

遮挡环境下多示例学习分块目标跟踪

才 华¹, 陈广秋¹, 刘广文¹, 程 帅¹, 于化东²

(1. 长春理工大学 电子信息工程学院, 长春 130022; 2. 长春理工大学 机电工程学院, 长春 130022)

摘 要:针对跟踪过程中遮挡导致错误累积而产生目标漂移甚至目标丢失的问题, 提出多示例学习分块目标跟踪方法。该方法以随机蕨为基础检测器, 通过多示例学习在线更新检测器, 提高检测器对目标变化的适应能力及学习的准确性。将目标均匀分成多个子块, 选取部分子块作为候选集合, 每个候选块分配一个检测器, 利用检测器完成每个候选块的跟踪, 根据候选块的信息确定目标最终位置。实时检测候选块的有效性, 替换无效的候选块, 提高跟踪的鲁棒性。在目标被遮挡等复杂条件下, 与目前主流跟踪算法进行了对比实验, 结果表明该算法能够有效解决目标漂移甚至跟踪丢失问题, 具有更高的跟踪精确度及更好的鲁棒性。

关键词:信息处理技术; 随机蕨分类器; 多示例学习; 分块; 无效子块更换

中图分类号: TP391.41 **文献标志码:** A **文章编号:** 1671-5497(2017)01-0281-07

DOI: 10.13229/j.cnki.jdxbgxb201701041

Novelty fragments-based target tracking with multiple instance learning under occlusions

CAI Hua¹, CHEN Guang-qiu¹, LIU Guang-wen¹, CHENG Shuai¹, YU Hua-dong²

(1. School of Electronic and Information Engineering, Changchun University of Science and Technology, Changchun 130022, China; 2. College of Mechanical and Electric Engineering, Changchun University of Science and Technology, Changchun 130022, China)

Abstract: To solve the problem that tracking algorithm may lead to drift or failure due to the accumulated error under the occlusion environment, a Multiple instance learning based Fragment Tracker (MFT) is proposed. In this MFT, the random ferns is used as the basic detector. To improve the adaption of the target change and the precision of the learning, the multiple instance learning is introduced to online update the detector. The object is segmented into fragments and parts of them are selected as the candidate set. The candidate block is tracked by the corresponding detector. The object can be finally located by the selected blocks. A real-time valid detection is made for the candidate blocks and the invalid block is replaced with an appropriate block to improve the robustness of the tracking. Experiments on variant challenging image sequence in the occlusion environment were carried out. Results show that, compared with the state-of-art trackers, the proposed MFT solves the problem of target drift and failure efficiently and has higher accuracy and better robust.

收稿日期: 2016-02-21.

基金项目: 吉林省科技发展计划项目(20130101179JC); 教育部留学基金委留学归国人员科研启动基金(教外师留1685).

作者简介: 才华(1977-), 男, 副教授, 博士. 研究方向: 图像处理与机器视觉. E-mail: caihua@cust.edu.cn

通信作者: 陈广秋(1977-), 男, 副教授, 博士. 研究方向: 图像(序列)配准与融合. E-mail: guangqiu_chen@126.com

Key words: information processing; random ferns detector; multiple instance learning; fragment; invalid block replacement

0 引言

如何有效处理遮挡是评价跟踪算法鲁棒性的关键指标^[1,2]。遮挡是目标跟踪过程中常见的情况,包括目标自身自遮挡和被背景中的物体遮挡,从目标自身角度看,可分为部分遮挡或全部遮挡^[3]。进入遮挡时,目标的信息逐渐丢失,如颜色、轮廓、运动和特征点等,而出遮挡时,这些信息又逐渐恢复。文献^[4]采用预测模型的方法如粒子滤波,对遮挡目标的位置进行预测,此类算法对目标长时间全部遮挡后,再出现在视场中无法重新定位。基于遮挡检测的跟踪算法^[5-8]可有效处理目标被部分遮挡的情况,但目标被全部遮挡或者消失后再次出现在视场中的情况,此类算法不能实现准确的定位及跟踪。检测-跟踪算法^[9-12]将跟踪与检测相结合,此类算法可有效处理目标全部遮挡后再出现在视场中,可准确定位。但目标被部分遮挡时,此类算法常发生目标漂移。基于子块法^[13,14]可有效处理部分遮挡问题,将目标划分为多个小块,根据部分子块信息确定目标的中心坐标以达到跟踪的目的。此类算法没有对子块信息进行更新及有效性检测,导致不能满足目标本身和外界条件的实时变化,利用无效子块对目标位置定位导致目标漂移问题,同时目标全部遮挡后再出现在视场中不能准确地跟踪。

为解决跟踪过程中部分遮挡产生目标漂移及目标全部遮挡再次出现视场中无法正确跟踪的问题,本文提出一种多示例学习分块目标跟踪(MFT)方法,该方法采用随机蕨作为基础检测器,结合全局多尺度扫描窗口,实现目标全部遮挡后重新出现在视场中的准确定位。为保证检测器对目标变化的适应能力及学习的准确性,避免目标漂移而导致跟踪失败,通过多示例学习在线更新随机蕨分类器。为解决部分遮挡问题,将目标分块处理,从中选取部分子块作为候选块,同时每个候选块分配一个检测器,利用检测器对每个候选块进行跟踪,根据候选块信息确定目标最终位置。实时检测候选块的有效性,利用有效的候选块实时替换和更新无效的候选块,从而在目标受到部分遮挡时,利用有效信息实现目标的准确跟踪。

1 多示例学习分块目标跟踪

1.1 随机蕨分类器及其在线更新

采用具有实时检测能力的随机蕨构建检测器,并结合多尺度滑动窗口^[15]完成当前帧目标的检测,解决目标被遮挡或消失后再次出现在视场中不能稳定跟踪问题。通过最近邻算法将检测器得到的图像块与在线模板匹配,将符合要求图像块作为最后的跟踪结果,同时在线更新模板库,以适应目标自身及外界环境变化。采用通过多示例学习在线更新随机蕨检测器,以保证检测器对目标变化的适应能力及学习的准确性,避免漂移而导致跟踪失败。

多示例学习算法^[16]对多示例包进行标注,包中至少含有一个正示例,则包为正样本,反之为负样本。训练集为 $\{(\mathbf{X}_1, y_1), \dots, (\mathbf{X}_n, y_n)\}$, 其中 $\mathbf{X}_i = \{x_{i1}, \dots, x_{im}\}, y_i \in \{0, 1\}$ 为包的标签。样本包标签定义为:

$$y_i = \max_j(y_{ij}) \quad (1)$$

式中: y_{ij} 为示例标签。

通过采用 boosting 方法训练分类器,使用示例似然概率 $p(y_i | x_{ij})$ 的 Noisy-OR 模型计算包的似然概率 $p(y_i | \mathbf{X}_i)$:

$$p(y_i | \mathbf{X}_i) = 1 - \prod_j (1 - p(y_i | x_{ij})) \quad (2)$$

各示例的似然概率计算公式为:

$$p(y | x) = \sigma(H(x)) \quad (3)$$

式中: $\sigma(x) = \frac{1}{1 + e^{-x}}$; $H(x)$ 为强分类器, $H(x)$

$= \sum_{k=1}^K \alpha_k h_k(x)$, α_k 为权值, $h_k(x)$ 为弱分类器。

随机蕨检测器检测到被跟踪的目标,并对目标图像扭曲、旋转、缩放建立完整的目标全视角数据库,以此作为正样本,选取目标周围的背景区域作为负样本。算法 1 为多示例学习在线更新过程,为保证跟踪实时性及准确性,多示例学习将随机蕨中的每一个蕨看作弱分类器 h , 在线构建一个含有 M 个弱分类器池,利用训练数据更新所有的弱分类器 h , 从中选取最优的 K 个弱分类器构建强分类器,即:

$$h_k = \operatorname{argmax}_{h \in \{h_1, \dots, h_m\}} L(H_{k-1} + h) \quad (4)$$

式中: $L = \sum_i (\log p(y_i | \mathbf{X}_i))$ 。

算法 1 多示例学习在线更新

输入: 数据集 $\{X_i, y_i\}_{i=1}^N, X_i = \{x_{i1}, x_{i2}, \dots, x_{iS}\}, y_i \in \{0, 1\}$, 使用数据集更新 M 个弱分类器,
初始化 $H_{ij} = 0, i = 1, 2, \dots, N, j = 1, 2, \dots, S$
for $k = 1$ to K do
 for $m = 1$ to M do
 $p_{ij}^m = \sigma(H_{ij} + h_m(x_{ij}))$
 $p_i^m = 1 - \prod_j (1 - p_{ij}^m)$
 $L^m = \sum_i (y_i \log(p_i^m) + (1 - y_i) \log(1 - p_i^m))$
 end for
 $m^* = \arg\max L^m$
 $h_k(x) \leftarrow h_{m^*}(x)$
 $H_{ij} = H_{ij} + h_k(x)$
end for
输出: 分类器 $H(x) = \sum \alpha_k h_k(x)$, 其中, $p(y | x) = \sigma(H(x))$

1.2 目标分块、定位及无效候选块更新

将目标均匀分割为 K 个子块, 图 1 为分割结果图。均匀分割后, 得到包含 K 个子块的集合 $P = \{p_1, p_2, \dots, p_K\}$, 试验中 $K = 8$ 。计算每个滑动窗口 f_j 和 p_i 之间特征向量的相似度 S_{ij} , 得到最大相似度 $ML_i = \max(S_{ij})$ 。选取 $M (M \leq K)$ 个 ML_i 值较大的子块构成候选块集合 $SP = \{sp_1, sp_2, \dots, sp_M\}$, 试验中 $M = 4$ 。



图 1 分割结果图

Fig. 1 Segmentation of object

目标及候选块表示为: 在第 t 帧, $O^t = (X^t, Y^t, W^t, H^t)$ 代表目标, 其中, X^t, Y^t 为中心坐标, W^t, H^t 为目标的高和宽。 $sp_k^t = (x_k^t, y_k^t, w_k^t, h_k^t)$ 为第 k 个候选块, 其中, (x_k^t, y_k^t) 为中心坐标, (w_k^t, h_k^t) 为候选块的高和宽。 $z_k^t = (\alpha x_k^t, \alpha y_k^t, r w_k^t, r h_k^t)$ 为第 k 个候选块和目标之间的偏移量, 其中, $(\alpha x_k^t, \alpha y_k^t)$ 为偏移量的中心坐标, $r w_k^t$ 为第 k 个候选块高与目标高之间的比值, $r h_k^t$ 为两者之间宽的比值, z_k^t 的计算公式为:

$$\begin{cases} \alpha x_k^t = X^t - x_k^t, \alpha y_k^t = Y^t - y_k^t \\ r w_k^t = \frac{W^t}{w_k^t}, r h_k^t = \frac{H^t}{h_k^t} \end{cases} \quad (5)$$

图 2 为目标最终位置确定过程。具体过程如下所示。

首先, 在第一帧手动选择被跟踪目标, 对目标分割并选择候选块, 计算 $O^1, sp_k^1, z_k^1, k = 1, 2, 3, 4$ 。

其次, 在第 t 帧, 每个候选块对应一个检测器, 完成候选块的跟踪, 得到 sp_k^t 在 $t+1$ 帧的跟踪结果 $sp_k^{t+1} = (x_k^{t+1}, y_k^{t+1}, w_k^{t+1}, h_k^{t+1})$, 黄色、红色、蓝色、绿色的实线区域分别为候选块 $sp_1^{t+1}, sp_2^{t+1}, sp_3^{t+1}, sp_4^{t+1}$ 。对应目标子块 $O_k^{t+1} = (X_k^{t+1}, Y_k^{t+1}, W_k^{t+1}, H_k^{t+1})$ 计算公式为:

$$\begin{cases} X_k^{t+1} = x_k^{t+1} + \alpha x_k^{t+1}, Y_k^{t+1} = y_k^{t+1} + \alpha y_k^{t+1} \\ W_k^{t+1} = w_k^{t+1} \cdot r w_k^{t+1}, H_k^{t+1} = h_k^{t+1} \cdot r h_k^{t+1} \end{cases} \quad (6)$$

虚线区域为对应的跟踪结果 $O_1^{t+1}, O_2^{t+1}, O_3^{t+1}, O_4^{t+1}$ 。目标最终位置(黑色加粗实线)计算公式为:

$$O^{t+1} = \sum_{i=1}^M w_i O_i^{t+1} \quad (7)$$

式中: M 为候选块个数。

权值 w_i 计算公式为:

$$w_i = a \exp(-|l^t(x, y) - l_i^{t+1}(x, y)|) \quad (8)$$

式中: $l^t(x, y)$ 为 t 时刻目标的中心坐标; $l_i^{t+1}(x, y)$ 为 $t+1$ 时刻候选块 i 的中心坐标; a 为归一化常量。

最后, 重新调整每个候选块相对于跟踪目标的偏移量, 将新得到的目标区域 O^{t+1} 重新分割, 根据式(5)重新计算候选块偏移量, 并利用多示例学习更新每个候选块对应的随机森林分类器。

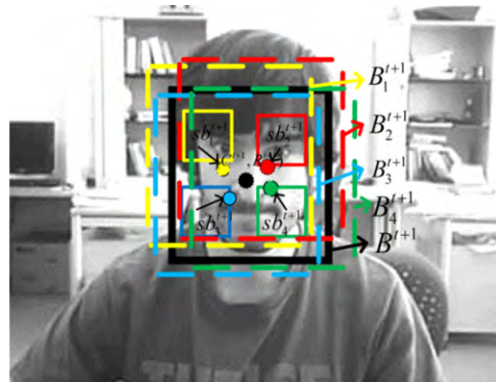


图 2 目标位置确定过程

Fig. 2 Process of locating object

为解决遮挡和目标外观变化的影响, 自适应更新候选块集合, 从而引入新的信息以适应目标外观的动态变化。利用随机森林分类器检测候选块的有效性, 如果候选块后验概率 $p_{avg} \leq 0.5$ 时, 表

示候选块无效,利用未使用的子块取代无效的候选块。

在第 t 帧,取代无效候选块 sp_k^t 过程:首先,对目标进行分块,得到未使用子块集合 $UP = \{up_1, up_2, \dots, up_N\}, N = K - M, UP \cup SP = P$ 。其次,提取 up_j 特征向量,计算特征向量相似度:

$$s_j^t = \text{sim}(x_j^t, x_j^q) \quad (9)$$

式中: sim 为相似度函数; x_j^t 为 up_j 的特征向量; x_j^q 为 up_j 最近一次第 q 帧被使用的特征向量(如果 up_j 没有被使用, $q = 1$)。最后,利用如下公式得到的 up_j 取代 sp_k^{t+1} :

$$up_\tau = \text{argmax}_j(s_j^t) \quad (10)$$

图 3 为候选块更新过程。在图 3(a)中,通过候选块集合得到目标位置。在图 3(b)中,当目标发生遮挡时,每个候选块依然有效,可通过候选块得到目标位置。在图 3(c)中,当目标受到遮挡,候选块 2、候选块 4 无效,被未使用集合中的子块 5、子块 7 所取代,目标依然可以成功跟踪。多示例学习分块跟踪算法如下:

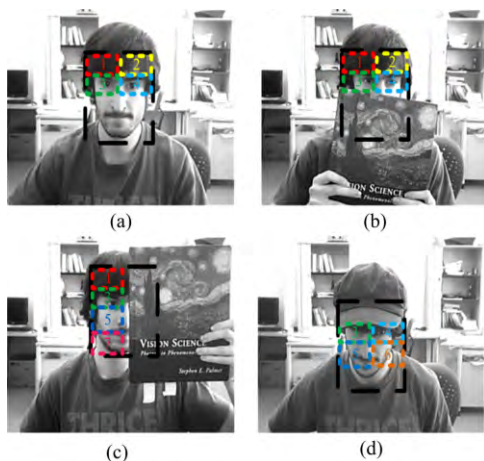


图 3 候选块更新过程

Fig. 3 Process of updating

算法 2 多示例学习分块目标跟踪算法

初始化:

将目标分为 K 个子块,得到 $P = \{p_1, p_2, \dots, p_K\}$,选择候选块集合 $SP = \{sp_1, sp_2, \dots, sp_M\} (M < K)$,计算第一帧偏移量集合 $\{z_k^1\}_{k=1}^M$,为每个候选块生成正样本集合 (x_k^+, y_k^+) 和负样本集合 $(x_k^-, y_k^-), k = 1, 2, \dots, M$,利用正、负样本集合为每个候选块训练随机森林分类器 $f_k, k = 1, 2, \dots, M$

跟踪:

```
for t=2 to H(图片总数) do
  for k=1 to M do
    利用随机森林分类器计算  $sp_k^t$ 
```

通过式(6)计算 O_k^t

多示例学习更新并重新训练分类器 f_k

end for

通过式(7)计算 O^t

通过式(5)调整偏移量集合

候选块更新:

```
for k=1 to M do
```

利用随机森林分类器检测每个候选块

```
  if  $p_{avg} < 0.5$  do
```

更新 $sp_k \leftarrow up_\tau$

为新的候选块生成正、负样本集合

break;

```
  end if
```

```
end for
```

```
end for
```

2 实验结果及分析

将本文算法与 8 个主流跟踪算法在不同图片序列上进行测试,对比跟踪算法为: CT^[17], CXT^[18], Frag^[13], TLD^[11], IVT^[19], MIL^[16], OAB^[20], MILRFD^[21], 这些算法参数均使用默认值,具体实现参考相应文献。选用 faceOcc2^[22] 等 8 组具有挑战的图片序列进行测试。8 种跟踪算法在每组图片序列跟踪结果的中心误差、重叠率如表 1 所示。

中心误差表示跟踪结果与真实目标的中心位置的欧氏距离,距离越小表示跟踪结果越有效、越精确。由表 1 可得, MFT 算法有 4 组为最优, 4 组为次优, mean 行为中心误差均值, MFT 算法的中心误差均值明显小于其他算法,表明跟踪结果具有较高的精确度。

重叠率为跟踪结果与真实位置之间的重叠比例,计算公式为 $\text{overlap} = (A \cap B) / (A \cup B)$, 其中 A 为目标真实位置的矩形框, B 为跟踪结果的矩形框。重叠率越高表明跟踪结果越接近真实位置,跟踪结果越准确。由表 1 可得, MFT 算法有 7 组为最优, mean 行为重叠率均值, MFT 算法的重叠率均值高于其他算法,可见 MFT 算法具有更高的准确度和鲁棒性。

表 2 为跟踪算法在 8 组图片序列处理速度,由表 2 可知: CT 算法的处理速度明显高于其他算法,这是因为 CT 算法是基于压缩感知理论,利用符合压缩感知条件的随机感知矩阵对多尺度高维图像特征进行降维,降维后的特征依然可以高概率地重构出高维特征,很大程度上降低了特征

表 1 中心误差和重叠率
Table 1 Center error and overlap ratio

图片序列	算 法								
	CT	CXT	Frag	TLD	IVT	MIL	OAB	MILRFD	MFT
faceOcc1	19.40	18.04	9.49	12.22	14.74	16.43	42.38	10.02	<u>5.68</u>
	(0.41)	(0.12)	(0.43)	(0.48)	(0.50)	(0.55)	(0.01)	(0.45)	(0.71)
faceOcc2	12.47	<u>4.81</u>	14.57	8.77	11.51	7.40	11.60	8.35	4.91
	(0.34)	(0.38)	(0.78)	(0.43)	(0.59)	(0.60)	(0.68)	(0.69)	(0.64)
tiger2	35.42	20.88	43.41	59.68	25.12	21.56	20.36	13.62	<u>7.42</u>
	(0.14)	(0.01)	(0.15)	(0.12)	(0.31)	(0.43)	(0.28)	(0.52)	(0.61)
subway	6.87	69.81	8.25	68.09	74.77	43.83	56.44	21.75	<u>3.82</u>
	(0.52)	(0.18)	(0.56)	(0.38)	(0.17)	(0.29)	(0.17)	(0.35)	(0.59)
woman	59.31	32.04	53.42	93.74	72.12	62.02	16.52	28.71	<u>8.23</u>
	(0.13)	(0.24)	(0.14)	(0.04)	(0.14)	(0.15)	(0.49)	(0.65)	(0.80)
walking2	41.76	23.84	17.33	<u>6.63</u>	33.58	30.52	16.04	25.68	8.11
	(0.26)	(0.35)	(0.48)	(0.45)	(0.30)	(0.26)	(0.37)	(0.47)	(0.51)
car	50.08	45.77	60.41	<u>10.07</u>	86.01	60.22	40.67	59.23	15.21
	(0.01)	(0.31)	(0.02)	(0.67)	(0.15)	(0.28)	(0.29)	(0.62)	(0.69)
carchase	90.06	70.12	80.34	<u>20.90</u>	100.34	97.67	85.58	62.45	21.43
	(0.01)	(0.05)	(0.01)	(0.53)	(0.01)	(0.22)	(0.26)	(0.49)	(0.56)
mean	39.38	35.66	24.41	35.01	52.27	42.46	36.20	28.72	<u>9.35</u>
	(0.22)	(0.21)	(0.32)	(0.39)	(0.27)	(0.35)	(0.32)	(0.53)	(0.64)

注:其中括号外数值为中心误差,括号内数值为重叠率,划线数值为最好结果,黑体数值为次好结果。

表 2 算法处理速度

Table 2 Processing speed of algorithm

算法	平均速度/(帧·s ⁻¹)	算法	平均速度/(帧·s ⁻¹)
CT	<u>29.7</u>	MIL	9.8
CXT	8.6	OAB	7.6
Frag	4.1	MILRFD	11.7
TLD	9.7	MFT	13.9
IVT	7.2		

注:划线数值为最好结果;黑体数值为次好结果。

提取的时间复杂度,而且分类过程仅采用简单的朴素贝叶斯分类器,提高了算法的运算速度。

图 4 为 8 种跟踪算法在 8 组图片序列中某些帧的跟踪效果对比跟踪结果。在 faceOcc1 和 faceOcc2 序列中背景基本保持不变,目标在跟踪过程中发生多次和长时间部分遮挡,导致 OAB、TLD、CT、CXT 算法都出现严重的目标漂移现象甚至跟踪失败,在目标大部分区域被遮挡时,MFT

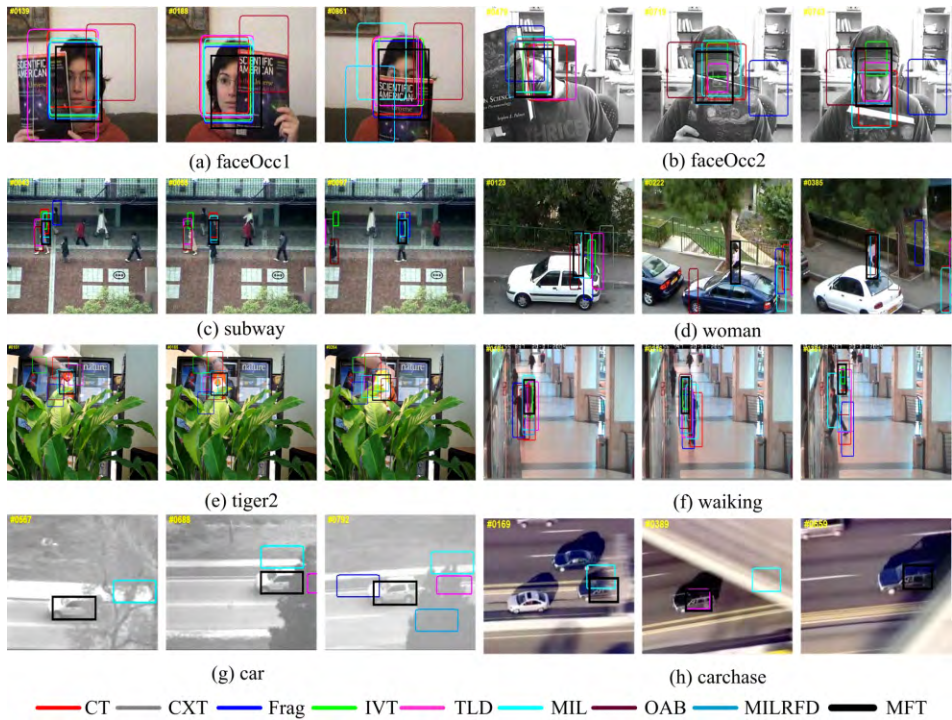


图 4 跟踪结果图

Fig. 4 Tracking result

算法仍能够正确跟踪,这表明 MFT 算法可有效处理长时间遮挡,实现稳定的跟踪。

在 subway 和 woman 序列中,目标在行走过程中不断被其他物体所遮挡,且观测角度不断变化,导致有些算法在跟踪过程的前几帧就出现错误跟踪,如 IVT、OAB、TLD、Frag 算法,CT 算法出现目标漂移问题。在后续帧中只有 MFT 算法能完成整个序列的跟踪,并实现稳定的跟踪。可见,在目标被不断遮挡的条件下,MFT 算法能够实现正确跟踪。

在 walking2 和 tiger2 序列中目标在运动过程中,不断被遮挡,同时伴有运动模糊条件的影响。导致 CT、CXT、Frag、OAB、MIL 在跟踪前期就已经跟踪失败。目标在遮挡结束后,受到遮挡物的干扰,MFT 算法能正确跟踪目标,其他算法都已跟踪失败。结果表明 MFT 算法在遮挡、模糊拖影等条件下具有较好的跟踪效果。

在 car 和 carchase 序列中,目标在运动过程中常发生部分遮挡及全部遮挡,且在视场中消失或者遮挡后又重新出现,在目标被部分遮挡时,部分算法出现目标漂移问题。当目标被全部遮挡或消失后再次出现时,大部分算法无法重新定位到目标,导致跟踪失败,MFT 算法在这两种情况下,均能实现正确的跟踪。可见,在部分遮挡及目标再次出现在视场中,MFT 算法均能实现稳定的跟踪。

3 结束语

提出了以随机藤作为基础检测器的多示例学习分块跟踪算法。利用随机藤检测器和滑动窗口完成目标跟踪,通过多示例学习算法在线更新检测器。将目标分块处理,为每个子块分配随机藤检测器进行跟踪,通过每个子块信息确定目标最终位置。实时检测无效子块并进行更新。通过几组图片序列进行测试,在遮挡等复杂环境下,本文算法的中心误差为 9.35、重叠率为 64%,平均帧率为 13.9 frame/s,实验结果表明,本文算法具有更高的精确度和鲁棒性,有效解决了目标被遮挡导致的目标漂移和跟踪丢失问题,具有极高的实用价值。本文算法对跟踪非刚性或者变形较大的物体跟踪效果较差,如何解决这一问题将是下一步的研究工作。

参考文献:

[1] 薛陈,朱明,刘春香. 遮挡情况下目标跟踪算法综

述[J]. 中国光学与应用光学, 2009, 2(5):388-394.

Xue Chen, Zhu Ming, Liu Chun-xiang. Review of tracking algorithms under occlusions[J]. Chinese Journal of Optics and Applied Optics, 2009, 2(5): 388-394.

[2] 颜佳,吴敏渊. 遮挡环境下采用在线 Boosting 的目标跟踪[J]. 光学精密工程, 2012, 20(2):439-446.

Yan Jia, Wu Min-yuan. On-line boosting based target tracking under occlusion[J]. Optics and Precision Engineering, 2012, 20(2):439-446.

[3] Alper Y, Omar J, Mubarak S. Object tracking: a survey[J]. ACM Computing Surveys, 2006, 38(4): 112-123.

[4] Medeiros H, Park J, Kak A. A parallel color-based particle filter for object tracking[C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Anchorage, USA, 2008: 1-8.

[5] Zhang T Z, Ghanem B, Xu C, et al. Object tracking by occlusion detection via structured sparse learning [C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Portland, USA, 2013: 1033-1040.

[6] Zhang S P, Wu J P, Tian Y, et al. Robust visual tracking based on occlusion detection and particle redistribution[C]// The Second International Conference on Internet Multimedia Computing and Service, New York, USA, 2010: 159-162.

[7] Mei X, Ling H B, Wu Y, et al. Minimum error bounded efficient L1 tracker with occlusion detection [C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, USA, 2011: 1257-1264.

[8] 陈东成,朱明,高文,等. 在线加权多示例学习实时目标跟踪[J]. 光学精密工程, 2014, 22(6): 1661-1667.

Chen Dong-cheng, Zhu Ming, Gao Wen, et al. Real-time object tracking via online weighted multiple instance learning[J]. Optics and Precision Engineering, 2014, 22(6): 1661-1667.

[9] Andriluka M, Roth S, Schiele B. People-tracking-by-detection and people-detection-by-tracking[C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Anchorage, USA, 2008: 1-8.

[10] Gall J, Yao A, Razavi N, et al. Hough forests for object detection, tracking, and action recognition [J]. IEEE Trans Pattern Analysis and Machine Intelligence, 2011, 33(11): 2188-2202.

- [11] Kalal Z, Matas J, Mikolajczyk K. Online learning of robust object detectors during unstable tracking [C]// IEEE International Conference on Computer Vision Workshops, Kyoto, Japan, 2009: 1417-1424.
- [12] 贾松敏, 文林风, 王丽佳. 基于多模板回归加权均值漂移的人体目标跟踪[J]. 光学精密工程, 2016, 24(9): 2339-2346.
Jia Song-min, Wen Lin-feng, Wang Li-jia. Person tracking based on multi-template regression weighted mean shift[J]. Optics and Precision Engineering, 2016, 24(9): 2339-2346.
- [13] Adam A, Rivlin E, Shimshoni I. Robust fragment-based tracking using the integral histogram[C]// IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, USA, 2006: 798-805.
- [14] Naik N, Patil S, Joshi M. A fragment based scale adaptive tracker with partial occlusion handling[C]// IEEE Region 10 Conference, Singapore, 2009: 1-6.
- [15] Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(7): 1409-1422.
- [16] Babenko B, Yang M H, Belongie S. Robust object tracking with online multiple instance learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(8): 1619-1632.
- [17] Zhang K, Zhang L, Yang M H. Fast compressive tracking[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(10): 2002-2015.
- [18] Dinh T B, Vo N, Medion G. Context tracker: exploring supporters and distracters in unconstrained environments[C]// IEEE Conference on Computer Vision and Pattern Recognition, Providence, USA, 2011: 1177-1184.
- [19] Ross D A, Lim J, Lin R S, et al. Incremental learning for robust visual tracking[J]. International Journal of Computer Vision, 2008, 77(1-3): 125-141.
- [20] Grabner H, Grabner M, Bischof H. Real-time tracking via on-line boosting[C]// British Machine Vision Conference, Edinburgh, Britain, 2006: 47-56.
- [21] 罗艳, 项俊, 严明君, 等. 基于多示例学习和随机森林检测的在线目标跟踪[J]. 电子与信息学报, 2014, 36(7): 1605-1611.
Luo Yan, Xiang Jun, Yan Ming-jun, et al. Online target tracking based on multiple instance learning and random ferns detection[J]. Journal of Electronics & Information Technology, 2014, 36(7): 1605-1611.
- [22] Wu Y, Lim J, Yang M H. Online object tracking: a benchmark [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 36(4): 2411-2418.