

算力限制平台下的长时目标跟踪

王向军^{1,2}, 郭志翼^{1,2}

- (1. 天津大学 精密测试技术及仪器国家重点实验室, 天津 300072;
2. 天津大学 微光机电系统技术教育部重点实验室, 天津 300072)

摘要: 为了满足某些算力受到限制的应用场景的长时跟踪需求, 如以 C64x+ DSP 为核心的嵌入式系统, 提出了一种由连续跟踪环节与目标检测环节两部分构成的低时间复杂度长时跟踪算法, 连续跟踪环节基于自适应更新的时空上下文算法(STC), 目标检测环节使用归一化互相关匹配算法。在没有目标出视场、目标快速移动等特殊跟踪场景时, 连续跟踪环节输出跟踪结果, 在跟踪失败后, 目标检测环节对全幅图像进行处理, 只要目标出现在图像中, 便可以重新锁定目标。经实验验证, 目标检测环节可以在目标出现后准确检测到目标, 满足了长时跟踪的要求。同时, 目标检测环节在跟踪不可靠时的辅助定位也提升了连续跟踪环节的鲁棒性, 使用 OTB2013 数据集测试, 本算法的精确度较 STC 算法提升了 4.95%。

关键词: 长时跟踪; 算力限制; 时空上下文算法; 模板匹配

中图分类号: TP391.4 **文献标志码:** A **DOI:** 10.3788/IRLA202049.0126003

Long-term target tracking algorithms in force-constrained platform

Wang Xiangjun^{1,2}, Guo Zhiyi^{1,2}

- (1. State Key Laboratory of Precision Measuring Technology and Instruments, Tianjin University, Tianjin 300072, China;
2. MOEMS Education Ministry Key Laboratory, Tianjin University, Tianjin 300072, China)

Abstract: In order to meet the long-term tracking requirements of platforms with weak computing power, such as the embedded system with C64x+ DSP as the computing core, a long-term tracking algorithm with low time complexity was proposed, which consisted of two parts. One part was the continuous tracking part and the other part was the target detection part. The continuous tracking part was based on adaptive update spatio-temporal context algorithm (STC), and the target detection part used normalized cross-correlation matching algorithm. If there were no special tracking scenarios such as field of view and fast moving target, the continuous tracking part outputted tracking results. After tracking failure, the whole image was processed in the target detection part, and the target was re-locked as long as the target appeared in the image again. Experiments show that target detection can accurately detect the target after it appeared, which meets the requirements of long-term tracking. At the same time, the robust of continuous tracking is also improved because the target location is redefined by the target detection part, when the result is not reliable. Using OTB2013 data set to test, the accuracy of this algorithm is 4.95% higher than that of STC algorithm.

Key words: long-term tracking; computational constraints; STC algorithm; template matching

收稿日期: 2019-09-05; 修订日期: 2019-10-15

基金项目: 国家自然科学基金(51575388)

作者简介: 王向军(1955-), 男, 教授, 博士, 主要从事光电传感与测试、计算机视觉与图像分析、微小型光机电系统及 MEMS 方面的研究。Email: xdocuxjw@vip.163.com

0 引言

目标跟踪作为计算机视觉研究领域中的热点之一^[1],也在智能视频监控、智能机器人等领域有着广泛的应用^[2-3],同时,很多复杂的计算机视觉问题都可以分解成目标跟踪问题及其他问题的组合,因此,研究目标跟踪问题具有重要意义。目标跟踪问题又分为短时跟踪与长时跟踪两个问题,短时跟踪一般指几分钟以内的跟踪,长时跟踪指十几分钟及以上的跟踪,并且长时跟踪常常面临复杂的跟踪场景,如遮挡、目标出视场^[4-5]等。长时跟踪算法的典型算法是 2011 年 Zdenek Kalal^[6]提出的 TLD 算法,该算法包括追踪器、检测器和分类器三部分,检测器可以估计追踪器的误差,并使用检测器的结果更新追踪器,可以在目标形变、部分遮挡时更稳定地跟踪目标。近年来,深度学习也开始在目标跟踪领域显现其优势,2018 年,Zheng Zhu^[7]等人提出了 DasiamRPN 算法,该算法分为模板分支和检测分支,采用单样本检测的方式,并使用数据集增强了模型的效果,使得该算法在长时跟踪领域有着出色的性能。2019 年,Bo Li^[8]等人提出了 SiamRPN++ 算法,通过修改采样策略,发挥了深层网络的优势。2019 年,Martin Danelljan 等人提出了 ATOM 目标跟踪算法,该算法将目标跟踪分为分类与评价两个阶段,分别用于大致定位与精确定位,其中精确定位借鉴了目标检测问题中的 IoUNet 网络,该算法性能较 DasiamRPN 有所增强,速度也较快,在 GPU 下能到到每秒 30 帧^[9]。虽然 SiamRPN++、ATOM 等算法已具有很好的性能。但在实际应用中,某些平台存储空间有限,处理器不支持浮点运算^[10],运算速度慢,如以 C64x+ DSP 为核心的嵌入式平台,这些算法在这些平台下远远无法满足实时性的要求。然而这些平台在现实生活中又有着巨大的应用空间,如可穿戴设备,无人机等应用场合。为此,需要尽可能简化跟踪算法的框架,降低算法的运算复杂度。

为了在保证满足长时目标跟踪需求的前提下尽可能降低运算复杂度,文中提出了一种长时目标跟踪算法,该算法包括连续跟踪环节与目标检测环节。连续跟踪环节基于时空上下文算法(STC),目标检测环节采用归一化互相关匹配算法。

1 时空上下文算法

时空上下文算法是基于贝叶斯框架^[11],该算法的优势在于利用了目标周围的背景信息,它对目标及局部区域进行建模,计算得到置信图,置信图表示目标所在位置的可能性,置信图的最大响应值所在位置即为目标位置,可以构造置信图的模型如公式(1)所示:

$$c(x)=be^{-\left|\frac{x-x^*}{\alpha}\right|^\beta} \quad (1)$$

由贝叶斯框架,可得置信图的求解公式如下所示:

$$c(x)=\sum_{c(z) \in X^c} P(x|c(z), o)P(c(z)|o) \quad (2)$$

式中: x 为目标位置; X^c 为上下文特征集; o 表示目标出现在视场; $c(x)$ 为置信图。 $P(x|c(z), o)$ 为条件概率定义如下:

$$P(x|c(z), o)=h^\infty(x-z) \quad (3)$$

式中: $h^\infty(x-z)$ 为空间上下文模型,反映了目标点 x 与周围区域内点 z 的空间关系。

$P(c(z)|o)$ 是先验概率,定义如下:

$$P(c(z)|o)=I(z)\omega_\sigma(z-x^*) \quad (4)$$

式中: $I(z)$ 为输入图像的灰度特征; $\omega_\sigma(z-x^*)$ 为权重函数,该权重函数定义如下:

$$\omega_\sigma(z)=ae^{-\frac{|z|^2}{\sigma^2}} \quad (5)$$

式中: a 为归一化常数; σ 为尺度参数。

联立公式(1)~(4),可得到置信图计算公式如下所示:

$$be^{-\left|\frac{x-x^*}{\alpha}\right|^\beta}=\sum_{c(z) \in X^c} h^\infty(x-z)I(z)\omega_\sigma(z-x^*) \quad (6)$$

使用傅里叶变换简化计算,可得空间上下文计算公式如下所示:

$$h^\infty(x)=F^{-1}\left(\frac{F\left(be^{-\left|\frac{x-x^*}{\alpha}\right|^\beta}\right)}{F(I(z)\omega_\sigma(z-x^*))}\right) \quad (7)$$

由空间上下文模型可得时空上下文模型 H_t^{stc} 如下所示:

$$\begin{cases} H_{t+1}^{\text{stc}}=(1-\rho)H_t^{\text{stc}}+\rho h_t^{\text{stc}} & t>1 \\ H_{t+1}^{\text{stc}}=H_t^{\text{stc}} & t=1 \end{cases} \quad (8)$$

由 $t+1$ 帧的时空上下文模型可以计算得到 $t+1$ 帧的置信图,从而得到目标位置,实现跟踪。

2 文中算法

文中提出了一种长时目标跟踪算法,在常规跟踪状态,文中使用自适应更新的时空上下文算法(STC)跟踪目标,针对长时跟踪目标可能丢失及目标出视场的应用需求,文中提出使归一化互相关匹配(NCC)全幅检索并定位目标。

2.1 自适应更新的时空上下文模型

文中的时空上下文算法需要计算两个时空上下文模型,分别是评估时空上下文模型和定位时空上下文模型,然后用这两个时空上下文模型分别计算评估置信图和定位置信图。由于置信图反映的是目标出现在图中各个像素点的概率,文中使用评估置信图的最大响应值来评价跟踪的可靠性,定位置信图的最大响应值用以计算目标位置。第一帧时,取定位时空上下文模型的计算结果作为评估时空上下文模型的计算结果,处理第一帧以外的图像帧时,当评估置信图最大响应值大于阈值 T_1 时更新定位时空上下文模型,更新公式如公式(8)所示,当评估置信图最大响应值大于阈值 $T_2(T_2 > T_1)$ 时,更新评估时空上下文模型。为了更好地体现场景的变化,由于第一帧图像计算得到的评估置信图的最大响应值 T 可以反映目标物的信息,因此,本算法将阈值 T_1 设置为 $n_1 T (0 < n_1 < 1)$,将阈值 T_2 设置为 $n_2 T (0 < n_2 < 1)$ 。

2.2 目标检测环节

文中目标检测环节采用归一化互相关匹配算法(NCC)^[12]来检测目标。这是一种广泛应用于机器视觉中的模板定位方法。

NCC 方法通过计算模板与输入图像的相关系数,来评价模板与图像的相似程度,相关系数越大说明图像与模板越相似,计算方法如公式(9)所示。

对全幅图像求相关系数可以得到相关系数矩阵,矩阵中最大值为最大相关系数。本算法中,当最大相关系数大于阈值 T_4 时说明检测成功,取相对应位置为目标位置,然后使用连续跟踪环节继续跟踪,否则对下一帧图像继续进行目标检测,直至最大相关系数大于阈值 T_4 。

$$\rho(x, y) = \frac{\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n (S_{x,y}(i, j) - \bar{S}_{x,y})(g(i, j) - \bar{g})}{\sqrt{\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n (S_{x,y}(i, j) - \bar{S}_{x,y})^2}}$$

$$\frac{1}{\sqrt{\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n (g(i, j) - \bar{g})^2}} \quad (9)$$

该式表示的是输入图像中以坐标 (x, y) 为左上角顶点的 $m \times n$ 区域内的待匹配图像与模板的相似度。其中, m 和 n 分别为模板的高和宽, $S_{x,y}(i, j)$ 表示输入图像中以 (x, y) 为坐标原点, 相对坐标 (i, j) 的像素点的灰度值, $g(i, j)$ 表示模板图像中坐标 (i, j) 的像素点的灰度值, $\bar{S}_{x,y}$ 表示 $S_{x,y}(i, j)$ 的均值, \bar{g} 表示模板图像的灰度均值。

2.3 算法框架

文中算法连续跟踪环节是自适应更新的时空上下文算法,在没有出现快速移动、目标遮挡、目标出视场等特殊场景时,连续跟踪环节起主要作用。然而在长时跟踪场景下,很可能出现特殊场景,如快速移动、目标出视场、目标遮挡等,这时连续跟踪环节便会跟踪失败,从而丢失目标。在跟踪失败特别是目标出视场后,目标可能出现在图像的任意位置,为了重新锁定目标,必须对全幅图像进行处理,采用目标检测的方法重新锁定目标。如果全幅图像分辨率较高,处理过程将面临着巨大的运算量,为了满足算力限制场景下的实时处理需求,文中提出使用模板匹配来完成重新锁定目标的要求。

系统开始跟踪时,首先启动自适应更新的时空上下文算法,并将第一帧计算得到的时空上下文模型保存为评估时空上下文模型,同时保存第一帧图像中的目标模板,文中使用连续跟踪环节处理每一帧图像时都需要计算评估置信图和定位置信图,时空上下文算法计算得到评估置信图的最大响应值小于阈值 T_3 时,说明此时跟踪结果不可靠,目标丢失,此时启动目标检测环节,使用归一化互相关匹配算法,检测目标。由于第一帧图像计算得到的评估置信图的最大响应值 T 可以反映目标物的信息,因此,文中算法将阈值 T_3 设置为 $n_3 T (0 < n_3 < 1)$ 。

设置检测成功的阈值指标为 $T_4(T_4$ 为常数,是根据实验场景调节的经验值),当目标不在当前图像中时,计算得到的最大相关系数会小于阈值 T_4 ,此时读入下一帧图像,并直接进入目标检测环节,直接检测目标,直至计算得到的最大相关系数大于 T_4 。然后取最大相关系数所在位置为目标位置,继续使用连续跟踪环节进行跟踪,在评估置信图最大响应值小于

T_3 时才再次启动目标检测环节。

3 实验验证

3.1 实验平台及评价指标

文中实验使用 MATLAB R2016a 进行, 处理器为 Inter(R) Core(TM) i5-3320M CPU, 内存为 8 GB。使用精度 (Precision) 和成功率 (Success) 评价跟踪情况, 精度表示以欧式距离衡量的中心点与实际中心的偏离程度。成功率表示预测目标与实际目标的重合程度。

3.2 实验过程及结果

实验使用 OTB2013 数据集进行测试, 该测试数据集可以反映文中算法在不同场景下跟踪的综合性

能。由于文中算法是针对长时跟踪场景的跟踪算法, 为此, 测试实验重点关注了在特殊跟踪场景, 特别是目标出视场情况下的跟踪效果。当目标被遮挡时可以模拟目标出视场的情况, Jogging 序列是一个典型的能够反映目标遮挡的序列, 使用 CT、STC、CSK 算法和文中算法进行跟踪, 只有文中算法可以在遮挡发生后继续跟踪, 跟踪效果如图 1(Jogging) 所示。

文中算法对快速移动等场景也有很好的鲁棒性。Deer 序列是一个典型的目标快速移动的序列, 使用文中算法, 跟踪精度可以达到 0.789。当使用 STC 算法进行跟踪时, 在第 4 帧跟踪失败, 文中算法同样跟踪失败, 但在第 11 帧时, 文中算法重新检测到目标继续跟踪, 跟踪情况如图 1(Deer) 所示。

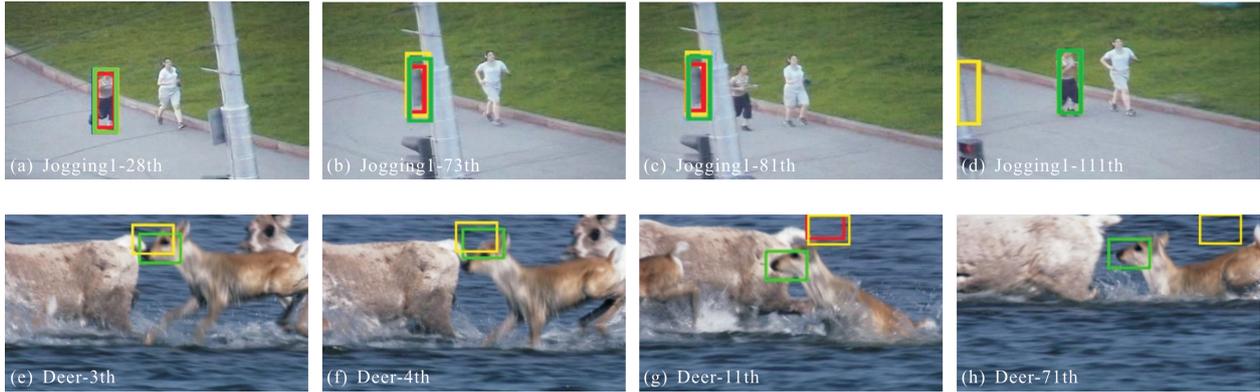


图 1 跟踪效果图, 绿色框、红色框、蓝色框、黄色框分别为文中算法、STC 算法、CSK 算法、CT 算法的跟踪效果

Fig.1 Figure of tracking effect, green box, red box, blue box and yellow box are our algorithm, STC algorithm, CSK algorithm and CT algorithm, respectively

文中算法不仅具备跟踪失败后重新检测目标的性能, 也可以在连续跟踪环节跟踪不可靠时辅助衡量跟踪的正确性, 因此也可以应用于短时跟踪场合, 使用 OTB2013^[7] 测试框架进行测试, 得到的测试结果如图 2 所示, 从图中可以看出, 文中算法的精度较 STC 算法提升了 4.95%, 效果有一定程度的改善。

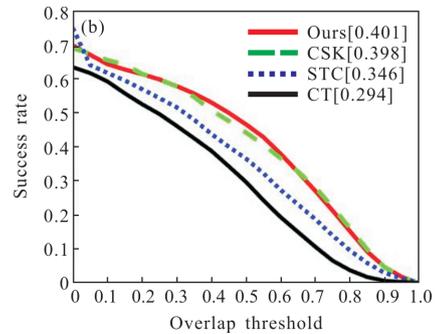
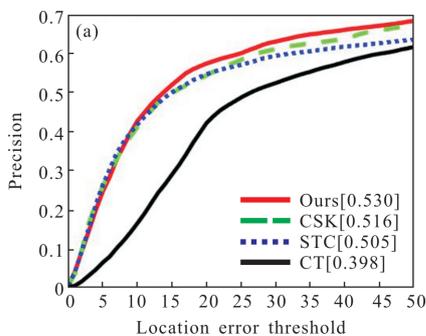


图 2 精度与成功率测试结果

Fig.2 Test results of accuracy and success rate



3.3 运算速度分析

Yi Wu^[13] 等人曾经对比过几种主流目标跟踪算法的处理帧率, 其中 TLD 算法为 28.1 帧, CT 算法为 64.4 帧, CSK 算法为 362 帧。为了便于比较, 文中选

取几个典型的视频序列,在同一实验平台,分别使用 CSK 算法和文中的算法进行实验,比较两种算法的处理帧率,结果如表 1 所示。

表 1 运算帧率对比表(单位:fps)

Tab.1 Operational frame rate contrast table
(Unit: fps)

	CarDark	Girl	Deer	Jogging
CSK	457	314	121	174
Ours	504	420	100	227

从上表可以看出,文中算法是一种运算时间复杂度较低的目标跟踪算法,具备应用于算力受到限制的应用平台的基础。

4 算力限制场景的应用

文中算法结构简单,时空上下文算法与归一化互相关匹配都是常用的计算量较低的算法。为了检

验文中的实际应用效果,文中将算法移植到了以 C64x+ DSP 为核心的嵌入式平台,C64x+ DSP 是主频 600 MHz 的 DSP 芯片。经实验验证,图像读取等基础功能环节消耗 6 M 时钟数,连续跟踪环节共消耗 5.3 M 时钟数,目标检测环节消耗 4.3M 时钟数,连续跟踪时帧率为 53 帧,目标检测重定位时帧率为 38 帧。

在实物实验中,文中通过抖动实验系统来营造目标丢失,目标出视场的场景,实验结果如图 3(Airplane)序列所示。图中绿色框为跟踪框,为了更好的显示实验效果,文中使用红色圆圈手动标注了目标位置。图 3(a)Airplane 是初始跟踪场景,图 3(b)Airplane 是抖动后目标丢失的场景,图 3(c)Airplane 是目标出视场的场景,图 3(d)Airplane 表示当目标返回视场后跟踪框重新定位到了目标。从该序列跟踪结果来看,本算法具备跟踪失败后在全幅图像中重新检索定位目标的能力。而图 3 中 Car 序列体现了文中算法跟踪稳定的特性。

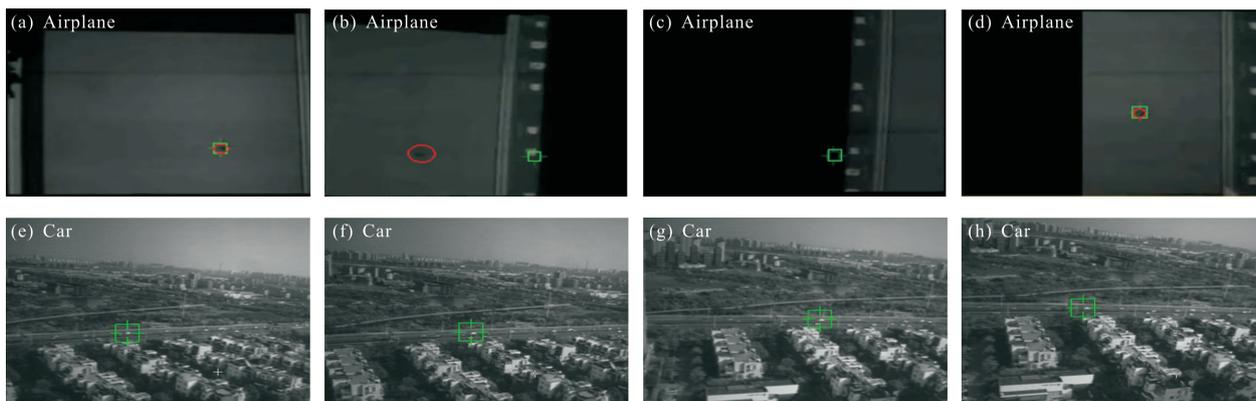


图 3 嵌入式平台跟踪效果图

Fig.3 Tracking Effect Diagram of Embedded Platform

5 结论

文中提出了一种长时目标跟踪算法,算法包括以自适应更新的时空上下文算法为基础连续跟踪环节和以归一化互相关模板匹配为核心的目标检测环节。文中算法可以在目标出视场后检测到出视场的状态,并在目标进视场后重新锁定目标并继续跟踪,这样的特性适用于长时跟踪场景。此外,目标检测环节在连续跟踪环节跟踪结果不可靠时可以再次定位目标,提升了短时跟踪的鲁棒性,经 OTB2013 数据集测试,文中算法的跟踪精度较时空上下文算

法提升了 4.95%,而且文中算法结构简单,运算速度较快,在 i5-3320M CPU 上可达到 500 帧,可以应用于算力限制的平台,如以 C64x+ DSP 为核心的嵌入式系统。在该嵌入式系统下进行实验,经过测算,在算法启动目标检测环节时,图像处理帧率可达 38 帧,满足实时性需求。

参考文献:

- [1] Gao Wen, Zhu Ming, He Baigen, et al. Overview of target tracking technology[J]. *Chinese Optics*, 2013, 7(3): 365-375. (in Chinese)
高文,朱明,贺柏根,等.目标跟踪技术综述[J].中国光学,

- 2013, 7(3): 365–375.
- [2] Guo Jingming, He Xin, Yang Jie, et al. Infrared target tracking based on template adaptive Mean Shift [J]. *Infrared and Laser Engineering*, 2014, 43(4): 1087–1093. (in Chinese) 郭敬明, 何昕, 杨杰, 等. 模板自适应的 Mean Shift 红外目标跟踪[J]. 红外与激光工程, 2014, 43(4): 1087–1093.
- [3] Luo Haibo, Xu Linyun, Hui Bin, et al. Status and prospect of target tracking based on deep learning [J]. *Infrared and Laser Engineering*, 2017, 46(5): 0502002. (in Chinese) 罗海波, 许凌云, 惠斌, 等. 基于深度学习的目标跟踪方法研究现状与展望[J]. 红外与激光工程, 2017, 46(5): 0502002.
- [4] Yang Dedong, Cai Yuzhu, Mao Ning, et al. Long-term object tracking based on kernelized correlation filters [J]. *Optics and Precision Engineering*, 2016, 24 (8): 2037 – 2049. (in Chinese) 杨德东, 蔡玉柱, 毛宁, 等. 采用核相关滤波器的长期目标跟踪[J]. 光学 精密工程, 2016, 24(8): 2037–2049.
- [5] Yan Hui, Xu Tingfa, Wu Qingqing, et al. Multi-object tracking based on multi-feature joint matching [J]. *Chinese Optics*, 2013, 6(2):163–170. (in Chinese) 闫辉, 许廷发, 吴青青, 等. 多特征融合匹配的多目标跟踪[J]. 中国光学, 2013, 6(2): 163–170.
- [6] Kalal Z, Mikolajczyk K, Matas J. Tracking-learning-detection [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, 34(7): 1409–1422.
- [7] Zhu Z, Wang Q, Li B, et al. Distractor-aware siamese networks for visual object tracking [C]//European Conference on Computer Vision. Springer, Cham, 2018: 103–119.
- [8] Li B, Wu W, Wang Q, et al. SiamRPN++: Evolution of siamese visual tracking with very deep networks [J]. arXiv preprint arXiv, 2018: 1812.11703.
- [9] Danelljan M, Bhat G, Khan F S, et al. Atom: Accurate tracking by overlap maximization [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019: 4660–4669.
- [10] Xiong Jingying, Dai Ming. Design of tracker for mobile smart devices [J]. *Optics and Precision Engineering*, 2017, 25(12): 3152–3159. (in Chinese) 熊晶莹, 戴明. 适应移动智能设备的目标跟踪器 [J]. 光学精密工程, 2017, 25(12): 3152–3159.
- [11] Zhang K, Zhang L, Liu Q, et al. Fast visual tracking via dense spatio-temporal context learning [C]//European Conference on Computer Vision. Springer, Cham, 2014: 127–141.
- [12] Rao Y R, Prathapani N, Nagabhooshanam E. Application of normalized cross correlation to image registration [J]. *International Journal of Research in Engineering and Technology*, 2014, 3(5): 12–16.
- [13] Wu Y, Lim J, Yang M H. Online object tracking: A benchmark [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013: 2411–2418.