计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
[doi: 10.15888/j.cnki.csa.009744] [CSTR: 32024.14.csa.009744]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于孪生网络的高效无人机目标跟踪①

王建浩, 叶　明, 姚佳烽

(南京航空航天大学 机电学院, 南京 210016)
通信作者: 叶　明, E-mail: yeming5@nuaa.edu.cn

摘　要: 在视觉跟踪领域, 大多数基于深度学习的跟踪器过分地强调精度, 而忽视了算法速度. 因此, 这些算法在移动平台上的部署 (无人机), 受到了阻碍. 在本文中, 提出了一种基于 Siamese 的深度交叉指导跟踪器 (SiamDCG). 为了更好地在边缘计算设备上部署, 在 MobileNetV3-small 的基础上设计了独特的 backbone 结构. 此外, 针对无人机场景的复杂性, 传统使用狄拉克 δ 分布预测目标框的方式有很大的弊端, 为了克服边界框存在的模糊效应, SiamDCG 将回归框分支转为预测偏移量的分布, 并且用学习到的分布去指导分类的准确性. 在多个无人机 benchmark 上的优秀表现, 都显示了其鲁棒性与高效性. 在 Intel i5 12 代 CPU 上, SiamDCG 运行速度是 SiamRPN++的 167 倍, 使用的参数仅为它的 1/98, FLOPs 是 1/410 .

关键词: 目标跟踪; 轻量级网络; 孪生网络; 无人机

引用格式: 王建浩,叶明,姚佳烽.基于孪生网络的高效无人机目标跟踪.计算机系统应用. http://www.c-s-a.org.cn/1003-3254/9744.html

Efficient Tracking for UAVs Based on Siamese Network

WANG Jian-Hao, YE Ming, YAO Jia-Feng

(College of Mechanical & Electrical Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China)

Abstract: In the field of visual tracking, most deep learning-based trackers overemphasize accuracy while overlooking efficiency, thereby hindering their deployment on mobile platforms such as drones. In this study, a deep cross guidance Siamese network (SiamDCG) is put forward. To better deploy on edge computing devices, a unique backbone structure based on MobileNetV3-small is devised. Given the complexity of drone scenarios, the traditional method of regressing target boxes using Dirac δ distribution has significant drawbacks. To overcome the blurring effects inherent in bounding boxes, the regression branch is converted into predicting offset distribution, and the learned distribution is used to guide classification accuracy. Excellent performances on multiple aerial tracking benchmarks demonstrate the proposed approach's robustness and efficiency. On an Intel i5 12th generation CPU, SiamDCG runs 167 times faster than SiamRPN++, while using 98 times fewer parameters and 410 times fewer FLOPs.

Key words: object tracking; lightweight network; Siamese network; unmanned aerial vehicle (UAV)

1　引言

目标跟踪是计算机视觉的基本任务之一. 由于无人机 (UAV) 具有优越的综合性能, 针对 UAV 视角的空中跟踪已经引起了相当多的关注, 如航空摄影[1]、目标跟踪[2]、室内避障[3]中的位置. 然而与一般跟踪任务不同的是, 由于空中计算资源有限, UAV 在空中的跟踪算法, 还需要考虑功耗以及算法实时性的问题. 此外针对 UAV 视频的跟踪需要面对一些严峻的挑战, 如快速运动[4]、低分辨率[5]、遮挡[6]等. 目前关于目标跟踪的主要方式有两种, 对 CPU 友好的基于相关滤波器

① 基金项目: 国家自然科学基金面上项目 (62271251)
收稿时间: 2024-06-17; 修改时间: 2024-07-10; 采用时间: 2024-08-01; csa 在线出版时间: 2024-11-15

(discriminative correlation filter, DCF) 跟踪器[7–9], 以及对于 GPU 友好的基于深度学习 (deep learning, DL) 的方法[10–12]. 由于 UAV 端侧的计算能力有限, 实际中 DCF 跟踪器由于其高效率被广泛应用, 然而基于 DCF 的跟踪器精度无法满足复杂的 UAV 的高空视频场景. 另一方面, 随着研究者们的深入, 为了获得高精度, 基于 DL 的跟踪器变得愈发复杂, 难以满足端侧的实时性的要求.

为了解决这些问题, 本文提出了一个基于孪生网络的 anchor-free 网络 SiamDCG (deep cross guidance Siamese network). 为了减少模型的参数, 并且让模型对 CPU 更加友好, 选择用 MobileNetV3-small[13]作为 backbone. 此外为了实现模型的轻量化, 我们只把跟踪问题分为两个问题: 分类与框回归. 该算法通过分类分支来预测前/背景, 用回归分支来预测精准的回归框. 传统认为的预测框是 Dirac δ 分布, 即是确定的, 然而面对实际的跟踪场景, 特别是高空视频中存在的边界模糊问题, 这样的假设是有局限性的, 因此我们让网络的回归分支去学习边界框的分布, 并且将回归分支学习到的这种分布去指导分类质量, 克服了传统的边界框回归方法的局限性. 在公开数据集上验证了我们的模型的高效性, 如图 1 所示, 其中, 圆的直径与模型的参数大小成正比. 在无人机数据集 DTB70[14]上 SiamDCG 的 Precision 超过了 SiamRPN++[15], SiamFC++[16]等基于 CNN 的大型孪生网络, 以及基于 Transformer 结构的 TCTrack[6], SiamAPN++[5]等网络, 比 SiamRPN++高 2.8%, 同时使用的参数量是 SiamRPN++的 1/98, FLOPs 是其 1/410.

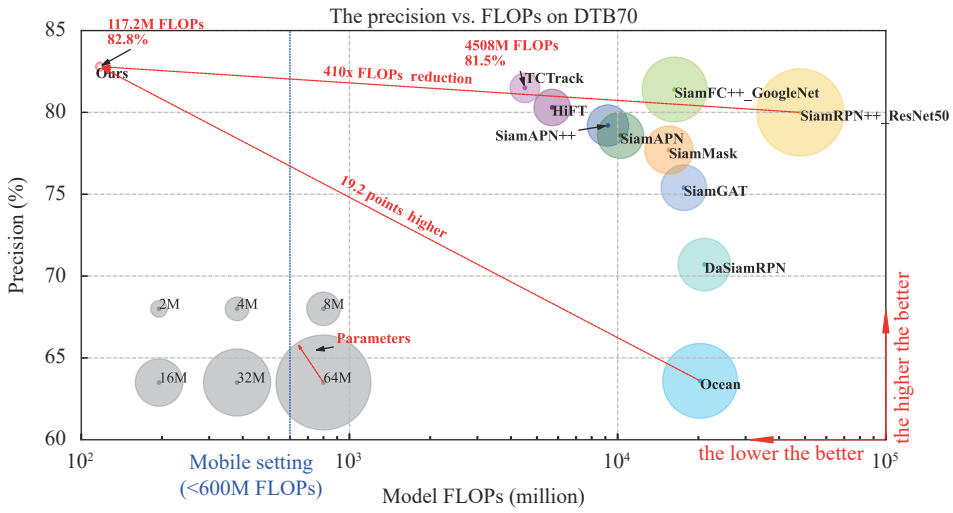


图 1 SiamDCG 与一些 SOTA 跟踪器在 DTB70 [14]上的 Precision 对比

2 基于孪生网络的 DL 单目标跟踪网络

近年来基于孪生网络的跟踪网络以其端到端的训练能力和高效率受到学界的广泛关注[15,22–24]. SiamFC[22]首先采用孪生网络作为特征提取的 backbone 部分, 并且引入了最开始的互相关的操作, 由于 SiamFC 简单的设计, 在精确度以及速度方面成为当时的 SOTA (state-of-the-art). 而因此引起了研究者们对于孪生跟踪网络的关注, 此后基于预设锚框的跟踪器显现出了更好的性能, SiamRPN[23]是最早引入预设锚框的跟踪器, 该算法通过引入区域建议网络 (RPN), 以此来应对不同尺度的目标, SiamRPN++[15]通过更强大的特征提取网络 (ResNet50) 进一步提高了性能.

而与上述基于锚框的跟踪器不同的是, SiamBAN[25], Ocean[21], SiamFC++[16]这些 anchor-free-based 方法不需要人为的预先设计先验框, 因此没有预设锚框这部分的超参数. 受目标检测领域 FCOS[26]的无锚框设计的影响, ARCF[8]引入了一个额外的质量评估模块以达到更高的精度, 但是这个额外的分支需要独立训练, 并且在推理阶段也需要结合其他分支, 这反而导致了训练与推理的不一致.

为了减少模型的复杂度, SiamDCG 在特征融合后直接分为 classification 和 regression 两个分支. 但是目前大部分的跟踪器都广泛认为目标框是 Dirac δ 分布[15,16,18,19,21], 因此在一些特定的场景, 比如说低分辨率, 相机移动等难跟踪的场景, 基于 anchor-free 的跟踪器很难学习到困难样本, 因此本文借鉴 GFL[27]的思想, 将目标回归框预测分支改为去学习目标框的概率分布,

这样增加了边界框回归的灵活性.

3 方法

图 2 显示了 SiamDCG 的网络结构, 与之前的基于孪生网络的跟踪器一样主要由 3 部分构成: 特征提取, 特征融合以及预测头. 在这部分, 主要对特征提取的 backbone 以及头预测网络进行详细讲述, 特征融合模块, 本文沿用 SiamRPN++的 DW-Corr[15].

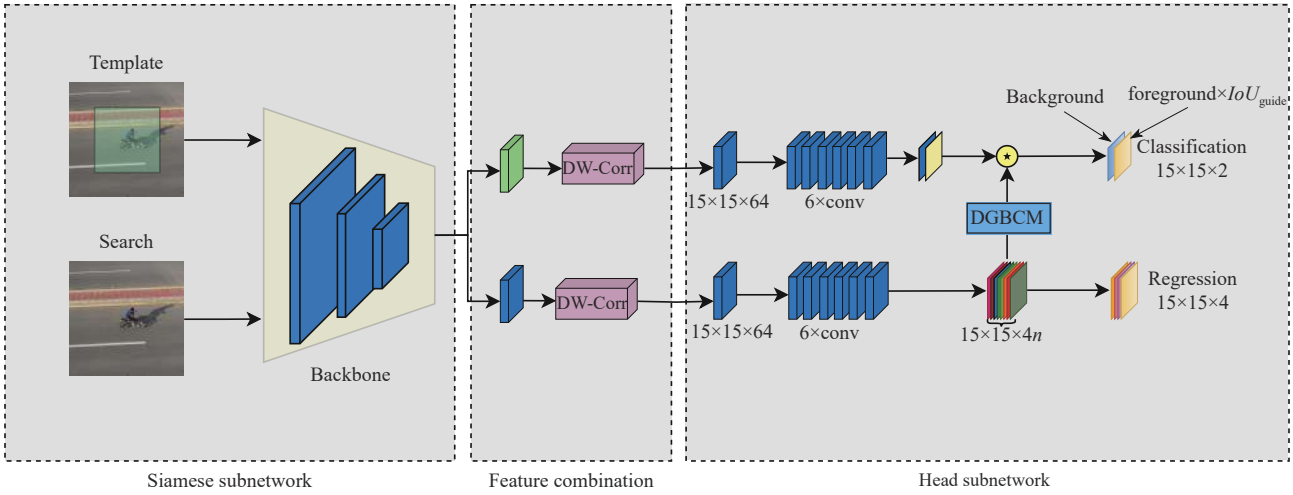


图 2 SiamDCG 整体结构

3.1 特征提取网络

为设计轻量化的更适合 CPU 计算的网络结构, 本文选择 MobileNetV3-small 作为 baseline. 在 SiamDW[28] 中作者分析了以前的基于孪生网络的跟踪器特征提取部分无法利用深度网络的优势的原因, 并给出了几点关于 backbone 的设计建议, 本文根据文献 [28]中给出的 guideline 在 MobileNetV3-small 的基础上设计了适合跟踪的轻量级 backbone.

1) 基于孪生网络跟踪器的特征提取网络步长不能太长, 表 1 中, SE 代表该块中是否存在 squeeze-and-excite, NL 表示所使用的非线性类型, 这里 HS 代表 h-swish, RE 表示 ReLU, NBN 表示没有 BN 归一化操作, s 表示步长. 从表 1 可以看出 MobileNetV3-small 的网络步长都为 1 或 2, 满足小步长的要求, 这样减小了定位的偏差.

2) 最后一层的感受野最好是 search 的 60%–80%, 因为大的感受野虽然增加了目标的上下文信息, 但是对于单目标跟踪而言, 反而减少了最重要的关于目标分类的判别信息和局部信息. 这里根据感受野的计算公式:

$$(N-1)_{\mathrm{RF}} = f(N_{\mathrm{RF}}, s, k) = (N_{\mathrm{RF}} - 1) \times s + k \qquad (1)$$

其中, RF是感受野, $(N-1)_{\mathrm{RF}}$ 是 N–1 层的感受野, s 是步长, k 是卷积核的大小.

可以计算出 Layer10 的感受野为 159. $159 \div 255 \approx 0.62 \in [0.6, 0.8]$满足要求, 因此这里取 Layer10 作为输出. 并且去除了最后一层的 padding 操作, 给出 backbone 的结构图, 如图 3 所示.

表 1 MobileNetV3-small 的结构表

Layer	Input	Operator	Exp size	#out	SE	NL	s
1	$224^2\times3$	Conv2d, 3×3	—	16	—	HS	2
2	$112^2\times16$	Bneck, 3×3	16	16	√	RE	2
3	$56^2\times16$	Bneck, 3×3	72	24	—	RE	2
4	$28^2\times24$	Bneck, 3×3	88	24	—	RE	1
5	$28^2\times24$	Bneck, 5×5	96	40	√	HS	2
6	$14^2\times40$	Bneck, 5×5	240	40	√	HS	1
7	$14^2\times40$	Bneck, 5×5	240	40	√	HS	1
8	$14^2\times40$	Bneck, 5×5	120	48	√	HS	1
9	$14^2\times40$	Bneck, 5×5	144	48	√	HS	1
10	$14^2\times40$	Bneck, 5×5	288	96	√	HS	2
11	$7^2\times96$	Bneck, 5×5	576	96	√	HS	1
12	$7^2\times96$	Bneck, 5×5	576	96	√	HS	1
13	$7^2\times96$	Conv2d, 1×1	—	576	√	HS	1
14	$7^2\times576$	Pool, 7×7	—	—	—	—	1
15	$1^2\times576$	Conv2d, 1×1, NBN	—	1 280	—	HS	1
16	$1^2\times1280$	Conv2d, 1×1, NBN	—	k	—	—	1

3.2 预测头

1) Regression 分支

与传统的 Dirac δ 分布不同, 在很多复杂环境中, 目

标回归出来的框具有很强的不确定性, 因此本文提出的算法让网络去学习边界框的概率分布, 而为了简化计算, 这里将概率分布直接预设为连续的离散值, 让网络去学习多个离散偏移量的分布. 首先对于在分类分支预测的前景 (目标) 的特征图中, 点(i,j)映射到搜索的原图中的坐标为$P_{i,j}=\left(p_x^{i,j},p_y^{i,j}\right)$, 目标的真实坐标为$G_{i,j}=\left(x_0^{i,j},y_0^{i,j},x_1^{i,j},y_1^{i,j}\right)$, 因此预测框相当于候选位置的距离表示为$O_{i,j}=\left(l^{i,j},t^{i,j},r^{i,j},b^{i,j}\right)$, 其中,

$$\begin{bmatrix} l^{i,j} \\ t^{i,j} \\ r^{i,j} \\ b^{i,j} \end{bmatrix} = \begin{bmatrix} p_x^{i,j}-x_0^{i,j} \\ p_y^{i,j}-y_0^{i,j} \\ x_1^{i,j}-p_x^{i,j} \\ y_1^{i,j}-p_y^{i,j} \end{bmatrix} \tag{2}$$

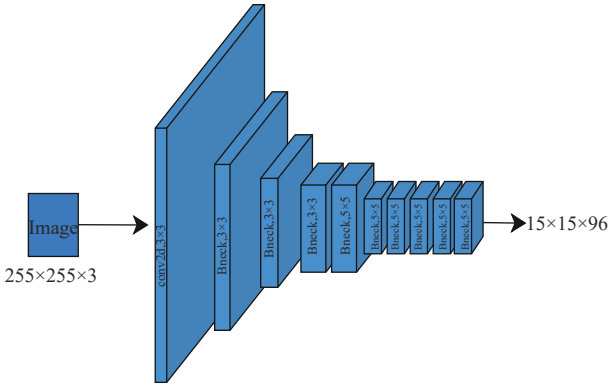


图 3　Backbone 结构图

通常而言, 在传统的 Dirac δ 分布中我们假定向量$O_{i,j}$某一个方向的分量为g, g的定义如式 (3) 所示:

$$g=\int_{-\infty}^{+\infty}\delta(x-g)\,x\mathrm{d}x \tag{3}$$

其中, x代表该方向上的所有可能的位置. 根据 Dirac δ分布的性质, 它必须满足$\int_{-\infty}^{+\infty}\delta(x-g)\,x\mathrm{d}x=1$的条件. 对于一个步长为$s$的跟踪器而言, 我们假设搜索区域的大小为$(W,H)$, 且目标可能分布在整个的搜索区域, 因此回归分支需要满足:

$$\begin{cases} y_{\max}=\max\left(O_{i,j}\right)/s \\ y_{\min}=\min\left(O_{i,j}\right)/s \end{cases} \tag{4}$$

其中, $\max\left(O_{i,j}\right)$应当为$\max(W,H)$的一半, 这里我们的网络的 search 大小为$W=H=255$, 因此$y_{\max}=127$, 而$\min\left(O_{i,j}\right)=0$.

因此我们设定的回归值的范围为$[y_{\min},y_{\max}]$. 为了方便起见, 在本文中我们假设离散的间距为 1, 则离散范围为 $\{y_{\min},y_{\min}+1,\cdots,y_{\max}-1,y_{\max}\}$. 本文的跟踪器$stride=16$, 因此离散的范围为$\{0,1,\cdots,8\}$. 效仿 Dirac δ 分布式, 将积分转为离散形式表示, 我们将向量g变化成式 (5) 所示:

$$g=\int_{y_{\min}}^{y_{\max}}I(x)\,x\mathrm{d}x=\sum_{i=0}^{n}I(y_i)\,y_i \tag{5}$$

其中, $I(x)$指的是离散范围内分布函数, n为离散值的数量.

为了将回归限制在$[y_{\min},y_{\max}]$的范围内, 分布函数应该满足$\sum_{i=0}^{n}I(y_i)=1$. 因此$I(y_i)$很容易能够通过$n+1$个 Softmax 层$S(\cdot)$来构成, 为了简单起见, 后面将$I(y_i)$表示为S_i. 而根据分布函数的特点, 将回归预测框的任务转为一系列的离散集合的概率预测问题. 从而在一定程度上规避了传统 Dirac δ 分布的局限性.

为了更好地预测这一离散的分布, 考虑到分布应当是集中在回归目标附近, 为了引导回归框更精准的预测出分布, 在文献 [27]中引入了 distribution focal loss (DFL), 让网络聚焦于最接近于标签y附近的y_i,y_i+1这两个分布值. 而由于边界框学习仅是在正样本中, 因此没有类不平衡的情况, 因此本文去掉了 focal loss[27] 中存在的缩放因子, 最终引入了式 (6) 所示交叉熵损失来指导这一分布:

$$L_{\mathrm{dfl}(S_i,S_{i+1})}=-((y_{i+1}-y)\log(S_i)+(y-y_i)\log(S_{i+1})) \tag{6}$$

为了提高包围盒回归的精度, 采用 CIoU[29]损失作为指导性度量. CIoU 包含 3 个关键因素, 即重叠区域、中心点距离和纵横比, 使得回归损失计算更加精确, 即使在与目标框重叠或包含的情况下也能加快收敛速度, 公式如下:

$$L_{\mathrm{CIoU}}=1-IoU+\frac{\rho^2(b,b^{\mathrm{gt}})}{c^2}+\alpha v \tag{7}$$

其中, b, b^{gt}分别表示预测框B和目标框B^{gt}的中心点, $\rho(\cdot)$是欧氏距离 (Euclidean distance), c是覆盖两个框的最小封闭 box 的对角线长度, α是平衡比例系数, v是用来衡量预测框和目标框之间的比例尺度一致性, α和v的计算公式如式 (8) 和式 (9) 所示:

$$\alpha=\frac{v}{(1-IoU)+v} \tag{8}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{\omega^{gt}}{h^{gt}} - \arctan\frac{\omega}{h}\right)^2 \tag{9}$$

其中, IoU代表预测框与目标框的 IoU 分数, w^{gt}, h^{gt}分别代表目标框的宽以及长, w, h分别代表预测框的宽以及长.

2) Classification 分支

受文献 [30]启发, 我们在网络中使用了一个分类-回归联合感知的分类策略, 本文称之为 Classification-IoU (C-IoU) 联合分类策略. 正如前文所提到的, 让网络学习边界框的分布而不仅是一个定值, 而这种分布对于分类的分支而言也是有指导意义的, 具体来说, 当预测到的定位质量较好的时候, 体现在预测框分布图中就是分布比较“尖锐”, 那么定位质量自然是比较高的, 反之如果分布是比较“平坦”则代表本身定位质量就较差, 文献中对于“尖锐”与“平坦”是通过计算 Top-k+mean 值得到的, 如式 (10) 所示. 因此文献中巧妙地将定位的质量分布联合分类感知, 最终在目标检测领域取得了一定的成功.

$$F = Concat(Topk(I^{\omega}), mean(Topk(I^{\omega}))) \tag{10}$$

其中, 因为回归的是 4 条边的偏移量, 这里将左, 右, 上和下的偏移记为$\{l,r,t,b\}$, 将ω边的离散分布为: $I^{\omega} = [I^{\omega}(y_0), I^{\omega}(y_1), \cdots, I^{\omega}(y_n)], \omega \in \{l,t,r,b\}$, 其中$F \in R^{4(k+1)}$代表基本统计特征, $Concat(\cdot)$代表通道连接, $Topk(\cdot)$就是选择分布的 Top-k值, $mean(\cdot)$代表几个 Top-k的均值.

因此本文在头预测模块加入了分布指导的二分类模块 (distribution-guided binary classification module, DGBCM), 用于将回归框分支学习到的一般分布I的统计信息传递到分类分支中, 具体如图 2 中的 DGBCM 所示. 而对于跟踪而言分类只有两类: 前景/背景, 由于背景区域处于一个没有边界的开放空间中, 因此本文中保持背景区域的特征向量即第 1 个通道的预测值不变, 而对于第 2 通道而言, 本来是用来预测前景的, 考虑到回归质量的联合预测, 将前景的预测值乘上分布质量, 这将减轻训练的困难, 提高分类的质量. DGBCM 的结构如式 (11) 所示:

$$cls_{\text{C-IoU}} = cls_{\text{foreground}} \otimes \sigma(W_2\delta(W_1 F)) \tag{11}$$

其中, F就是式 (10) 中提到的统计特征, W_1, W_2是两个点卷积核, $W_1 \in R^{p\times 4(k+1)}, W_2 \in R^{1\times p}$, k是 Top-k的参数, p代表隐藏层的通道尺寸 (在本文中$k=4, p=64$). 此外δ和σ分别代表 ReLU 和 Sigmoid, $cls_{\text{foreground}}$代表原来分类分支的第 2 通道 (用于预测前景的特征图), $cls_{\text{C-IoU}}$就是 Classification-IoU 的分布质量融合分类结果.

提出的 C-IoU 的分类分支需要整个训练集的图像参与训练, 因此存在类不平衡的问题, 本文效仿文献 [30]使用了 quality focal loss (QFL) 来指导 C-IoU 分类分支的训练:

$$L_{\text{qfl}} = -\left|\hat{y} - y_p\right|^{\beta}\left((1-\hat{y})\log\left(1-y_p\right) + \hat{y}\log\left(y_p\right)\right) \tag{12}$$

其中, β是一个超参数, 本文中取 2.0, $\hat{y}$代表每个位置的分类 GT 标签, y_p代表每个位置的预测标签.

最后, SiamDCG 的损失函数即为:

$$L = L_{\text{qfl}} + \lambda_1 L_{\text{CIoU}} + \lambda_2 L_{\text{dfl}} \tag{13}$$

其中, λ_1和λ_2是超参数, 本文中$\lambda_1 = 0.25$, $\lambda_2 = 3.0$.

4 实验分析

4.1 实验细节

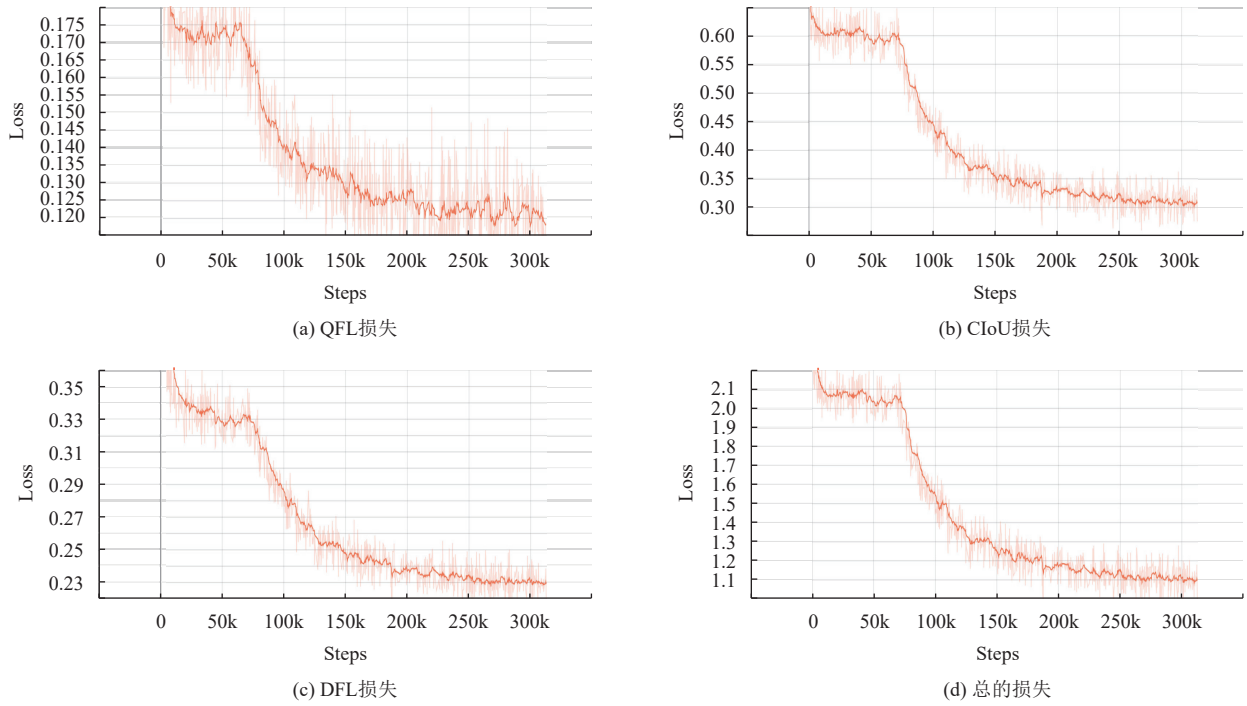
实验环境: 本文算法是使用 Python 版本为 3.8.17 和 PyTorch 2.0.1 框架实现的, 在 Windows 10 操作系统上, 使用 CPU 型号为 13th Gen Intel(R) Core™ i9-13900K, GPU 为 GeForce RTX 4090 进行实验, 网络的 search 大小为 255×255×3 (宽, 高, 通道数), template 的大小为 127×127×3 (宽, 高, 通道数), 孪生网络共享 backbone 部分, 如前文所述使用裁剪过的 MobileNetV3-small 结构, 并且利用文献 [13]中给出的预训练权重进行训练.

SiamDCG 所用到的训练集有微软的上下文通用数据集 (COCO)[31], ImageNet DET[32], ImageNet 视频目标检测 (VID)[32], YouTube-BB[33], 2019 年的大规模单目标跟踪高质量基准数据集 (LaSOT)[34]和通用目标跟踪基准数据集 (GOT-10k)[35]. 一共训练了 50 个 epochs, 前 5 个 epochs 学习率从 0.001 上升到 0.005, 之后的 45 个 epochs 使用从 0.005 衰减到 0.000 5, 使用随机梯度下降法训练网络, 动量设置为 0.9, 权重衰减为 0.005, 在前 10 个 epochs 冻结 backbone 的参数, 并且在之后的 40 个 epochs 中解冻 backbone, 进行整体参数训练, 所提出的网络训练是以端到端的方式进行训练的. 具体的损失收敛曲线见图 4 所示.

4.2 与其他的轻量级跟踪算法对比

在这部分中, 本文将 SiamDCG 在标准的空中跟踪基准上与现有的 19 个轻量级跟踪器进行对比.

UAV123: UAV123[36]包含 123 个低航空跟拍、具有挑战性的序列, 总共超过 110 000 帧, 可用于从无人机的角度测试的跟踪器的泛化能力. UAV123 的性能评估可以验证跟踪器在最常见的空中跟踪条件下的跟踪性能. 如图 4 所示, SiamDCG 超过了以往的轻量级空中跟踪器, 值得一提的是 SiamDCG 的 Success 超过了 HiFT 2.3%, Preicision 领先了 0.6%.



(a) QFL损失 (b) CIoU损失

(c) DFL损失 (d) 总的损失

图 4 训练的损失收敛曲线

DTB70: DTB70[14] 包括 70 个严重的运动场景在各种具有挑战性无人机航拍视频序列. 能够很好地评估我们跟踪器的鲁棒性. 结果如图 4 所示, 可以看到不论是 Success 还是 Precision SiamDCG 都取得了 SOTA 的表现, 不仅如此, 相对于第 2 名的 TCTrack, SiamDCG 在 Success 方面领先 1.9%, 在 Precision 方面领先 1.3%.

UAV123_10fps: UAV123_10fps[36]也是由 123 个视频序列构成, 不同的是 UAV123_10fps 的物体的运动更为迅速, 变化更快, 也进一步增加了跟踪的难度, 但是也更贴近实际 UAV 飞行的场景, 从图 4 中可以看出, 与其他跟踪器相比 SiamDCG 具有更好的鲁棒性, 不论在 Success 还是 Preicision 方面都超过了第 2 名 TCTrack.

此外对于复杂空中跟踪场景, 正如本文前面所提到的, 无人机的特殊工作环境进一步加剧了跟踪的难度, 特别是一些快速运动, 低分辨率, 相似物体干扰以及实时的光照变化等情况. 如图 5 所示, SiamDCG 与其他跟踪器在几种具有挑战性的条件下的鲁棒性, 在面对快速移动 (fast motion, FM), 低分辨率 (low resolution, LR), 相似物体干扰 (similar object, SO), 明暗变化 (illumination variation, IV) 等这些无人机常见的难跟踪的场景中, 可以看到 SiamDCG 都取得了 SOTA 的表现. 值得注意的是, 在 IV 以及 SO 的挑战中, SiamDCG 远超过了第 2 名的 TCTrack, TCTrack 由于充分利用了从第 1 帧到当前帧的历史信息从而使得跟踪器具有一定的鲁棒性[6], 但是对于边界框的回归还是局限于 Dirac δ 分布, 因此对于边界不明确的场景, 如 IV, 虽然前后帧的信息在一定程度上弥补了这些变化带来的误差, 但是不足以弥补目标在跟踪过程中的变化所损失的信息, 此外对于 SO, 本文引入的 DGBCM 通过回归框的分布进一步指导分类, 提高了分类的准确率, 因此对于相似物体而言, SiamDCG 更具优势.

UAVTrack112_L: 为了验证本文提出的跟踪器的

长期跟踪性能, 在 UAVTrack112_L[37]基准上进行了测试评价. UAVTrack112_L 是目前最长的针对无人机的长期空中跟踪基准, 具有丰富的空中视频场景, 如城市、乡村、海洋等, 总量超过 60k 帧. 表 2 显示了 SiamDCG 与其他 SOTA 跟踪器的对比. 其中, 加粗代表性能最优, 斜体表示第 2, 下划线第 3, 部分结果数据来源于文献 [6]. 可以看到, SiamDCG 全面超过了其他 SOTA 跟踪器.

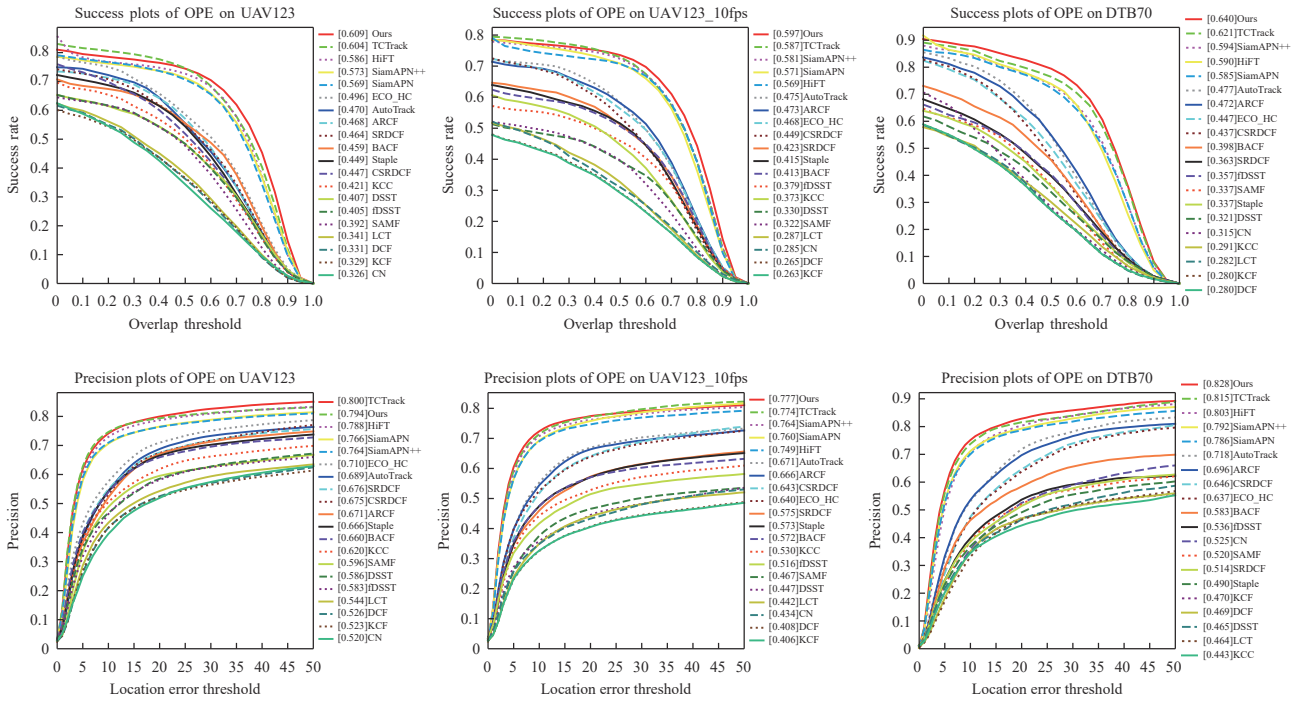


图 5 所有跟踪器在 3 个著名的空中跟踪基准上的性能对比

表 2 各个跟踪器在 UAVTrack112_L 上的性能对比

Trackers	Success↑	Precision↑	Trackers	Success↑	Precision↑
AutoTrack[7]	0.405	0.675	C-COT[38]	0.422	0.691
ARCF[8]	0.399	0.640	UDT+[39]	0.405	0.637
STRCF[40]	0.360	0.609	ECO[41]	0.436	0.684
UDT[39]	0.388	0.620	SiamRPN++[15]	<u>0.559</u>	<u>0.773</u>
SRDCF[42]	0.320	0.508	SiamFC[22]	0.452	0.690
CoKCF[43]	0.283	0.520	DaSiamRPN[19]	0.479	0.729
BACF[44]	0.358	0.593	SiamAPN++[5]	0.537	0.735
DSiam[11]	0.321	0.512	TCTrack[6]	*0.582*	*0.786*
HiFT[20]	0.551	0.734	Ours	**0.603**	**0.794**

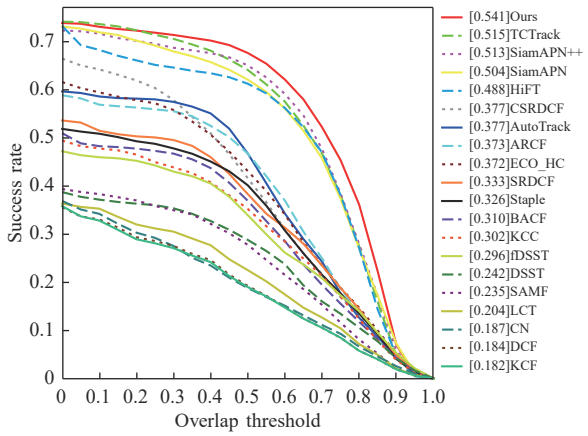
4.3 与基于深度学习的算法对比

为了能够很好地评估 SiamDCG 的性能, 本文选择与其他 12 个 SOTA 的深度跟踪器进行对比, 部分结果来源于文献 [6,45]. 实验的结果如图 6 所示, 可以看到 SiamDCG 在 DTB70 上 Precision 以及 Success 都取得了 SOTA 的成绩, 其中 Precision 方面比第 2 名 TCTrack 高了 1.3%, 比 Ocean 高了 19.2%, Success 方面比 SiamFC++_GoogleNet 高 0.3%, 比 SiamDW 高 18.7%.

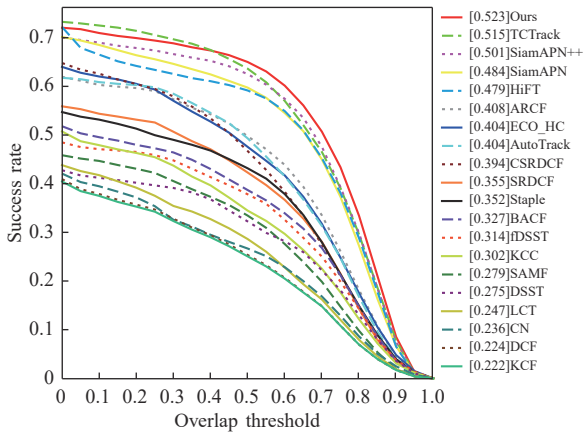
此外表 3 给出这些跟踪器的模型大小以及性能对比. 其中加粗代表性能最优, 斜体表示第 2, 下划线第 3. 可以看到虽然 Success 方面 SiamDCG 领先 SiamFC++并不多, 但是 SiamDCG 的 FLOPs 只有 SiamFC++的约 1/15, Parameters 只有 SiamFC++的 1/60. 值得一提的是, 相对于深度跟踪器 SiamRPN++_ResNet50[16], SiamDCG 在 Precision 以及 Success 方面分别领先 2.8%, 0.7%, 而相对于 SiamRPN++_ResNet50, SiamDCG 减少了 410 倍的 FLOPs 以及 98 倍的 Parameters.

为了进一步证明本文提出的跟踪器的强大性能, 针对快速运动, 低分辨率, 相似物体干扰以及光照不足等较难跟踪场景, 在 DTB70 上可视化了部分视频序列的跟踪效果, 对比了几个目前的 SOTA 空中跟踪算法, 如图 7 所示.

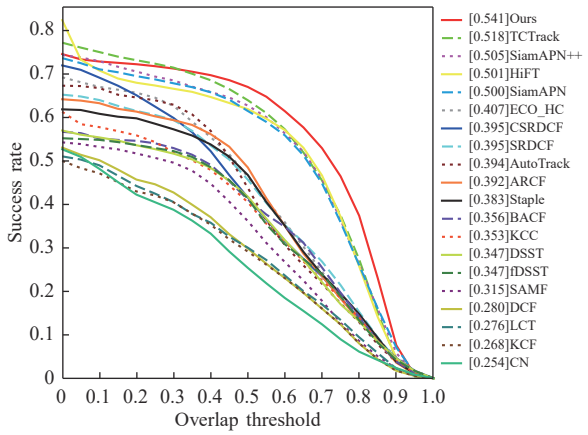
图 8(a), (c), (d) 显示了 SiamDCG 在面对相似物体干扰时的鲁棒性, 这一点可以归功于 SiamDCG 对于分类分支的 *IoU* 指导, 使得分类向量对于前背景的判断更为准确. 此外针对图 8(b) 的低分辨率的快速运动物体, SiamDCG 也表现得非常出色.



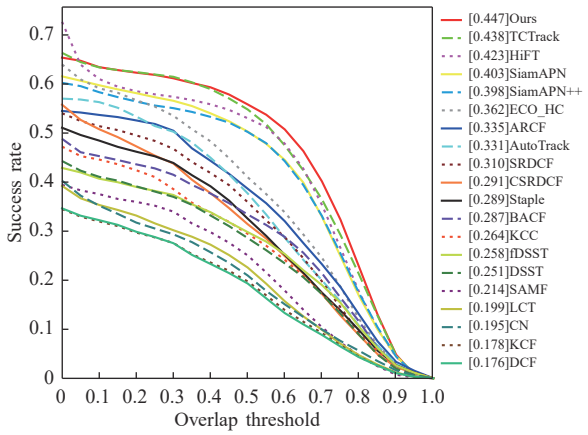
(a) Success plots of OPE on UAV123_10fps-Similar object



(b) Success plots of OPE on UAV123_10fps-Fast motion



(c) Success plots of OPE on UAV123-Illumination variation



(d) Success plots of OPE on UAV123-Low resolution

图 6 SiamDCG 在各种具有挑战性的场景中的成功率曲线

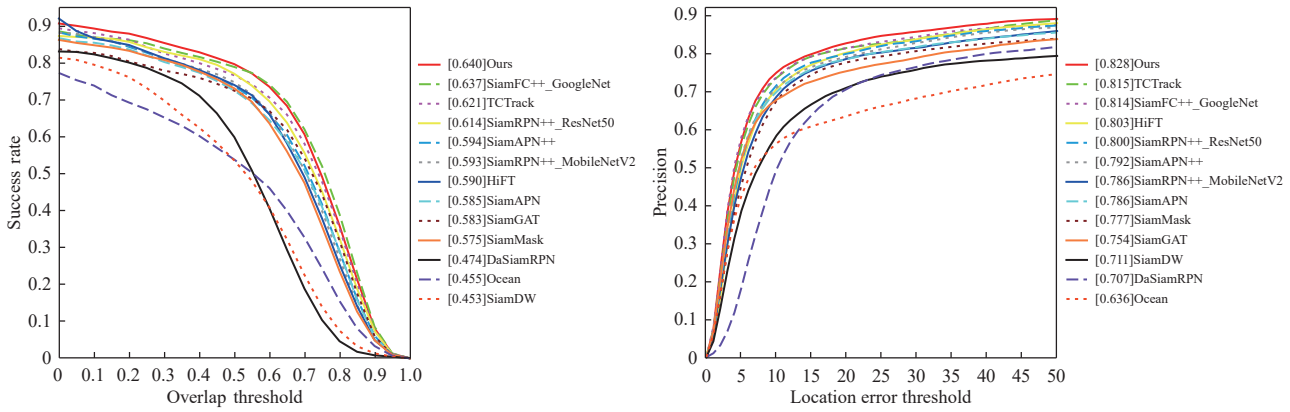
表 3 各个跟踪器在 DTB70[35] benchmark 上的表现, DTB70 上各个跟踪器的模型大小 (Parameters&FLOPs) 以及性能对比

Trackers	Precision↑	Success↑	FLOPs (G)↓	Parameters (M)↓
Ocean[21]	0.636	0.455	20.3	25.9
DaSiamRPN[19]	0.707	0.474	21.1	19.6
SiamDW[28]	0.711	0.453	36.0	45.8
SiamGAT[17]	0.754	0.583	17.7	14.8
SiamMask[18]	0.777	0.575	15.5	16.6
SiamAPN[4]	0.786	0.585	10.2	15.1
SiamAPN++[5]	0.792	0.594	9.2	12.2
SiamRPN++ (MobileNetV2)[15]	0.786	0.593	7.1	26.25
SiamRPN++ (ResNet50) [15]	0.800	0.614	48.0	54.0
HiFT[20]	0.803	0.590	5.7	9.5
SiamFC++ (GoogleNet)[16]	0.814	*0.637*	16.3	30.0
TCTrack[6]	*0.815*	0.621	*4.5*	*6.3*
Ours	**0.828**	**0.640**	**0.1**	**0.5**

5 消融实验与速度测试

从表 4 中可以看出, 其中 I 是基于传统 Dirac δ 分布的模型, II 是通过在回归分支引入离散积分的方式预测边界框, III 是最终引入 DGBCM 的回归离散预测模型, 通过消融实验 I 和 II 可以看出, 相对于使传统的 Dirac δ 分布, 本文在回归分支引入的离散积分回归后, 由于打破了之前的 Dirac δ 分布的缺陷, 确实改善了模型的能力. 同时通过 II 和 III 的比较可以看出 DGBCM 的引入更进一步提升了模型的性能.

此外为了显示本文提出模型的轻量级优势, 分别在 Intel 的 i5 12 代, 7 代以及 RK3588S 上进行了模型速度测试, 如表 5 所示, 使用 Onnxruntime 部署, 3 000 帧取平均值, 可以发现 SiamDCG 在 CPU 上比 SiamRPN++ (ResNet50) 快了接近 150 倍, 并且在中端的 CPU 上可以实现实时运行.



(a) Success plots of OPE on DTB70 (b) Precision plots of OPE on DTB70

图 7 各个跟踪器在 DTB70[35] benchmark 上的表现

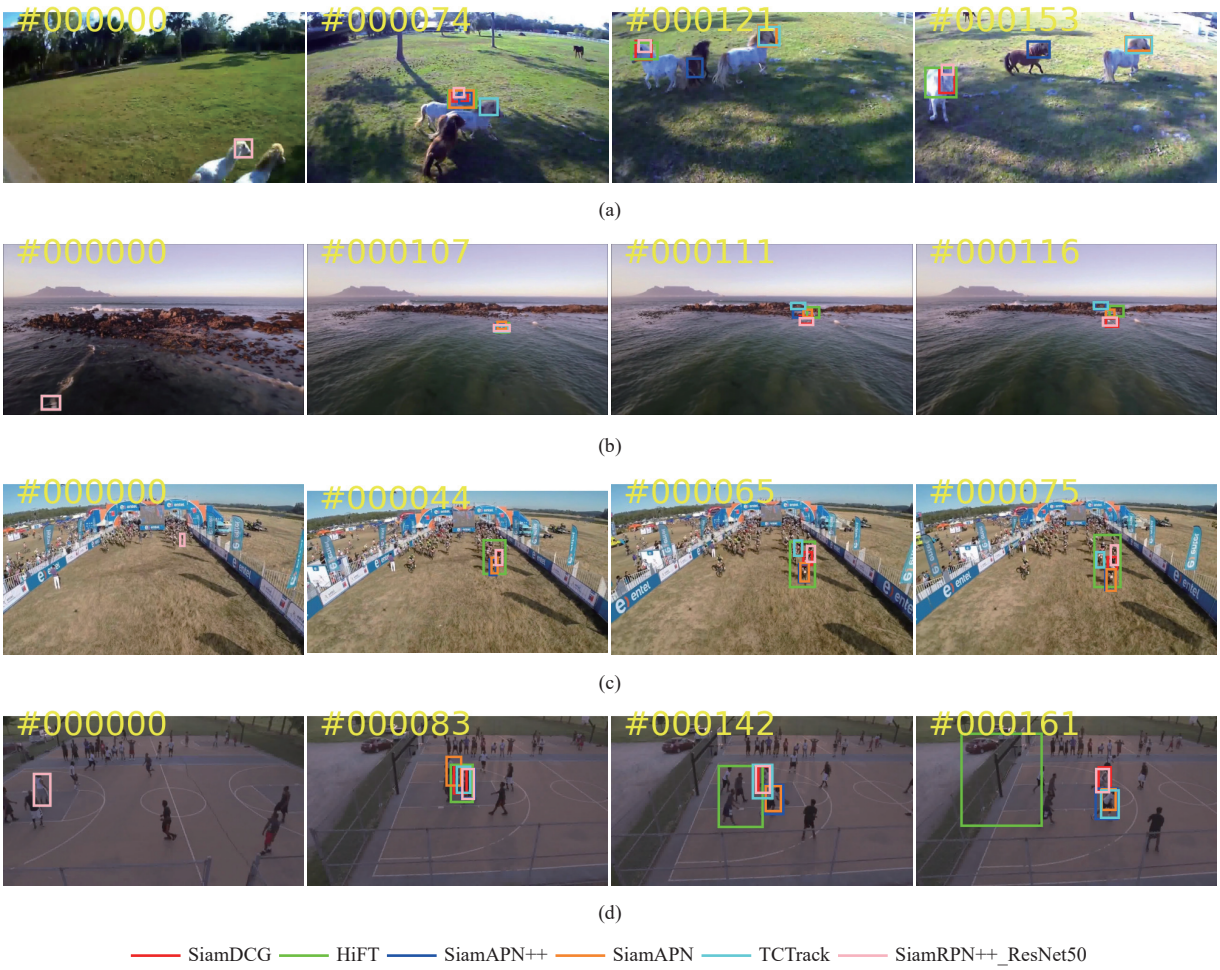


图 8 SiamDCG 在各种具有挑战性的场景中展现出了显著的稳健性

表 4 在 DTB70 上的消融实验

模型	Precision↑	Success↑
I	0.751	0.600
II	0.799	0.615
III	0.828	0.640

6 总结

本文提出了一种基于孪生网络的轻量级双分支跟踪网络 SiamDCG, 通过设计轻量级的 backbone, 大大减小了模型的大小. 此外 SiamDCG 通过学习回归框的

离散分布, 克服了传统的基于 Dirac δ 的回归框预测的局限性. 大量的实验结果表明, 本文提出的 SiamDCG 在多个无人机 benchmark 上取得了 SOTA 的性能, 此外在 CPU 上的速度测试进一步证明了该模型的优秀性能.

表 5 SiamDCG 和 SiamRPN++ (ResNet50) 在不同 CPU 上的速度

CPU	Model	Speed (f/s)↑	Model size (MB)↓
i5-12490	SiamRPN++	4.87	211.1
	SiamDCG	372.80	3.5
i5-7300	SiamRPN++	1.28	211.1
	SiamDCG	100.89	3.5
RK3588s (A76)	SiamRPN++	0.33	211.1
	SiamDCG	50.08	3.5

参考文献

1 Odelga M, Stegagno P, Kochanek N, *et al*. A self-contained teleoperated quadrotor: On-board state-estimation and indoor obstacle avoidance. Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA). Brisbane: IEEE, 2018. 7840–7847.

2 Bonatti R, Ho C, Wang WS, *et al*. Towards a robust aerial cinematography platform: Localizing and tracking moving targets in unstructured environments. Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Macao: IEEE, 2019. 229–236.

3 Cheng H, Lin LS, Zheng ZQ, *et al*. An autonomous vision-based target tracking system for rotorcraft unmanned aerial vehicles. Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vancouver: IEEE, 2017. 1732–1738.

4 Fu CH, Cao ZA, Li YM, *et al*. Siamese anchor proposal network for high-speed aerial tracking. Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA). Xi'an: IEEE, 2021. 510–516.

5 Cao ZA, Fu CH, Ye JJ, *et al*. SiamAPN++: Siamese attentional aggregation network for real-time UAV tracking. Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Prague: IEEE, 2021. 3086–3092.

6 Cao ZA, Huang ZY, Pan L, *et al*. TCTrack: Temporal contexts for aerial tracking. Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022. 14778–14788.

7 Li YM, Fu CH, Ding FQ, *et al*. AutoTrack: Towards high-performance visual tracking for UAV with automatic spatio-temporal regularization. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 11920–11929.

8 Huang ZY, Fu CH, Li YM, *et al*. Learning aberrance repressed correlation filters for real-time UAV tracking. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 2891–2900.

9 Li F, Fu CH, Lin FL, *et al*. Training-set distillation for real-time UAV object tracking. Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA). Paris: IEEE, 2020. 9715–9721.

10 Wang Q, Teng Z, Xing JL, *et al*. Learning attentions: Residual attentional Siamese network for high performance online visual tracking. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 4854–4863.

11 Guo Q, Feng W, Zhou C, *et al*. Learning dynamic Siamese network for visual object tracking. Proceedings of the 2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 1781–1789.

12 Li Q, Qin ZK, Zhang WB, *et al*. Siamese keypoint prediction network for visual object tracking. arXiv:2006.04078, 2020.

13 Howard A, Sandler M, Chen B, *et al*. Searching for MobileNetV3. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 1314–1324.

14 Li SY, Yeung DY. Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models. Proceedings of the 31st AAAI Conference on Artificial Intelligence. San Francisco: AAAI, 2017. 4140–4146.

15 Li B, Wu W, Wang Q, *et al*. SiamRPN++: Evolution of Siamese visual tracking with very deep networks. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 4277–4286.

16 Xu YD, Wang ZY, Li ZX, *et al*. SiamFC++: Towards robust and accurate visual tracking with target estimation guidelines. Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York: AAAI, 2020. 12549–12556.

17 Guo DY, Shao YY, Cui Y, *et al*. Graph attention tracking. Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 9538–9547.

18 Hu WM, Wang Q, Zhang L, *et al*. SiamMask: A framework for fast online object tracking and segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence,

2023, 45(3): 3072–3089.

19 Zhu Z, Wang Q, Li B, *et al*. Distractor-aware Siamese networks for visual object tracking. Proceedings of the 15th European Conference on Computer Vision (ECCV). Munich: Springer, 2018. 103–119.

20 Cao ZA, Fu CH, Ye JJ, *et al*. HiFT: Hierarchical feature transformer for aerial tracking. Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021. 15437–15446.

21 Zhang ZP, Peng HW, Fu JL, *et al*. Ocean: Object-aware anchor-free tracking. Proceedings of the 16th European Conference on Computer Vision. Glasgow: Springer, 2020. 771–787.

22 Bertinetto L, Valmadre J, Henriques JF, *et al*. Fully-convolutional Siamese networks for object tracking. Proceedings of the 2016 European Conference on Computer Vision. Amsterdam: Springer, 2016. 850–865.

23 Li B, Yan JJ, Wu W, *et al*. High performance visual tracking with Siamese region proposal network. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 8971–8980.

24 Du SD, Wang SP. An overview of correlation-filter-based object tracking. IEEE Transactions on Computational Social Systems, 2022, 9(1): 18–31. [doi: 10.1109/TCSS.2021.3093298]

25 Chen ZD, Zhong BN, Li GR, *et al*. SiamBAN: Target-aware tracking with Siamese box adaptive network. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(4): 5158–5173.

26 Tian Z, Shen CH, Chen H, *et al*. FCOS: Fully convolutional one-stage object detection. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 9626–9635.

27 Li X, Wang WH, Wu LJ, *et al*. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. Proceedings of the 34th International Conference on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2020. 1763.

28 Zhang ZP, Peng HW. Deeper and wider Siamese networks for real-time visual tracking. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 4586–4595.

29 Zheng ZH, Wang P, Liu W, *et al*. Distance-IoU loss: Faster and better learning for bounding box regression. Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York: AAAI, 2020. 12993–13000.

30 Li X, Wang WH, Hu XL, *et al*. Generalized focal loss V2: Learning reliable localization quality estimation for dense object detection. Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 11627–11636.

31 Lin TY, Maire M, Belongie S, *et al*. Microsoft COCO: Common objects in context. Proceedings of the 13th European Conference on Computer Vision. Zurich: Springer, 2014. 740–755.

32 Russakovsky O, Deng J, Su H, *et al*. Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 2015, 115(3): 211–252. [doi: 10.1007/s11263-015-0816-y]

33 Real E, Shlens J, Mazzocchi S, *et al*. YouTube-bounding boxes: A large high-precision human-annotated data set for object detection in video. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 7464–7473.

34 Fan H, Lin LT, Yang F, *et al*. LaSOT: A high-quality benchmark for large-scale single object tracking. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 5369–5378.

35 Huang LH, Zhao X, Huang KQ. GOT-10k: A large high-diversity benchmark for generic object tracking in the wild. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(5): 1562–1577. [doi: 10.1109/TPAMI.2019.2957464]

36 Mueller M, Smith N, Ghanem B. A benchmark and simulator for UAV tracking. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 445–461.

37 Fu CH, Cao ZA, Li YM, *et al*. Onboard real-time aerial tracking with efficient Siamese anchor proposal network. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60: 5606913.

38 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.

39 Wang N, Song YB, Ma C, *et al*. Unsupervised deep tracking. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 1308–1317.

40 Li F, Yao YJ, Li PH, *et al*. Integrating boundary and center correlation filters for visual tracking with aspect ratio

variation. Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops. Venice: IEEE, 2017. 2001–2009.

41 Danelljan M, Bhat G, Khan FS, *et al*. ECO: Efficient convolution operators for tracking. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 6931–6939.

42 Danelljan M, Häger G, Khan FS, *et al*. Learning spatially regularized correlation filters for visual tracking. Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015. 4310–4318.

43 Zhang L, Suganthan PN. Robust visual tracking via co-trained kernelized correlation filters. Pattern Recognition, 2017, 69: 82–93.

44 Galoogahi HK, Fagg A, Lucey S. Learning background-aware correlation filters for visual tracking. Proceedings of the 2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017. 1144–1152.

45 Fu CH, Lu KH, Zheng GZ, *et al*. Siamese object tracking for unmanned aerial vehicle: A review and comprehensive analysis. Artificial Intelligence Review, 2023, 56(1): 1417–1477.

(校对责编: 孙君艳)