

控制与决策

Control and Decision

基于空域抑制与轨迹关联优化的目标跟踪算法

陈志旺, 高一铭, 吕昌昊, 刘禄阳, 彭勇

引用本文:

陈志旺, 高一铭, 吕昌昊, 等. 基于空域抑制与轨迹关联优化的目标跟踪算法[J]. *控制与决策*, 2026, 41(1): 165-174.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2025.0608>

您可能感兴趣的其他文章

Articles you may be interested in

基于两阶段迭代优化的空天观测资源协同任务规划方法

A two-stage iterative optimization method for the coordinated task planning of space and air observation resources
控制与决策. 2021, 36(5): 1147-1156 <https://doi.org/10.13195/j.kzyjc.2019.1193>

基于Frenet坐标系的自动驾驶轨迹规划与优化算法

Trajectory planning and optimization algorithm for automated driving based on Frenet coordinate system
控制与决策. 2021, 36(4): 815-824 <https://doi.org/10.13195/j.kzyjc.2019.0748>

天临空协同对地观测任务规划模型与并行竞争模因算法

Planning model and parallel competing memetic algorithm for space-near space-air based cooperative earth observation missions
控制与决策. 2021, 36(3): 523-533 <https://doi.org/10.13195/j.kzyjc.2020.0732>

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>

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>

基于空域抑制与轨迹关联优化的目标跟踪算法

陈志旺^{1,2†}, 高一铭^{1,2}, 吕昌昊³, 刘禄阳^{1,2}, 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 河北省工业计算机控制工程重点实验室, 河北 秦皇岛 066004;
3. 燕山大学 河北省电力电子节能与传动控制重点实验室, 河北 秦皇岛 066004;
4. 燕山大学 电气工程学院, 河北 秦皇岛 066004)

摘要: 传统目标跟踪算法更新模板的方式较为单一, 无法在目标发生剧烈形变时有效更新, 易导致算法对于剧烈形变和遮挡场景适应性不足. 鉴于此, 提出一种基于空域抑制与轨迹关联优化的目标跟踪方法. 首先, 在特征提取部分采用改进后的 ResNet-50 网络, 引入空域抑制注意力 (SIA), 通过设计能量函数为具有空域抑制效应的神经元优化权重分配, 增强对目标特征的关注强度; 然后, 加入选择性查询回忆策略 (SQR), 将 Transformer 训练重点放在后期阶段, 避免解码器缺乏训练重点和多层解码结构的级联错误; 接着, 构建长期-短期-轨迹框架 (LST), 引入轨迹关联度 (DTC), 在传统模板更新算法的基础上加入 DTC 预测头, 通过最小成本流算法 (MCF) 建模全局轨迹关联性, 并使用连续二次规划进行反向传播, 更新轨迹参考模板; 最后, 在 GOT-10k、LaSOT、TrackingNet、NfS30、UAV123 和 OTB100 等公开数据集上进行实验评估, 实验结果验证了所提出算法的有效性.

关键词: 目标跟踪; Transformer 结构; 空域抑制注意力; 选择性查询回忆; 轨迹关联度; 最小成本流

中图分类号: TP391.4 文献标志码: A

DOI: 10.13195/j.kzyjc.2025.0608

引用格式: 陈志旺, 高一铭, 吕昌昊, 等. 基于空域抑制与轨迹关联优化的目标跟踪算法 [J]. 控制与决策, 2026, 41(1): 165-174.

Object tracking algorithm based on spatial inhibition and trajectory correlation optimization

CHEN Zhi-wang^{1,2†}, GAO Yi-ming^{1,2}, LV Chang-hao³, LIU Lu-yang^{1,2}, PENG Yong⁴

1. Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Qinhuangdao 066004, China;
2. Hebei Key Laboratory of Industrial Computer Control Engineering, Yanshan University, Qinhuangdao 066004, China;
3. Key Laboratory of Power Electronics for Energy Conservation and Drive Control of Hebei Province, Yanshan University, Qinhuangdao 066004, China;
4. School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China)

Abstract: Traditional target tracking algorithms rely on a single template update method, which fails to handle severe deformation and occlusion, thus limiting their adaptability in challenging scenarios. Therefore, we propose an object tracking method based on spatial inhibition and trajectory association optimization. First, the improved ResNet-50 network is used in the feature extraction part of the algorithm. Spatial inhibition attention (SIA) is introduced. By using an energy function, the weight allocation of neurons exhibiting spatial inhibition is optimized to enhance attention to target features. Then, a selective query recall (SQR) strategy is added. The training focuses on the later stages of Transformer training to avoid a lack of training focus in the decoder and prevent cascading errors in the multi-layer decoding sequence. Furthermore, we construct a long-short-trajectory (LST) framework, and introduce a degree of trajectory correlation (DTC). Based on the traditional template update algorithm, a DTC prediction head is proposed. The global trajectory association is modeled through the minimum cost flow (MCF) algorithm, and the continuous quadratic programming constraints are used for back-propagation to update the trajectory reference template. Finally, the proposed method is experimentally evaluated on public datasets such as GOT-10k, LaSOT, TrackingNet, NfS30, UAV123, and OTB100, and the results verify its effectiveness.

Keywords: target tracking; Transformer architecture; spatial inhibition attention; selective query recall; degree of trajectory correlation; minimum cost flow

收稿日期: 2025-06-09; 录用日期: 2025-09-29.

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); 河北省级重点实验室绩效补助经费项目 (22567612H).

†通信作者. E-mail: czwaaron@ysu.edu.cn.

0 引言

目标跟踪是计算机视觉的核心任务,旨在通过视频初始帧提供的目标信息,在后续帧持续定位目标的位置和尺度.然而,由于跟踪目标的自身因素(如形变、尺度变化、快速移动)以及外界环境干扰(如相似物干扰、遮挡)等因素的影响,实现高精度和高准确率的目标跟踪仍然是一项具有挑战性的任务.

当前,基于深度学习的主流目标跟踪方法可分为孪生网络和 Transformer 两大技术路线,具体如下:

1) 孪生网络通过双分支特征匹配实现高效跟踪: SiameseFC^[1]首次将全卷积孪生结构引入该领域,将目标跟踪转化为给定模板与候选图像匹配问题,在保证实时性的同时实现了高效的相似性匹配,奠定了后续孪生网络算法的基础; SiameseRPN^[2]以 SiameseFC 为基础,融合 Fast R-CNN 的改进方案,实现了高精度目标定位和尺度自适应; DaSiamRPN^[3]引入了干扰物感知机制,通过大规模数据集构造不同难度的负样本数据对,提升了模型判别能力,缓解了遮挡或目标消失时产生的漂移问题; SiamFC++^[4]针对孪生网络存在的问题,提出了目标估计的4项核心准则,构建了高效且鲁棒的跟踪框架.

2) Transformer 结构侧重于凭借全局注意力机制捕捉图像中的长距离关系: ViT^[5]将纯 Transformer 结构 ViT 引入了计算机视觉领域,将图像视为序列化的块,输入 Transformer 进行全局建模,处理图像中不同位置的上下文依赖关系; Swin Transformer^[6]引入了局部窗口自注意力与跨窗口连接机制,在保留 ViT 全局建模能力的同时降低了计算复杂度,并引入了分层特征金字塔结构以实现多尺度建模; STARK^[7]则提出了一种面向视觉跟踪的 Transformer 端到端结构,通过全局时空建模和动态模板更新策略,预测目标边界框而无需预定义锚框或后处理步骤,大大简化了跟踪过程; OSTrack^[8]通过自监督 MAE 预训练 ViT 主干联合优化特征学习和融合,并设计了早期候选消除模块动态剔除搜索区域中的背景干扰,在保持实时性的同时实现了多场景下的高精度跟踪.

目标外观的持续变化仍然可能会破坏注意力权重分布.为应对动态外观变化,研究者提出模板更新策略以动态调整目标表征. PrDiMP^[9]根据输入图像预测目标的条件概率密度,引入最小化 KL 散度训练回归网络,提高了算法适应目标形变和遮挡下外观变化的能力; STARK 通过动态模板更新结合 Transformer 的全局建模能力,在跟踪过程中调整了

目标模板以应对形变、遮挡等外观变化.这些方法通过动态调整模板特征来缓解外观失配问题.

上述基于不同架构的跟踪方法将目标跟踪视为目标图像与搜索区域间的模板匹配问题,遵循模板更新机制以期缓解目标外观变化对于算法跟踪精度和成功率造成的影响,但是,对于跟踪过程中帧节点间的空间依赖性并未充分研究,过于简单的模板更新机制会导致剧烈形变和遮挡场景下的目标丢失^[10].针对上述挑战,本文提出一种基于空域抑制与轨迹关联优化的目标跟踪框架,主要内容如下:

1) 设计空域抑制注意力 SIA 模块,通过能量函数筛选高质量模板特征,抑制噪声干扰;

2) 开发选择性查询回忆 SQR 策略,在 Transformer 结构的解码器中融合历史查询状态,避免级联误差传播;

3) 构建长期-短期-轨迹框架 LST,联合最小成本流算法动态求解并更新轨迹参考模板,将长期和短期参考模板与轨迹参考模板相结合,避免因目标剧烈形变导致目标丢失.

1 本文算法

所提出算法整体流程如图1所示,由特征提取网络、Transformer 结构、预测头结构和模板更新模块组成.所提出算法以初始帧作为长期参考模板,将若干中间帧作为短期参考模板.为增强网络特征提取的能力,引入基于神经科学中神经元的空域效应构建的 SIA 模块,为具有空域抑制效应的神经元赋予更高的重要性,更精准地关注和跟踪感兴趣的目标;为避免传统 Transformer 结构的多层级联错误,引入选择性查询 SQR 策略,在 Transformer 结构的解码器中融合历史查询状态,将训练重点放在后期阶段,缓解前序阶段错误对当前预测的误导;为解决目标发生剧烈形变时无法通过传统模板更新模块有效更新的问题,构建长期-短期-轨迹框架 LST,引入轨迹参考模板,将长期和短期参考模板与轨迹参考相结合,及时对目标形变情况进行更新,避免真实目标丢失.

1.1 特征提取网络

ResNet-50 作为深度卷积网络的代表性模型,在图像分类和目标检测等任务中性能卓越.现有研究多基于其结构进行改进,常通过并联或串联空间和通道注意力机制来优化特征权重分配.然而,这类方法不仅引入了额外参数,也未能充分模拟人脑注意力中空域抑制的核心机制——活跃神经元对周边神经元的抑制,以增强局部显著性.

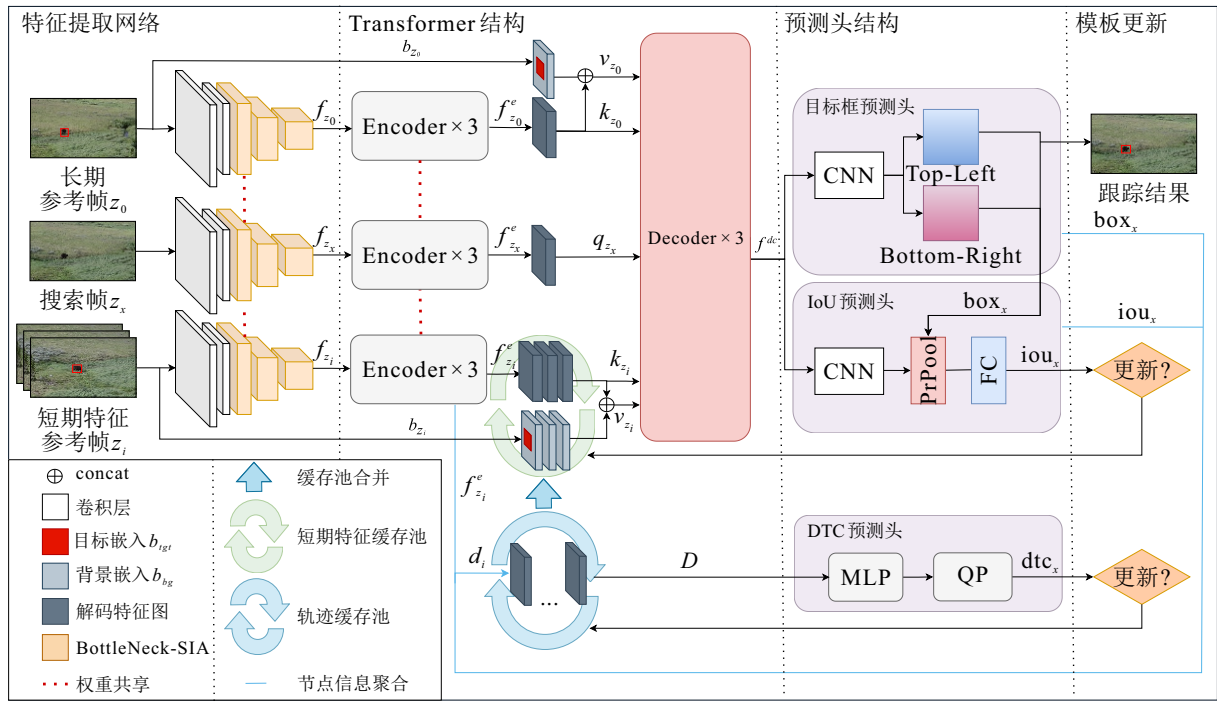


图1 本文算法整体流程

基于此, 所提出算法在 ResNet-50 中 Bottleneck 残差块的 3×3 卷积层后嵌入 SIA 模块, 构成如图 2 所示的 Bottleneck-SIA 残差块. 该模块借鉴神经科学中的空域抑制机制, 通过构建能量函数来评估特征图中各神经元的重要性, 并推导出能量函数的快速闭式解, 用于表征神经元权重, 从而增强关键特征的显著性.

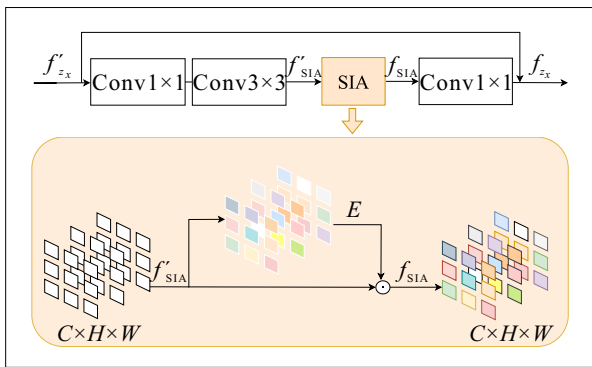


图2 Bottleneck-SIA 残差块

以搜索帧分支为例进行说明, 其修改后的 ResNet-50 结构最后一层中单个 Bottleneck-SIA 残差块如图 2 所示. 该残差块包括卷积核分别为 1×1 、 3×3 和 1×1 的 3 个卷积层和 1 个 SIA 模块. 其中: 特征图 f'_{z_x} 为上一层 Bottleneck-SIA 残差块的输出, 在图 2 所示残差块中经 1×1 和 3×3 两个卷积层, 输出为 $f'_{SIA} = W_{R_2}(W_{R_1}(f'_{z_x}))$, W_{R_1} 、 W_{R_2} 分别是 ResNet-50 中卷积核为 1×1 和 3×3 的卷积层权重. SIA 模块以 f'_{SIA} 为输入, 对该特征图推断各神经元权

重并赋值, 最后得到该模块的输出 f_{SIA} . f_{z_x} 为该残差块的总输出, ResNet-50 网络整体由类似结构的多个残差块叠加构成.

在残差块中, 输入特征 f'_{z_x} 经卷积层、SIA 模块的整体计算过程可表示为

$$f_{z_x} = f'_{z_x} + W_{R_3}(F_{SIA}(W_{R_2}(W_{R_1}(f'_{z_x})))) \quad (1)$$

其中: W_{R_1} 、 W_{R_2} 和 W_{R_3} 分别为 ResNet-50 中卷积核为 1×1 、 3×3 和 1×1 的卷积层权重; $F_{SIA}()$ 为 SIA 模块, 其算法具体实现过程如下.

采用 SVM 算法评估神经元的线性可分性, 定义线性决策边界为 $\omega_s x + b_s = 0$, 则上下间隔边界可定义为 $\omega_s x + b_s = y_1$ 和 $\omega_s x + b_s = y_2$, 其中 y_1 和 y_2 为两个不同的值. 将输入特征 $f'_{SIA} \in R^{C \times H \times W}$ 中单个通道上的目标神经元 n_s 和其他神经元 n_i 代入上述公式进行整合, 可得到 SIA 模块中所定义的能量函数, 即

$$e_s(\omega_s, b_s, y_1, y_2, n_s, n_i) = \frac{1}{M-1} \sum_{i=1}^{M-1} (y_1 - \hat{n}_i)^2 + (y_2 - \hat{n}_s)^2 \quad (2)$$

这里: 该函数主要作用为度量目标神经元 n_s 与其他神经元 n_i 的线性可分性, $\hat{n}_s = \omega_s n_s + b_s$ 和 $\hat{n}_i = \omega_s n_i + b_s$ 分别为 n_s 和 n_i 的线性变换, n_s 和 n_i 分别为输入特征 f'_{SIA} 上单个通道的目标神经元和其他神经元, ω_s 和 b_s 分别为基于目标神经元 n_s 计算能量函数时线性变换的权重和偏置, $n_i \neq n_s$, 且 i 为空间维度上的索引; $M = H \times W$ 为该通道上神经元的数量.

当 $\hat{n}_i = y_1$ 且 $\hat{n}_s = y_2$ 时, 式 (2) 达到最小值, 即

等价于找到目标神经元 n_s 与同一通道中所有其他神经元间的线性可分性。

对 y_1 和 y_2 采用二值标记法, 分别设为1和-1, 并在式(2)中加入正则化项, 最终的能量函数为

$$e_s(w_s, b_s, n_s, n_i) = \frac{1}{M-1} \sum_{i=1}^{M-1} (1 - (\omega_s n_i + b_s))^2 + (-1 - (\omega_s n_s + b_s))^2 + \lambda_{\text{SIA}} \omega_s^2, \quad (3)$$

其中正则化项 $\lambda_{\text{SIA}} \omega_s^2$ 为添加到损失函数中的惩罚项, 主要用于减少模型的复杂度, 避免过拟合, 并提高模型的泛化能力。

理论上, 每个通道有 M 个能量函数. 对于式(3)而言, 其对 ω_s 和 b_s 有一个快速的闭型解, 可以很容易地通过下式计算:

$$\omega_s = -\frac{2(n_s - \mu_s)}{(n_s - \mu_s)^2 + 2\sigma_s^2 + 2\lambda_{\text{SIA}}}, \quad (4)$$

$$b_s = -\frac{1}{2}(n_s + \mu_s)\omega_s, \quad (5)$$

其中 $\mu_s = \frac{1}{M-1} \sum_{i=1}^{M-1} n_i$ 和 $\sigma_s^2 = \frac{1}{M-1} \sum_{i=1}^{M-1} (n_i - \mu_s)^2$ 分别为其他神经元的均值和方差。

假设单个通道中的所有神经元均遵循相同的分布, 则基于这一假设, 可以计算该通道上所有神经元的均值 $\bar{\mu}$ 和方差 $\bar{\sigma}^2$, 并对该通道上的所有神经元重复使用, 避免了对每个位置进行均值和方差的迭代计算, 降低了计算成本. 将式(4)和(5)得到的 ω_s 和 b_s 的闭型解代入式(3), 可得到

$$e_s^*(n_s, n_i) = \frac{4(\bar{\sigma}^2 + \lambda_{\text{SIA}})}{(n_s - \bar{\mu})^2 + 2\bar{\sigma}^2 + 2\lambda_{\text{SIA}}}, \quad (6)$$

其中 $\bar{\mu} = \frac{1}{M} \sum_{i=1}^M n_i$ 和 $\bar{\sigma}^2 = \frac{1}{M} \sum_{i=1}^M (n_i - \bar{\mu})^2$ 分别为全部神经元的均值和方差. 式(6)表明对于目标神经元 n_s 而言, 能量 e_s^* 越低, 与周围神经元的区别越大, 对于算法而言越重要, 则每个神经元的重要性可由 $E = \frac{1}{e_s^*}$ 得到。

根据神经科学理论中的研究表明, 哺乳动物大脑中的注意力调节通常表现为对神经元反应的增益(即缩放)效应^[11]. 因此, 本文使用逐元素相乘运算进行特征细化, 在公式中的符号表示为 \odot .

综上, SIA 模块的整体函数 $F_{\text{SIA}}()$ 可细化为

$$f_{\text{SIA}} = F_{\text{SIA}}(f'_{\text{SIA}}) = \text{sigmoid}(E) \odot f'_{\text{SIA}}. \quad (7)$$

1.2 Transformer 结构

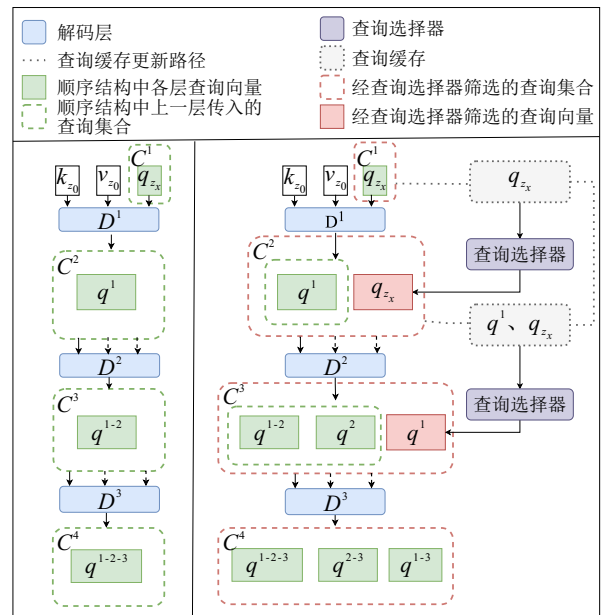
在本文的 Transformer 结构中, 其输入特征由长期参考模板、搜索帧和短期参考模板 3 个分支传入,

分别为长期参考特征 f_{z_0} 、搜索帧的特征 f_{z_x} 和短期参考特征 f_{z_i} , 3 个分支经编码器进一步提取特征, 并经由解码器输出特征图 f^{dc} , 用于预测任务。

以搜索帧为例, 如图 1 所示: 在搜索帧分支上, Transformer 编码器由多层多头注意力和前馈神经网络组成, 其以特征提取阶段中得到的搜索帧特征 f_{z_x} 作为输入, 通过多头自注意力机制和前馈网络对输入特征图进一步加强和提取, 将输入特征编码为富含全局上下文信息的高维特征表示 $f_{z_x}^e$, 经线性变换后为解码器在进行相似性度量操作时提供初始查询向量 q_{z_x} . 对于长期和短期参考模板分支而言, Transformer 编码器则分别输出编码特征 $f_{z_0}^e$ 和 $f_{z_i}^e$, 在引入相应的目标-背景嵌入 b_{z_0} 和 b_{z_i} 后可得到初始键值对向量 v_{z_0} 、 k_{z_0} 、 v_{z_i} 和 k_{z_i} .

对于 Transformer 的解码器而言, 该结构共包括两个交叉注意力分支: 长期交叉注意力和短期交叉注意力. 二者各司其职, 分别将表示长期特征信息的键值对向量 v_{z_0} 和 k_{z_0} , 以及表示短期特征信息的键值对向量 v_{z_i} 和 k_{z_i} 传入搜索帧提取的初始查询向量 q_{z_x} 中进行特征融合, 从而解码信息。

以 Transformer 解码器中关于长期参考模板与搜索帧的特征融合过程为示例, 原始 Transformer 中解码器与 SQR 结构对比如图 3 所示, 其中图 3(a)为原始 Transformer 解码器的典型结构. 在传统目标跟踪任务中: 解码层的初始输入 q_{z_x} 为初始查询向量, q^1 为经第 1 层解码层后输出的中间查询向量, 其余中间查询向量定义以此类推; D^s 为不同的解码阶段, $s \in \{1, 2, \dots\}$. 假设以初始查询向量 q_{z_x} 作为一个基



(a) 原始解码结构

(b) SQR 解码结构

图3 原始 Transformer 解码器与 SQR 结构对比

本单位, 对其依次应用单解码阶段 D^s , 则查询向量在解码层中将进行连续更新。

沿原始 Transformer 解码器的基本路径对初始查询向量 q_{z_x} 进行细化, 以 3 层解码器为例, q^{1-2-3} 为经各层细化后的最终查询, 则该过程可表示为

$$q^{1-2-3} = D^3(D^2(D^1(q_{z_x}, k_{z_0}, v_{z_0}))). \quad (8)$$

然而, 对于传统 Transformer 而言, 其解码器的设计仍然存在局限性. 如 Yu 等^[12] 指出, 尽管解码器各层以相同方式处理查询向量, 但是, 不同层对于结果的重要性不同, 后期解码层需要承担更关键的预测责任, 传统 Transformer 的解码机制对各层施加均等约束, 缺乏将训练重点放在后期阶段的机制。

针对上述问题, 所提出 SQR 策略进行相应的改进, 其结构如图 3(b) 所示. 具体而言, SQR 通过改变信息级联传递的策略, 将搜索帧分支的特征信息 f_{z_x} 经编码器处理后作为查询向量 q_{z_x} 传入解码器, 在解码器前向传播过程中添加逐层更新的查询缓存池和相应的查询选择器, 系统性地缓存历史解码阶段输出的中间查询状态, 并将这些中间查询作为额外输入, 选择性地注入下游阶段。

SQR 结构主要通过以下方式优化解码过程:

- 1) 不均监督机制: 设计区别于传统级联传递的不均监督机制, 增强对后期层的监督;
- 2) 跨层查询融合: 引入逐层更新的查询缓存池, 累积并选择性传递早期层中间查询至后期层;
- 3) 动态路径剪枝: 设计查询选择器, 拒绝输入跳过太多层的早期查询, 避免同一层内查询间学习差异过大, 有选择地将查询保存并转发到每个阶段。

查询选择器具体实现过程可表示为

$$C^1 = \{q_{z_x}\}, \quad (9)$$

$$C^2 = \{q_{z_x}, q^1\}, \quad (10)$$

$$C^s = \{D^s(q) | q \in C^{s-1}\} \cup \{D^{s-1}(q) | q \in C^{s-2}\}. \quad (11)$$

其中: $s \in \{1, 2, \dots\}$, C^s 为第 s 层解码器输入的查询集合。

1.3 预测头结构

所提出算法包括目标框预测头、IoU 预测头和 DTC 预测头, 共同构成长期-短期-轨迹跟踪框架。

1.3.1 目标框预测头和 IoU 预测头

对于目标框预测部分而言, Transformer 解码的特征 f^{dc} 将被送到目标框预测头, 通过计算两个角点概率分布的期望得到当前搜索帧的预测框坐标。

为使得模型适应跟踪过程中的外观渐变, 算法

通过选择可靠特征参考来保持短期特征缓存池的最新状态, 即对于每个预测的边界框 box_x , IoU 预测头通过结合 f^{dc} 与 box_x , 经由全连接层产生 IoU 预测值 iou_x , 使用该 IoU 预测值与设定的 IoU 阈值比较, 决定是否将当前搜索帧 z_x 作为新的短期参考特征。

为解决上述传统模板更新的结构无法在目标发生剧烈形变时对目标进行更新, 易导致跟踪漂移, 引发真实目标丢失的问题, 所提出算法引入了 DTC 预测头。

1.3.2 DTC 预测头模型构建

本文引入最小成本流算法, 以初始帧节点信息为源点, 以当前搜索帧信息节点为汇点, 在当前轨迹参考缓存池从源点到汇点的全部轨迹 TRACK 中, 寻求最优轨迹 T^* 各节点间的流量分配方案, 进而计算得到搜索帧与其他帧间的轨迹匹配程度 dte_x 。

首先, 在 DTC 预测头部分, 将 Transformer 编码器提取出的各帧图片外观特征 $f_{z_i}^e$ 传入 DTC 预测头提取帧特征信息, 并以 iou_x 作为置信度信息, 与帧空间信息集合 box_x 相结合, 构建帧节点信息集合 $D = \{d_1, d_2, \dots, d_i\}$, 其中 $d_i = (f_{z_i}^e, \text{box}_i, \text{iou}_i)$ 为第 i 帧参考帧节点信息表示. 定义每帧图片的特征信息、空间信息和置信度信息集合表示流图节点、边缘连接跨帧的信息. 然后, 将帧节点集合 D 传入 MLP 回归节点间的连接概率, 作为最小成本流网络中的转换成本 c^{tran} , 与由 IoU 预测值 iou_x 定义的检测成本 c^{det} , 进入成本 c^{en} 以及退出成本 c^{ex} 一起构成代价函数 C . 最后, 通过构建最小代价网络流模型得到最优解 x^* , 并在反向传播的过程中学习代价函数 C . 在测试时, 模型通过学习好的代价函数 C 在最小成本流网络生成最优解 x^* , 从而得到最优轨迹 T^* 以及当前搜索帧与最优轨迹 T^* 的轨迹关联程度 dte_x 。

所提出算法将轨迹关联程度 dte_x 作为 DTC 预测头的输出, 通过与设定的 DTC 阈值进行对比, 决定是否对轨迹参考缓存池进行更新。

DTC 预测头的目标是根据贝叶斯规则 $P(T|D) = \frac{P(D|T)P(T)}{P(D)}$, 在参考帧节点集合中寻求一组 k 个参考帧节点组成的轨迹序列 T , 使得在给定输入参考帧节点集合 D 时 T 的后验概率最大. 其中: T 为 D 的子集, 即 $T = \{d_{t_1}, d_{t_2}, \dots, d_{t_k} | d_{t_k} = d_i \text{ for some } d_i \in D\}$, d_{t_k} 为 T 中第 k 个参考帧节点的信息集合; $P(D)$ 为一个不影响解的常数. 假设输入的参考帧节点 d_i 是相互独立的, 则 DTC 预测头的目标是优化

$$T^* = \arg \max_{\text{TRACK}} P(T) \cdot \prod_i P(d_i|T). \quad (12)$$

这里: $P(d_i|T)$ 为在轨迹 T 内观察到检测 d_i 的似然概率, $P(T)$ 为轨迹空间中观测到轨迹 T 的概率. 为简化该模型, 限制节点间的状态依赖关系, 所提出算法对轨迹 T 作一阶马尔可夫假设, 则 $P(T)$ 可分解为

$$P(T) = P_{\text{en}}(d_{t_1}) \left(\prod_{j=1}^{k-1} P_{\text{tran}}(d_{t_{j+1}}|d_{t_j}) \right) P_{\text{ex}}(d_{t_k}). \quad (13)$$

其中: $P_{\text{en}}(d_{t_1})$ 和 $P_{\text{ex}}(d_{t_k})$ 分别为起始节点 d_{t_1} 和终止节点 d_{t_k} 的观测概率, 表示在长为 k 的轨迹 T 中, 初始帧节点 d_{t_1} 处进入, 搜索帧节点 d_{t_k} 退出; $P_{\text{tran}}(d_{t_{j+1}}|d_{t_j})$ 为帧节点 $d_{t_{j+1}}$ 和帧节点 d_{t_j} 的节点转移概率.

对最优轨迹概率公式 T^* 整理后, 可得到

$$T^* = \arg \max_{\text{TRACK}} P_{\text{en}}(d_{t_1}) \left(\prod_{j=1}^{k-1} P_{\text{tran}}(d_{t_{j+1}}|d_{t_j}) \right) \times P_{\text{ex}}(d_{t_k}) \times \prod_i P(d_i|T). \quad (14)$$

对该目标函数进行负对数变换, 整理可得到

$$-\ln T^* = \arg \min_{\text{TRACK}} \left[-\ln P_{\text{en}}(d_{t_1}) - \sum_{j=1}^{k-1} \ln P_{\text{tran}}(d_{t_{j+1}}|d_{t_j}) - \ln P_{\text{ex}}(d_{t_k}) - \sum_{i \in D} \ln P(d_i|T) \right]. \quad (15)$$

引入向量 $x = (x_1, \dots, x_n, x_{n+1}, \dots, x_{n+m}, x_{\text{en}}, x_{\text{ex}})$ 表示 D 中参考帧节点、节点间的边和输入输出节点的决策值. 其中: n 为流图中帧节点的个数, m 为流图中跨帧的边数, x_{en} 和 x_{ex} 分别为输入输出参考帧节点的决策值. 结合流量守恒约束以及容量约束可将上述目标函数转化为最小成本流模型, 如下所示:

$$\begin{aligned} x^* &= \arg \min_{x \in X} C^T x; \\ \text{s.t. } Ax &= b, \quad Gx \leq h. \end{aligned} \quad (16)$$

其中: C 为与决策向量 x 相对应的代价矩阵, 可写为 $C = (c_1^{\text{det}}, c_2^{\text{det}}, \dots, c_n^{\text{det}}, c_{n+1}^{\text{tran}}, \dots, c_{n+m}^{\text{tran}}, c^{\text{en}}, c^{\text{ex}})$; $A \in R^{m \times n}$ 、 $b \in R^m$ 、 $G \in R^{m \times n}$ 和 $h \in R^m$ 分别为定义等式约束和不等式约束的矩阵和向量.

对于代价函数 C 而言, 其每部分具体定义如下:

1) 检测成本 c_i^{det} : 检测 d_i 成本为 $c_i^{\text{det}} = -\hat{\text{IoU}}$, 其中 $\hat{\text{IoU}}$ 为 IoU 预测头计算输出的 IoU 预测值;

2) 进入成本 c^{en} /退出成本 c^{ex} : 训练中更新的参数, 表示参考帧节点进入和退出轨迹的代价;

3) 转换成本 c_i^{tran} : 对于帧节点 d_i 和 d_{i+1} 而言, 两个节点间的转换成本被定义为 $c_i^{\text{tran}} = P(d_{i+1}|d_i)$

$= \text{MLP}(d_{i+1}, d_i; w)$, 其中 w 为 MLP 网络的可学习参数.

由此可得到当前代价函数下的最优解 x^* , 其中 $x^* = (x_1^*, \dots, x_n^*, x_{n+1}^*, \dots, x_{n+m}^*, x_{\text{en}}^*, x_{\text{ex}}^*)$, 且 $x^* \in \{0, 1\}^{n+m+2}$.

1.3.3 基于最小成本流的反向传播算法

为学习第 1.3.2 节中的代价矩阵 C , 需要计算损失函数 L 对代价矩阵中参数 w 的梯度 $\frac{dL}{dw} = \frac{dL}{dx^*} \frac{dx^*}{dc} \frac{dc}{dw}$. 然而, 对于 $\frac{dx^*}{dc}$ 的计算面临以下问题:

- 1) Argmin 算子的微分困难;
- 2) 线性规划解的约束复杂性阻碍梯度计算;
- 3) 整数线性规划解的离散性.

所提出算法参考 Amos^[13] 中的计算方法, 通过在式 (16) 的原线性目标中加入 Tikhonov 阻尼项 γ ($\gamma > 0$), 构建二次规划函数 $f(x) = C^T x + \gamma x^T I x$, 从而利用最优 KKT 条件构建 KKT 方程组并求解最优值, 实现反向传播, 具体如下所示:

$$\begin{aligned} x^* &= \arg \min_{x \in X} x^T Q x + C^T x; \\ \text{s.t. } Ax &= b, \quad Gx \leq h. \end{aligned} \quad (17)$$

其中: $Q = \gamma I$, I 为单位矩阵, 且 $\gamma > 0$. 式 (17) 目标函数为严格凸函数, 约束条件为线性约束, 可通过 KKT 条件确定最优解 x^* , 且 $x^* \in [0, 1]^{n+m+2}$ 为连续值.

为构建 KKT 方程组, 对式 (17) 构建拉格朗日函数, 有

$$L(x, \lambda, v) = x^T Q x + C^T x + \lambda^T (Gx - h) + v^T (Ax - b). \quad (18)$$

根据 KKT 条件可构建如下方程:

- 1) 原始可行性: $Ax = b, Gx \leq h$;
- 2) 对偶可行性: $\lambda \geq 0$;
- 3) 互补松弛性: $\lambda^T (Gx - h) = 0$;
- 4) 梯度条件: $\nabla_x L(x, \lambda, v) = 0$.

根据上述条件构建 KKT 方程组并求解, 可得到在当前代价函数 C 下的最优解 x^* , 以及反向传播过程中所需要的导数 $\frac{dx^*}{dc}$, 完成训练过程.

定义当前搜索帧 z_x 对应节点的轨迹匹配程度 dte_x 为最优轨迹中各边观测概率的均值, 则 DTC 预测头的输出值 dte_x 可表示为

$$\text{dte}_x = \frac{1}{k} \sum_{i=1}^{k-1} x_{t_i}^*, \quad (19)$$

其中 $x_{t_i}^*$ 为最优轨迹 T^* 中参考帧节点 $d_{t_i}^*$ 与 $d_{t_{i+1}}^*$ 的边概率值, 且 $x_{t_i}^* \in x_N^*$.

1.4 模板更新

所提出算法涉及的参考模板包括长期参考模板、短期参考模板和轨迹参考模板. 其中: 长期参考模板为初始帧 z_0 , 短期参考模板和轨迹参考模板分别通过 IoU 预测头和 DTC 预测头选择符合要求的帧更新模板.

对于使用 IoU 预测头得到的预测结果 iou_x , 当 iou_x 大于阈值 iou_p 时, 考虑更新短期参考缓存池, 阈值 iou_p 的设置参考 ATOM^[14] 的实验, 取 $iou_p = 0.75$. 即对于 IoU 预测头得到的预测结果 iou_x 而言, 当满足如下关系时, 对短期参考缓存池进行更新:

$$iou_x \geq iou_p. \quad (20)$$

对于 DTC 预测头分支而言, 当搜索帧的轨迹关联程度 dte_x 大于阈值 dte_p 时, 则考虑更新轨迹参考缓存池. 通过对该阈值设计实验可以发现, 当 $dte_p = 0.9$ 时会取得较好的跟踪结果, 即当满足如下关系时, 所提出算法将对轨迹参考缓存池进行更新:

$$dte_x \geq dte_p. \quad (21)$$

设置短期参考缓存池容量为 3 帧, 轨迹参考缓存池容量为 30 帧. 当缓存池存满时, 对于短期参考缓存池而言, 将时间最早的一帧参考帧剔除, 以此保证该缓存池所存放的参考帧可提供最新的特征参考; 对于轨迹参考缓存池而言, 则将 dte 值得分最低的一帧参考帧剔除, 以此保证该缓存池内所存放参考帧信息能够最大程度地还原目标移动的真实轨迹.

2 实验分析

2.1 实验配置与细节

本文实验在 NVIDIA RTX 3090 GPU 上进行, 显卡驱动版本为 525.125.06, 操作系统为 64 位 Ubuntu 20.04, CUDA 版本为 11.3, 使用 Python 3.8.18 和 PyTorch 1.7.1 实现. 为保证模型的泛化性, 采用 TrackingNet^[15]、COCO^[16]、GOT-10k^[17] 和 LaSOT^[18] 数据集进行训练, 在 GOT-10k、NfS30、OTB100^[19]、UAV123^[20]、LaSOT 和 TrackingNet 共 6 个数据集上进行测试. 从搜索帧中裁出 52 倍于目标面积的搜索区域图像块, 并将其分辨率调整为 320×320 像素, 参考帧采用相同的裁剪流程. 目标预测的损失由边界框预测值与真值间的 GIoU 损失和 $L1$ 损失组合来定义, 超参数 $\lambda_{giou} = 2$, $\lambda_{L1} = 5$. IoU 预测和 DTC 预测的损失由 $L2$ 损失定义, 两个预测头损失函数的超参数分别为 λ_{iou} 和 λ_{dte} , 数值为 2. 整个训练过程对特征提取模块、Transformer 模块和 Head 模块训练 500 个 epoch, 训练数据的最小单元由 1 张长期参考帧、3 张短期参考帧、30 张轨迹参考帧和 1 张搜索帧组

成, 且短期参考帧更新阈值 $iou_p = 0.75$, 轨迹参考更新阈值 $dte_p = 0.9$. 每个批次采样 32 组最小单元, 每个 epoch 采样 60000 组最小单元. 使用 AdamW 优化器对网络进行优化, 权值衰减系数为 10^{-4} . 学习率在第 400 个 epoch 后下降 10 倍.

2.2 消融实验

为验证 SIA 模块、SQR 策略和 LST 框架的有效性, 在 GOT-10k 上对各模块设计不同模块的消融实验, 实验结果如表 1 所示, 其中“√”表示启用模块. 由表 1 可见, 各模块对于算法的结果产生不同程度的积极作用.

表1 不同模块消融实验

实验	LST	SIA	SQR	AO	SR _{0.5}	SR _{0.75}
1	—	—	—	0.684	0.789	0.591
2	√	—	—	0.692	0.806	0.616
3	√	√	—	0.716	0.821	0.645
4	√	√	√	0.736	0.845	0.679

为探究 DTC 预测头中不同阈值对性能的影响, 在 GOT-10k 数据集上对不同阈值设计实验, 结果以 AO 作为评价指标, 实验情况如表 2 所示.

表2 DTC 阈值分析

DTC阈值	0.600	0.700	0.750	0.800	0.850	0.900	0.950
AO	0.685	0.689	0.687	0.689	0.688	0.692	0.687

由表 2 可见, 在该组实验中, 当阈值为 0.900 时 AO 值达到了最高值, 表明在该阈值下算法可达到最佳性能, 因此, 所提出算法选择 0.900 作为 DTC 预测头的更新阈值 dte_p .

为了探究 SIA 模块在 BottleNeck-SIA 中的位置对算法结果的影响, 本文设计 4 组对比实验并在 Got-10k 数据集上进行评估. 实验分别在以下位置插入 SIA 模块: 1) BottleNeck 残差块第 1 个 1×1 卷积层之前; 2) 3×3 卷积层之前; 3) 第 2 个 1×1 卷积层之前; 4) 第 2 个 1×1 卷积层之后. 实验结果如表 3 所示.

表3 SIA 不同嵌入位置分析

位置	AO	SR _{0.5}	SR _{0.75}
1	0.689	0.803	0.602
2	0.693	0.809	0.622
3	0.716	0.821	0.645
4	0.704	0.810	0.632

分析表 3 的实验结果可以看出, 当 SIA 模块处于位置 3, 即第 2 个 1×1 卷积层之前时, AO 与 SR_{0.75} 均优于其他位置的结果, 表明该模块在此位置对于

局部特征提取的能力较佳,实现了较好的跟踪效果.

为验证 SQR 模块在不同解码层数中的性能差异,对其设计 3 组实验,分别验证解码层为 1、2 和 3 时,算法的性能差异,结果如表 4 所示.

表4 SQR 策略不同层分析

层数	AO	SR _{0.5}	SR _{0.75}
1	0.716	0.821	0.645
2	0.724	0.829	0.654
3	0.736	0.845	0.679

由表 4 实验数据可见, SQR 对于深层解码器具有正向作用,可通过聚合浅层信息与深层信息,纠正前层传播时产生的错误查询,将训练重点放在解码器的后期阶段,提高算法精度和成功率.

2.3 对比实验

为进一步验证所提出算法的跟踪性能,将所提出算法和较为先进的目标跟踪算法在 GOT-10k、NfS30、OTB100、UAV123、LaSOT 和 TrackingNet 共 6 个数据集上与经典算法进行对比实验.

2.3.1 GOT-10k 基准评估

GOT-10k 是一个专为通用目标跟踪设计的大规模基准数据集.将所提出算法与 GRM、Video、OSTrack、MixFormer、SparseTT、SwinTrack、AiATrack、CSWinTT 共 8 个经典算法在 GOT-10k 数据集上进行对比,实验结果如表 5 所示.

表5 GOT-10k 数据集算法对比

算法名称	AO	SR _{0.5}	SR _{0.75}
GRM	0.734	0.829	0.704
Video	0.729	0.819	0.698
OSTrack	0.737	0.832	0.708
MixFormer	0.707	0.800	0.678
SparseTT	0.693	0.791	0.638
SwinTrack	0.694	0.780	0.643
AiATrack	0.696	0.800	0.632
CSWinTT	0.694	0.789	0.654
ours	0.736	0.845	0.679

由表 5 可知:所提出算法在与各经典算法的实验数据对比中,在 SR_{0.5} 上均取得了最好的分数;在 AO 上仅与得分最高的 OSTrack 相差 0.1%,而在 SR_{0.75} 上较 OSTrack 低 2.9%,这是因为 GRM 算法是基于纯 Transformer 架构进行设计的,使用 ViT 结构通过自注意力精确分离边缘像素,更适合高精度跟踪需求场景,所提出算法在加入 DTC 预测头后,提升了短期运动连续性,缓解了目标剧烈形变时模板更新不及时的问题,相比于 OSTrack 在 SR_{0.5} 具有明显优势.

2.3.2 TrackingNet 基准评估

TrackingNet 是专为野外复杂场景下的目标跟踪任务设计的.表 6 为所提出算法分别与 STARK-ST50、DTT、TrSiam、TransT 和 SiamAttn 等经典算法在 TrackingNet 测试集上的测试结果.由表 6 可以发现,所提出算法相比于其他算法具有一定的竞争力.

表6 TrackingNet 数据集算法对比

算法名称	AUC	P _{Norm}	P
STARK-ST50	0.813	0.861	0.781
DTT	0.796	0.850	0.789
TransT	0.814	0.867	0.803
TrSiam	0.781	0.829	0.727
SiamAttn	0.752	0.817	—
ours	0.830	0.879	0.805

2.3.3 小规模数据集基准评估

将所提出算法在 NfS30、OTB100 和 UAV123 共 3 个小规模数据集上与 SiamRPN++、SeqTrack、STARK-ST50、TransT 和 PrDiMP50 共 5 个较经典的目标跟踪算法进行对比实验,实验结果如表 7 所示.

表7 NfS30、OTB100 与 UAV123 数据集算法对比

算法名称	NfS30	OTB100	UAV123
SiamRPN++	50.2	69.6	61.3
SeqTrack	—	68.3	68.5
STARK-ST50	65.2	68.5	69.1
TransT	65.7	69.4	69.1
PrDiMP50	63.5	69.6	68.0
ours	68.2	69.7	68.9

由表 7 可知,所提出算法在上述 3 个数据集上与不同架构的算法比较时仍然具有一定的竞争力,虽然在 UAV123 数据集上相较于 STARK-ST50 与 TransT 低 0.2,但是,在 NfS30 和 OTB100 数据集上所提出算法均达到了较优的水平.分析其原因如下: UAV123 作为无人机俯视视角数据集,目标在图像中占比较小; STARK-ST50 可通过高分辨率分支显式保留更高分辨率的特征图来提升细节感知; TransT 则使用交叉注意力细化定位,对于 UAV123 数据集的表现稍优.

2.3.4 LaSOT 基准评估

LaSOT 是计算机视觉领域专为单目标跟踪任务设计的大规模基准数据集.图 4 为所提出算法分别与 CSWinTT、SiamTPN、DiMP、SiamFC++、LTMU、GlobalTrack、SiamGTA、ATOM、SiamMask 和 GFSDCF 共 12 个经典算法在 LaSOT 测试集上不同

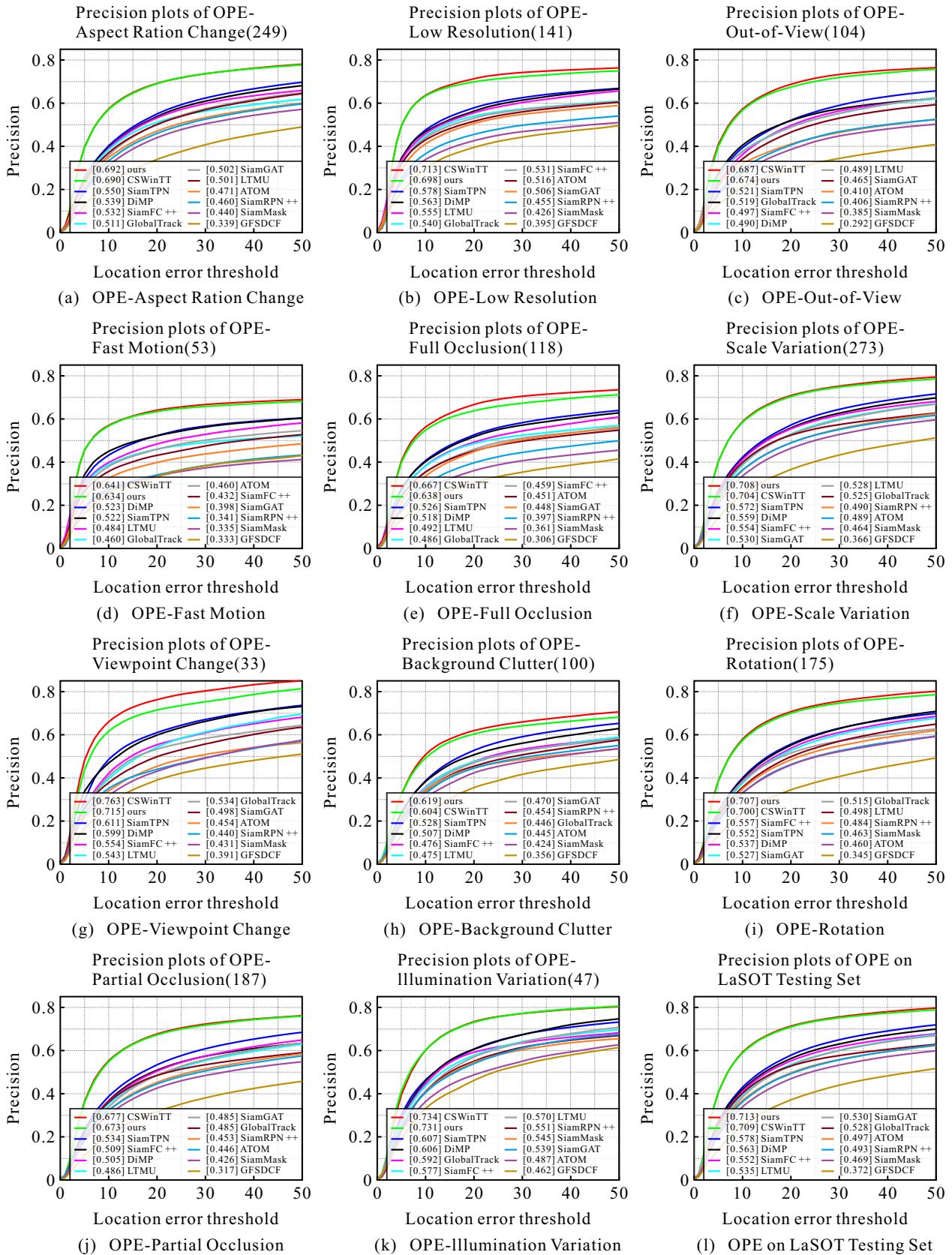


图4 Lasot 数据集算法对比

挑战的精度测试结果. 通过以上实验对比可看出, 所提出算法在各测试中均处于较优水平.

3 结论

本文提出了一种基于空域抑制与轨迹关联优化

的目标跟踪方法. 首先, 算法特征提取部分采用了改进后的 ResNet-50 网络, 引入了空域抑制注意力模块 SIA, 通过能量函数为具有空域抑制效应的神经元优化权重分配, 增强了对目标特征的关注强度; 然后, 引入了选择性查询回忆策略 SQR, 用于在

Transformer 解码阶段跨层融合中间查询, 抑制级联误差传播; 接着, 所提出算法构建了长期-短期-轨迹框架 LST, 该框架将初始帧作为长期参考, 通过 IoU 预测值筛选可靠中间帧更新短期参考; 最后, 为避免碎片化轨迹, 联合最小成本流算法建模全局轨迹关联性, 通过连续二次规划约束更新轨迹参考模板. 所提出算法在 GOT-10k、LaSOT、TrackingNet、NfS30、UAV123 和 OTB100 等公开数据集上进行了实验评估, 结果验证了所提出算法的良好性能.

参考文献 (References)

- [1] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]. IEEE Conference on Computer Vision and Pattern Recognition. Boston, 2015: 3431-3440.
- [2] Li B, Yan J J, Wu W, et al. High performance visual tracking with Siamese Region proposal network[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, 2018: 8971-8980.
- [3] 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. Munich, 2018: 101-117.
- [4] Xu Y D, Wang Z Y, Li Z X, et al. SiamFC ++: Towards robust and accurate visual tracking with target estimation guidelines[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, 34(7): 12549-12556.
- [5] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16×16 words: Transformers for image recognition at scale[J/OL]. 2020, arXiv: 2010.11929.
- [6] Liu Z, Lin Y T, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]. IEEE/CVF International Conference on Computer Vision. Montreal, 2021: 9992-10002.
- [7] Yan B, Peng H W, Fu J L, et al. Learning spatio-temporal transformer for visual tracking[C]. IEEE/CVF International Conference on Computer Vision. Montreal, 2021: 10428-10437.
- [8] Ye B T, Chang H, Ma B P, et al. Joint feature learning and relation modeling for tracking: A one-stream framework[C]. European Conference on Computer Vision. Tel Aviv, 2022: 341-357.
- [9] Danelljan M, van Gool L, Timofte R. Probabilistic regression for visual tracking[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, 2020: 7183-7192.
- [10] 仲训泉, 范东嘉, 仲训昱, 等. 融合多模板注意力深度网络的自适应目标框跟踪算法[J]. *控制与决策*, 2024, 39(4): 1123-1132.
(Zhong X G, Fan D J, Zhong X Y, et al. Adaptive target box tracking algorithm by integrating multi-template attention deep network[J]. *Control and Decision*, 2024, 39(4): 1123-1132.)
- [11] 胡一凡, 李国齐, 吴郁杰, 等. 脉冲神经网络研究进展综述[J]. *控制与决策*, 2021, 36(1): 1-26.
(Hu Y F, Li G Q, Wu Y J, et al. Spiking neural networks: A survey on recent advances and new directions[J]. *Control and Decision*, 2021, 36(1): 1-26.)
- [12] Yu R, Du D W, LaLonde R, et al. Cascade transformers for end-to-end person search[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans, 2022: 7257-7266.
- [13] Ji Y, Bai H, Ge C, et al. Amos: A large-scale abdominal multi-organ benchmark for versatile medical image segmentation[J]. *Advances in Neural Information Processing Systems*, 2022, 35: 36722-36732.
- [14] Danelljan M, Bhat G, Khan F S, et al. ATOM: Accurate tracking by overlap maximization[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, 2019: 4655-4664.
- [15] Muller M, Bibi A, Giancola S, et al. TrackingNet: A large-scale dataset and benchmark for object tracking in the wild[C]. Proceedings of the European Conference on Computer Vision. Munich, 2018: 300-317.
- [16] Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: Common objects in context[C]. *Computer Vision — ECCV 2014*. Zurich, 2014: 740-755.
- [17] Huang L H, Zhao X, Huang K Q. GOT-10k: A large high-diversity benchmark for generic object tracking in the wild[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021, 43(5): 1562-1577.
- [18] Fan H, Lin L T, Yang F, et al. LaSOT: A high-quality benchmark for large-scale single object tracking[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, 2019: 5369-5378.
- [19] Madge S N, Prabhakaran V C, Shome D, et al. Orbital tuberculosis: A review of the literature[J]. *Orbit*, 2008, 27(4): 267-277.
- [20] Yu X U, Jian C. UAV object tracking for air-ground targets based on status detection and Kalman filter[J]. *Acta Aeronautica et Astronautica Sinica*, 2024, 45(16): 232-251.

作者简介

陈志旺 (1978-), 男, 副教授, 博士, 主要研究方向为运动物体目标跟踪与检测、多旋翼飞行器导航与控制, E-mail: czwaaaron@ysu.edu.cn;

高一铭 (2000-), 女, 硕士生, 主要研究方向为计算机视觉中的目标跟踪, E-mail: 2083664580@qq.com;

吕昌昊 (1996-), 男, 硕士生, 主要研究方向为智能电网的优化和控制, E-mail: 316998054@qq.com;

刘禄阳 (2001-), 男, 硕士生, 主要研究方向为计算机视觉中的目标跟踪, E-mail: 2132915520@qq.com;

彭勇 (1963-), 男, 教授, 博士, 博士生导师, 主要研究方向为生物机器人控制、计算机视觉中的目标检测, E-mail: PY81@sina.com.