doi:10.3772/j.issn.1002-0470.2023.06.006

基于无锚框的孪生网络目标跟踪改进算法①

张立国② 张 升③ 章玉鹏 耿星硕 金 梅

(燕山大学电气工程学院 秦皇岛 066000)

摘 要 视觉目标跟踪在车辆、人机交互以及监控等领域应用广泛,虽然近年来取得了很大的进展,但是在跟踪过程中,仍然存在许多的干扰因素。针对跟踪过程存在目标尺度和长宽的比例会随着目标或跟踪设备的变化而变化以及背景干扰的问题,设计了一种基于无锚框的孪生神经网络的跟踪方法。首先,改进了特征提取网络,提高了跟踪的准确性。其次,增加了非局部感知网络,能够更好地利用模板和搜索分支更深度的特征。对于分类来说,增加了选择分支,用于抑制较低的得分,选择更高更准确的得分,从而能够进行更好的回归预测。其采样策略也不同于之前的网络,并对损失部分进行了优化。在对网络进行整体的训练及实验之后,该算法能够很好地跟踪目标,提高了跟踪的成功率和精确度。

关键词 目标跟踪;特征提取;孪生神经网络;精度

0 引 言

目标跟踪无论是在安防还是机器人领域,一直是人们研究的重点。目前的目标跟踪一般指的是在视频的连续帧中,不需要任何目标对象的先验知识即能用来跟踪任意感兴趣目标。通过初始化视频场景中的感兴趣区域,目标跟踪算法需要寻找该区域中的指定目标在后续视频帧里面的位置变化。但是目标跟踪仍然面临许多问题[1],如何在具有挑战性的场景下准确、高效地检测和定位目标的遮挡、失视、变形、背景杂波和其他变化[2],越来越成为人们研究的热点。

现代追踪器大致可以分为2 个分支。第1 个分支是基于相关滤波器,它利用循环相关的特性,在傅里叶域中训练回归器,其广泛应用于跟踪领域。最近的基于相关滤波的方法利用深度特征来提高精度。第2 个分支旨在使用非常强的深度特征,而不更新模型。但是,由于没有使用特定领域的信息,这些方法的性能往往不如基于相关滤波器的方法。2016 年,全卷积孪生网络(fully convolutional siamese networks, SiamFC)[3]算法被提出,孪生卷积网络被引入到目标跟踪领域[4],通过目标帧与模板帧的匹配,求得目标的位置。然而,它的运行速度却很慢。2019 年,SiamRPN ++ (siamese region proposal network ++)[5]利用目标检测中的区域侯选网络(region proposal network, RPN)以及级联的思想将目标跟踪的精度提升至0.960。2020 年,SiamFC ++ (siamese fully convolutional network ++)[6]跟踪网络舍弃了那些预定义的锚框从而让网络能够直接得到被跟踪目标的边框[7],这极大提高了目标跟踪的精确度和效率。然而,当前目标跟踪的瓶颈在于:视频跟踪时往往会存在背景干扰现象[8],目标的尺度和长宽比也会随着目标或摄像机的移动和目标外观的变化而变化,这使得准确估计目标尺度和高宽比以及追踪目标变得很难[9]。

针对上述问题,本文设计了一个无锚框的具有深度特征提取的孪生卷积网络跟踪器,通过对特征网络进行优化,进行深度特征提取,增加了非局部感

① 河北省中央引导地方科技发展专项(199477141G)和河北省科学技术研究与发展计划科技支撑(20310302D)资助项目。
② 男,1978 年生,博士,副教授;研究方向:图像处理,计算机视觉,故障诊断,虚拟现实;E-mail:zlgtime@163.com。
③ 通信作者,E-mail: 278383534@qq.com。
(收稿日期:2022-04-24)

知模块。在跟踪器的最后,增加用于对分类目标进行准确估计的独立选择分支,选取更加合适的回归特征、更加精确的追踪目标,平衡了准确率和效率,进一步提高了精度。

1 基于孪生卷积网络的目标跟踪

1.1 基于 ResNet-50 的特征提取

传统的 SiamFC 网络结构如图 1 所示。

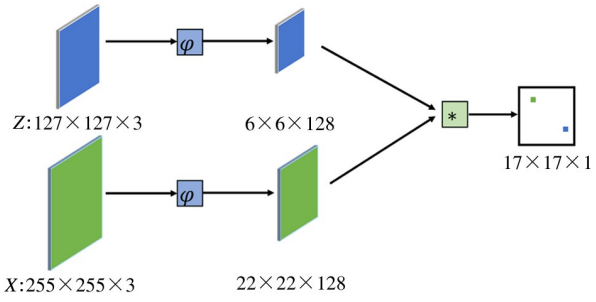

图 1 SiamFC 网络结构

网络由 2 个分支构成,一个是模板分支,输入记为 Z,尺寸为 $127\times127\times3$;另外一个是搜索分支,输入记为 X,尺寸为 $255\times255\times3$。2 个分支共享网络的参数,对 2 个输入进行 φ 变换,分别输出特征图 $\varphi_Z(6\times6\times128)$ 和 $\varphi_X(22\times22\times128)$,对 φ_Z 和 φ_X 进行互相关操作(求卷积),得到了响应图 R,计算过程为

$$R = \varphi_Z * \varphi_X \tag{1}$$

其中,$*$ 代表互相关操作,R 为响应图,再对生成的响应图进行双三次线性插值生成 272×272 的图像来确定目标的位置。

在 SiamFC 之后,许多以 AlexNet 为基准的孪生网络跟踪算法也相继提出,后来许多人也尝试着使用深层次的网络。然而实验发现,使用已经预训练好的深层网络反而会降低跟踪的精度。因此,本文采用 ResNet-50[10] 用作为主干网络(backbone),ResNet-50 结构图如图 2 所示。

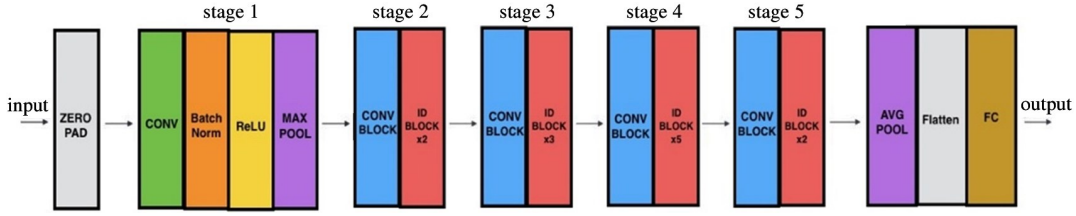

图 2 ResNet-50 结构图

尽管 ResNet-50 可以用来学习抽象的特征,但是目标特征的分辨率被降低了。跟踪器预测的时候需要详细的信息,针对这个问题,本文在最后 2 个卷积块中去掉了下采样这一步骤。为了增加感受野,使用了空洞卷积,受文献[11]的启发,采用不同的扩张率,在卷积层 4 和卷积层 5 中把步距都设为 1,在卷积层 4 中把扩张率设为 2,在卷积层 5 中把扩张率设为 4。

1.2 网络整体结构

孪生网络 2 个分支在网络中共享参数,确保 2 个分支进行相同的变换,整体架构如图 3 和图 4 所示。

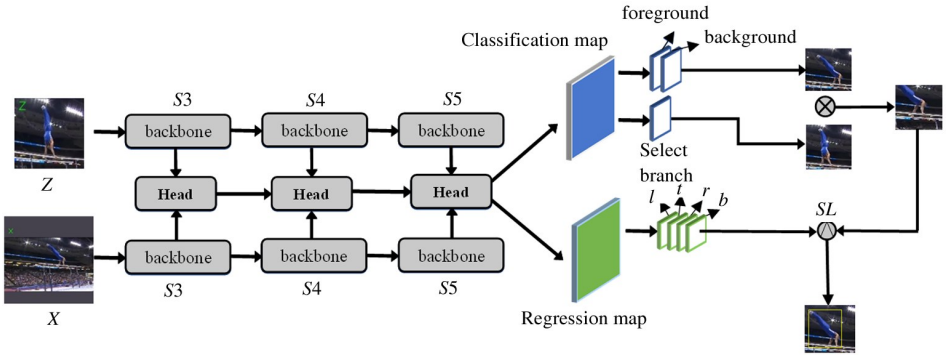

图 3 网络整体结构

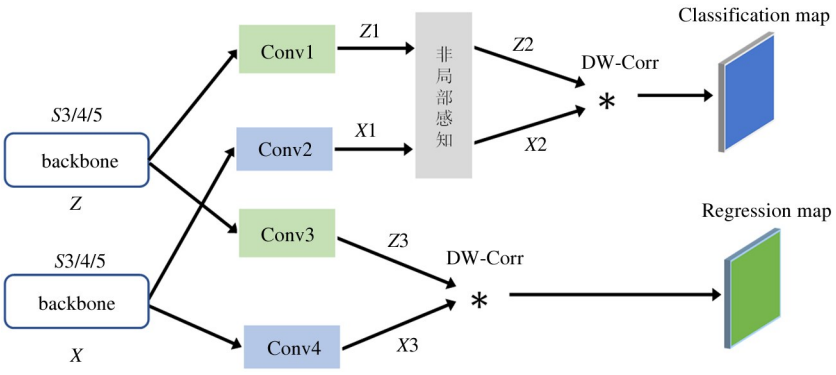

图 4　主干网络结构

图中 *S*3、*S*4 和 *S*5 为主干网络的特征图，*SL* 为独立的选择分支，模板和搜索分支通过卷积等操作得到模板特征 *Z*1 和搜索特征 *X*1，通过卷积层 3 和 4 之后得到回归的特征，在分类特征上增加非局部感知网络，得到分类特征图 *X*2 和 *Z*2。Classification Map 和 Regression Map 为头模块输出的特征图，DW-Corr 为深度交叉相关操作。

整体框架由孪生网络和多个自适应头组成，不需要预定义的候选框。在进行分类后，在全卷积网络中直接回归边界框，其中 *SL* 为质量选择分支。网络可以预测相关特征图上每个空间位置的四维向量，即从边界框到搜索区域对应的特征位置中心点的相对偏移量。

1.3　质量选择分支

在一般网络中，置信度和定位精度没有很好的相关性[12]，距离目标中心远的位置经常容易产生质量比较低的预测边界框[13]，直接使用分类置信度来选择边界框会导致定位精度下降，从而降低跟踪的性能。因此选择和分类分支独立的质量选择分支，即在卷积分类的基础上添加 1 ×1 卷积层，输出定义为

$$SLS = \sqrt{\frac{\min(l,r)}{\max(l,r)} \times \frac{\min(t,b)}{\max(t,b)}} \tag{2}$$

其中，l、t、r、b 的含义将在下文做具体阐述。将它的输出 *SLS* 和相应预测的分类分数乘起来选择最终框的分数，那些远离物体中心的边界框所占的权重就会下降，从而提高跟踪精度。

1.4　非局部感知模块

孪生网络是通过对大量图像进行训练来学习目标的跟踪特征。但是，这些特征的辨别力较弱，当类似的干扰物体出现时，跟踪器很容易被误导。为了应对在跟踪过程中干扰物以及背景等对特征带来的影响，增强搜索分支的识别能力，并且因为在不同的特征通道中，语义是不一样的，所以增加了非局部感知模块（non-local means module，NL），在模块中把模板的信息加入到了搜索分支中，从而提高搜索分支的识别能力，模块的网络结构如图 5 所示。

NL 主要利用每个通道的平均值、最大值以及不同通道的相关性，通过整合这几个位置的信息得到非局部感知网络的权重信息。对于模板分支 *Z*1，把平均全局池化特征 V_z、最大池化 Z_z 和通道的相关信息 R_z 拼接起来得到了响应 y_z。生成通道相关信息 R_z 时，调整 *Z*1 生成 Z_1^R，通过卷积形成了 T_{Z1} 和 T_{Z2}，然后将其相乘就得到了通道间的相关信息，即获得了某个通道和其他通道之间的关系。再通过最大池化和全局平均池化，就得到了 y_z。最后经过 Sigmoid 得到了 A_Z，将其与 *Z*1 进行聚合之后得到了 *Z*2。对于搜索分支 *X*1，类似 *Z*1，得到了模板通道之间的相关信息，然后和搜索分支组合到一起，得到了响应 y_x，再执行与模板分支相同的操作，得到 *X*2。

原始的跟踪器没有目标相关信息的监督，搜索分支并不能保证那些与目标相关的区域不受干扰物的影响得到最大的关注。非局部感知模块的主要作用是引入了全局信息和局部关联信息。该模块的这种关联信息相互作用可以减少背景干扰物对搜索分支的负面影响，从而有助于在搜索区域中定位目标。因此，采用非局部目标感知网络来学习特征权重的跟踪器可以通过对特征通道重要性的再分配来增强

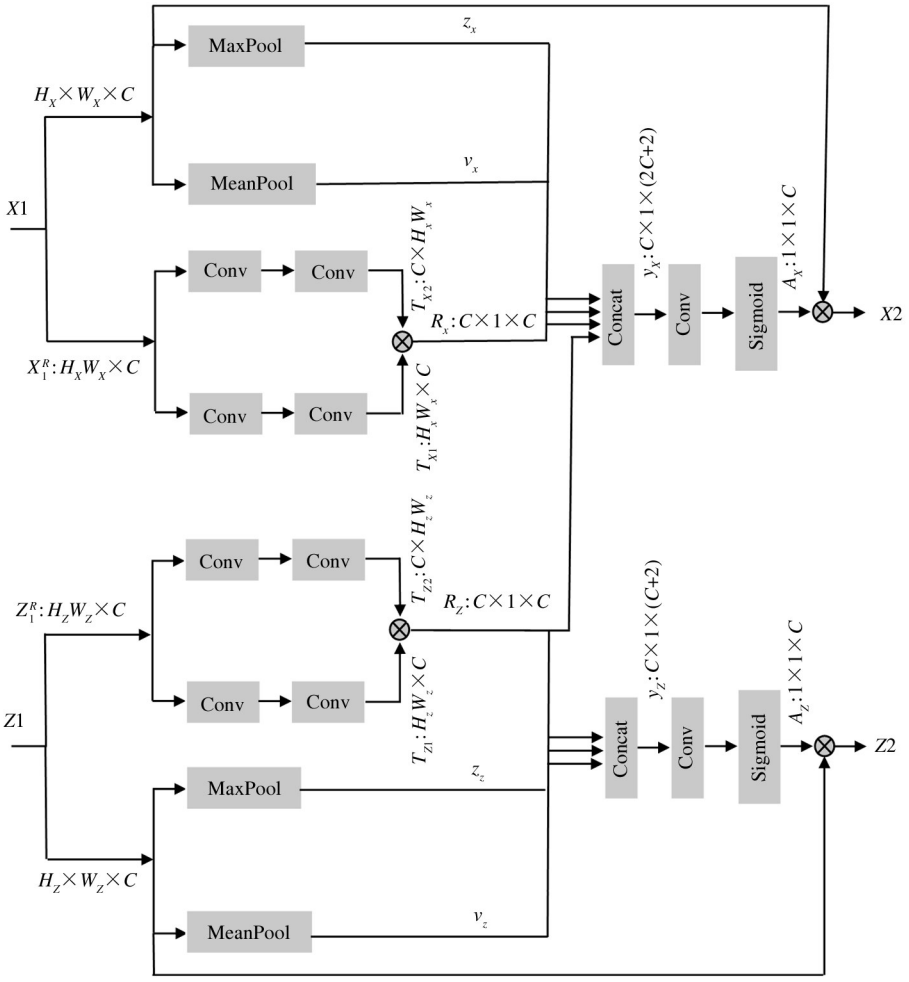

图 5　非局部感知模块

网络对目标的关注效果,进而提高跟踪器性能。后续的消融研究将进一步验证这一想法。

1.5　头模块

如图 4 所示,把网络的输出特性用 $\varnothing(Z)$ 和 $\varnothing(X)$ 来表示,头部模块(Head)由分类和回归 2 部分组成,之后调整 $\varnothing(Z)$ 和 $\varnothing(X)$ 到模块 $[\varnothing(Z)]_{\text{cls}}$、$[\varnothing(Z)]_{\text{reg}}$ 和 $[\varnothing(X)]_{\text{cls}}$、$[\varnothing(X)]_{\text{reg}}$。分类模块主要用来进行前景和背景分类,回归模块主要输出 4 个通道进行边界框的预测,每个模块使用深度互相关组合而成。

$$P^{\text{cls}}_{w\times h\times 2} = [\varnothing(X)]_{\text{cls}} * [\varnothing(Z)]_{\text{cls}} \tag{3}$$

$$P^{\text{reg}}_{w\times h\times 4} = [\varnothing(X)]_{\text{reg}} * [\varnothing(Z)]_{\text{reg}} \tag{4}$$

式中,$*$ 表示 $[\varnothing(Z)]_{\text{reg}}$ 或 $[\varnothing(Z)]_{\text{cls}}$ 作为卷积核进行卷积操作,$P^{\text{cls}}_{w\times h\times 2}$ 表示分类图,$P^{\text{reg}}_{w\times h\times 4}$ 表示回归图。分类图 $P^{\text{cls}}_{w\times h\times 2}$ 和回归图 $P^{\text{reg}}_{w\times h\times 4}$ 中的每一个位置,都能将其映射到搜索补丁,比如 (i, j) 对应搜索补丁上的位置是 $\{\frac{w_m}{2} + (i - \frac{w}{2}) \times s, \frac{h_m}{2} + (j - \frac{h}{2}) \times s\}$,将其表示为 (x, y),其中 w_m 和 h_m 是搜索补丁的宽高,s 表示网络的步距,w 和 h 分别为特征图的宽和高。

1.6　特征融合

实验中考虑了聚合多层深度特征来进行跟踪[14],虽然 backbone 的 conv3、conv4 和 conv5 空间分辨率相同,但它们的扩展速率不同,导致感受野差异较大,捕获的信息存在差异,所以使用多个自适应头进行预测,分别取出搜索分支和模板分支中第 3、4、5 卷积模块的卷积结果,选取模板分支特征图的 7 × 7 区域大小以减小计算量。在对模板图像进行特性提取时,根据目标中心点得到模板补丁 127 × 127,骨干网络后 3 层输出特征图的大小为 15 × 15,此时选取中心[4 : 11]的区域,可以代表目标区

域。相对于搜索分支，通过相同的骨干网络，后 3 层得到大小为 31 ×31 的特征图。然后将模板分支和搜索分支的后 3 层特征分别进行深度互相关操作，最后将得到的结果进行加权融合。

$$P_{w\times h\times 2}^{\text{cls-all}} = \sum_{l=3}^{5} \alpha_l P_l^{\text{cls}} \tag{5}$$

$$P_{w\times h\times 4}^{\text{reg-all}} = \sum_{l=3}^{5} \beta_l P_l^{\text{reg}} \tag{6}$$

其中，α、β 表示每一个特征图对应的权值，经实验可得 α =1、β =2 时，可以取得比较理想的效果。

1.7　网络损失

在对孪生网络提取的不同分支特征进行互相关操作后，设计了分类网络和回归网络，接着将跟踪器的训练损失分为分类和回归损失。

1.7.1　边界框回归

受 SiamRPN[15] 的启发，本实验中，负样本数量比基于有锚框的负样本少，但是总体来说负样本的数量还是比正样本大得多。实验时从 1 对图像里面选择 16 个正样本和 48 个负样本。正负样本的选取如图 6 所示。

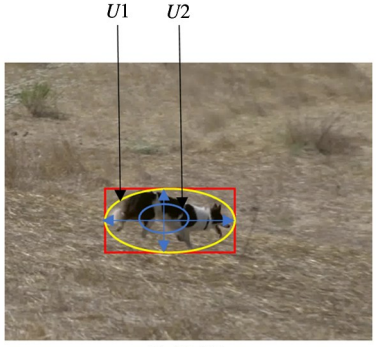

图 6　样本选取

每一个搜索补丁上的跟踪目标都用真实框做一个标记，真实框的高、宽、左上、中心、和右下角点的坐标分别用 g_w、g_h、(g_{x_1},g_{y_1})、(g_{x_c},g_{y_c}) 以及 (g_{x_2},g_{y_2}) 表示。以 (g_{x_c},g_{y_c}) 为中心，以 $\frac{g_w}{2}$、$\frac{g_h}{2}$ 为轴长，可以得到椭圆 $U1$：

$$\frac{(x-g_{x_c})^2}{\left(\frac{g_w}{2}\right)^2}+\frac{(y-g_{y_c})^2}{\left(\frac{g_h}{2}\right)^2}=1 \tag{7}$$

以(g_{x_c},g_{y_c}) 为中心，以 $\frac{g_w}{6}$、$\frac{g_h}{6}$ 为轴长，可以得到椭圆 $U2$：

$$\frac{(x-g_{x_c})^2}{\left(\frac{g_w}{6}\right)^2}+\frac{(y-g_{y_c})^2}{\left(\frac{g_h}{6}\right)^2}=1 \tag{8}$$

(x, y) 落在 $U2$ 内，记为正标签，落在 $U1$ 之外，就记为负标签，落在 $U1$ 和 $U2$ 之间，则将其忽略。用正标签的 (x, y) 对边界框进行回归，对于回归来说，在对分类得分图上得分最大处的位置进行选择之后，就对应着回归分支对目标边框的估计值。回归图上每个位置对应的 4 个偏移值可以不需要预定义的锚框来预测目标边界框位置。把网络预测目标边框的 4 个边到目标真实边框的距离表示为向量 $\boldsymbol{q}$ = (l,t,r,b)，表示如下：

$$l = x - g_{x_1} \tag{9}$$

$$t = x - g_{y_1} \tag{10}$$

$$r = g_{x_2} - x \tag{11}$$

$$b = g_{y_2} - y \tag{12}$$

其中，l、t、r、b 是各个位置到边界框 4 条边的距离。在回归训练中就可以把偏差坐标图转化成预测框，然后挑选最好的预测框进行跟踪。

1.7.2　损失函数

在选定正负样本之后，将损失函数定义如下。

$$L = \lambda_1 L_{\text{cls}} + \lambda_2 L_{\text{reg}} \tag{13}$$

其中 λ 为超参数，在网络训练过程中，发现令 λ_1 = $1,\lambda_2$ = 2，可以取得很好的效果。其中 L_{cls} 为分类损失，L_{reg} 为回归损失。

分类损失表示如下。

$$L_{\text{cls}} = \begin{cases} -\alpha_t(1-p_t)^{\beta}\log(p_t) & y = 1 \\ -(1-\alpha_t)^{\beta}\log(1-p_t) & y = 0 \end{cases} \tag{14}$$

其中，p_t 是网络的估计值，y 表示正负样本时的值，α 取 0.06、β 取 0.02。

对于回归损失 L_{reg}，将其定义为 CIoU 损失。

$$L_{\text{CIoU}} = 1 - \text{IoU} + \frac{\rho^2(b,\ b^{gt})}{c^2} + \alpha\nu \tag{15}$$

其中，

$$V = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{16}$$

其中，IoU 为预测框和目标真实框的交并比，b 和 b^{gt} 分别表示预测框和目标真实框的中心，ρ^2 则表示 2 个点的欧氏距离，c 表示包含 2 个框最小框的斜对

角线长度。α 为平衡系数,w^{gt} 和 h^{gt} 表示目标真实框的宽高,w 和 h 表示预测框的宽高,$(\arctan\frac{w^{gt}}{h^{gt}}-\arctan\frac{w}{h})^2$ 取值是$(0,\frac{\pi^2}{4})$。

2 实验与分析

2.1 网络训练

实验主要使用了数据集 GOT10K[16]、COCO[17]、ImageNet VID[18] 和 ImageNet DET[19],使用在 ImageNet 上预训练后改进的 ResNet-50 作为 backbone 训练模型,采用随机梯度下降法进行优化。重量衰减设置为 0.0001,动量为 0.9,共 20 个 epoch,batch_size 设置为 64,每个 epoch 中采用 40 万对图像样本对进行训练。硬件方面,是在处理器为 Intel(R) Core(TM) i5-10400F CPU@2.90 GHz,RAM 为16 GB,显卡为 RTX 3060 的电脑上进行的;软件方面,在 Window 10 上使用 Python 以 Pytorch 为框架进行。

2.2 数据集与评价指标

本文使用的数据集是目标跟踪的标准数据集 OTB100[20] 和 UAV123[21]。OTB100 是一个广泛使用的公共基准数据集,其包含着尺度变化、背景杂波等 11 个情况下的 100 个视频序列。UAV123 是使用无人机拍摄的场景数据集,包含了从低空航拍视角的 123 个序列,帧数超过了 110K,其中的序列均已被完全标注,对象主要有快速运动、尺度变化、光照变化和遮挡这些问题,使得跟踪变得十分有挑战性。

本文的实验主要使用精确度、成功率、速度 3 个指标对提出的算法进行分析。

(1)精确度

精确度就是跟踪预测的目标框和目标真实框的重叠程度,数值越大,表示精确度越高,公式如下:

$$\phi_t(i)=\frac{1}{N}\sum_{k=1}^{N}\phi_t(i,k) \tag{17}$$

其中,$\phi_t(i,k)$ 代表经过 k 次后,第 t 帧图像的精确性,N 代表重复次数,所以平均准确率为

$$\rho_A(i)=\frac{1}{M}\sum\nolimits_{t=1}^{M}\phi_t(i) \tag{18}$$

其中 M 代表跟踪的有效图像的数量。

(2)成功率

使用预测边界框和真实边界框之间的交并比来表示成功率,通过重叠率(overlap ratio, OR)来表示预测边界框和真实边界框的重叠比率,公式如下。

$$OR=\frac{P\cap G}{P\cup G} \tag{19}$$

其中,OR 表示交并比,P 指的是预测的边界框区域,G 指的是真实边界框区域。

(3)速度

对于目标跟踪来说,跟踪的速度是指每秒钟算法平均处理的帧数,其值越大代表速度越快。

2.3 实验结果与分析比较

为了进一步测试本文所提算法的有效性,将本文算法在数据集 OTB100 和 UAV12 上与主流的算法 SiamFC[3]、SiamRPN[15]、SiamFC ++[5] 和 SiamCAR[22] 进行对比评估。本文方法与其他跟踪算法在 OTB100 和 UAV123 上跟踪的评估结果如表 1 和表 2 所示。

表 1 不同跟踪方法在 OTB100 上的对比结果

跟踪器	成功率	精确度	FPS
SiamFC	0.556	0.745	85
SiamRPN	0.569	0.763	180
SiamFC ++	0.616	0.823	210
SiamCAR	0.617	0.834	52
本文方法	0.641	0.851	45

表 2 不同跟踪方法在 UAV123 上的对比结果

跟踪器	成功率	精确度	FPS
SiamFC	0.497	0.722	75
SiamRPN	0.523	0.726	167
SiamFC ++	0.567	0.775	203
SiamCAR	0.526	0.763	50
本文方法	0.579	0.777	44

在数据集 OTB100 上对不同的算法受尺度变化、背景杂波等因素影响下进行测试,绘制成功率图和精度图如图 7 ~9 所示。

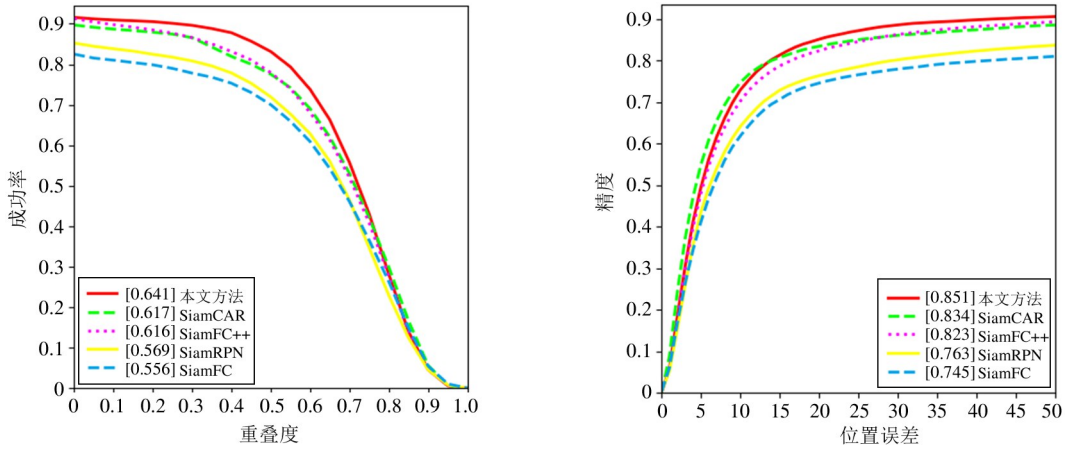

图 7　在 OTB 上不同算法的成功率和精确度曲线图

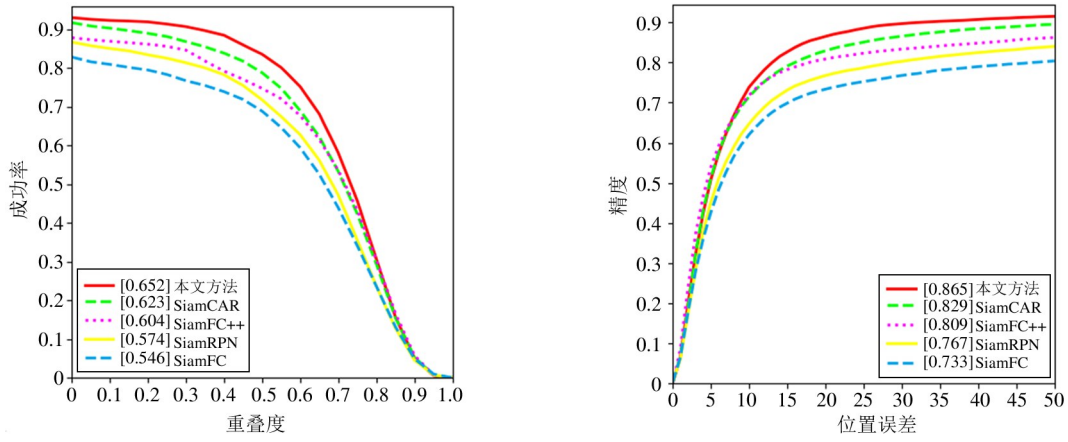

图 8　尺度变化下的成功率和精确度曲线图

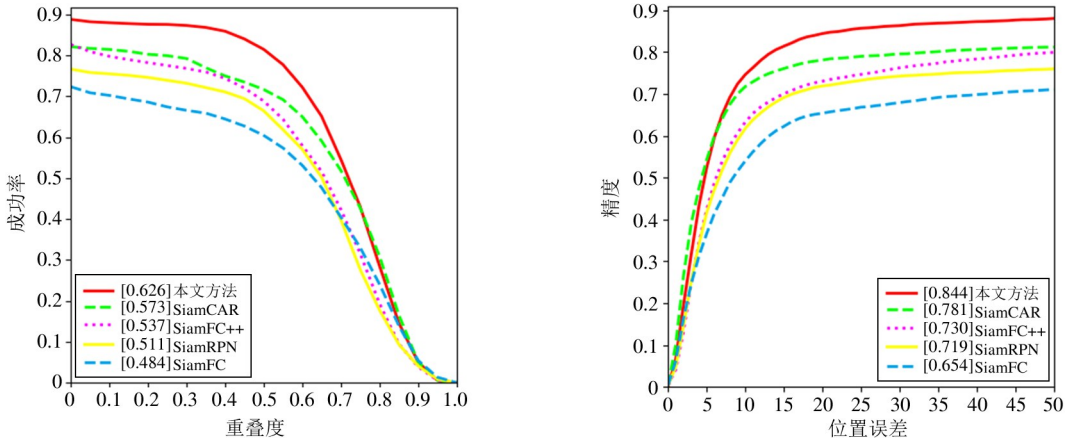

图 9　背景杂波下的成功率和精确度曲线图

在数据集 UAV123 上绘制成功率图和精度图如图 10所示。

表 1 和表 2 显示本文方法和其他算法在精确度和成功率上的评估结果。从表中可以看出，在对特征提取网络进行改进以及增加了非局部感知网络和选择分支之后，精确度和成功率都得到了提升，本文的精确度达到了 85% 和 77%，超过了其他具有竞争力的算法，并且跟踪的速度可以达到实时的要求。

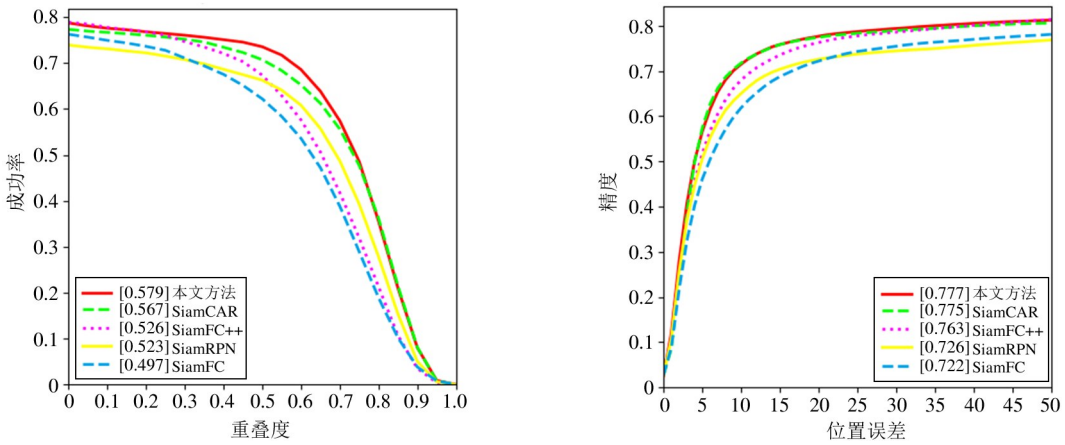

图 10　不同算法的成功率和精确度曲线图

从图 7 和图 10 中可以看出,本文算法在数据集 OTB100 和 UAV123 上的成功率和精确度评估曲线取得了较好的结果,在 SiamFC ++ 之前的网络,由于浅层轻量型网络的表达能力有限,基于孪生网络的跟踪器取得的性能较差,在使用改进后的深层网络 ResNet-50 作为特征提取网络后,其成功率和精确度都得到了提升。从图 8 和图 9 中可以看出,同其他主流算法相比,在添加了非局部感知模块、多特征融合以及选择分支之后,本文的跟踪器可以很好地应对尺度变化和背景杂波等干扰因素带来的影响。总体来说,相较于其他 4 种算法,本文的跟踪性能获得了较大提升。

图 11 为本文算法与其他 4 种算法的定性比较,在典型的视频序列特别是在尺度变化和背景干扰的情况下,对各个算法进行分析。从图中可以看出,随着尺度变化和背景干扰遮挡情况的发生,另外 4 种算法都发生了跟踪漂移的现象,有的甚至出现了跟错目标的情况,相比之下本文的方法可以准确地跟踪目标。

在图 11(a)的第 475 帧和 636 帧,图 11(b)的第 113 帧、361 帧和 581 帧,以及图 11(c)的第 369 帧和 416 帧,只有本文算法可以准确地跟踪目标,应对干扰因素带来的影响,这说明改进的特征提取网络、多层特征融合、局部感知网络及增加的选择分支可以使跟踪器更加关注目标的尺度变化及其位置这些因素带来的干扰,从而更加准确地跟踪目标。

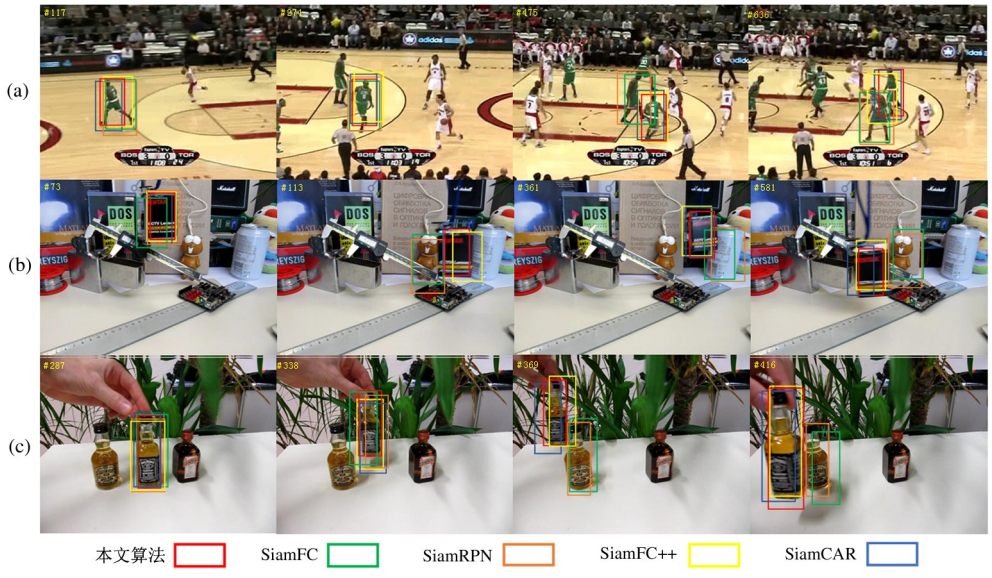

图 11　跟踪结果可视化

2.4 消融研究

为了充分评估本文算法的有效性,在标准的跟踪基准数据集 OTB100 中进行消融研究,验证模块的有效性,比较结果如表 3 所示。

表 3 消融分析

实验	NL							SL	精确度	成功率
	Z1			X1						
	V_z	Z_z	R_z	V_X	Z_X	R_X	R_z			
1	√			√				√	0.785	0.597
2	√			√					0.792	0.601
3	√	√		√	√				0.793	0.605
4			√			√			0.798	0.608
5			√			√	√		0.803	0.617
6	√	√	√	√	√	√	√		0.819	0.616
7	√	√	√	√	√	√	√	√	0.851	0.641

消融实验对非局部感知模块 NL 的平均全局池化特征、最大池化和通道的相关信息及选择分支 SL 进行了研究。将实验 1 和 2 进行对比,说明添加了 SL 之后,算法精度开始有所提升;从实验 2、3 和 4 可以得出,平均全局池化、最大池化和相关信息对跟踪的性能都有影响;从实验 4 中可以看出,当存在 R_Z 和 R_X 的情况下,对跟踪的影响最大,接近 80%;从实验 5 和 6 中可以看出,在搜索分支中添加了模板分支 R_Z 后,算法精度获得了提升;从实验 6 和 7 的对比可以看出,当同时添加了非局部感知模块 NL 和选择分支 SL 之后,跟踪结果取得了最好的提升,说明本文的方法能够提高跟踪器的精确度和准确率。

3 结 论

本文针对当前目标跟踪领域存在的尺度变化和复杂背景的问题,提出了一个以孪生神经网络为基础、结合非局部感知网络和改进的 RseNet-50 网络的无锚框孪生网络跟踪器。在具有挑战性的数据集 OTB100 和 UAV123 上进行基准测试,发现本文的跟踪器能够进行更深层次的特征提取,增加的非局部感知网络能够对跟踪特征进行增强,提高了算法有效性,同时添加了选择分支,可以用来选取更好的跟踪框。实验结果证明,本文的方法可以达到较好的跟踪效果,在应对尺度变化、背景干扰这些影响因素下跟踪的准确度和成功率都有所提高。

参考文献

[1] XU K Y, SHU P, BAO H. Tracking algorithm combining attention and feature fusion network modulation[J]. Laserand Optoelectronics Progress, 2022, 59(12): 121-133.

[2] XU Y, XIONG Y, HUANG W, et al. Deformable siamese attention networks for visual object tracking[C]// IEEE Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2021: 132-153.

[3] XIE G R, QU Y, JIANG R Q. A review of tracking algorithms based on anti-occluded target model[J]. Laser and Optoelectronics Progress, 2022, 59(8): 815-827.

[4] TAN F, MU P A, MA Z X. Multi-target tracking algorithm based on YOLOv3 detection and feature point matching[J]. Journal of Metrology, 2021, 42(2): 157-162.

[5] LI B, WU W, WANG Q, et al. Evolution of siamese visual tracking with very deep networks[J]//IEEE Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2019: 715-140.

[6] XU Y, WANG Z, LI Z, et al. Towards robust and accurate visual tracking with target estimation guidelines[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Seattle: AAAI Press, 2020, 34(7): 12549-12556.

[7] 陈志旺,张忠新,宋娟,等. 基于目标感知特征筛选的孪生网络跟踪算法[J]. 光学学报,2021,40(9):915-932.

[8] 陈卫东,陈磊,邓志巍,等. 基于混合相关滤波信息融合再检测的目标跟踪算法[J]. 计量学报,2020,40(6):1006-1012.

[9] 应巨林,曾铮,曾敏,等.基5G技术的农作物视频监控平台的研究与应用[J].农业与技术,2020,40(15):52-54.

[10] HE K M, XIANG Y Z, SHAO Q R, et al. Deep residual learning for image recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2018: 770-778.

[11] QIANG W, ZHU T, JUN L X, et al. Residual attentional siamese network for high performance online visual tracking[C]//IEEE Conference on Computer Vision and Pattern Recognition. Seattle: USA, 2019:4854-4863.

[12] ZHU Z, WANG Q, LI B, et al. Distractor-aware siamese networks for visual object tracking[C]//Proceedings of the European Conference on Computer Vision. Amsterdam: ECCV, 2019:2-8.

[13] GAO J Y, ZHANG T Z, XU C S. Graph convolutional tracking[J]. Conference on Computer Vision and Pattern Recognition, 2019,2:3-9.

[14] 王浩,李大智.基于5G网络的城市轨道交通全自动无人驾驶列车控制系统研究[J].信息技术与信息化,2020(8):193-194,19.

[15] LIN O, DENG J, SU H, et al. Imagenet large scale visual recognition challenge[J]. International Journal of Computer Vision, 2015,115(3):211-252.

[16] HUANG L, ZHAO X, HUANG K. A large high-diversity benchmark for generic object tracking in the wild[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019,43(5):1562-1577.

[17] CHENG S H, WANG Y. Single fish tracking under occlusion and light change[J]. Journal of Metrology, 2021,42(2):171-177.

[18] LIU Z D, DONG L Q, ZHAO Y J, et al. Adaptive model tracking algorithm for fast-moving targets in video[J]. Acta Optica Sinica, 2021,41(18):181-196.

[19] REAL E, SHLENS J, MAZZOCCHI S, et al. A large high-precision human annotated data set for object detection in video[C]//IEEE Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2017: 574-587.

[20] MULLER M, BIBI A, GIANCOLA S, et al. A large scale dataset and benchmark for object tracking in the wild[C]//Proceedings of the European Conference on Computer Vision. Munich: ECCV, 2018:300-317.

[21] MATTHIAS M, NEIL S, BERNARD G. A benchmark and simulator for UAV tracking[C]//Proceedings of the European Conference on Computer Vision. Amsterdam: ECCV, 2016:445-461.

[22] GUO D, WANG J, CUI Y, et al. Siamese fully convolutional classification and regression for visual tracking[C]//IEEE Conference on Computer Vision and Pattern Recognition. Nanjing: IEEE, 2020:6269-6277.

An improved target tracking algorithm based on frameless twin networks

ZHANG Liguo, ZHANG Sheng, ZHANG Yupeng, GENG Xingshuo, JIN Mei

(School of Electrical Engineering, Yanshan University, Qinhuangdao 066000)

Abastrct

Visual target tracking technology is widely applied in vehicles, human-computer interactions, monitoring and other fields. Despite great progress that has been made in recent years, the current visual target tracking methods still suffer from many interference factors that affect the tracking process. To cope with the problem that the scale and the length-to-width ratio of the target vary with changes of targets or the tracking devices and background interference in the tracking process, a tracking method based on frameless twin neural network is designed. First of all, the feature extraction network is improved to increase tracking accuracy. Meanwhile, non-local sensing network is introduced, which can make better use of the template and deeper features of search branch. For classification, the selection branch is incorporated to suppress low scores and select higher and more accurate scores, which enable better regression prediction. In addition, the sampling strategy differs from the previous network, and the loss function is optimized. With the whole network training and experiments conducted on the network, the algorithm performs better in target tracking with higher success rate and accuracy.

Key words: target tracking, feature extraction, siamese neural network, accuracy