

基于深度学习的嵌入式目标追踪研究进展^①

董 博^{1,2}, 陈 华¹, 龚 勇¹

¹(中国电子科技南湖研究院, 嘉兴 314001)

²(中国科学技术大学 先进技术研究院, 合肥 230031)

通信作者: 龚 勇, E-mail: gongyong@cnaeit.com



摘 要: 作为计算机视觉领域的基本问题之一, 目标追踪具有广泛的应用场景. 随着硬件算力和深度学习方法的进步, 常规的深度学习目标追踪方法精度越来越高, 但其模型参数量庞大, 计算资源和能耗需求高. 近年来, 随着无人机和智能物联网应用的蓬勃发展, 如何在存储空间和算力有限、低功耗需求的嵌入式硬件环境中进行实时目标追踪, 成为当前研究的热点. 本文对面向嵌入式应用的目标追踪方法进行了分析综述, 包括相关滤波结合深度学习的目标追踪方法、基于轻量神经网络的目标跟踪方法, 并总结了深度学习模型部署流程和无人机等领域的嵌入式目标追踪典型应用实例, 最后对未来研究重点进行了展望.

关键词: 目标追踪; 嵌入式系统; 轻量神经网络; 相关滤波; 深度学习; 边缘 AI

引用格式: 董博, 陈华, 龚勇. 基于深度学习的嵌入式目标追踪研究进展. 计算机系统应用, 2023, 32(1): 12-28. <http://www.c-s-a.org.cn/1003-3254/8914.html>

Research Progress of Object Tracking by Deep Learning in Embedded System

DONG Bo^{1,2}, CHEN Hua¹, GONG Yong¹

¹(China Nanhu Academy of Electronics and Information Technology, Jiaxing 314001, China)

²(Institute of Advanced Technology, University of Science and Technology of China, Hefei 230031, China)

Abstract: Object tracking, a basic problem in computer vision, has a wide range of application scenarios. Due to the advance in the computational capacity of hardware and deep learning methods, conventional deep learning methods for object tracking have higher precision, but they face the problems of massive model parameters and high demand for computational resources and power consumption. In recent years, with the booming development of unmanned aerial vehicle (UAV) and Internet of Things (IoT) applications, a great deal of research focuses on how to achieve real-time tracking in embedded hardware environment with limited storage space and computational capacity and low power consumption. Firstly, object tracking algorithms in the embedded environment, including the ones combining correlation filters with deep learning and those based on lightweight neural networks, are analyzed and discussed. Secondly, deployment procedures of deep learning models and classical embedded object tracking applications, such as those in UAVs, are summarized. Finally, future research directions are given.

Key words: object tracking; embedded system; lightweight neural network; correlation filter; deep learning; edge AI

近年来深度神经网络迅速发展, 人们将深度神经网络应用于实际问题中发现能有效地解决问题, 因而深度学习方法的应用场景也越来越广泛, 如人脸识别、行人

检测、自动驾驶、智能机器人等. 计算机视觉作为主要应用方向之一, 而目标追踪又是视觉领域一个备受关注和极具挑战性的基本方向之一, 因而在近几年目标追踪

① 基金项目: 浙江省“领雁”研发攻关计划 (2022C01098)

收稿时间: 2022-04-24; 修改时间: 2022-05-22; 采用时间: 2022-07-20; csa 在线出版时间: 2022-09-23

CNKI 网络首发时间: 2022-11-15

的研究并不少。目标追踪是根据视频中的图像序列计算跟踪目标轨迹的任务,目标常在跟踪过程中大小改变,发生形变、位移变化和旋转变化及超出视野范围等,且跟踪任务常受到环境的干扰,比如遮挡、光照变化等,另外也经常受到采集图像装置的影响,比如设备抖动和快速移动、低清晰度等。因此目标追踪要在计算目标轨迹的过程中解决上述问题而变得困难。

一个完整的目标追踪任务主要包含目标描述、特征选择、目标检测和目标追踪4个子问题。为了解决这4个子问题,传统方法和深度学习方法为此做了大量努力。传统方法往往根据经验进行目标描述,如点、基础几何形状、骨架、轮廓剪影等^[1],然后手工设计和选择特征,最后配上与目标描述相符的检测算法和跟踪算法来完成任务,速度往往在数十帧每秒。由于传统方法在目标表示和特征提取上过于简单,不注重利用背景信息和时空信息,且往往做出的很多假设与真实世界不相符,所以解决追踪问题中所面临困难的能力不强,相关方法已逐步淘汰。当然,在一些场景下仍可使用一些传统方法,比如:卡尔曼滤波、粒子滤波、均值漂移、光流法。而神经网络的目标建模能力强和鲁棒性较好,能自动提取特征,且能较为充分地利用背景信息和时空信息,且结合相关滤波或轻量化方法往往能拥有较好的实时性,故现在深度学习方法被广泛地应用于目标追踪任务。

随着近几年物联网应用、移动互联网应用和无人机应用的兴起,从计算机视觉会议的论文来看,如何在计算资源、存储资源和能耗受限的边缘设备上部署深度神经网络模型这一问题被广泛关注,因为边缘设备往往要求整个模型只能有几兆大小,运算量只能有几百兆浮点运算量。神经网络的参数和结构存在冗余,所以研究者们提出了大量的模型压缩方法^[2],主要分为剪枝、量化、低秩分解、知识蒸馏4大类。其中剪枝和量化方法的研究相对另两者较多,应用也相对较多。这些方法往往针对训练完的模型进行压缩,基本都需要引入微调操作,这就比较耗时。另外一种思路就是手工设计紧凑或轻量的神经网络,然后直接用作提取目标特征的模型。相比前面所述的模型压缩方法,这种方法就比较节省训练时间。虽然设计紧凑的网络往往比较困难,但有一些优秀的轻量神经网络。相比前面的手工设计轻量神经网络,神经网络架构搜索的自动化轻量神经网络设计则不需要人为设计网络结构,并且往往

将以往轻量神经网络中使用的块结构设计定义为搜索空间,使用NAS或NAS的优化算法,如one-shot NAS,来搜索轻量神经网络。

虽然深度学习目标追踪方法纷繁复杂,模型压缩方法与轻量神经网络也数量众多,但在嵌入式环境下结合两者进行目标追踪却是当前计算机视觉的应用研究热点。本文将针对这个研究方向进行分析总结,希望给从事相关工作的研究者们提供帮助和指导,同时也给从业者们一个未来方向。本文结构如下:第1节主要介绍结合相关滤波与深度学习的目标追踪方法;第2节主要介绍基于轻量神经网络的目标追踪方法;第3节总结网络模型部署流程和嵌入式目标追踪应用实例;第4节总结全文并展望未来研究方向。

1 结合相关滤波与深度学习的追踪方法

2010年Bolme等人^[3]提出MOSSE方法首次将相关滤波方法用于目标追踪任务,根据卷积定理利用快速傅里叶变换(FFT)将卷积运算变成点乘运算从而大大提高了运算速度,此外通过最小化平方和误差函数得到最优的滤波器模板。此后,2012年Henriques等人^[4]提出的CSK方法对MOSSE方法做了改进,其中利用循环移位的采样方法进一步减少计算量,同时求解滤波器模板的过程中引入正则项并利用核技巧求解,减少相关滤波结果发生过拟合的可能性;2014年Danelljan等人^[5]在CSK方法基础上扩展特征到多通道颜色特征;2015年Henriques等人^[6]在CSK方法基础上扩展特征到多通道特征。多通道特征的岭回归相关滤波器目标函数如式(1)所示,其中“ \otimes ”表示卷积运算, W_i 表示求解的滤波器目标, X_i 表示第*i*个通道的特征图, Y_i 是理想的高斯响应, $l \geq 0$ 是用来防止过拟合的正则惩罚项系数。至此,相关滤波追踪器的大体框架基本建立,如图1所示。首先余弦窗口用来平滑窗口边界效应,根据卷积定理FFT变换能将空域的卷积运算变为时域的按位乘法运算,“ \odot ”表示按位乘法运算;然后再完成相关性计算后,使用IFFT得到空间置信图,最亮的地方即为目标的新位置;最后使用新位置提取的特征配合期望输出来更新相关滤波器。

$$\min \sum_i \|W_i \otimes X_i - Y_i\|^2 + l \sum_i \|W_i\|_2^2 \quad (1)$$

相关滤波方法由于在频域进行计算所以运算速度快,但传统的基于相关滤波的目标跟踪方法中手工设计

的特征具有较差的目标建模能力,导致追踪器在现实场景下精度欠佳,故后续大部分基于相关滤波的追踪器基本在特征形式、相关滤波的目标函数形式、训练形式做出改变.采用卷积神经网络(CNN)自动提取较好的目标特征,刚好能弥补以往相关滤波目标追踪方法的不足,故在实际应用中经常把两者结合起来使用.而为了进一步提高追踪速度满足实时应用,近年来兴起的孪生神经网络(SNN)是一个速度优秀的模型,故经常也将相关滤波方法与SNN进行组合来构建追踪器,这是一种判别式的追踪模型,如图2所示.首先,滑动窗口在搜索图像上选取与模板相同大小的图像块;然后,选定图像

块和模板经过CNN提取特征;最后,提取的特征进行互相关运算得到分数图.其中,“*”表示互相关运算,并且两个CNN共享参数.因为结合相关滤波与深度学习的目标追踪方法在嵌入式中地位重要,故本节从完善相关滤波追踪器框架,或者用卷积层对等表示相关滤波器构成端到端的追踪网络,或者相关滤波追踪网络作为追踪器的子网络这3方面介绍结合相关滤波与深度学习的目标追踪方法,本节方法分类表如表1所示.从表1可得,相关滤波追踪算法过往通过结合CNN和SNN来提升算法的精度和速度,而现在主要使用其追踪框架或追踪框架子部件构建灵活的追踪模型.

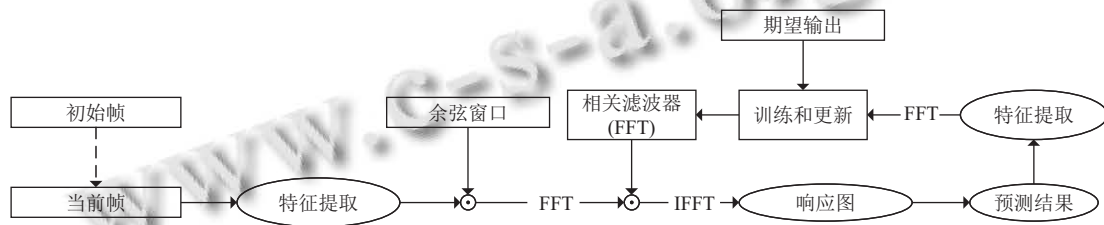


图1 相关滤波追踪器通用框架

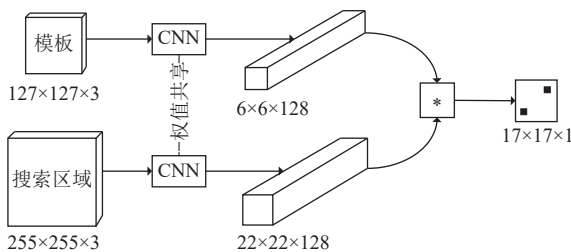


图2 孪生神经网络追踪器通用框架

1.1 完善相关滤波追踪器框架

在相关滤波追踪算法中采用手工设计特征对目标的建模能力有限,在跟踪过程中经常由于上一帧的噪声干扰发生目标漂移现象,故需要蕴含信息更丰富的目标表示模型或目标函数,主要通过改变特征形式、优化追踪结果或调整相关滤波目标函数实现.

表1 结合相关滤波与深度学习的追踪方法分类表

大分类	小分类	追踪方法	关注问题
完善相关滤波追踪器框架	改变特征形式	HCFT、DeepSRDCF、TRACA、CFCF、IBCCF、KAOT、MKCT、ASRCF、ECO、GFS-DCF等	模型的健壮性
	优化追踪结果	HDT、WECO、UPDT、MCCT、MLCFT	模型的健壮性
	调整相关滤波目标函数	DeepSTRCF、DeepCSRDCF、CCOT、BEVT等	模型的健壮性
	带粒子滤波的相关滤波器	DeepHPFT、MCPF、DCPF、DCPF2	模型的速度、健壮性
端到端的相关滤波追踪网络	结合SNN与相关滤波层的网络	CFNet、DCFNet、DSiamM、i-Siam、UDT等	模型的速度、健壮性
	自定义的相关滤波网络	CREST、UCT、adaDDCF、D3S、ROAM	模型的灵活性
相关滤波追踪器为追踪器子网络	—	PTAV、ACFN、CFSRL、CF-CNN、CF-SiamFC、CODA	模型的灵活性

1.1.1 改变特征形式

改变特征形式主要有3种方法,分别是改变特征设计方式、增加特征数目、引入特征选择方法.

(1) 改变特征设计方式

HCFT方法^[7]融合高层卷积特征和低层卷积特征,

主要通过VGGNet和AlexNet的不同卷积层上训练相关滤波器探讨特征对追踪器性能的影响,重点解决外形变化和滤波器易发生目标漂移的问题,但在处理长期遮挡和区分暗前景与亮背景问题上存在显著不足. DeepSRDCF方法^[8]探讨卷积特征与手工设计特

征在标准 DCF 框架和 SRDCF 方法中对追踪结果的影响,发现 VGG2048 的第 1 层卷积特征更利于追踪. TRACA 方法^[9]和 AEPCF 方法^[10]则引入自动编码器压缩特征,前者通过无监督的方式粗略聚类训练样本,每一类训练一个自动编码器,并通过上下文关注网络进行选择,以此来改善速度和精度;后者在源域和目标域分别使用自动编码器缓解样本的不足,并采用长短期更新策略提升速度. CFCF 方法^[11]与 DSNet 方法^[12]提出相关滤波器的特征学习方法,前者主要改善相关滤波追踪器中以分类为目的的特征提取网络,将学习深度全卷积特征问题形式化定义,并通过新的反向传播算法训练该网络,以此实现特征自主设计;后者提出深浅学习网络 (DSNet) 来学习多层同分辨率的卷积特征,以此实现高效的在线追踪.

(2) 增加特征数目

DTO 方法^[13]在 HCFT 中加入类别信息来提升精度. HCFTs 方法^[14]和 IBCCF 方法^[15]主要解决 HCFT 的缺陷,前者主要构建一个相关滤波器长期记忆目标外形变化,主要解决尺寸估计和再检测问题,一定程度上弥补 HCFT 的缺陷,但在应对长期遮挡、剧烈尺寸变化和面内旋转仍有不足;后者重点在于额外使用估计目标边界和中心的相关滤波器解决目标纵横比变化问题,比起 HCFT 方法能应对旋转问题. STSGS 方法^[16]综合考虑目标在帧内的外形相关性和帧间的运动显著性来进行精确定位,但追踪过程中由于目标外形变化和复杂场景而精确地分割前后景能力存在不足. IMM-DFT 方法^[17]则使用交互多样模型 (IMM) 融合在不同卷积层的相关滤波器和在线学习的运动模型得到混合系统来改善处理目标外形变化和背景干扰能力,但在目标再追踪能力上有所缺陷. KAOT 方法^[18]在周期性选择的关键帧上学习关键相关滤波器来表达上下文信息,以此减轻滤波器腐败和边界效应,提升速度,但在关键帧的选择方法和处理目标再追踪及旋转问题上存在不足. MKCT 方法^[19]使用卷积特征和手工设计的特征训练多个权重滤波器和核相关滤波器来实现相关滤波器的动态权重,并采用梯度数值相似导数 (GMSD) 来进行上下文分析来进一步使用上下文信息,对相关滤波结果会采用分辨率增强运算来加强响应并抑制背景信息,以此加强处理外形变化能力,但在处理快速移动、再追踪和全遮挡问题上存在不足. ASRCF 方法^[20]使用在多尺度的浅层特征上训练的额外相关滤波器来

快速估计目标尺寸增强模型的健壮性,且 KCF、SRDCF、BACF 为 ASRCF 的特例.

(3) 引入特征选择方法

ECO 方法^[21]在 CCOT^[22]基础上引入因式分解的卷积算子减少模型参数数量,使用训练样本空间的紧凑生成模型来减少样本数量,以此来降低 CCOT 方法中模型复杂度和训练空间大小,同时优化模型更新策略提高速度;ORHF 方法^[23]和 TRACA 方法则引入特征选择网络. CPT 方法^[24]则是在 ECO 上使用通道剪枝过的 VGG16 提取特征,并且使用平均特征能量比方法自适应地进行特征通道剪枝,改善速度. GFS-DCF 方法^[25]中考虑组特征选择方式来减少特征的空间和通道维度来实现特征压缩,以此提高速度和精度.

1.1.2 优化追踪结果

HDT 方法^[26]将多个弱追踪器对冲成一个强追踪器. WECO 方法^[27]和 UPDT 方法^[28]改进 ECO 的卷积特征融合方式,前者对不同卷积特征所得的置信分数进行加权聚合;后者认为浅层特征能增强模型鲁棒性,而深层特征能改善模型的定位精度,提出新的策略来融合深层特征和浅层特征的预测结果. MCCT 方法^[29]采用鲁棒性评估策略对不同特征训练的 DCF 追踪器做出选择,并且考虑多个结果之间差异性来更新追踪器. MLCFT 方法^[30]用基于相对熵的方法混合多个相关滤波器的响应地图来得到可能位置,给再检测模块进行最佳位置选择,保证速度的同时还可处理快速移动、光照变化和遮挡问题.

1.1.3 调整相关滤波目标函数

调整相关滤波目标函数主要有两种形式,一种是改变目标函数的权重矩阵,一种是改变目标函数的正则项,但并不意味着改变目标函数的权重矩阵不会改变正则项,只表示方法的主要倾向性.

(1) 改变目标函数的权重矩阵

CCOT 方法和 DRT 方法^[31]提出新的目标函数,前者将单分辨率特征推广到多分辨率特征,并实现亚像素级的定位,但计算复杂度太高;后者为了增加模型的可靠性在目标函数中将滤波器权重矩阵表示为基滤波器权重和区域可靠性权重矩阵的点乘积,但处理复杂背景、目标快速形变和旋转上存在不足. BEVT 方法^[32]在目标函数中采用裁剪矩阵来提取更多的背景信息,同时在正则项中加入背景信息惩罚矩阵来抑制背景噪声,主要用来缓解相关滤波器的边界效应,同时采用多

帧的方法增强建模能力,但惩罚项的设计由于计算方法引入了干扰和应对背景杂乱问题存在不足。SASR方法^[33]在目标函数中使用新的采样矩阵提取背景信息和空间正则项抑制无用的背景信息,在应对快速运动、目标超出视野问题和背景块选取策略上存在不足。DeepFWDCF方法^[34]在目标函数中使用权值矩阵区别对待靠近和远离目标的区域解决目标漂移和不同形状问题。RPCF方法^[35]则在目标函数中使用表示神经网络平均池化层的矩阵。DeepCSRDCF方法^[36]在目标函数中使用空间可靠性矩阵来打破搜索区域为矩形的限制;而DeepTACF方法^[37]在目标函数中引入目标附近背景区域的抑制权重矩阵来优化矩形预测框。

(2) 改变目标函数的正则项

DeepSTRCF方法^[38]在目标函数中整合时间和空间正则项,比SRDCF方法处理外形变化能力更强,并且处理边界效应问题不会影响速度。DeepCSRDCF方法在目标函数中加入空间和特征通道可靠性正则项,空间可靠性的目的在于一定程度上使搜索区域任意大小,而通道可靠性的目的在于在定位中加权每个特征通道滤波结果,在应对变形、旋转、短暂的全遮挡问题上比SRDCF方法更好。GFS-DCF方法在目标函数中补充组空间正则项、组通道正则项和时间平滑假设正则项。WAEF方法^[39]则是对ECO方法中提出了增强算子处理输入图像减轻干扰,同时在目标函数中用吉洪诺夫正则化来表达目标的帧间相关性,相比ECO方法在应对遮挡、光照变化、尺寸变化上有了改善。

1.1.4 带粒子滤波的相关滤波器

DeepHPFT方法^[40]中使用粒子滤波平移层和粒子滤波尺寸层的分层粒子滤波结构来构建相关滤波器,用粒子滤波尺寸层和手工设计特征来估计目标大小。MCPF方法^[41]使用特征之间的依赖来学习相关滤波器,并且使用粒子滤波来处理目标的尺寸变化问题,相比CF2、HDT处理目标尺寸变化能力更强,并且速度更快。DCPF方法^[42]在HCFT基础上加入粒子滤波器估计的位置信息来改良处理变形、光照、旋转、短暂目标消失的能力;DCPF2方法^[43]改进DCPF方法,使用粒子滤波器解决目标边界框大小固定的问题,并且采用自适应的相关滤波器来产生多个目标模型。

1.2 端到端的相关滤波追踪网络

由于特征提取网络往往以分类为目的而进行训练,

这样的通用特征表示模型不适合特定的追踪任务,而分类模型也比较复杂,对实时应用不够友好;此外,手动来添加额外的辅助信息或者人为地修改相关滤波的目标函数来完成追踪任务增加了计算复杂度,对实时应用不够友好。为了解决上述的域不稳定性问题,考虑到相关滤波的目标函数本质上是在频域上求均方差,则可以直觉上用卷积层去对等表示相关滤波器,这样特征提取网络和相关滤波器则能构成端到端的网络一同训练。近年来端到端的相关滤波追踪网络有两个趋势,一个是结合相关滤波层与SNN的网络;另一个是结合CNN和相关滤波层的自定义网络。

1.2.1 结合SNN与相关滤波层的网络

CFNet^[44]将相关滤波器解释为可微分的卷积层,之后与SNN的一个分支结合起来构成端到端的网络;DCFNet^[45]将判别相关滤波器解释为特殊的相关滤波层来替换SiamFC的相关函数。MTHCF^[46]在CFNet基础上使用多任务学习的方法分别对高卷积层和低卷积层提取的特征构建相关滤波层来获得更加丰富的目标表示能力;IMLCF^[47]改变DCFNet的SNN分支,一个分支用于特征提取,另一个分支用于融合特征,再作为输入送入判别相关滤波层。DSiamM^[48]使用相关滤波层处理目标的外形变化和抑制背景干扰;i-Siam^[49]对提取的特征进行负信号抑制和背景抑制来提升追踪精度,使用多样多模板方法解决因目标外形变化造成的模板漂移问题。FICFNet^[50]在CFNet基础上增加特征融合模块和通道注意力机制对融合后的不同通道特征赋予权重。FlowTrack^[51]提出端到端的流相关追踪网络来整合光流估计、特征提取和聚合、相关滤波层,使用新的时空注意力机制来聚合特征;RASNet^[52]使用多种注意力机制对模板特征进行处理,采用加权互相关运算来产生追踪结果。UDT^[53]将非监督学习用于孪生相关滤波网络,一个分支用于向前追踪,一个分支用于向后验证追踪结果,两个分支的目标轨迹应一致,同时为了减轻非监督学习带来的影响,向后验证中使用多帧验证方法和成本敏感损失。

1.2.2 自定义的相关滤波网络

CREST^[54]将特征提取、相关滤波层、模型更新整合在一起构成端到端的网络,使用残差学习来解决目标外形变化造成的模型退化问题;UCT^[55]将特征提取和追踪过程都视为卷积操作,依据峰对噪音比来更新模型,使用相关滤波器处理目标尺寸变化;adaDDCF^[56]

在提取特征后接 Fisher 判别分析层来提升正负样本差异,再配合深度相关滤波层来适应目标外形变化;D3S^[57]中使用几何不变模型、几何受限的欧几里得模型和细分处理方法得到分割图来完成追踪任务和分割任务;ROAM^[58]使用元学习方法训练循环神经优化器来训练由响应生成器和边界回归器组成的追踪器。

1.3 相关滤波追踪器为追踪网络子网络

这部分网络与前两类的区别主要在于使用完整的相关滤波框架,同时串联或并联别的完整网络来完成追踪任务。PTAV^[59]由追踪器和验证器组成,追踪器是每时每刻都在运行完整的相关滤波器框架,验证器是追踪结果不可靠时才会启动的孪生神经网络;ACFN^[60]参照神经形态学的注意力机制使用注意力机制网络对多个相关滤波器结果进行选择;CFSRL^[61]使用强化学习的方法训练策略网络来对历史帧产生的相关滤波器与当前帧的特征运算产生的响应地图做出选择;CF-CNN^[62]是3层特征模型,其中两层特征和相关滤波器运算,另外一层特征用卷积算子运算,综合三者得到追踪结果;CF-SiamFC^[63]解决 SiamFC 中追踪同类目标的会漂移的问题,使用 SiamFC 产生候选位置,再将候选图像块变为 HOG 特征输入 DSST 来得到追踪结果;CODA^[64]受到 DTO 思想的启发来解决域的不适定性问题,在 HCFT 基础上对提取的特征进一步学习,配合 SSD 的检测结果来矫正跟踪结果。

2 基于轻量神经网络的目标追踪算法

近年来物联网和移动互联网蓬勃发展,其中丰富的应用场景迫切需要工作者提出在小型物联网设备、

只含 CPU 物联网设备或多应用物联网设备上精度优秀、计算量更低和耗能更少的目标追踪方法。尽管第1节中结合相关滤波和深度学习的追踪方法能耗比较为优越,但在上述设备上目标追踪仍比较困难,例如 Raspberry Pi,主要原因在于第1节中跟踪算法的特征提取网络较为笨重。据目前的研究看来,现在主要有两种研究思路,一种是传统追踪算法^[1]或相关滤波追踪算法的特征表示使用传统的 HOG 特征与颜色特征,例如 CN 特征,而检测器使用轻量神经网络检测器增强检测性能;另一种是近两年出现的轻量孪生网络追踪器。本节将从基于轻量检测网络的追踪算法与轻量的孪生追踪网络两方面介绍基于轻量神经网络的目标追踪算法。

2.1 基于轻量检测网络的追踪算法

基于轻量检测网络的目标追踪结构图如图3所示,首先,经改造的轻量检测网络用第1帧检测结果来初始化追踪器;然后,追踪器执行追踪任务并根据追踪结果来更新自身。另外当严重错误发生时,可以调用轻量检测网络来重新初始化追踪器。图3中,检测器上方的虚线框表示可选的分割网络,追踪器上方的数据关联算法常见于多目标追踪任务。其主要思想是通过精度优越检测器和速度较快追踪器的联合使用来提高追踪算法的性能,检测器虽然是神经网络,具有较大的计算量,但并不是每进行一次追踪都需要运行检测器,而追踪算法的目标特征表示采用手工设计特征而极大减少了计算量,在一定程度上可以得到精度和速度平衡的追踪算法。本方法在小型物联网设备和只含 CPU 的物联网设备上目标追踪应用得较为广泛。

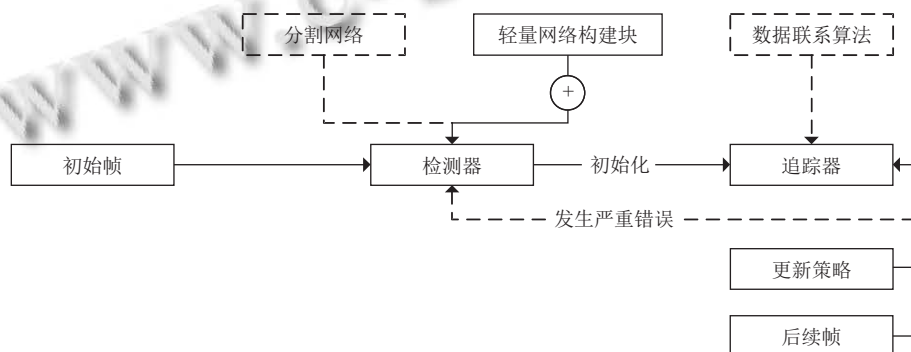


图3 基于轻量检测网络的目标追踪框架图

基于轻量检测网络的追踪算法具体可以分为结合轻量检测网络与传统追踪方法^[65-78]、结合轻量检测网

络与相关滤波方法^[75,76,78-80]、结合轻量检测网络与孪生追踪网络^[81-83]。本节将从轻量目标检测网络和本方

法常用追踪算法两方面介绍基于轻量检测网络的追踪算法。

2.1.1 轻量目标检测网络

目标检测问题不是本文核心内容,但又是基于轻量检测网络的追踪算法核心内容,故本节简要总结轻量目标检测网络的设计要点。

图3中的轻量检测网络通常在一阶段检测网络的基础上进行修改,如SSD、SSD-Lite、YOLO。借鉴轻量神经网络的优秀块结构设计调整骨干网和调整特征金字塔网络特征融合方式,例如YOLOv3的FPN、NanoDet的PAFPN;同时根据追踪类别数目对检测头分类类别数目进行修改,单目标追踪还可删除整个检测头的类别信息。轻量检测网络也常采用深度可分离卷积替换一些普通卷积层去降低模型参数量,如YOLOX-Nano、PP-YOLO Tiny。目前,研究者已经提出一些用NAS设计目标检测网络的研究,也提出一些anchor-free的动态标签赋予方法,如SimOTA、TAL。近年来常用的轻量检测网络如表2所示。

表2 常见的轻量目标检测网络

模型	图像尺寸	mAP (%)	参数量 (M)	计算量 (GFLOPs)	测试集
YOLOv4-Tiny	416	21.7*	6.06	6.69	COCO val [◇]
PPYOLO-Tiny	416	22.7*	4.20	—	COCO val [◇]
YOLOX-Tiny	416	32.8*	5.06	6.45	COCO val [◇]
YOLOX-Nano	416	25.3*	0.91	1.08	COCO val [◇]
YOLOv5n	640	28.4*	1.9	4.5	COCO val [◇]
YOLOv5n6	1280	34.0*	3.2	4.6	COCO val [◇]
MobileNetV1-SSDLite	300	22.2*	5.1	1.30	COCO test-dev [△]
MobileNetV2-SSDLite	300	22.1*	4.3	0.80	COCO test-dev [△]
NanoDet-m	320	20.6*	0.95	0.72	COCO val [◇]
NanoDet-Plus-m	416	30.4*	1.17	1.52	COCO val [◇]
EfficientDet-D0	512	34.6	3.9	≈2.5	COCO test-dev [◇]

注: *表示mAP 0.5:0.95,无*则不确定; △表示COCO 2015, ◇表示COCO 2017。

在骨干网的调整上,研究者经常使用SqueezeNet、MobileNet、ShuffleNet、ResNet等轻量神经网络作为轻量检测网络特征提取的骨干网,例如MobileNet-SSD、ShuffleNetV2-YOLOv5。YOLO从YOLOv4开始,骨干网Darknet也使用Focus、CSP或SPP来加强所检测目标的特征表达能力。

图3中检测器上方的虚线框表示特定的模型,往

往在特定应用的目标检测中是需要的。机器人应用和自动驾驶中常使用分割网络,如UNET、SegNet;头部跟踪应用中的头部检测网络,如DCNN。

根据特定的应用场景和部署环境,缩小检测模型存储尺寸和减少模型计算量具有在硬件平台上低传输功耗开销和加速推理的优势,所以在一些情况下也会对检测模型采取通道剪枝和量化的方法进行模型压缩。目前工业上常用INT8量化算法来压缩模型,INT8量化内容将在第3节进行介绍。

2.1.2 本方法常用追踪算法

图3中追踪器常用算法有Boosting、MEDIANFLOW、MIL、TLD、KCF、MOSSE和GOTURN、SiamFC网络,其中,Boosting、MEDIANFLOW、MIL、TLD为传统追踪算法,MOSSE、KCF为相关滤波追踪算法,GOTURN、SiamFC为孪生网络追踪算法,以上算法除SiamFC外在OpenCV库中均有开源实现。而图3中追踪器上方的虚线框表示多目标追踪任务的数据关联算法,目的在于匹配帧间多个目标,防止发生同类目标漂移现象,例如经典的匈牙利匹配算法、KM算法等。

由于数据关联算法不在本文讨论范围之内,而相关滤波追踪算法与孪生网络追踪算法已在第1节进行了总结,故本小节将简要总结嵌入式里常用的几种传统追踪算法。

Boosting算法利用AdaBoost的思想来实现目标追踪。追踪过程中初始指定的目标为正样本,并选取周围的像素块作为负样本,分类器给出分类得分,并选取分数最高的为目标新位置。在新的一帧图像中,以该位置选择正负样本来重新训练分类器。缺点在于性能一般,追踪失败反馈难度大。

MEDIANFLOW追踪算法基于LK光流追踪算法,假设时间上的正向追踪和反向追踪得到的轨迹应该一致,最小化前反向误差(FB),同时根据FB误差选择最佳追踪点来决定目标在新一帧图像中的尺寸和位置,基本上是取中值。优点在于可以跟踪目标的尺寸变化,追踪失败反馈敏锐,缺点在于追踪快速运动物体和遮挡物体性能差。

MIL追踪算法与Boosting相比,正样本为初始指定目标附近的样本,大多数表示正样本的目标中心并不居中,训练以样本包为单位。优点在于能处理部分遮挡,但与Boosting一样,追踪失败反馈难度大。

TLD 追踪算法与前面的算法相比, 类似于一种追踪模式, 分为追踪器、检测器、学习器. 追踪器常用 MEDIANFLOW 追踪算法; 检测器主要是估计追踪器的误差, 使用多帧图像、级联分类策略和积分图快速计算方法产生正负样本送给学习器得到检测结果, 以此在误差较大的情况下更正追踪器结果; 学习器用 P-N 学习算法迭代训练分类器改善检测器的精度. 优点在于长时间追踪、处理目标遮挡和目标消失问题, 但同类目标漂移现象严重.

2.2 轻量的孪生追踪网络

从 VOT 比赛的情况来看, 尽管研究者提出许多深度学习追踪网络, 但 2018 年前的深度学习追踪算法不具有优越性. 近几年来, 深度学习追踪算法中的孪生追踪网络表现较好, 具有代表性的孪生追踪网络如表 3 所示. 尽管这些算法精度较好, 但其模型体积和计算量并不适合在端侧芯片中进行部署, 往往需要云端芯片进行加速, 如 Ocean. 直到近两年, 才出现适合在多应用物联网设备上部署的追踪网络, 例如手机, 这些追踪网络往往具有几兆大小和几百兆浮点运算量. 本小节将介绍轻量的孪生追踪网络, 包括 LightTrack^[84]、E.T.Track^[85]、FEAR-XS^[86].

表 3 经典的孪生追踪网络

算法	计算量 (GFLOPs)	参数量 (M)	EAO (%)	测试集
SiamFC (A)	2.7	2.3	27.4	VOT-15
SiamFC++ (G)	17.5	13.9	28.8	VOT-19
SiamRPN++ (M)	7.0	11.2	28.8	VOT-19
SiamRPN++ (R)	≈49	53.951	23.9	VOT-ST2021
Ocean (offline) (R)	20.3	25.9	32.7	VOT-19
ATOM	—	8.4	30.1	VOT-19

注: A代表AlexNet, G代表GoogLeNet, R代表ResNet50, M代表MobileNetV2.

LightTrack 追踪器^[84] 主要利用 one-shot NAS 在由深度可分离卷积和翻转移动瓶颈卷积构成的搜索空间上来设计更轻量和高效率的追踪器, 比 SiamRPN++ 和 Ocean 具有更少的参数量和运算量. 主要有预训练骨干超网, 训练追踪超网, 使用进化算法搜索追踪超网 3 步; 第 1 步中使用 ImageNet 和交叉熵损失函数预训练骨干超网, 并且用单路径均匀采样方法使一数据块只更新其中一条路径的权重; 第 2 步使用追踪数据集的训练集、交叉熵分类损失函数和目标边界框的 IoU 损失函数来微调骨干超网权重和训练追踪头权重, 同第

1 步使用单路径均匀采样方法; 第 3 步中共享第 2 步中得到的网络权重, 使用进化算法并且重新计算 BN 层结果在追踪数据集的验证集上计算所得追踪网络精度, 最终得到符合预设参数量和计算量的最优追踪网络.

E.T.Track^[85] 主要受 scaled dot-production attention 启发提出计算量低的 Exemplar Transformer 来设计高效实时的 Transformer 追踪器, Exemplar Transformer 替换标准 Transformer 的 Attention 为 Exemplar Attention, 如式 (2) 所示, 如图 4 所示. $\Psi_S(X)$ 表示输入空间维度降到 S 的平均池化运算, 目的在于减少计算量, 原因在于视觉任务输入的相邻空间表示相同物体, 故只需要粗略查询就可以利用 exemplar value; \hat{W}_K^T 代表捕获数据集信息的一小组范例表示, 可在数据集共享, 共 E 个. $(\Psi_S(X)W_Q)(\hat{W}_K^T)$ 共同粗略表示数据集的全局时空信息, 而“ \otimes ”表示卷积运算, 代表输入的局部信息. 此外, 作者将 LightTrack 的追踪头中卷积层全换为 Exemplar Transformer, 以常数倍的计算代价在换来精度提升.

$$A(x) = \text{Softmax} \left(\frac{(\Psi_S(X)W_Q)(\hat{W}_K^T)}{\sqrt{d_k}} \right) (W_V \otimes X) \quad (2)$$

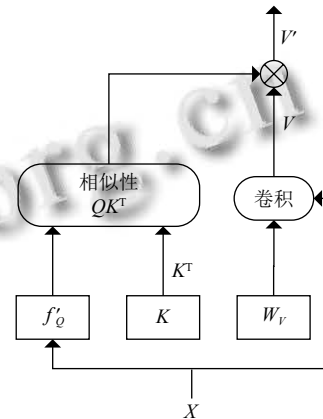


图 4 Exemplar Attention

FEAR-XS^[86] 主要提出双模板表示模块来适应目标外形变化和逐像素融合模块来更好地回归目标边界框. 特征提取骨干网为 NAS 搜索算法所得 FBNet 网络 (ImageNet 中训练) 的前 4 个阶段加上 AdjustLayer (256 个卷积层和 BN 层组成的 filter). 逐像素融合模块是对孪生追踪网络两分支相似性运算的代替, 在孪生追踪网络中具有普适性, 如图 5 所示. 一般通过在线训练来增强模型抗目标外形变化能力, 而双模板表示模

块是一种离线训练方法. 训练时捕获视频序列中目标变化特征获得动态模板的特征, 以可学习 ω 权重线性插入到静态模板的特征, 最终得到目标模板的特征; 而推断时则与搜索区域特征经过不同池化算子后计算余弦相似度来更新动态模板. 同时, 损失函数为 triplet loss、IoU loss 与 focal loss 的线性组合.

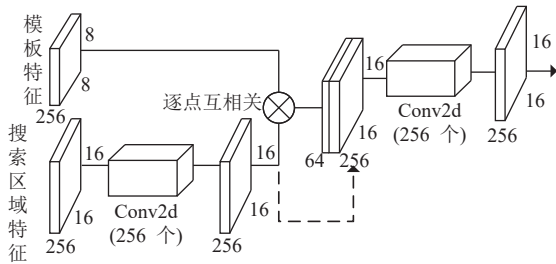


图5 逐像素融合模块

3 嵌入式目标追踪应用

物联网和移动互联网时代, 智能手机、穿戴设备和智能家居的普及和使用给人们的日常生活带来极大便利, 而在这些计算资源、存储资源和能耗受限的小型设备上发挥神经网络的优势而进行目标追踪是近几年的新兴领域. 本节首先总结网络模型部署流程, 然后总结嵌入式目标追踪应用实例.

3.1 网络模型部署流程

本节主要简要总结在嵌入式工程环境下如何部署深度学习追踪算法的网络模型及所面临的困难. 一般而言, 深度学习模型的部署流程如图6所示. 其中, 运用框架训练模型是指用广泛使用的深度学习框架来构建选定的模型, 常见框架有 TensorFlow、PyTorch、Caffe, 接下来分别总结部署平台、压缩工具和部署工具(推理引擎).

3.1.1 部署平台

部署平台一般指模型的实际运行芯片, 主要有手机芯片和嵌入式 AI 开发板两种, 由设计人员的预算、功耗和应用场景决定.

一般而言, 嵌入式 AI 开发板是通用处理器与人工智能处理器的组合异构架构, 具有操作系统, 常见的处理器如表4所示. 其中, 通用处理器常见 ARM 公司专门为边缘人工智能计算设计的 A 系列 CPU, Intel 的凌动处理器与 Movidius VPU, 并且具有 FPU 加快浮点数运算; 而不同产商具有不同架构的人工智能处理器, 比

如 FPGA、DSP、NPU、KPU、VPU、Edge TPU、ARM Mali GPU 等, 这也意味着具有不同的指令集和计算模式, 如 RK1808 的 NPU 支持 INT8、INT16、FP16, BM1880 的 Edge TPU 只支持 INT8, 与压缩工具和部署工具紧密挂钩, 决定模型是否需要压缩和附加模型压缩条件. 此外, 它们往往支持常用深度学习框架得到的模型格式, 如 ONNX 等. 对于手机芯片, 总情况与嵌入式 AI 开发板类似, 只不过一般采用大小核设计和 ARM 公司设计的 A 系列 CPU 和 Mali-T GPU, 还具有自研的设计架构、自研的处理器和矩阵运算加速器, 只不过核数更多, 性能更强, 功耗也越高, 如华为麒麟系列芯片、高通骁龙系列芯片、联发科天玑系列芯片、三星猎户座芯片和苹果 A 系列芯片.

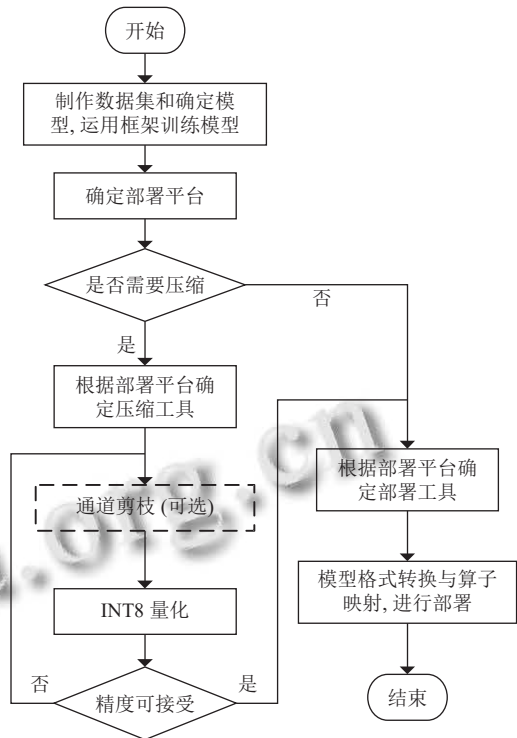


图6 深度学习模型的部署流程

目前看来, 因为智能物联网是有应用价值的新兴领域, 故国内外公司在嵌入式 AI 芯片上均有布局和产品, 国外手机芯片产品较多, 而国内嵌入式 AI 开发板产品较多, 尽管国内公司自研了众多 AI 加速器, 但芯片的加工和通用处理器仍是国外技术为主. 此外, 不同的芯片具有不同的架构和指令集, 导致压缩和部署工具种类繁多.

3.1.2 压缩工具和部署工具

目前, 智能物联网是人工智能最有希望落地的领

域,大公司致力于加大研究投入来争夺该领域的至高点,掌握市场的主导权.这就产生了如表4所示的众多具有不同指令集和结构的芯片,支持AI推理的位数也

不尽相同,同时部署需要减小模型体积和提高推理速度.鉴于上述原因,厂家提出自研的模型压缩工具和部署工具.

表4 嵌入式AI开发板的处理器

型号	通用处理器	主频(Hz)	人工智能处理器架构	RAM	主要用途
STM32747系列	ARM Cortex M7+M4	480 M+240 M	DSP	1 MB	分类任务
STM32MP1系列	ARM Cortex A7+M4	800 M+209 M	DSP	708 KB,可扩展1 GB	分类和检测任务
RK1808	ARM Cortex A35	1.6 G	NPU	2 MB,可扩展4 GB	智能物联网
RK3399Pro	ARM Cortex A72+A53	1.8 G+1.4 G	Mali-T GPU+NPU	200 KB,可扩展4 GB	智能物联网
Raspberry Pi 4B	ARM Cortex-A72	1.5 G	—	1、2、4或8 GB	实时性低的视觉任务
MYC-Y7Z010/20-V2	ARM Cortex-A9	667 M-766 M	FPGA	512 MB,最高1 GB	控制任务
K210	RISC-V Dual Core	400 M	KPU	8 MB	机器视觉任务
旭日3	ARM Cortex-A53+R5	—	BPU	最高4 GB	智能摄像头
BM1880	ARM Cortex-A53+RSIC-V Single Core	1.5 G+1 G	Edge TPU	2 MB,可扩展4 GB	音频识别、机器视觉
NVIDIA Jetson Nano	ARM Cortex-A57	1.43 G	GPU	最高4 GB	智能物联网
MLU220	ARM Cortex-A55*4	1.5 G	NPU	8 GB	多人工智能应用

注:RAM一列中如果小于10 MB,则表示为处理器自带的SRAM存储空间;其他容量则表示可扩展的最大的DRAM存储空间,一般为LPDDR的内存规格.

目前,INT8量化^[87,88]是模型压缩中理论完备的方法,是工程环境中受欢迎的方法,是量化工具广泛支持的方法,也是部署平台广泛支持的推理计算模式.本小节先总结INT8量化方法,再总结压缩工具和部署工具.

(1) INT8量化方法

完备的INT8量化理论最早由Google提出^[87],只量化对网络精度影响小的权值和特征值(激活值),采用线性缩放的方式将权值和特征值放到 $[0,255]$ 或 $[-127,127]$,如量化式(3)所示,引入了舍入误差.线性缩放系数 s 代表一层网络的一个卷积核或所有卷积核或特征图一个通道或全部通道的最小表示间隔,为浮点数,采用式(4)计算; z_0 代表0点的INT8整数值,对称量化为0,非对称量化采用式(5)计算; q 代表放缩区间的数值,为整数; r 表示量化张量中的实际原数值; $round$ 表示舍入算子.

$$q = round\left(\frac{r}{s} + z_0\right) \quad (3)$$

式(4)中 r_{min} 和 r_{max} 表示让量化结果尽可能分布满整个量化区间所选的原张量范围,张量中小于 r_{min} 的数用 r_{min} 代替,大于 r_{max} 的数用 r_{max} 代替,这引入了截断误差. n 表示量化的位数,这里 $n=8$.

$$s = \frac{r_{max} - r_{min}}{2^n - 1} \quad (4)$$

式(5)一般仅在非对称量化时才会使用, q_{max} 表示量化范围的最大值,这里 $q_{max}=255$.

$$z_0 = round\left(q_{max} - \frac{r_{max}}{s}\right) \quad (5)$$

式(6)为量化结果映射到浮点域的公式,在量化模拟中使用, \hat{r} 表示映射结果.

$$\hat{r} = s(q - z_0) \quad (6)$$

一般而言,先使用后训练量化方法(PTQ),流程如图7所示.图7中制作校正集用来统计激活值的数值范围,方便确定激活值的量化范围,NVIDIA建议从验证集中选取500到1000个样本.此外,模型需要再修改来让量化工具自动地进行量化模拟,例如高通的AIMET量化工具要求模型每个模块必须是module类和替换激活函数为ReLU等,并且当有些算子和模型结构不被工具支持需要设计者进行算子映射和结构映射.另外,确定量化方法一般是在对称量化和非对称量化之间进行选择,例如文献[88]和NVIDIA建议权值使用对称量化,激活值使用非对称量化,同时确定使用固定数位量化还是混合数位量化方案;量化模拟前需要折叠BN层和ReLU激活函数来达到更好的量化效果,一般由工具根据模型结构自动加入伪量化节点来进行量化模拟.

图7中确定权值量化范围,文献[87]中直接使用原权值的最小最大值,而文献[88]使用均方误差最小化量化权值前后的误差确定,TensorRT使用KL散度最小化量化权值前后分布的差异性确定.此外,在确定激活值的量化范围之前,需要用校正集来统计模型特

征值(激活值)分布情况,文献[87]使用指数移动平均(EMA)方法在统计过程中确定特征值量化范围,文献[88]提出均方差和基于BN层的量化特征值范围方法.量化之后的参数将由式(3)–式(5)计算得出,方便量化模型在推理中进行精度评估.此外,校正量化误差为可选操作,文献[88]提出 cross-layer equation (CLE) 和吸收高偏置值来改善和校正权值分布差异大的结果,尤其是深度可分离卷积的权值,也提出偏置校正和 AdaRound 来减少量化误差.

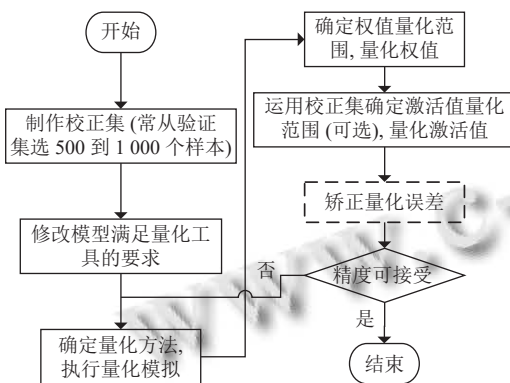


图7 PTQ 量化流程

图7中量化模型精度可能无法得到保证,这就需要考虑改变量化方法和量化范围确定方法,或者使用校正量化误差方法,甚至使用量化感知训练(QAT)方法. QAT方法与PTQ方法区别在于首先制作训练集,然后在训练过程中通过伪量化节点模拟量化误差来同时量化权值、特征值和校正量化误差.一般而言,模型前向计算使用式(6)将INT8量化结果转为浮点数进行推理,然而由于舍入算子round导数为0,故在未量化的浮点权重上使用梯度下降算法(STE)来优化模型.此外,在更低位数的权值或激活值量化上,如INT4,基本上要使用QAT方法[88].

(2) 量化工具与部署工具

众多芯片厂商为了良好地适配自研的边缘AI芯片和抢占市场而提出自研的量化工具与部署工具.量化工具主要是在软件环境下通过量化模拟得到模型的量化版本,而部署工具主要将模型的量化版本部署到边缘AI芯片上.一般而言,量化工具与部署工具由部署平台决定,且量化工具的结果面向对应的部署工具,故量化工具和部署工具常配套使用,如表5所示.

表5 量化工具与部署工具

所属公司	量化工具	量化支持框架	部署工具	部署支持平台
英特尔	NNCF	P、T	OpenVINO	英特尔的CPU、显卡、VPU
高通	AIMET	P、T	SNPE/QNN	骁龙CPU、Ardeno GPU、Hexagon DSP
联发科	NeuroPilot ML Kits	TFL	NeuroPilot-Micro/ NeuroPilot SDK	联发科的APU、MT系列MCU、天玑系列芯片、曦力系列芯片
英伟达	TensorRT	T、P(O)、O	TensorRT	英伟达GPU (CUDA计算能力不低于5.0)
Meta	PyTorch	P	LibTorch	ARM、x86 CPU
百度	PaddleSlim	P(O)、T、C、Pa	Paddle Lite	ARMv7a、ARMv8a、x86 CPU、ARM Mali G、高通 Ardeno GPU、苹果A系列、百度和英特尔FPGA、百度昆仑、华为NPU、瑞芯微NPU、比特大陆TPU、联发科APU等
腾讯	NCNN	P、T、M、O、K、D等	NCNN	英伟达、AMD、英特尔、高通、苹果、ARM的CPU和GPU平台
阿里	MNN	T、C、O、P(O)、M(O)	MNN	安卓和iOS平台
商汤	MQBench	P	OpenVINO、SNPE、Vitis、TensorRT、Tengine等	部署工具所支持的所有平台

注: 量化支持框架一列中P指PyTorch, T指TensorFlow, O指ONNX, TFL指TF Lite, C指代Caffe, M指代MXNet, Pa指Paddle, K指Keras, D表示DarkNet, P(O)指借助ONNX的PyTorch, M(O)指借助ONNX的MXNet.

目前看来,国内公司在工具的部署平台广泛支持性上做了大量努力,而嵌入式AI开发板的工具较手机芯片工具并不完善.同时,嵌入式AI开发板厂商提供的量化工具功能单一和受众人群众少,并不具有竞争力,推荐使用百度或腾讯工具.此外,尽管不同公司提供的量化工具比深度学习框架的量化功能要完善,但大多

数与自研的部署工具联系紧密,无法与其他公司的量化工具一同使用,这就带来了量化方法和量化模拟的局限性,导致很多结构无法进行量化和部分模型的量化版本精度具有随机性.这对芯片的应用开发者并不友好,故迫切需要提出量化规范来对接不同的部署工具和开发一款具有众多量化算法和量化模拟方法的量

化工具,商汤的MQBench在这方面较为优秀。

3.2 嵌入式目标追踪应用实例

嵌入式目标追踪应用常见于机器人应用、无人机应用、自动驾驶、安全系统和监控系统,本小节将从以上几方面总结近几年出现的嵌入式目标追踪应用实例,并重点说明应用实例中的检测模块和追踪模块。

3.2.1 机器人应用

Zhang 等人^[77]使用头部检测模块和运动评估模块设计宇航员舱内跟踪系统,头部检测模块使用修改了多尺寸预测层的轻量 DCNN 模型,而头部运动估计模块是使用检测结果和深度图像的概率预测模型。Ahmed 等人^[75]主要将以往用正视图的监控系统推广到使用俯视图的监视系统,提出一种用俯视图进行监控的机器人框架,框架中包含 MobileNet-SSD 或 YOLO 检测模型和不同的追踪算法 (GOTURN、MEDIANFLOW、TLD、KCF、MIL 和 Boosting) 的搭配,并且会根据预设条件选择不同的搭配完成追踪任务。

3.2.2 无人机应用

Le 等人^[80]使用深度学习算法和相关滤波方法构建无人机跟随人的应用,首先基于 MobileNetV2-SSDLite 构建检测器来检测人,然后配合 MOSSE 追踪器完成跟随任务,部署平台为 Raspberry Pi。Dinh 等人^[82]提出能长期和实时追踪车辆的无人机系统,该系统的追踪器由 Siam-FC 网络和 YOLO 网络组成的非对称的孪生网络,其中 Siam-FC 网络的模板分支比搜索分支多出判别相关滤波操作而获取更好的区分分数图,而 YOLO 网络在 Siam-FC 网络的结果低于阈值的情况下用来检测搜索区域的目标产生增强分数图并结合区分分数图来改善追踪结果和进行模板更新。Yang 等人^[89]使用负载均衡方式提出分层的无人机跟踪目标框架来考虑精度和时间平衡,CNN 模型的低层嵌入到无人机中,而模型的高层交给边缘移动计算装置,优化框架的加权和成本保证精度和时间平衡。

3.2.3 自动驾驶应用

RetinaTrack^[90]修改 RetinaNet 检测器能提取实例级别的特征,并且修改 RetinaNet 的后特征金字塔预测子网络,构成一阶段多目标追踪网络。Chang 等人^[91]提出一阶段实例分割的多任务多目标追踪网络 YolTrack,其中使用带特征金字塔网络 (FPN) 的 ShuffleNetV2 作为骨干网,并且扩展两个解码器来生成实例段和嵌入向量,此外采用分割掩膜对特征图进行逻辑与运算来

提高追踪精度。

3.2.4 安全监测系统应用

Vu 等人^[67]在无人机上使用分割网络 UNET 和分类网络 MobileNet 处理红外相机拍摄的天坑图,并且使用匈牙利算法来完成对天坑的追踪来保证居民和建筑安全。Xiao 等人^[71]提出变电站异物检测与追踪系统,主要由采用背景消除算法的移动目标检测器、YOLOv4 检测器和 MEDIANFLOW 追踪器构成,追踪器只有在检测失败时以一定的频率运行。Dahal 等人^[92]使用 CNN 网络和 LSTM 网络构建拖车援助系统,该系统的缺陷在于难以建模多样的拖车外观,由于摄像头的局限而导致角度估计不准确,在小物体和刹车灯的反射光处理上易产生错误。

3.2.5 人车监测系统应用

本小节主要从无类别监控系统、人监控系统和车辆监控系统 3 方面说明监控系统应用。

(1) 无类别监控系统

无类别监控系统主要监控人或车辆,可以细分为多目标追踪系统和单目标追踪系统。监控系统在现实中有广泛的应用场景,比如监控系统可以用于自动驾驶领域来避让行人和车辆。

多目标追踪系统往往需要数据关联算法来对追踪的目标进行联系。Fernández-Sanjurjo 等人^[78]提出实时多目标追踪的嵌入式 GPU 系统,该系统主要由检测器、追踪器和数据关联模块组成,检测器主要是在 YOLOv3 基础上修改了检测头,追踪器主要是处理外形变化的核相关滤波器和处理运动信息的卡尔曼滤波器,数据联系模块使用特征联系和位置联系。Tu 等人^[65]提出了 REMOT,包括使用基于外形模型的 angular triplet loss 的关联测度和运动联系作为数据联系依据,并且使用策略优化深度网络层的并行处理能力和压缩模型。Blanco-Filgueira 等人^[93]在 NVIDIA Jetson TX2 开发工具包的帮助下使用 GOTURN 追踪器结合基于像素的面向硬件自适应分割算法检测移动物体完成多目标追踪。Liu 等人^[70]使用改变过滤器数量和 Fire 模块的 SqueezeNet 实现的一阶段多盒检测器 (SSD) 和自定义的关联算法实现道路上多目标追踪。而 Lu 等人^[69]使用摄像头和雷达传感器混合算法来改善跟踪精度,其中使用 SSD-Lite 或 PVANet-Lite 识别目标和使用无迹卡尔曼滤波和卡尔曼滤波来追踪目标。

单目标跟踪系统目前看来往往采用孪生神经网络和相关滤波的判别式方法。Yang 等人^[79]使用 8 位整数

量化的 MobileNet-SSD 网络和关注目标尺寸变化和遮挡问题的核相关滤波器在嵌入式开发板上完成单目标追踪,并在 IndustriPi 板上做了测试. Mao 等人^[81]将 Siam-FC 网络的标准卷积变成深度可分离卷积构建追踪器,并且根据过滤器的方差对深度卷积层的过滤器进行剪枝. MiniTracker^[83]使用剪枝量化的 Siam-FC 的 8 位参数网络构建追踪器并且将卷积核大小统一为 3×3.

(2) 行人监控系统

Zhu 等人^[73]使用单眼深度估计和二维空间中目标检测来完成三维空间中人的定位,并用卡尔曼滤波算法完成人的跟踪任务,其中检测网络使用 MobileNet-SSD. Liu 等人^[66]提出的地铁乘客检测算法 (MPD) 中使用位置和移动方向信息来判断地铁上乘客是否下地铁,该算法由 MetroNexts 检测网络和光流算法组成,前者的核心是由多分支扩张卷积模块 (MDC)、侧卷积模块 (LC) 和位置感知注意模块 (PA) 组成的多尺度注意卷积块 (MASC),后者用来预测乘客移动的方向. Vu 等人^[74]使用 Tiny-YOLOv2 改进而来的 YLHD 检测器搭配卢卡斯-卡纳德光流法追踪器来完成行人监控.

(3) 车辆监控系统

Chen 等人^[68]提出用深度学习方法在边缘设备上智能交通流量检测的方案,该方案分为基于 YOLOv3 的目标检测网络、基于 DeepSORT 多目标追踪算法的追踪网络和实时车辆追踪器计数器. 而 Sreekumar 等人^[76]提出交通模式收集和分析模型 (TPCAM) 分别来收集路口交通数据和统计路口交通数据,其中模型的目标检测与分类模块基于 YOLOv2 构建,其中多目标追踪算法引擎包括卡斯-卡纳德光流法、MEDIANFLOW、KCF、TLD. 另外, Hsu 等人^[72]在带监视摄像头的嵌入式系统中实现多车追踪和计算,系统中使用 PVANet-Lite 网络检测车辆和卡尔曼滤波追踪车辆.

4 总结与展望

本文首先概括性介绍了目标追踪任务、追踪算法和嵌入式目标追踪的研究背景;其次,从完善相关滤波追踪器框架、端到端的相关滤波追踪网络和相关滤波网络作为追踪器的子网络 3 方面分类总结深度学习追踪算法的重要一类——结合相关滤波与深度学习的跟踪方法;然后,从基于轻量检测网络的追踪算法和轻量的孪生追踪网络两方面介绍总结基于轻量神经网络的目标追踪方法;最后总结网络模型部署流程和按照无人

机应用、机器人应用、自动驾驶、安全系统和监控系统分类归纳了近几年兴起的嵌入式目标追踪应用实例,主要面向边缘嵌入式设备.

目前,虽然目标追踪算法和轻量化方法发展已经日益成熟,但嵌入式目标追踪算法的实际部署上仍然是轻量的目标检测网络与传统追踪算法或者相关滤波算法搭配为主,而轻量的孪生追踪网络仍处于起步阶段.同时,国内外公司在边缘 AI 芯片上均有布局,尽管国内公司在工具的适配性上有一定努力,但量化工具和部署工具的紧耦合性在一定程度上阻碍智能物联网的发展.最后,大部分追踪应用都是在手机芯片和成熟的嵌入式 GPU 开发板上完成,例如 NVIDIA Jetson XT2,其他开发板的追踪应用实例相对较少.

近年来轻量神经网络设计中也开始采用自注意力机制、ViT、NAS 等方法,训练上也开始采用稀疏训练、自监督训练的方法,如何在嵌入式目标追踪算法中使用这些方法具有广泛的研究空间.同时,使用 NAS 构造孪生追踪网络在嵌入式深度学习追踪网络领域是一个新兴的有价值的研究方向.亟需制定量化规范和开发适用性广的量化工具和部署工具,并且易于硬件部署的量化 ViT 方法具有研究价值.

嵌入式目标追踪算法在无人机和监控系统上应用比较多,尤其是无人机应用.这是由于无人机方便搭载雷达、红外摄像头、彩色摄像头等成像装置和具有安全、人机交互等良好特性,故无人机具有更广泛的应用场景,如:基础设施的检查和维护、军事等.无人机应用需要算法具有自适应性和背景感知能力,而多无人机应用还需要关注算法的鲁棒性、安全机制和无人机自主决策能力,无人机目标追踪算法仍是一个值得深入研究的方向.

参考文献

- 1 Yilmaz A, Javed O, Shah M. Object tracking: A survey. *ACM Computing Surveys*, 2006, 38(4): 13–57. [doi: [10.1145/1177352.1177355](https://doi.org/10.1145/1177352.1177355)]
- 2 Choudhary T, Mishra V, Goswami A, et al. A comprehensive survey on model compression and acceleration. *Artificial Intelligence Review*, 2020, 53(7): 5113–5155. [doi: [10.1007/s10462-020-09816-7](https://doi.org/10.1007/s10462-020-09816-7)]
- 3 Bolme DS, Beveridge JR, Draper BA, et al. Visual object tracking using adaptive correlation filters. *Proceedings of IEEE Computer Society Conference on Computer Vision and*

- Pattern Recognition. San Francisco: IEEE, 2010. 2544–2550.
- 4 Henriques JF, Caseiro R, Martins P, *et al.* Exploiting the circulant structure of tracking-by-detection with kernels. Proceedings of the 12th European Conference on Computer Vision. Florence: Springer, 2012. 702–715.
 - 5 Danelljan M, Khan FS, Felsberg M, *et al.* Adaptive color attributes for real-time visual tracking. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014. 1090–1097.
 - 6 Henriques JF, Caseiro R, Martins P, *et al.* High-speed tracking with kernelized correlation filters. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(3): 583–596. [doi: [10.1109/TPAMI.2014.2345390](https://doi.org/10.1109/TPAMI.2014.2345390)]
 - 7 Ma C, Huang JB, Yang XK, *et al.* Hierarchical convolutional features for visual tracking. Proceedings of IEEE International Conference on Computer Vision. Santiago: IEEE, 2015. 3074–3082.
 - 8 Danelljan M, Häger G, Khan FS, *et al.* Convolutional features for correlation filter based visual tracking. Proceedings of IEEE International Conference on Computer Vision Workshop. Santiago: IEEE, 2015. 621–629.
 - 9 Choi J, Chang HJ, Fischer T, *et al.* Context-aware deep feature compression for high-speed visual tracking. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 479–488.
 - 10 Cheng X, Zhang YF, Zhou L, *et al.* Visual tracking via auto-encoder pair correlation filter. IEEE Transactions on Industrial Electronics, 2020, 67(4): 3288–3297. [doi: [10.1109/TIE.2019.2913815](https://doi.org/10.1109/TIE.2019.2913815)]
 - 11 Gundogdu E, Alatan AA. Good features to correlate for visual tracking. IEEE Transactions on Image Processing, 2018, 27(5): 2526–2540. [doi: [10.1109/TIP.2018.2806280](https://doi.org/10.1109/TIP.2018.2806280)]
 - 12 Wu QQ, Yan Y, Liang YJ, *et al.* DSNet: Deep and shallow feature learning for efficient visual tracking. Proceedings of the 14th Asian Conference on Computer Vision. Perth: Springer, 2019. 119–134.
 - 13 Wang XY, Li HX, Li Y, *et al.* Deep tracking with objectness. Proceedings of IEEE International Conference on Image Processing. Beijing: IEEE, 2017. 660–664.
 - 14 Ma C, Huang JB, Yang XK, *et al.* Robust visual tracking via hierarchical convolutional features. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(11): 2709–2723. [doi: [10.1109/TPAMI.2018.2865311](https://doi.org/10.1109/TPAMI.2018.2865311)]
 - 15 Li F, Yao YJ, Li PH, *et al.* Integrating boundary and center correlation filters for visual tracking with aspect ratio variation. Proceedings of IEEE International Conference on Computer Vision Workshops. Venice: IEEE, 2017. 2001–2009.
 - 16 Zhang P, Zhuo T, Huang W, *et al.* Online object tracking based on CNN with spatial-temporal saliency guided sampling. Neurocomputing, 2017, 257: 115–127. [doi: [10.1016/j.neucom.2016.10.073](https://doi.org/10.1016/j.neucom.2016.10.073)]
 - 17 Tang FH, Lu XK, Zhang XY, *et al.* Deep feature tracking based on interactive multiple model. Neurocomputing, 2019, 333: 29–40. [doi: [10.1016/j.neucom.2018.12.035](https://doi.org/10.1016/j.neucom.2018.12.035)]
 - 18 Li YM, Fu CH, Huang ZY, *et al.* Keyfilter-aware real-time UAV object tracking. Proceedings of IEEE International Conference on Robotics and Automation. Paris: IEEE, 2020. 193–199.
 - 19 Fu CH, He YJ, Lin FL, *et al.* Robust multi-kernelized correlators for UAV tracking with adaptive context analysis and dynamic weighted filters. Neural Computing and Applications, 2020, 32(16): 12591–12607. [doi: [10.1007/s00521-020-04716-x](https://doi.org/10.1007/s00521-020-04716-x)]
 - 20 Dai KN, Wang D, Lu HC, *et al.* Visual tracking via adaptive spatially-regularized correlation filters. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 4665–4674.
 - 21 Danelljan M, Bhat G, Khan FS, *et al.* ECO: Efficient convolution operators for tracking. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 6931–6939.
 - 22 Danelljan M, Robinson A, Khan FS, *et al.* Beyond correlation filters: Learning continuous convolution operators for visual tracking. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 472–488.
 - 23 Liu MJ, Jin CB, Yang B, *et al.* Occlusion-robust object tracking based on the confidence of online selected hierarchical features. IET Image Processing, 2018, 12(11): 2023–2029. [doi: [10.1049/iet-ipr.2018.5454](https://doi.org/10.1049/iet-ipr.2018.5454)]
 - 24 Che MQ, Wang RL, Lu Y, *et al.* Channel pruning for visual tracking. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2019. 70–82.
 - 25 Xu TY, Feng ZH, Wu XJ, *et al.* Joint group feature selection and discriminative filter learning for robust visual object tracking. Proceedings of IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 7949–7959.
 - 26 Qi YK, Zhang SP, Qin L, *et al.* Hedged deep tracking. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 4303–4311.
 - 27 He ZQ, Fan YR, Zhuang JF, *et al.* Correlation filters with weighted convolution responses. Proceedings of IEEE

- International Conference on Computer Vision Workshops. Venice: IEEE, 2017. 1992–2000.
- 28 Bhat G, Johnander J, Danelljan M, *et al.* Unveiling the power of deep tracking. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2018. 493–509.
- 29 Wang N, Zhou WG, Tian Q, *et al.* Multi-cue correlation filters for robust visual tracking. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 4844–4853.
- 30 Ma YP, Yuan C, Gao P, *et al.* Efficient multi-level correlating for visual tracking. Proceedings of the 14th Asian Conference on Computer Vision. Perth: Springer, 2019. 452–465.
- 31 Sun C, Wang D, Lu HC, *et al.* Correlation tracking via joint discrimination and reliability learning. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 489–497.
- 32 Fu CH, Huang ZY, Li YM, *et al.* Boundary effect-aware visual tracking for UAV with online enhanced background learning and multi-frame consensus verification. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. Macao: IEEE, 2019. 4415–4422.
- 33 Fu CH, Xiong WJ, Lin FL, *et al.* Surrounding-aware correlation filter for UAV tracking with selective spatial regularization. Signal Processing, 2020, 167: 107324. [doi: [10.1016/j.sigpro.2019.107324](https://doi.org/10.1016/j.sigpro.2019.107324)]
- 34 Du F, Liu P, Zhao W, *et al.* Spatial-temporal adaptive feature weighted correlation filter for visual tracking. Signal Processing: Image Communication, 2018, 67: 58–70. [doi: [10.1016/j.image.2018.05.013](https://doi.org/10.1016/j.image.2018.05.013)]
- 35 Sun YX, Sun C, Wang D, *et al.* ROI pooled correlation filters for visual tracking. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 5776–5784.
- 36 Lukežić A, Vojir T, Zajc LC, *et al.* Discriminative correlation filter with channel and spatial reliability. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 4847–4856.
- 37 Li DD, Wen GJ, Kuai YL, *et al.* Learning target-aware correlation filters for visual tracking. Journal of Visual Communication and Image Representation, 2019, 58: 149–159. [doi: [10.1016/j.jvcir.2018.11.036](https://doi.org/10.1016/j.jvcir.2018.11.036)]
- 38 Li F, Tian C, Zuo WM, *et al.* Learning spatial-temporal regularized correlation filters for visual tracking. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 4904–4913.
- 39 Rout L, Mishra D, Gorthi RKSS. WAEF: Weighted aggregation with enhancement filter for visual object tracking. Proceedings of the 15th European Conference on Computer Vision. Munich: Springer, 2019. 83–99.
- 40 Li SJ, Zhao S, Cheng B, *et al.* Robust visual tracking via hierarchical particle filter and ensemble deep features. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 30(1): 179–191. [doi: [10.1109/TCSVT.2018.2889457](https://doi.org/10.1109/TCSVT.2018.2889457)]
- 41 Zhang TZ, Xu CS, Yang MH. Multi-task correlation particle filter for robust object tracking. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 4819–4827.
- 42 Mozhdehi RJ, Medeiros H. Deep convolutional particle filter for visual tracking. Proceedings of IEEE International Conference on Image Processing. Beijing: IEEE, 2017. 3650–3654.
- 43 Mozhdehi RJ, Reznichenko Y, Siddique A, *et al.* Deep convolutional particle filter with adaptive correlation maps for visual tracking. Proceedings of the 25th IEEE International Conference on Image Processing. Athens: IEEE, 2018. 798–802.
- 44 Valmadre J, Bertinetto L, Henriques J, *et al.* End-to-end representation learning for correlation filter based tracking. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 5000–5008.
- 45 Wang Q, Gao J, Xing JL, *et al.* DCFNet: Discriminant correlation filters network for visual tracking. arXiv:1704.04057, 2017.
- 46 Kuai YL, Wen GJ, Li DD. Multi-task hierarchical feature learning for real-time visual tracking. IEEE Sensors Journal, 2019, 19(5): 1961–1968. [doi: [10.1109/JSEN.2018.2883593](https://doi.org/10.1109/JSEN.2018.2883593)]
- 47 Liu GE, Liu GZ. Integrating multi-level convolutional features for correlation filter tracking. Proceedings of the 25th IEEE International Conference on Image Processing. Athens: IEEE, 2018. 3029–3033.
- 48 Guo Q, Feng W, Zhou C, *et al.* Learning dynamic siamese network for visual object tracking. Proceedings of IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 1781–1789.
- 49 Tan WR, Lai SH. i-Siam: Improving siamese tracker with distractors suppression and long-term strategies. Proceedings of IEEE/CVF International Conference on Computer Vision Workshop. Seoul: IEEE, 2019. 55–63.
- 50 Li DD, Wen GJ, Kuai YL, *et al.* End-to-end feature integration for correlation filter tracking with channel attention. IEEE Signal Processing Letters, 2018, 25(12): 1815–1819. [doi: [10.1109/LSP.2018.2877008](https://doi.org/10.1109/LSP.2018.2877008)]

- 51 Zhu Z, Wu W, Zou W, *et al.* End-to-end flow correlation tracking with spatial-temporal attention. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 548–557.
- 52 Wang Q, Teng Z, Xing J, *et al.* Learning attentions: Residual attentional siamese network for high performance online visual tracking. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 4854–4863.
- 53 Wang N, Song YB, Ma C, *et al.* Unsupervised deep tracking. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 1308–1317.
- 54 Song YB, Ma C, Gong LJ, *et al.* CREST: Convolutional residual learning for visual tracking. Proceedings of IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 2574–2583.
- 55 Zhu Z, Huang G, Zou W, *et al.* UCT: Learning unified convolutional networks for real-time visual tracking. Proceedings of IEEE International Conference on Computer Vision Workshops. Venice: IEEE, 2017. 1973–1982.
- 56 Han ZJ, Wang P, Ye QX. Adaptive discriminative deep correlation filter for visual object tracking. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 30(1): 155–166. [doi: [10.1109/TCSVT.2018.2888492](https://doi.org/10.1109/TCSVT.2018.2888492)]
- 57 Lukežič A, Matas J, Kristan M. D3S—A discriminative single shot segmentation tracker. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 7131–7140.
- 58 Yang TY, Xu PF, Hu RB, *et al.* ROAM: Recurrently optimizing tracking model. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 6717–6726.
- 59 Fan H, Ling HB. Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking. Proceedings of IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 5487–5495.
- 60 Choi J, Chang HJ, Yun S, *et al.* Attentional correlation filter network for adaptive visual tracking. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 4828–4837.
- 61 Xie YC, Xiao JM, Huang KZ, *et al.* Correlation filter selection for visual tracking using reinforcement learning. IEEE Transactions on Circuits and Systems for Video Technology, 2020, 30(1): 192–204. [doi: [10.1109/TCSVT.2018.2889488](https://doi.org/10.1109/TCSVT.2018.2889488)]
- 62 Ma C, Xu Y, Ni BB, *et al.* When correlation filters meet convolutional neural networks for visual tracking. IEEE Signal Processing Letters, 2016, 23(10): 1454–1458. [doi: [10.1109/LSP.2016.2601691](https://doi.org/10.1109/LSP.2016.2601691)]
- 63 Kuai YL, Wen GJ, Li DD. When correlation filters meet fully-convolutional siamese networks for distractor-aware tracking. Signal Processing: Image Communication, 2018, 64: 107–117. [doi: [10.1016/j.image.2018.03.002](https://doi.org/10.1016/j.image.2018.03.002)]
- 64 Li HX, Wang XY, Shen FM, *et al.* Real-time deep tracking via corrective domain adaptation. IEEE Transactions on Circuits and Systems for Video Technology, 2019, 29(9): 2600–2612. [doi: [10.1109/TCSVT.2019.2923639](https://doi.org/10.1109/TCSVT.2019.2923639)]
- 65 Tu JZ, Chen CL, Xu QM, *et al.* Resource-efficient visual multiobject tracking on embedded device. IEEE Internet of Things Journal, 2022, 9(11): 8531–8543. [doi: [10.1109/JIOT.2021.3115102](https://doi.org/10.1109/JIOT.2021.3115102)]
- 66 Liu QL, Guo Q, Wang W, *et al.* An automatic detection algorithm of metro passenger boarding and alighting based on deep learning and optical flow. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 5006613.
- 67 Vu HN, Pham C, Dung NM, *et al.* Detecting and tracking sinkholes using multi-level convolutional neural networks and data association. IEEE Access, 2020, 8: 132625–132641. [doi: [10.1109/ACCESS.2020.3010885](https://doi.org/10.1109/ACCESS.2020.3010885)]
- 68 Chen C, Liu B, Wan SH, *et al.* An edge traffic flow detection scheme based on deep learning in an intelligent transportation system. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(3): 1840–1852. [doi: [10.1109/TITS.2020.3025687](https://doi.org/10.1109/TITS.2020.3025687)]
- 69 Lu JX, Lin JC, Vinay MS, *et al.* Fusion technology of radar and RGB camera sensors for object detection and tracking and its embedded system implementation. Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference. Auckland: IEEE, 2020. 1234–1242.
- 70 Liu MJ, Jin CB, Park D, *et al.* Integrated detection and tracking for ADAS using deep neural network. Proceedings of IEEE Conference on Multimedia Information Processing and Retrieval. San Jose: IEEE, 2019. 71–76.
- 71 Xiao C, Sun H, Cai YM, *et al.* Method for detecting and tracking foreign objects in substation videos based on embedded AI. Proceedings of Power System and Green Energy Conference. Shanghai: IEEE, 2021. 583–587.
- 72 Hsu YH, Chang SY, Guo JI. A multiple vehicle tracking and counting method and its realization on an embedded system with a surveillance camera. Proceedings of IEEE International Conference on Consumer Electronics—Taiwan. Taichung: IEEE, 2018. 1–2.

- 73 Zhu YP, Wang T, Zhu SQ. Real-time monocular 3D people localization and tracking on embedded system. Proceedings of the 6th IEEE International Conference on Advanced Robotics and Mechatronics. Chongqing: IEEE, 2021. 797–802.
- 74 Vu QD, Chung ST. Real-time robust human tracking based on Lucas-Kanade optical flow and deep detection for embedded surveillance. Proceedings of the 8th International Conference of Information and Communication Technology for Embedded Systems. Chonburi: IEEE, 2017. 1–6.
- 75 Ahmed I, Din S, Jeon G, *et al.* Towards collaborative robotics in top view surveillance: A framework for multiple object tracking by detection using deep learning. IEEE/CAA Journal of Automatica Sinica, 2021, 8(7): 1253–1270. [doi: [10.1109/JAS.2020.1003453](https://doi.org/10.1109/JAS.2020.1003453)]
- 76 Sreekumar UK, Devaraj R, Li Q, *et al.* TPCAM: Real-time traffic pattern collection and analysis model based on deep learning. Proceedings of IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation. San Francisco: IEEE, 2017. 1–4.
- 77 Zhang R, Zhang YL, Zhang XY. Tracking in-cabin astronauts using deep learning and head motion clues. IEEE Access, 2021, 9: 2680–2693. [doi: [10.1109/ACCESS.2020.3046730](https://doi.org/10.1109/ACCESS.2020.3046730)]
- 78 Fernández-Sanjurjo M, Mucientes M, Brea VM. Real-time multiple object visual tracking for embedded GPU systems. IEEE Internet of Things Journal, 2021, 8(11): 9177–9188. [doi: [10.1109/JIOT.2021.3056239](https://doi.org/10.1109/JIOT.2021.3056239)]
- 79 Yang CL, Chen ZW, Zhang Y, *et al.* Design of embedded target tracking system based on MobileNet and KCF. Proceedings of the 15th IEEE Conference on Industrial Electronics and Applications. Kristiansand: IEEE, 2020. 1578–1581.
- 80 Le MC, Le MH. Human detection and tracking for autonomous human-following quadcopter. Proceedings of International Conference on System Science and Engineering. Dong Hoi: IEEE, 2019. 6–11.
- 81 Mao YH, He ZZ, Ma Z, *et al.* Efficient convolution neural networks for object tracking using separable convolution and filter pruning. IEEE Access, 2019, 7: 106466–106474. [doi: [10.1109/ACCESS.2019.2932733](https://doi.org/10.1109/ACCESS.2019.2932733)]
- 82 Dinh M, Morris B, Kim Y. UAS-based object tracking via deep learning. Proceedings of the 9th IEEE Annual Computing and Communication Workshop and Conference. Las Vegas: IEEE, 2019. 217–275.
- 83 Zhang BY, Li X, Han J, *et al.* MiniTracker: A lightweight CNN-based system for visual object tracking on embedded device. Proceedings of the 23rd IEEE International Conference on Digital Signal Processing. Shanghai: IEEE, 2018. 1–5.
- 84 Yan B, Peng HW, Wu K, *et al.* LightTrack: Finding lightweight neural networks for object tracking via one-shot architecture search. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 15175–15184.
- 85 Blatter P, Kanakis M, Danelljan M, *et al.* Efficient visual tracking with exemplar transformers. arXiv:2112.09686, 2022.
- 86 Borsuk V, Vei R, Kupyn O, *et al.* FEAR: Fast, efficient, accurate and robust visual tracker. arXiv:2112.07957, 2022.
- 87 Jacob B, Kligys S, Chen B, *et al.* Quantization and training of neural networks for efficient integer-arithmetic-only inference. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 2704–2713.
- 88 Nagel M, Fournarakis M, Amjad RA, *et al.* A white paper on neural network quantization. arXiv:2106.08295, 2021.
- 89 Yang B, Cao XL, Yuen C, *et al.* Offloading optimization in edge computing for deep-learning-enabled target tracking by Internet of UAVs. IEEE Internet of Things Journal, 2021, 8(12): 9878–9893. [doi: [10.1109/JIOT.2020.3016694](https://doi.org/10.1109/JIOT.2020.3016694)]
- 90 Lu ZC, Rathod V, Votel R, *et al.* RetinaTrack: Online single stage joint detection and tracking. Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 14656–14666.
- 91 Chang XP, Pan HH, Sun WC, *et al.* YolTrack: Multitask learning based real-time multiobject tracking and segmentation for autonomous vehicles. IEEE Transactions on Neural Networks and Learning Systems, 2021, 32(12): 5323–5333. [doi: [10.1109/TNNLS.2021.3056383](https://doi.org/10.1109/TNNLS.2021.3056383)]
- 92 Dahal A, Hossen J, Sumanth C, *et al.* DeepTrailerAssist: Deep learning based trailer detection, tracking and articulation angle estimation on automotive rear-view camera. Proceedings of IEEE/CVF International Conference on Computer Vision Workshop. Seoul: IEEE, 2019. 2339–2346.
- 93 Blanco-Filgueira B, García-Lesta D, Fernández-Sanjurjo M, *et al.* Deep learning-based multiple object visual tracking on embedded system for IoT and mobile edge computing applications. IEEE Internet of Things Journal, 2019, 6(3): 5423–5431. [doi: [10.1109/JIOT.2019.2902141](https://doi.org/10.1109/JIOT.2019.2902141)]

(校对责编: 孙君艳)