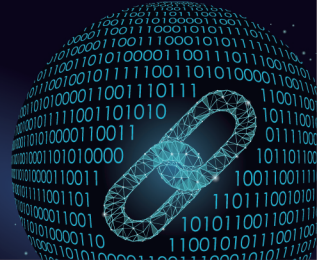




中国科技期刊卓越行动计划项目入选期刊

控制与决策

CONTROL AND DECISION



基于图网络与IoU感知的孪生网络目标跟踪算法

陈志旺, 刁华康, 袁宇, 吕昌昊, 彭勇

引用本文:

陈志旺, 刁华康, 袁宇, 吕昌昊, 彭勇. 基于图网络与IoU感知的孪生网络目标跟踪算法[J]. 控制与决策, 2024, 39(9): 2867–2875.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2023.0414>

您可能感兴趣的其他文章

Articles you may be interested in

基于条件对抗生成孪生网络的目标跟踪

Conditional generative adversarial siamese networks for object tracking

控制与决策. 2021, 36(5): 1110–1118 <https://doi.org/10.13195/j.kzyjc.2019.1215>

复杂背景下全景视频运动小目标检测算法

Panoramic video motion small target detection algorithm in complex background

控制与决策. 2021, 36(1): 249–256 <https://doi.org/10.13195/j.kzyjc.2019.0686>

尺度自适应的多特征融合相关滤波目标跟踪算法

Scale adaptation and multi-feature fusion correlation filtering object tracking algorithm

控制与决策. 2021, 36(2): 429–435 <https://doi.org/10.13195/j.kzyjc.2019.0445>

基于MobileNet的多目标跟踪深度学习算法

Deep learning algorithm based on MobileNet for multi-target tracking

控制与决策. 2021, 36(8): 1991–1996 <https://doi.org/10.13195/j.kzyjc.2019.1424>

Anchor-free的尺度自适应行人检测算法

Anchor-free scale adaptive pedestrian detection algorithm

控制与决策. 2021, 36(2): 295–302 <https://doi.org/10.13195/j.kzyjc.2020.0124>

基于图网络与IoU感知的孪生网络目标跟踪算法

陈志旺^{1,2†}, 刁华康^{2,5}, 袁宇², 吕昌昊³, 彭勇⁴

1. 燕山大学 智能控制系统与智能装备教育部工程研究中心, 河北 秦皇岛 066004;
2. 燕山大学 工业计算机控制工程河北省重点实验室, 河北 秦皇岛 066004;
3. 燕山大学 电力电子节能与传动控制河北省重点实验室, 河北 秦皇岛 066004;
4. 燕山大学 电气工程学院, 河北 秦皇岛 066004; 5. 航空工业天津航空机电有限公司, 天津 300308)

摘要: 传统孪生网络目标跟踪算法采用互相关或者深度互相关的方式对模板帧与检测帧进行相似性度量, 无法有效适应极端的目标形变. 以无锚点框的目标跟踪算法为基础, 设计一种基于图网络的 IoU 感知目标跟踪算法. 首先, 以 Resnet50 为基础, 在每个残差结构后引入通道自适应调整模块 NCAM, 构造轻量高效且具有通道自适应的特征提取网络. 其次, 基于图网络设计一种新的模板帧与检测帧的相似性计算方式, 将特征图像素点视为图网络的节点, 对模板特征与检测特征的图网络节点进行相似性计算以有效应对目标极端形变. 最后, 在分类部分采用 IoU 感知的分类损失函数在分类分支与回归分支之间建立联系, 改变以往孪生网络目标跟踪算法训练与测试不一致的情况; 在回归部分选用 CIoU 损失计算离线训练阶段的回归损失, 得到更加精准的边界框. OTB2015、UAV123、VOT2018 与 VOT2019 数据集上的实验结果验证了所提算法的有效性.

关键词: 目标跟踪; 孪生网络; 图网络; IoU 感知; 通道自适应调整; CIoU

中图分类号: TP391.4

文献标志码: A

DOI: 10.13195/j.kzyjc.2023.0414

引用格式: 陈志旺, 刁华康, 袁宇, 等. 基于图网络与 IoU 感知的孪生网络目标跟踪算法[J]. 控制与决策, 2024, 39(9): 2867-2875.

Siamese network object tracking algorithm based on graph network and IoU-aware

CHEN Zhi-wang^{1,2†}, DIAO Hua-kang^{2,5}, YUAN Yu², LV Chang-hao³, PENG Yong⁴

(1. Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Qinhuangdao 066004, China; 2. Key Laboratory of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao 066004, China; 3. Key Lab of Power Electronics for Energy Conservation and Motor Drive of Hebei Province, Yanshan University, Qinhuangdao 066004, China; 4. School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China; 5. Tianjin Aviation Electromechanical Co., Ltd, Tianjin 300308, China)

Abstract: Traditional Siamese network target tracking algorithms use cross-correlation or deep cross-correlation to calculate the similarity measurement between template frame and detection frame, which can't effectively adapt to extreme target deformation. In this paper, an IoU-aware target tracking algorithm based on graph network is designed based on the target tracking algorithm without anchor point frame. Firstly, based on Resnet50, the normalization-based channel attention module (NCAM) is introduced after each residual structure to construct a lightweight and efficient feature extraction network with channel adaptive. Then, a new similarity calculation method of template frame and detection frame is designed based on the graph network. The pixel of the feature map is regarded as the node of the graph network, and the similarity calculation is carried out on the node of the graph network with template feature and detection feature to effectively deal with the extreme deformation of the target. Finally, in the classification and regression part, the IoU-perceived classification loss function is used in the classification part to establish a connection between the classification branch and the regression branch, which changes the inconsistency between the training and testing of the Siamese network target tracking algorithm in the past. In the regression part, CIoU loss is used to calculate the regression loss in the off-line training stage, and a more accurate boundary box is obtained. Experimental results on the OTB2015, UAV123, VOT2018 and VOT2019 data sets demonstrate the effectiveness of the proposed algorithm.

Keywords: object tracking; Siamese network; graph network; IoU-aware; channel adaptive adjustment; CIoU

收稿日期: 2023-04-04; 录用日期: 2023-08-16.

基金项目: 国家自然科学基金项目(61573305); 河北省自然科学基金项目(F2022203038, F2019203511).

责任编辑: 彭木根.

†通讯作者. E-mail: czwaaron@ysu.edu.cn.

0 引言

孪生网络目标跟踪是当前目标跟踪研究的热点算法之一^[1]. 孪生网络目标跟踪的原理是将目标跟踪问题转化为初始帧与检测帧相似度计算的问题. 将模板帧的特征与检测帧的特征进行比对,从而得到目标在检测帧中的位置,继而实现跟踪目标.

当前孪生网络目标跟踪算法根据是否含有锚点框可以分为: 1) 基于锚点框的目标跟踪算法; 2) 无锚点框目标跟踪算法. 有锚点框目标跟踪算法包含 SiamFC^[2]、SiamRPN^[3]、DaSiamRPN^[4]、SiamDW^[5]、SiamRPN++^[6]、SiamMask^[7]等. 最早的孪生网络目标跟踪算法是基于全卷积孪生网络目标跟踪算法(SiamFC), 它利用全卷积的网络对模板帧与检测帧进行相似度的计算, 得到单通道相应图, 但是算法特征提取能力较弱^[8]. 随着区域候选网络(region proposal network, RPN)在目标检测领域的兴起, 学者们将RPN与孪生网络架构结合提出了孪生区域候选网络(SiamRPN), 算法可适应目标尺度带来的变化, 提高跟踪的精度与速度. 为了能够更好地抵抗相似物的干扰影响, 提高跟踪器的跟踪精度, 基于干扰物感知的孪生网络目标跟踪算法(DaSiamRPN)被提出, 并取得了较好的成效. SiamDW使用更加深层次的特征提取网络Resnet, 这意味着更加深层次的语义信息被提取, 并应用于后续的分类与回归. SiamRPN++将改进后的Resnet50作为特征提取网络, 采用多层特征融合的方式(depthwise cross correlation)将Resnet50所提取的3、4、5层特征融合得到最终用于分类与回归的特征图, 成功地将深层特征提取网络应用于目标跟踪. SiamMask^[7]是在SiamFC的基础上增加了Mask分支, 解决了视频目标跟踪与目标分割的问题, 利用优化模块提高分割算法精度, 该算法只利用初始帧的一个边界就可以实现无类型差别的实时目标分割与跟踪.

由于在孪生网络目标跟踪的回归网络中存在锚点框, 增加了算法的参数量, 导致算法的训练难度增加, 因此将无锚点框的策略引入到目标跟踪的回归网络部分. 无锚点框目标跟踪算法包含 SiamFC++^[9]、SiamCAR^[10]、SiamBAN^[11]、Ocean++^[12]算法等. SiamFC++是在SiamFC的基础上添加了位置回归和质量分数, 同时运用多种损失联合训练. SiamFC++首次将无锚点框的思想与孪生网络结合, 避免了多尺寸锚点框的复杂计算. 基于视觉跟踪的孪生全卷积分类与回归算法SiamCAR通过端到端的全卷积训练直接实现对每个位置的目标边界

框进行分类和回归, 避免大量先验框的设计, 简化了算法的复杂程度, 提升了跟踪效率. 同时, SiamCAR采用了一种新的特征融合方式, 将经过调整层的第3、4、5层进行了拼接操作, 将拼接的特征图输入到后面网络进行分类与回归. 孪生边框自适应网络SiamBAN将视觉跟踪问题视为一个并行的分类和回归问题, 因此直接对物体进行分类, 并在一个统一的全卷积网络中回归它们的边界框. 无锚点框的设计避免了与候选框相关的超参数, 使得SiamBAN更加灵活通用. 与上面两种跟踪网络不同的是, SiamBAN提出了一种新的正负样本判别方式, 使得划分更加准确, 提升了性能. 其次, 在特征融合方面区别于SiamCAR, SiamBAN采用的是加权融合, 通过训练得到3层特征的权重, 并对其进行融合. 在实时目标跟踪算法Ocean中, 作者提出一种新的特征融合方式, 在特征融合方面采用不同膨胀系数的膨胀卷积进行特征融合, 最后作者采用了类似ATOM^[13]和DIMP^[14]的在线分类器更新方式设计在线模块. 与基于锚点框的目标跟踪算法相比, 无锚点框目标跟踪算法解决了前者因为尺度估计引入大量超参数的问题, 提升了算法的速度与精度. 基于锚点框的目标跟踪算法由于先验框的存在, 其分类分支除了可以对目标进行粗定位之外, 还能确定目标的大致尺度, 这是无锚点框的目标跟踪算法不具备的优势.

上面所介绍的孪生网络目标跟踪算法在进行模板帧与检测帧之间的相似度量时, 多采用互相关或者深度互相关的方式进行. 然而, 无论是互相关还是深度互相关的方式, 都是通过检测帧与模板帧的全局匹配的方式, 很大程度上忽略了目标的形状与目标局部信息, 当目标出现较大形变或者遮挡时, 算法将无法保持原有精度. 另外, 孪生网络目标跟踪算法在推理阶段, 分类分支与回归分支相对独立缺少联系, 这将造成推理与训练的不一致性, 影响算法跟踪精度.

1 本文算法

本文算法由以下3部分组成: 特征提取模块、特征融合模块、分类与回归模块. 算法的整体框架如图1所示, 其中 Θ 代表varifocal_loss损失函数.

1.1 特征提取模块

本文采用的是以残差网络(Resnet50)为基础, 搭建通道自适应权重调整的特征提取网络Resnet-NCAM. 原Resnet50主要应用于分类任务, 但是在目标跟踪任务中并不会用到Resnet50最后2个卷积的下采样操作, 因此将其删除, 减少计算量. 同时删除

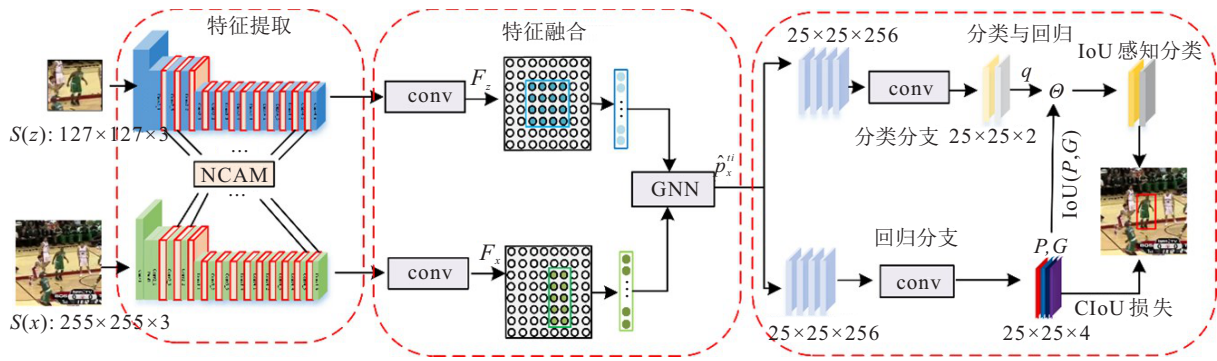


图1 本文算法跟踪框架

Resnet50的conv5卷积块,与SiamRPN++、SiamBAN、SiamCAR等不同,本文算法直接将conv4得到的特征图作为整个特征提取网络的输出,不再采用多特征融合的方式,减少了特征提取网络的参数量与计算量.其次,将Resnet50中的conv4卷积块的有效步长减少到8个像素值,并采用空洞卷积来增加感受野.

先前一些算法是通过抑制不重要的通道,而对

重要的通道赋予更多的权重,以此来进行特征图通道权重的调节.这种方法缺少对权值贡献因子的考虑,因此设计一种通道自适应调整模块NACM^[15]对残差结构进行调整.如图2所示,本文利用权值的贡献因子来进行通道权重的自适应调整,使用Batch_Normalization的缩放因子来表示权值的重要程度.

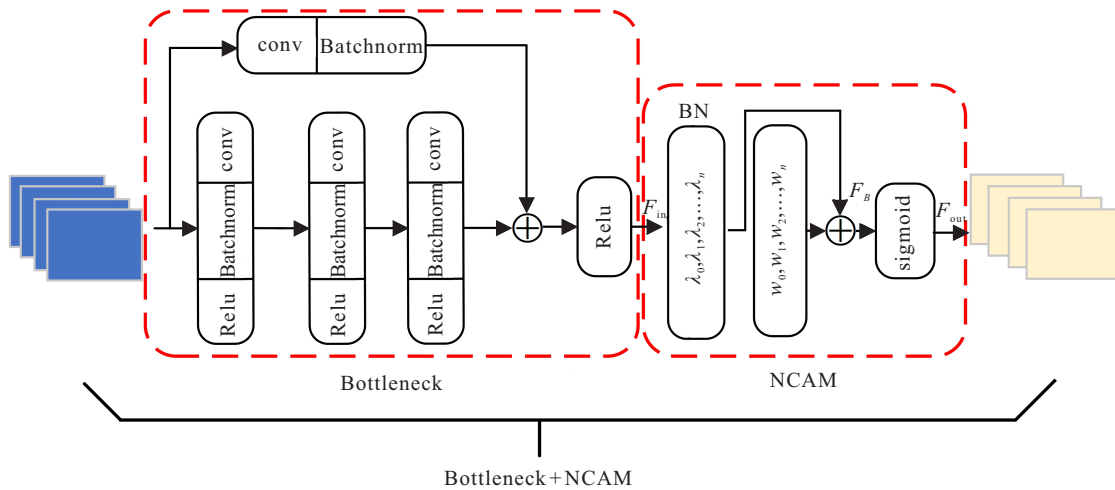


图2 残差块结构Bottleneck+NCAM

λ_c 可以代表每个通道的缩放因子,因此每个通道的权值可以通过下式得到:

$$w_c = \lambda_c / \sum_{c=1}^n \lambda_c. \quad (1)$$

其中: c 为当前特征图的通道数, n 为特征图的通道总数.将上述不同通道权重值乘上经过Batch_Normalization操作的特征 F_B , F_B 表示为

$$F_B = \text{Batch_Normalization}(F_{in}) = \lambda_c \frac{F_{in} - \mu_F}{\sqrt{\sigma_F^2 + \epsilon}} + \theta. \quad (2)$$

其中: μ_F 代表 F_{in} 的均值, σ_F^2 代表 F_{in} 的方差, θ 代表移动因子; λ_c 、 θ 是可训练的参数.

将结果输入到sigmoid函数中得到最终的输出特征,即

$$F_{out} = \text{sigmoid}(F_B w_c). \quad (3)$$

通过式(3)即可实现特征图的通道自适应权重调整.

1.2 基于图网络的特征融合

目标在跟踪过程中极易出现极端形变,大多数的孪生网络目标算法在面对目标极端形变时效果较差,这是因为在当前孪生网络跟踪算法中几乎都是将目标所在区域视为整体,以此来计算这个整体与检测帧各个区域的相似性,将相似程度最高的检测帧区域作为目标的所在位置.为了应对目标在跟踪过程中发生较为极端形状变化的情况,提出一种以图网络为基础的计算模板帧与检测帧相关性得分的新方式,利用二分图建立模板帧目标与检测帧各区域的局部拓扑对应关系.将模板帧与检测帧的特征图分别表示为

F_z 、 F_x . 整个模板帧与检测帧计算相关性流程如图3所示.

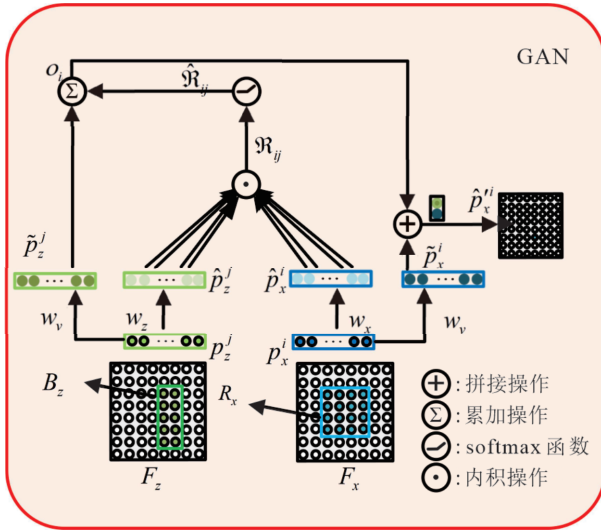


图3 图网络计算相关性示意图

为了更好地适应目标在后续跟踪阶段较为极端的形状变化,需要产生一个自适应变化的模板用于后续的跟踪. R_z 表示为模板帧中目标所在位置区域,并将此位置区域的特征 B_z 与检测帧区域 R_x 的特征 F_x 进行相似度计算,完成对目标的定位. 目标区域特征表示如下:

$$B_z = \begin{cases} F_z(\mu, \eta), & (\mu, \eta) \in R_z; \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

其中 (μ, η) 表示模板帧中目标所在位置区域坐标.

特征图 F_z 与 F_x 均可以表示为 $w \times h \times c$ 的形式, w 为特征图的宽, h 为特征图的高, c 为特征图通道数,特征图上的每一个 $1 \times 1 \times c$ 特征点作为图网络的每一个节点. 假设 M_z 为 B_z 在图网络中的节点集, $M_z = \{a | a = B_z(\mu, \eta), (\mu, \eta) \in R_z\}$; M_x 为 F_x 在图网络中的节点集, $M_x = \{b | b = F_x(\mu, \eta), (\mu, \eta) \in R_x\}$. 根据二部图的定义,构造完全二部图 $G(M, T)$,以此来计算模板帧目标区域节点与检测帧区域节点的相似性,定义 $M = M_x \cup M_z$, $T = ((u, v) | \forall u \in M_z, \forall v \in M_x)$. 为了将模板帧节点与检测帧节点区分开,进一步定义上述二部图的两个子图,分别是模板帧特征节点集 $G_z = (M_z, \emptyset)$ 与检测帧特征节点集 $G_x = (M_x, \emptyset)$. 对于每一组节点 (i, j) , $i \in G_x, j \in G_z$.

当检测帧中的区域与模板帧中目标区域越相似时越有可能是前景,并且同时应该向该前景区域传递更多的目标信息. 检测帧与模板帧的相关性得分与两个节点的特征相似度成正比,并使用特征向量的内积作为相似性度量的计算依据. 为了更好地自适应学习节点之间的相似性度量,首先对节点特征进行如

下线性变换:

$$\begin{cases} \hat{p}_x^i = w_x p_x^i, \\ \hat{p}_z^j = w_z p_z^j. \end{cases} \quad (5)$$

其中: w_x 、 w_z 分别表示对检测帧特征与模板帧特征进行线性变换的矩阵,通过 1×1 的卷积操作得到线性变化的权重矩阵; p_x^i 、 p_z^j 分别表示检测帧特征图节点 i 的特征向量与模板帧特征图节点 j 的特征向量; p_x^i 、 p_z^j 表示经过线性变换后的特征向量.

用线性变换后的特征向量的内积来表示检测帧与模板帧的相似性得分 \mathfrak{R}_{ij} ,并将相关性得分表示如下:

$$\mathfrak{R}_{ij} = g(\hat{p}_x^i, \hat{p}_z^j) = (\hat{p}_x^i)^T \cdot (\hat{p}_z^j). \quad (6)$$

为了更好地表示相关性分数,对 \mathfrak{R}_{ij} 进行softmax计算,有

$$\hat{\mathfrak{R}}_{ij} = \frac{e^{\mathfrak{R}_{ij}}}{\sum_{k \in M_z} e^{\mathfrak{R}_{ik}}}. \quad (7)$$

以节点 j 为基础, $\hat{\mathfrak{R}}_{ij}$ 可表示在当前跟踪算法中节点 j 应该给予节点 i 的关注程度. 利用从 M_z 中的所有节点依次传递到 M_x 中的第 i 个节点的关注度,计算节点 i 的聚合,表示如下:

$$\begin{aligned} \tilde{p}_z^j &= w_v p_z^j, \\ o_i &= \sum_{j \in M_z} \hat{\mathfrak{R}}_{ij} \tilde{p}_z^j, \end{aligned} \quad (8)$$

其中 w_v 为线性变换矩阵,同样通过 1×1 的卷积实现.

最后,将聚合的特征 o_i 与节点的特征 p_x^i 进行拼接操作融合,以此来增强特征表示:

$$\begin{aligned} \tilde{p}_x^i &= w_v p_x^i, \\ \hat{p}_x^i &= \text{ReLU}(\text{cat}(o_i, \tilde{p}_x^i)), \end{aligned} \quad (9)$$

其中 $\text{cat}()$ 为向量之间的叠加,产生后续特征响应图.

1.3 分类与回归

在本算法的结构中有两个分支,分别是分类分支与回归分支,分类分支的作用是完成目标与背景的分类,回归分支的作用是完成目标边界框的回归.

1.3.1 分类损失——IoU感知损失

孪生网络的分类与回归是通过不同的损失函数进行优化迭代,这导致分类任务与回归任务在训练过程中不一致,通常表现在分类得分高的正样本回归生成的预测框并不一定最接近真实框. 为了解决该问题, SiamFC++与SiamCAR采用的是增加中心度分支的方法. 但是这样的方法在整个算法运行过程中无疑增加了算法的计算量与参数量. 因此,本文在分类分支损失函数的基础上改变了原有分类

目标的定义方式——Ground-truth (正样本为1, 负样本为0), 而采用一种IoU感知方式, 将分类得分作为分类目标, 并将此作为后续训练的训练目标. 这种方式可以有效地将回归分支与分类分支通过IoU联系起来. 具体操作方法是原Ground-truth中的正样本由1替换为预测框 P 与真实框 G 之间的交并比, 表示为 $\text{IoU}(P, G)$, 并将该值定义为IoU感知分类得分(IoU-aware classification score, IACS). 负样本按照原Ground-truth定义方法将其定义为0. IACS可以帮助算法从回归候选框中选择出最准确的边界框, 并通过IACS将分类分支与回归分支建立联系, 有效避免分类任务与回归任务训练不一致的问题. 针对IACS分类损失, 损失函数采用 varifocal_loss^[16] 损失函数, 如下所示:

$$\text{cls_LOSS} = f_{\text{VFL}}(q, \text{IoU}(G, P)). \quad (10)$$

其中: q 定义为 IACS 分数. f_{VFL} 代表 varifocal_loss 损失函数, 定义为

$$f_{\text{VFL}}(q, \bar{q}) = \begin{cases} -\bar{q}(\bar{q} \log(q) + (1 - \bar{q}) \log(1 - q)), & q > 0; \\ -\alpha q^\beta \log(1 - q), & q = 0. \end{cases} \quad (11)$$

\bar{q} 代表目标分数. 对于正样本点, \bar{q} 代表预测框 P 与真实框 G 之间的交并比; 对于负样本点, $q = 0$. 其中 α 、 β 的值参考文献[15], 选择值 $\alpha = 0.75$, $\beta = 2.0$.

1.3.2 回归损失——CIoU

孪生网络目标跟踪算法是通过分类分支得到前景与背景的分类, 并在正样本点进行边界框的回归. 本文所提出的目标跟踪算法借鉴了FCOS算法的相关思想, 在正样本点直接回归边界框. 具体内容如图4所示.

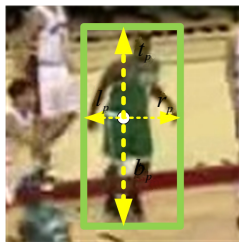


图4 边界框回归示意

在进行回归计算时, 由于 $\{l, t, r, b\}$ 均代表距离, 并可以采用每个变量单独回归, 直接使用均方差分别对4个距离值进行回归. 但是如果采用这种回归方式, 则忽略了整个回归边界框的完整性与4个距离值之间的内部关系. 因此提出交并比损失(IoU损失), 通过IoU损失可以充分地考虑到预测边界框与真实边界框之间的联系. IoU损失是一种常用的衡量两个形状之间的相似性度量的一种方法, 尤其IoU损失可微性的特点, 使其可以直接作为损失函数被优化. 其中最简单的是线性IoU损失, 其表达式为

$$\text{IoU_LOSS} = 1 - \text{IoU}. \quad (12)$$

随着对IoU损失认识的逐渐加深, 更多性能更好的IoU损失被提出, 例如GIoU损失^[17]、DIoU损失^[17]、CIoU损失^[17]等. 其中GIoU损失为避免在线性IoU损失中出现真实框与预测框两不相交而导致梯度无法回传的问题, 在损失函数中特意增加了预测框与真实框所包围的最小区域, 将该区域面积除以在该区域中除去两框面积后的剩余面积作为新的一项加在线性IoU后面. GIoU损失示意图如图5(a)所示, 计算公式如下:

$$\text{GIoU_LOSS} = 1 - \text{IoU} + \frac{S - P \cup G}{S}, \quad (13)$$

其中 S 代表预测框 P 与真实框 G 所围成的最小矩形面积. 但是上面所描述的GIoU损失仍旧存在问题, 当预测框与真实框存在包含关系时, GIoU损失退化成了线性IoU损失. GIoU损失通过预测框向真实框的不断靠拢来增加二者的重叠区域, 以此来实现预测框与真实框重叠的目的, 但是这种方式迭代速度相对较慢. DIoU损失引入了真实框与预测框中心点的欧氏距离, 同时为预测框的移动提供方向. 与GIoU损失不同的是, DIoU损失直接最小化了两框之间的距离, 使得收敛速度加快. DIoU损失示意图如图5(b)所示, 计算公式如下:

$$\text{DIoU_LOSS} = 1 - \text{IoU} + \frac{d}{d_A^2}. \quad (14)$$

其中: d 表示预测框与真实框中心点之间的欧几里得距离, d_A 表示预测框与真实框所包含最小矩形的对角线.

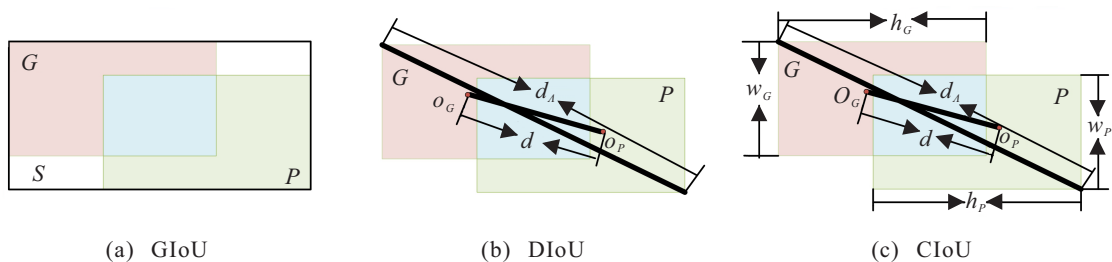


图5 不同回归损失函数示意图

CIoU与GIoU、DIoU相比不仅同样具有重叠面积、中心点之间的距离,更是含有真实框与预测框的纵横比.其中CIoU与GIoU相比增加了中心度距离惩罚项 $\mu\psi$,通过该项可以使目标框与真实框以更快速度接近,减少迭代次数.CIoU损失与其他两种IoU损失相比增加了形状比例惩罚项,可以使真实框与预测框之间的纵横比更加接近.本文算法中选择CIoU损失作为回归损失的损失函数.CIoU损失示意图如图5(c)所示,计算公式如下:

$$\begin{cases} \text{CIoU_LOSS} = 1 - \text{IoU} + \frac{d}{d_A^2} + \mu\psi, \\ \mu = \frac{4}{\pi^2} \left(\arctan \frac{w_G}{h_G} - \arctan \frac{w_P}{h_P} \right), \\ \psi = \frac{\mu}{1 - \text{IoU} + \mu}. \end{cases} \quad (15)$$

表1 OTB100数据集消融实验结果

base	图注意力	通道自适应调整	线性IoU损失	CIoU损失	交叉熵损失	IoU感知	精度	成功率
✓			✓		✓		0.656	0.871
✓	✓		✓		✓		0.683	0.897
✓	✓	✓	✓		✓		0.690	0.902
✓	✓	✓		✓	✓		0.697	0.910
✓	✓	✓		✓		✓	0.703	0.921

从表1可以看出:在基础算法上增加图注意力为算法带来了精度2.7%与成功率2.6%的提升;在加入图注意力的基础上,再在特征提取网络的每个残差块后面引入通道自适应调整模块,提升特征提取网络的提取能力,直接带来0.7%的精度提升与0.5%成功率的提升;在回归损失的选择上选用CIoU代替原来的线性IoU,得到了更加精准的边界框,给算法带来0.7%的精度提升与0.8%的成功率提升;选用基于IoU感知的分类损失函数保证了算法在训练与推理过程中的一致性,相对于原采用的交叉熵损失可以带来0.6%的精度提升与1.1%的成功率提升.

2.3 定量实验

在4个常用公开数据集OTB100、VOT2018、VOT2019、UAV123上与其他优秀的目标跟踪算法进行性能对比,下面为实验结果与分析.

2.3.1 OTB100数据实验

为了验证本文算法的有效性,用OTB100数据集进行检验,与当前流行的目标跟踪算法SiamRPN++, DaSiamRPN、SiamFC、SiamCAR、SiamBAN、TADT^[18]、MDNet^[19]、CFNet^[20]进行对比分析,结果如图6和图7所示.

2 实验结果与分析

为验证本文算法的有效性,实验采用OTB100、VOT2018、VOT2019、UAV123四个数据集进行实验验证.

2.1 实验平台

本文算法所做实验均在一台装有一张Nvidia GTX 3080TiGPU的主机上进行,处理器为Intel core (TM) i7-8700K,处理器的主频为3.70 GHz,操作系统为64的Ubuntu16.04,编程环境为python3.7,深度学习框架为PyTorch.

2.2 消融实验

为验证本文算法所加各个模块的有效性,在OTB100数据集上对各个模块分别进行消融实验,实验结果如表1所示.

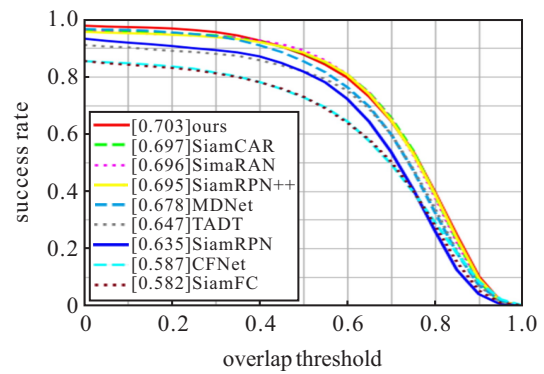


图6 不同算法在OTB100上的成功率

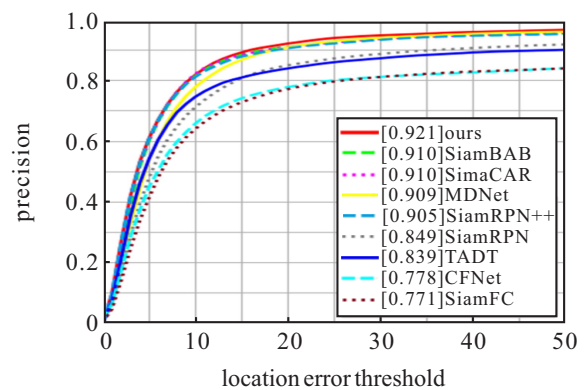


图7 不同算法在OTB100上的精度

从图6和图7可以看出,无论精度还是成功率,本文算法均优于其他算法.在成功率方面,本

文算法比同为 Anchor-Free 孪生网络目标跟踪算法的 SiamBAN 和 SiamCAR 分别高出 0.6% 和 0.7%; 与 Anchor-Base 的孪生网络目标跟踪算法 SiamRPN++、SiamRPN、SiamFC 相比分别高出 0.8%、6.8%、20.1%。在精度方面, 本文算法比除孪生网络之外的其他目标跟踪算法 MDNet、CFNet、TADT 分别高出 1.2%、14.3%、8.2%。在本文所列举的 8 种对比目标跟踪算法中, SiamBAN 和 SiamCAR 在精度方面效果最好, 为 0.910, 而本文算法比 SiamBAN 和 SiamCAR 高出 1.1%。

2.3.2 VOT2018数据实验

当前孪生网络目标跟踪算法在面对剧烈形变、大尺度变化等跟踪难点时, 表现出性能差、丢帧次数多等问题。基于此, 本文引入目标感知来感知目标的变化。为了更好地与目标感知相配合, 采用图网络相关知识来计算模板帧与检测帧的相似性, 以此完成后续的跟踪任务。同时, 精简特征提取网络, 通过在特征提取网络中引入通道自适应调整模块, 提升了特征提取的效果。将本文算法与当前具有代表性的算法 SiamFC、SiamRCNN^[21]、ATOM、SiamMargin^[22]、SiamRPN++、SiamDW、DaSiamRPN、UpdataNet^[23]、SiamMask、SiamCAR 进行比较, 实验结果如表 2 所示。

表 2 在 VOT2018 数据集的实验结果对比

tracker	A	R	EAO	LN	V _{FPS}
SiamFC	0.503	0.585	0.187	101	101
SiamRCNN	0.612	0.220	0.405	47	15
ATOM	0.590	0.204	0.401	44	30
SiamMargin	0.583	0.197	0.432	42	62
SiamRPN++	0.601	0.234	0.415	50	35
SiamDW	0.538	0.398	0.270	150	150
DaSiamRPN	0.586	0.276	0.383	59	160
UpdateNet	0.587	0.276	0.393	59	90
SiamMask	0.609	0.281	0.381	60	35
SiamCAR	0.574	0.197	0.423	—	35
ours	0.583	0.187	0.447	40	125

从表 2 的实验数据可以分析得出: 在 VOT2018 数据集的 EAO 方面, 本文算法排名第一, 与 SiamFC、SiamRCNN、ATOM、SiamMargin、SiamRPN++、SiamDW、DaSiamRPN、UpdataNet、SiamMask、SiamCAR 相比分别高出 26%、4.2%、4.6%、1.5%、0.3%、17.7%、6.4%、5.4%、6.6%、2.4%。与其他算法相比, 本文算法有着更高的运行速度, 这是因为本文算法相对于 SiamRPN++、SiamCAR 等, 采用了更加简洁的特征提取网络, 选用改进的 Resnet50, 在缩减原特征网络参数量、计算量的同时尽可能地保证特征提取效果。本文算法整体运行速度为 125FPS, 大约是 SiamCAR 运行速度的 3.57 倍, 已经完全达到实

时性的要求。采用基于图注意力相关的模板帧与检测帧的计算方式使得本文算法在面对跟踪物体较大形变时仍具有较好的跟踪效果, 并且优于上述算法中以 SiamRPN++ 为代表的大部分孪生网络目标跟踪算法。同时, 本文算法在精度与鲁棒性方面同样优于上述所提的大部分目标跟踪算法。由此可见, 本文算法在 VOT2018 数据集上取得了较好的结果, 且就其运动速度方面而言, 本文算法做到了运行速度与 EAO 较好的平衡。

2.3.3 VOT2019数据实验

将本文算法在 VOT2019 上进行测试与评估, 选取 SiamRPN++、SiamMask、ATOM、DCFST^[24]、SiamDW-ST^[24]、SiamCRE-RT^[24]、ARTCS^[24]、SPM^[24] 八种跟踪算法进行比较, 并且使用与 VOT2018 同样的评价指标对上述跟踪算法进行比较, 实验结果如表 3 所示。

表 3 在 VOT2019 数据集的实验结果对比

tracker	A	R	EAO	LN	V _{FPS}
SiamRPN++	0.599	0.482	0.285	96	35
SiamMask	0.594	0.461	0.287	92	35
ATOM	0.579	0.557	0.240	111	30
DCFST	0.585	0.376	0.317	75	—
SiamDW-ST	0.600	0.467	0.299	93	150
SiamCRF-RT	0.549	0.346	0.262	69	—
ARTCS	0.602	0.482	0.287	96	—
SPM	0.577	0.507	0.275	101	120
ours	0.588	0.421	0.321	84	128

根据表 3 的结果分析可知: 在 VOT2019 数据集的 EAO 方面, 本文算法在上述 9 种算法中排名第一, 比其他的目标跟踪算法高出 0.4%~8.1% 不等。由此可见, 本文算法在跟踪算法的综合评价指标方面取得了较好的结果。在精度方面, 本文算法也超过了上述算法中的大部分算法, 相较 DCFST、SPM、SiamCRF-RT、ATOM 分别高出 0.3%、1.1%、3.9%、0.9%。在算法鲁棒性方面, 本文算法也达到了中等偏上的水平, 与 SiamDW-ST、ARTCS、SiamMask、SiamRPN++、SPM、ATOM 相比分别低了 4.6%、6.1%、4%、6.1%、8.6%、13.6%。在实时性方面, 本文算法早已超过算法实时性的要求, 且与其他几种算法相比有较大的优势。

2.3.4 UAV123数据实验

将本文算法与 SiamBAN、Ocean、CGACD^[25]、SiamCAR、SiamRPN、SiamRPN++、SiamDW 七种算法在 UAV123 数据集进行实验对比, 以精度和成功率作为评价指标, 实验结果如图 8 和图 9 所示, 可以看出本文算法无论是在精度还是在成功率方面均排名第一。

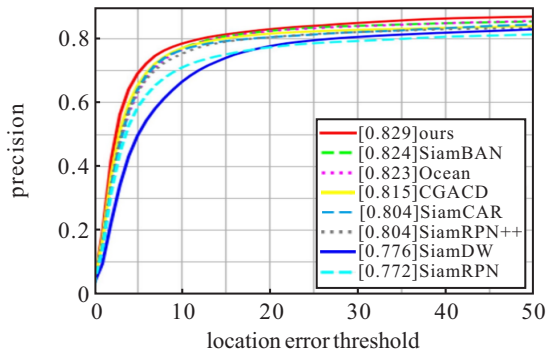


图8 不同算法在UAV123上的精度

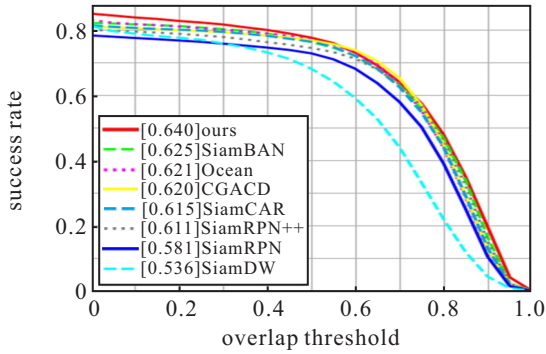


图9 不同算法在UAV123上的成功率

UAV123数据集共12种挑战属性,将上述算法在UAV123上的12种属性的精度和成功率的测试结果绘制如图10和图11所示雷达图。

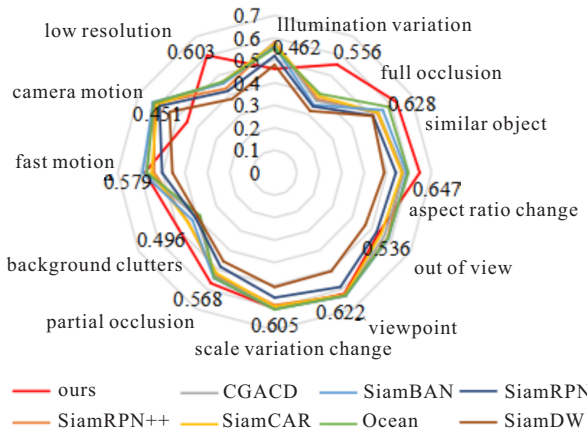


图10 不同算法在UAV123上不同属性的成功率雷达图

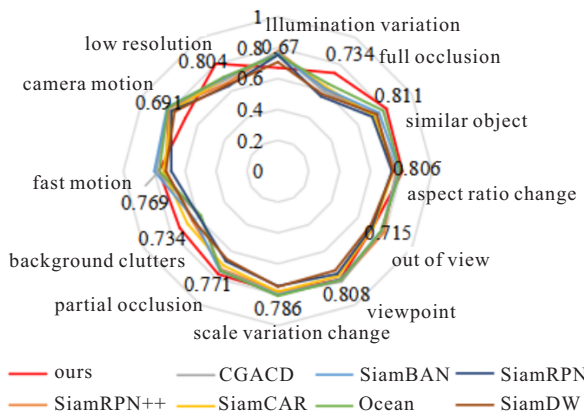


图11 不同算法在UAV123上不同属性的精度雷达图

从图10和图11可以看出,本文算法的成功率和精度在 full occlusion、similar object、aspect ratio change、partial occlusion、background clutters、low resolution等方面均超越其他7种算法,其他应用挑战中即使不是最好的,也基本保持前3. UAV123中的视频都是在飞行器上并以飞行器的视角进行拍摄的,大部分的目标物体比较小,但是本文算法相比于其他算法依然有较好的效果,这说明所提出的基于图网络的模板帧与检测帧相似度计算方式能够较好应对小目标物体的跟踪任务。

3 基于大疆Manifold 2-G的应用与部署

为了进一步研究本文算法的实用性,将本文算法移植到大疆Manifold 2-G嵌入式平台上.移植后的算法在面对目标跟踪中的一些难点问题,依旧可以表现出良好的鲁棒性.在整个实验过程中,选用CPU+GPU的异构部署模式.GPU占用率保持在99.0%左右,六核CPU的平均资源占用率为28.1%,并没有出现系统崩溃或者过热的情况.这代表移植后的算法可以在Manifold 2-G嵌入式平台平稳有效地运行,为目标跟踪算法的落地部署与应用提供参考.本文算法仍旧存在一些问题,例如算法的运行速度仅为5.7FPS,这是由算法参数量大造成的.为了实现实时性,在后续研究过程中将考虑采用更加强轻量级的特征提取网络,或者使用加速器对代码实行加速。

4 结论

本文所提出的孪生网络目标跟踪算法是以Anchor-free的孪生网络目标跟踪算法为基础提出的,主要由目标特征提取模块、特征融合模块、分类回归模块3部分组成.在特征提取模块中选择以Resnet50为基础,采用Resnet50的第4层特征为最后的特征输出,并引入通道自适应调整结构增强通道特征,将算法速度与同类型算法SiamCAR相比提升3~4倍.在特征融合时,为了避免采用互相关或者深度互相关等全局匹配的方式,设计了一种基于图网络计算模板帧与检测帧相似性的方式,使得算法在EAO方面提升2%~3%.通过在分类分支中采用IoU感知的分类损失函数,有效地在分类分支与回归分支之间建立联系,保证了模型推理与训练的一致性.在回归分支中选择CIoU损失函数的方式实现训练阶段边界框的精准回归.将本文算法在常用公开数据集上进行实验,取得了一定效果。

参考文献(References)

[1] 郭林, 刘冉, 蓝发籍, 等. 基于激光与UWB序列匹配的目标跟踪[J]. 控制与决策, 2024, 39(8): 2613-2621.

- (Guo L, Liu R, Lan F J, et al. Object tracking based on sequence matching between UWB and LiDAR[J]. Control and Decision, 2024, 39(8): 2613-2621.)
- [2] Bertinetto L, Valmadre J, Henriques J F, et al. Fully-convolutional Siamese networks for object tracking[C]. European Conference on Computer Vision. Cham: Springer, 2016: 850-865.
- [3] Li B, Yan J J, Wu W, et al. High performance visual tracking with Siamese region proposal network[C]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, 2018: 8971-8980.
- [4] Danelljan M, Robinson A, Khan F S, et al. Beyond correlation filters: Learning continuous convolution operators for visual tracking[C]. European Conference on Computer Vision. Cham: Springer, 2016: 472-488.
- [5] Yang K, Song H H, Zhang K H, et al. Deeper Siamese network with multi-level feature fusion for real-time visual tracking[J]. Electronics Letters, 2019, 55(13): 742-745.
- [6] Li B, Wu W, Wang Q, et al. SiamRPN: Evolution of Siamese visual tracking with very deep networks[C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, 2020: 4277-4286.
- [7] Wang Q, Zhang L, Bertinetto L, et al. Fast online object tracking and segmentation: A unifying approach[C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, 2020: 1328-1338.
- [8] 刘如浩, 张家想, 金辰曦, 等. 基于可变形卷积的孪生网络目标跟踪算法[J]. 控制与决策, 2022, 37(8): 2049-2055.
(Liu R H, Zhang J X, Jin C X, et al. Target tracking based on deformable convolution Siamese network[J]. Control and Decision, 2022, 37(8): 2049-2055.)
- [9] Xu Y D, Wang Z Y, Li Z X, et al. SiamFC++: Towards robust and accurate visual tracking with target estimation guidelines[J/OL]. 2019, arXiv: 1911.06188.
- [10] Guo D Y, Wang J, Cui Y, et al. SiamCAR: Siamese fully convolutional classification and regression for visual tracking[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 6268-6276.
- [11] Chen Z D, Zhong B N, Li G R, et al. Siamese box adaptive network for visual tracking[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 6667-6676.
- [12] Zhang Z P, Peng H W, Fu J L, et al. Ocean: object-aware anchor-free tracking[C]. European Conference on Computer Vision. Cham: Springer, 2020: 771-787.
- [13] Danelljan M, Bhat G, Khan F S, et al. ATOM: Accurate tracking by overlap maximization[J/OL]. 2018, arXiv: 1811.07628.
- [14] Bhat G, Danelljan M, Van Gool L, et al. Learning discriminative model prediction for tracking[C]. 2019 IEEE/CVF International Conference on Computer Vision. Seoul, 2020: 6181-6190.
- [15] Liu Y C, Shao Z R, Teng Y Y, et al. NAM: Normalization-based attention module[J/OL]. 2021, arXiv: 2111.12419.
- [16] Zhang H Y, Wang Y, Dayoub F, et al. VarifocalNet: an IoU-aware dense object detector[C]. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville, 2021: 8510-8519.
- [17] Rezatofghi H, Tsoi N, Gwak J Y, et al. Generalized intersection over union: Ametricanda loss for bounding box regression[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, 2019: 658666.
- [18] Li X, Ma C, Wu B Y, et al. Target-aware deep tracking[C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, 2020: 1369-1378.
- [19] Nam H, Han B. Learning multi-domain convolutional neural networks for visual tracking[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016: 4293-4302.
- [20] Valmadre J, Bertinetto L, Henriques J, et al. End-to-end representation learning for correlation filter based tracking[C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, 2017: 5000-5008.
- [21] Voigtlaender P, Luiten J, Torr P H S, et al. Siam R-CNN: Visual tracking by re-detection[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 6577-6587.
- [22] Wang Q, Zhang L, Bertinetto L, et al. Fast online object tracking and segmentation: A unifying approach[C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, 2020: 1328-1338.
- [23] Zhang L C, Gonzalez-Garcia A, Van De Weijer J, et al. Learning the model update for Siamese trackers[C]. 2019 IEEE/CVF International Conference on Computer Vision. Seoul, 2020: 4009-4018.
- [24] Wang Q, Gao J, Xing J L, et al. DCFNet: Discriminant correlation filters network for visual tracking[J/OL]. 2017, arXiv: 1704.04057.
- [25] Du F, Liu P, Zhao W, et al. Correlation-guided attention for corner detection based visual tracking[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 6835-6844.

作者简介

陈志旺(1978—), 男, 副教授, 博士, 硕士生导师, 从事运动物体目标检测与跟踪等研究, E-mail: czwaaron@ysu.edu.cn;

刁华康(1996—), 男, 硕士, 从事计算机视觉中目标跟踪的研究, E-mail: diaohk2022@163.com;

袁宇(1996—), 男, 硕士, 从事计算机视觉中目标检测的研究, E-mail: 1448366692@qq.com;

吕昌昊(1996—), 男, 硕士, 从事智能电网的优化运行的研究, E-mail: 316998054@qq.com;

彭勇(1963—), 男, 教授, 博士, 从事生物调控、脑电应用科学等研究, E-mail: PY81@sina.com.