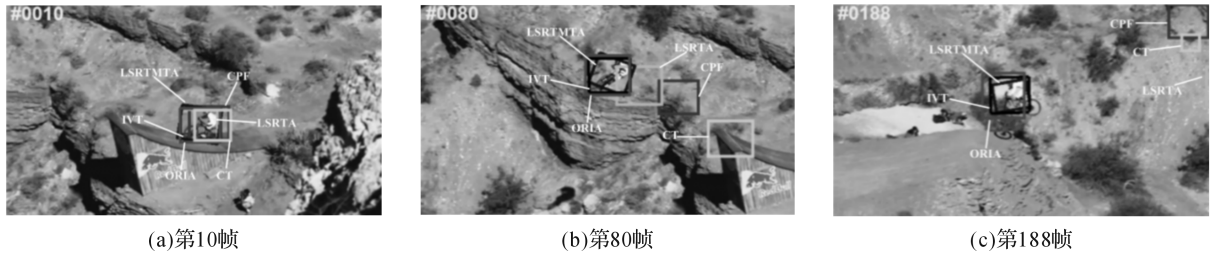


图10 mountainBike 视频运行效果

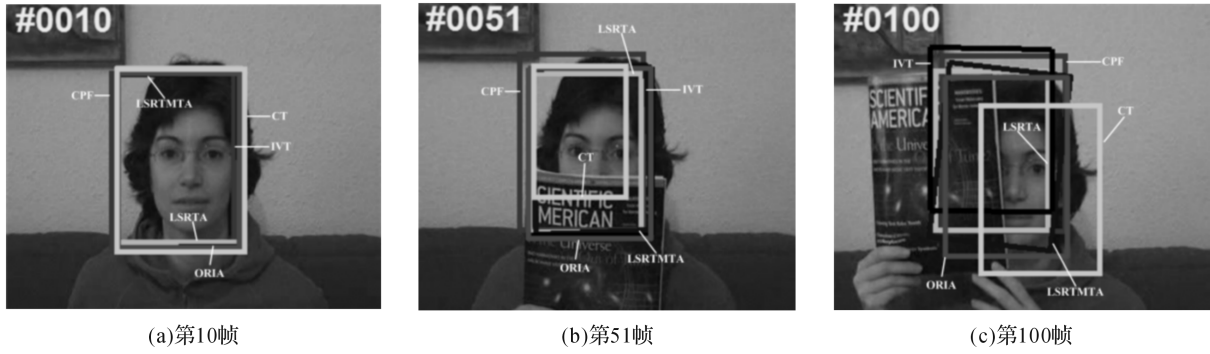


图11 faceoccl 视频运行效果

为了直观反映每一帧图像与实际跟踪目标的差距,本文定义了中心误差函数。中心误差值越小,代表目标跟踪越准确。某一帧图像中目标实际中心位置为 $M(x_1, y_1)$,跟踪标记的中心位置 $N(x_2, y_2)$,中心误差 $r(M, N)$ 的计算公式为:

$$r(M, N) = [(x_1 - x_2)^2 + (y_1 - y_2)^2]^{1/2} \quad (10)$$

分别对每组测试视频绘制6种算法的中心误差图,计算每组视频中6种算法下的平均中心误差。 x 轴代表第几组测试视频, y 轴代表每组视频中6种算法下的平均中心误差,得到平均中心误差,如图12所示。

计算50组视频中6种算法的总平均中心误差值,得到总平均中心误差、标准差与标准误见表1。

为了进一步反应算法跟踪效果与实际跟踪区域的覆盖程度,引用了覆盖率,表示实际目标区域与算法跟踪区域的交集部分占并集部分的比例。实际目标矩形区域 A ,算法跟踪矩形区域为 B , $s(C)$ 表示 C

区域的面积,覆盖率 $c(A, B)$ 计算公式为:

$$c(A, B) = \frac{s(A \cap B)}{s(A \cup B)} \quad (11)$$

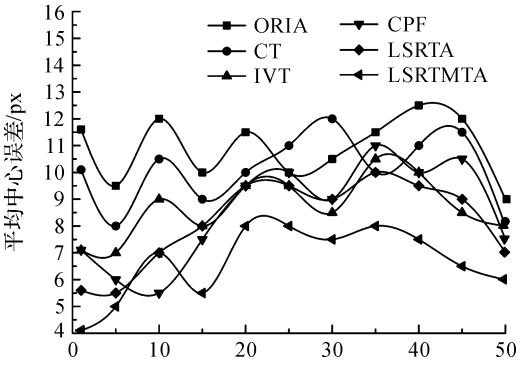


图12 6种算法下的平均中心误差

表1 6种算法的总平均中心误差

算法	ORIA	CT	IVT	CPF	LSRTA	LSRTMTA
总平均中心误差	11	10.4	8.8	8.7	8.1	6.8
标准差	1.20	1.39	1.05	1.92	1.47	1.10
标准误	0.38	0.44	0.33	0.61	0.46	0.35

分别对每组测试视频绘制 6 种算法的覆盖率图,计算每组视频中 6 种算法下的平均覆盖率, x 轴代表第几组测试视频, y 轴代表每组视频中 6 种算法下的平均覆盖率,得到平均覆盖率,如图 13 所示。

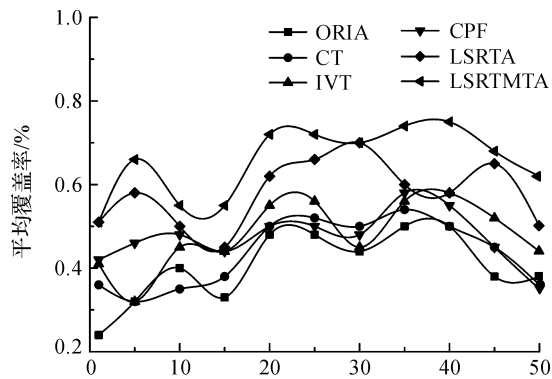
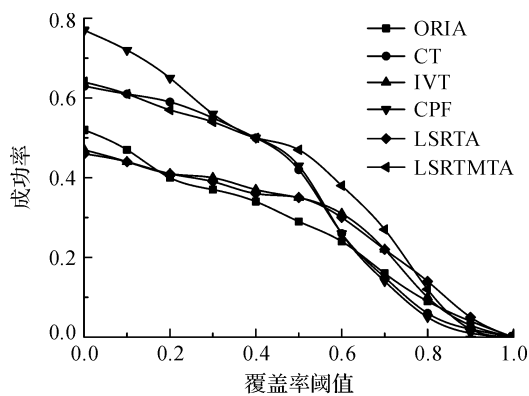


图 13 6 种算法下的平均覆盖率

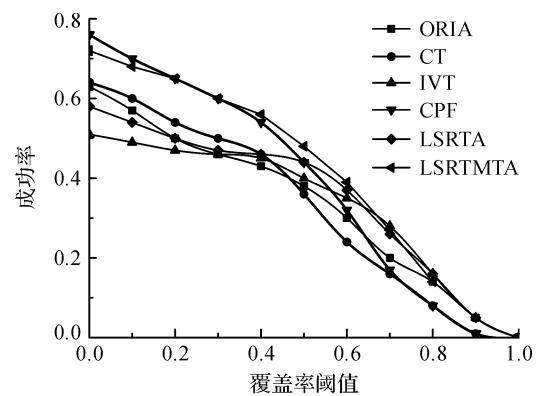
计算 50 组视频中 6 种算法的总平均覆盖率值,得到总平均覆盖率、标准差与标准误差见表 2。

表 2 6 种算法的总平均覆盖率

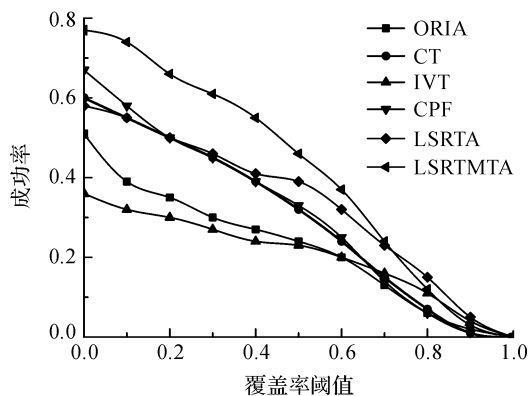
算法	ORIA	CT	IVT	CPF	LSRTA	LSRTMTA
总平均覆盖率	0.41	0.45	0.49	0.48	0.57	0.66
标准差	0.065	0.077	0.077	0.059	0.076	0.069
标准误差	0.021	0.024	0.024	0.019	0.024	0.022



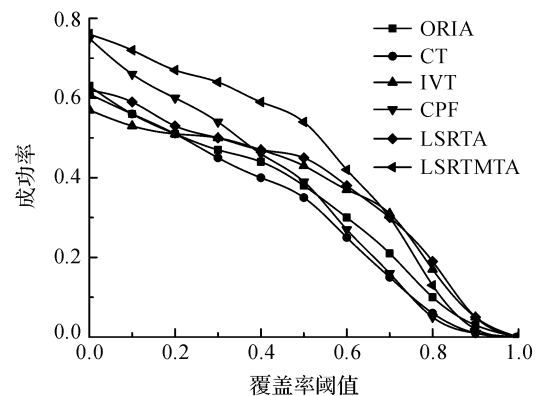
(a)外观变形



(b)目标被遮挡



(c)快速移动



(d)整个50组

图 14 6 种算法下的 OPE 成功率

为了整体评估算法的运行效果,引用了一次通过的评估(OPE)成功率图, x 轴表示覆盖阈值, y 轴表示成功率,首先对应一个给定的覆盖阈值 x_0 ,对于每一帧图片计算覆盖率,如果覆盖率大于覆盖阈值,则判断成功, $l(n)=1$,否则判断失败, $l(n)=0$, n 表示所有测试视频中所有图片的总数, $\text{sum}(l(n))$ 表示 n 张图片中 $l(n)$ 的和,成功率 $s(x,y)$ 公式为:

$$s(x,y) = \frac{\text{sum}(l(n))}{n} \quad (12)$$

其中:

$$l(n) = \begin{cases} 0 & f(c(\mathbf{A}, \mathbf{B}) \leq x_0) \\ 1 & f(c(\mathbf{A}, \mathbf{B}) > x_0) \end{cases}$$

在 OPE 成功率图中,函数曲线下的面积越大,代表目标跟踪的效果是越好。分别选取 19 组目标外观变形的视频、29 组目标在遮挡情况下的视频、17 组目标在快速移动的视频和整个 50 组视频,得到 6 种算法的 OPE 成功率,如图 14 所示。

选取整个 50 组视频,通过计算得到 6 种算法的 OPE 平均成功率,如表 3 所示。

最后为了说明文算法的实时性,选取了 boy 视频前 200 帧(从第二帧开始)的检测时间绘制算法处理时间图,如图 15 所示。

表3 6种算法的平均 OPE 成功率

算法	ORIA	CT	IVT	CPF	LSRTA	LSRTMTA
平均 OPE 成功率	0.33	0.31	0.36	0.36	0.38	0.44

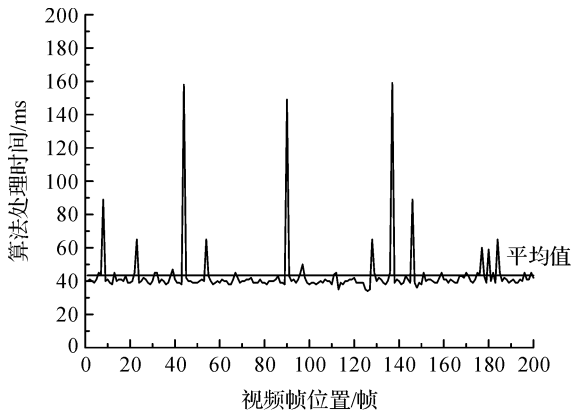


图15 采用本文算法对 boy 视频的跟踪处理时间

从图9—图11中可以明显地看出,在目标快速移动的 boy 视频、目标外观变形的 mountainBike 视频、目标遮挡的 faceoccl 视频中,本文提出的 LSRTMTA 跟踪算法在显示的那些图像中跟踪效果比较准确。

从图12中可以看出,对于50组测试数据,在6种算法中,LSRTMTA 算法平均中心误差曲线下的面积最小;从表1中可以看出,50组总平均中心误差值也最小,说明 LSRTMTA 算法的中心位置与实际目标的中心位置最接近。

从图13中可以看出,对于50组测试数据,在6种算法中,LSRTMTA 算法平均覆盖率曲线下的面积最大;从表2中可以看出,50组总平均覆盖率值也最大,说明 LSRTMTA 算法的矩形跟踪区域与实际目标矩形跟踪区域的覆盖程度最大。

从图14中可以看出,在19组目标外观变形的视频、29组目标在遮挡情况下的视频、17组目标在快速移动的视频和整个50组视频下,6种算法中 LSRTMTA 算法的 OPE 成功率曲线下的面积都最大;从表3中可以看出,在整个50组视频中,LSRTMTA 算法的平均 OPE 成功率值最大。

从图15中可以看出,选取 boy 视频前200帧的处理时间计算出的平均处理时间为43.4 ms (Windows 7, 4核),即每秒可处理约23帧图像,可以应用于实际场景实现实时目标跟踪。

综上所述,LSRTMTA 跟踪算法在6种算法中跟踪效果最好,改善了 LSRTA 算法的跟踪效果。

4 结 语

本文针对目标偏移时跟踪无法修正的问题,在

LSRTA 算法的基础上,加入模板匹配算法,提出了 LSRTMTA 算法以解决目标跟踪偏移问题。该算法通过不断计算新模板与当前模板的匹配值来判断是否发生偏移,当目标发生偏移时,停止 LSRTA 算法的跟踪,通过模板与帧图像之间匹配来重新确定目标位置,当确定目标位置后,再次进行 LSRTA 算法的跟踪。实验结果表明,该算法不仅保留了 LSRTA 算法的优点,还具有自动修正偏移功能,改善了跟踪效果,增加了人物跟踪的容错性。

参考文献:

- [1] 李博,张凌. 基于视觉显著性的监控视频动态目标跟踪[J]. 信息技术, 2014(4): 60-65.
- [2] 李霞. 基于 Mean-Shift 算法的目标跟踪技术研究[J]. 自动化与仪器仪表, 2016(4): 20-22.
- [3] 同鑫,熊红凯. 基于均值漂移的视频目标跟踪改进算法[J]. 信息技术, 2010(3): 40-43.
- [4] 李晶,范九伦. 一种基于卡尔曼滤波的运动物体跟踪算法[J]. 计算机应用研究, 2010, 27(8): 3162-3164.
- [5] 张明杰,康宝生. 一种基于图模型的粒子滤波跟踪方法[J]. 计算机应用研究, 2016, 33(2): 590-593.
- [6] 胡昭华,徐玉伟,赵孝磊,等. 基于支持向量机的多特征选择目标跟踪[J]. 应用科学学报, 2015, 33(5): 502-517.
- [7] Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2012, 34(7): 1409-1422.
- [8] Ross D A, Lim J, Lin R S, et al. Incremental learning for robust visual tracking[J]. International Journal of Computer Vision, 2008, 77(1): 125-141.
- [9] 孙涛,谷士文,费耀平. 基于 PCA 算法的人脸识别方法研究比较[J]. 现代电子技术, 2007, 30(1): 112-114.
- [10] Mei X, Ling H. Robust visual tracking using l1 minimization[C]//Proceedings of the IEEE International Conference on Computer Vision (ICCV09), 2009: 1436-1443.
- [11] Wang Q, Chen F, Xu W, et al. Online discriminative object tracking with local sparse representation[C]//Proceedings of the Workshop on the Applications of Computer (WACV12), 2012: 425-432.
- [12] Fan J, Liang R Z. Stochastic learning of multi-instance dictionary for earth mover's distance-based histogram comparison[J]. Neural Computing & Applications, 2016: 1-11.
- [13] 詹曙,王俊,杨福猛,等. 基于 Gabor 特征和字典学习的高斯混合稀疏表示图像识别[J]. 电子学报, 2015, 43(3): 523-528.
- [14] 胡昭华,徐玉伟,赵孝磊,等. 多特征联合的稀疏跟踪方

- 法[J]. 计算机应用, 2014, 34(8): 2380-2384.
- [15] 林晓帆, 林立文, 邓涛. 基于 SURF 描述子的遥感影像配准[J]. 计算机工程, 2010, 36(12): 216-218.
- [16] 钟华, 金国平, 郑林华. 基于扩展卡尔曼滤波的 FLANN 网络学习算法[J]. 信号处理, 2009, 25(10): 1555-1559.
- [17] 肖辉辉, 段艳明. 基于属性值相关距离的 KNN 算法的改进研究[J]. 计算机科学, 2013, 40(11): 157-187.
- [18] 张晓伟, 李明, 左磊, 等. 基于基追踪-Moore-Penrose 逆矩阵算法的稀疏信号重构[J]. 电子与信息学报, 2013, 35(2): 388-393.
- [19] Wu Y, Shen B, Ling H. Online robust image alignment via iterative convex optimization[C]//Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR12), 2012: 1808-1814.
- [20] Zhang K, Zhang L, Yang M H. Real-time compressive tracking[C]//Proceedings of the European Conference on Computer Vision (ECCV12), 2012: 864-877.
- [21] Perez P, Hue C, Vermaak J, et al. Color-Based probabilistic tracking[C]//Proceedings of the European Conference on Computer Vision (ECCV02), 2002: 661-675.

Research on template matching tracking algorithm based on local sparse representation

BAO Xiaolan, CHEN Lei, WAN Weixiang, ZHANG Junwei, GUI Jiangsheng

(School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China)

Abstract: In the target tracking process, when the target is sheltered excessively, moves fast is deformed largely, the target deviation phenomenon will occur to local sparse representation tracking algorithm(LSRTA) in the target tracking process. To solve this problem, based on the LSRTA, the template matching algorithm which combines surf, flann and knn algorithm was integrated, and a template matching tracking algorithm(LSRTMTA) based on local sparse representation was proposed to solve the target offset phenomenon. In the LSRTA tracking process, the offset is judged through calculating the matching value of the new template and the current template. When the target is offset, the LSRTA tracking stops. The target is repositioned by matching between the template and frame image. When the target position is determined, the LSRTA tracking starts again. Experimental results show that LSRTMTA not only retains the advantages of LSRTA, but also has the function of automatic correction of target offset, improve tracking effect and increases fault tolerance in the target tracking process.

Key words: target tracking; template matching algorithm; target tracking offset; automatic correction

(责任编辑: 康 锋)