http://bhxb.buaa.edu.cn jbuaa@buaa.edu.cn
DOI: 10.13700/j.bh.1001-5965.2024.0020

基于模板更新和双特征增强的视觉跟踪算法

丁奇帅[1,2,3]，雷帮军[1,2,3,*]，牟乾西[1,2,3]，吴正平[1,2,3]

(1. 水电工程智能视觉监测湖北省重点实验室，宜昌 443002；2. 三峡大学 计算机与信息学院，宜昌 443002；
3. 水电工程视觉监测宜昌市重点实验室，宜昌 443002)

摘 要：针对视觉跟踪中由于目标形变、翻转和遮挡而导致的跟踪失败问题，提出了一种基于图像结构相似性的模板更新算法，通过动态更新模板以适应目标在跟踪过程中的变化。同时，基于SiamMask网络设计了跟踪特征增强模块和分割特征增强模块。跟踪特征增强模块包括非局部操作和卷积下采样，用于建立上下文关联，增强目标特征，抑制背景干扰，提高跟踪鲁棒性，解决由于目标被遮挡而导致的特征减弱问题。分割特征增强模块引入卷积块注意力模块和可变形卷积，以提高网络对通道和空间特征的捕捉能力，自适应地学习目标的形状和轮廓信息，提升网络对跟踪目标的分割精度，进而提高跟踪准确率。实验表明：所提算法表现良好且稳定，与SiamMask相比，在VOT2016、VOT2018和VOT2019数据集上期望平均重叠率分别提升了0.052、0.053和0.025，鲁棒性分别提升了0.06、0.079和0.156，且达到了平均每秒91帧的实时速度。

关 键 词：目标跟踪；图像分割；SiamMask；模板更新；特征增强
中图分类号：TP391.4
文献标志码：A **文章编号**：1001-5965(2026)04-1096-11

目标跟踪在自动驾驶、体育分析、搜索救援、工业自动化和智能交通等领域有着广泛应用。跟踪的目的是在视频序列中，根据初始帧目标的位置，持续预测后续帧中目标的位置。现有的跟踪算法在部分场景中取得了不错的跟踪效果，但在目标形变、翻转和遮挡[1]等复杂场景下仍然面临着准确率下降、鲁棒性变差等诸多挑战。

当前，大多数跟踪算法都基于深度学习，尤其以孪生网络为范式，SiamFC[2]首次将孪生网络应用于目标跟踪领域，提升了跟踪算法的速度和准确率。Li等将区域提议网络(regional proposal network, RPN)从目标检测引入到孪生网络中，提出了SiamRPN[3]，以更精确地预测目标的位置。DaSiamRPN[4]引入了干扰感知网络(interference perception network, IPN)，着重解决正负样本不均衡问题。SiamRPN++[5]提出了深度互相关，解决搜索帧和模板帧参数数量不平衡的问题。为拓展孪生网络的产出能力，Wang等将SiamRPN[3]与图像分割技术相结合，提出了SiamMask[6]，该算法能够在跟踪的同时对目标进行像素级的分割，通过分割结果来实现更精确的跟踪。然而，以上算法在跟踪时都固定使用第1帧给定的模板图像，当目标遮挡、尺度变化或形状变化时，容易因搜索图像与模板图像不匹配而跟踪失败。

为适应目标的外观变化，一些跟踪算法开始采用模板更新策略。Meta-Tracker[7]采用元学习对模板图像在线更新，该算法可能会出现过度适应的问题，模型泛化到新的跟踪场景时效果变差。DSiam[8]提出了动态孪生网络，通过快速傅里叶变换的学习模型，使网络能在线学习目标的外观变化，但在面

收稿日期：2024-01-11；**录用日期**：2024-01-28；**网络出版时间**：2024-02-27 13:29
网络出版地址：link.cnki.net/urlid/11.2625.V.20240226.1633.001
基金项目：国家自然科学基金(61871258)；水电工程智能视觉监测湖北省重点实验室建设项目(2019ZYYD007)；宜昌市科技研究与开发项目(A201130225)
* **通信作者**. E-mail：bangjun.lei@ieee.org

引用格式：丁奇帅，雷帮军，牟乾西，等. 基于模板更新和双特征增强的视觉跟踪算法[J]. 北京航空航天大学学报，2026，52(4)：1096-1106. DING Q S，LEI B J，MOU Q X，et al. Visual tracking algorithm based on template updating and dual feature enhancement[J]. Journal of Beijing University of Aeronautics and Astronautics，2026，52(4)：1096-1106 (in Chinese).

对复杂背景和动态环境时, 鲁棒性会受到极大影响。UpdateNet[9] 通过跳跃连接形式实现残差学习, 输出对下一帧模板的预测, 该算法使用积累模板的特征, 当积累模板误差较多时, 会产生负面影响。

最近, 研究人员开始研究无锚框跟踪算法, 以提高目标状态估计的准确性, 如 SiamFC++[10]、SiamBAN[11] 和 SiamCAR[12] 等, 这些算法为目标跟踪提供了新思路, 但仅能进行跟踪, 不能像 SiamMask 一样实现对目标分割和跟踪的多任务学习。

综合上述分析, 本文基于 SiamMask 在目标跟踪上的优势及在模板更新方面的不足, 对其进行改进, 以提升其在复杂场景下的跟踪性能, 主要贡献如下:

1) 设计了更为简单高效的模板更新算法, 仅消耗少量计算资源, 无须引入新的网络, 通过计算当前预测的结果图像和模板图像之间的结构相似性[13](structural similarity, SSIM)来判别是否需要更新模板, 更新模板时从历史跟踪结果中匹配出更适合当前目标形态的图像作为新模板参与后续跟踪。

2) 设计了跟踪特征增强模块, 该模块包含非局部操作[14] 和卷积下采样, 通过捕获全局信息来有效增强目标所在位置的特征强度, 抑制背景干扰, 提升跟踪的鲁棒性。

3) 设计了分割特征增强模块, 该模块结合了卷积块注意力模块[15](convolutional block attention module, CBAM)和可变形卷积网络(deformable convolution network, DCN)[16], 以更好地捕捉通道和空间特征, 自适应地学习目标的形状和轮廓信息, 提高分割的精度, 进而提高跟踪的准确性。

4) 在多个具有挑战性的跟踪数据集(VOT2016、VOT2018 和 VOT2019)上进行实验, 相比于基准算法 SiamMask, 本文算法在鲁棒性和期望平均重叠率(expected average overlap, EAO)指标上取得了显著的提升。

1　本文算法

算法整体框架如图 1 所示。网络的输入由 2 部分构成: ①模板图像, 大小为 127×127×3; ②搜索图像, 大小为 255×255×3。特征提取网络采用 ResNet-50 的前 4 个阶段, 每个阶段的特征图对应图 1 中的 conv1、conv2、conv3 和 conv4, Adjust 卷积块的作用是将 conv4 的特征通道从 1 024 调节到 256。conv2 经过跟踪特征增强模块处理后与 Adjust 的结果相加, 通过深度互相关输出 17×17×256 的响应得分图, 得分最高点所在位置的一列特征矩阵作为候选响应窗口(response of candidate window, RoW), 大小为 1×1×256, RoW 通过特征融合来产生更准确的目标掩码。特征融合网络如图 2 所示。RoW 经过反卷积变换形状, 通过多次卷积和上采样, 与经过分割特征增强模块处理的低层特征 conv1、conv2 和 conv3 进行融合, 最终输出精确的分割和跟踪结果。

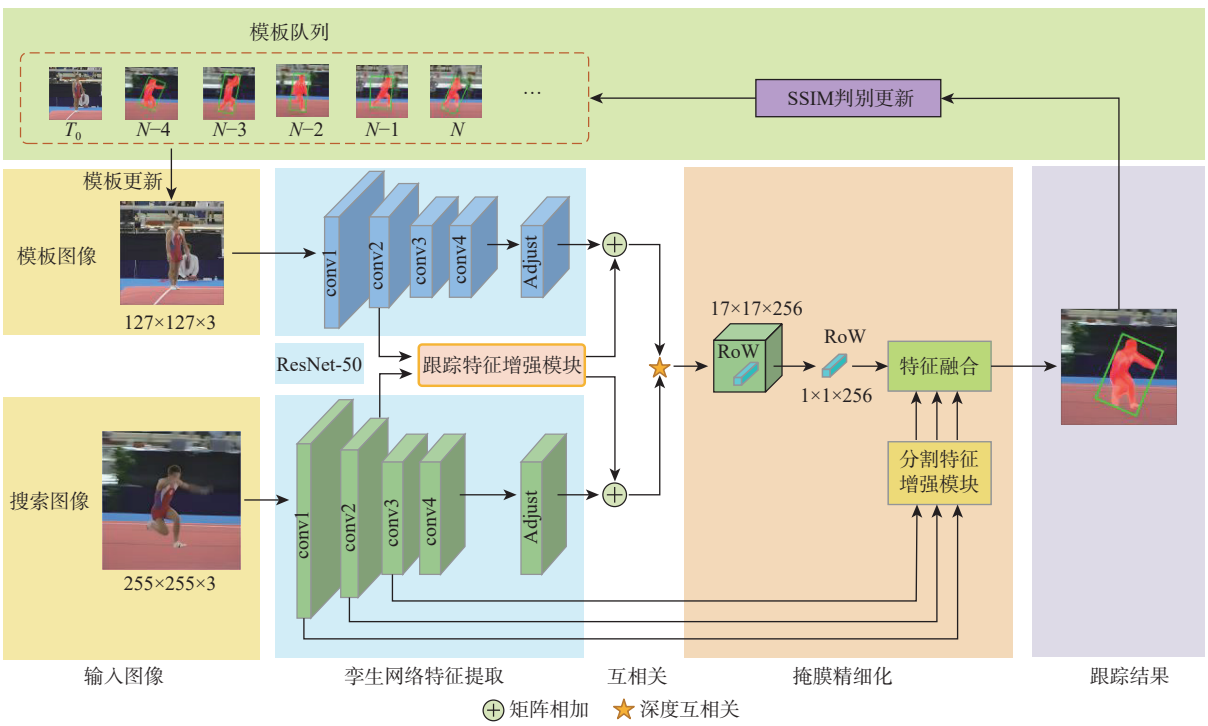

图 1　基于模板更新和双特征增强的视觉跟踪算法框架

Fig. 1　Framework of visual tracking algorithms based on template updating and dual-feature enhancement

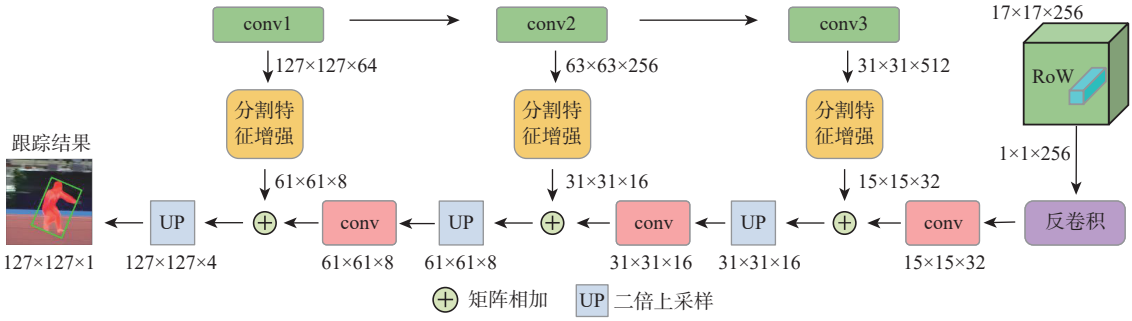

图 2 特征融合网络

Fig. 2 Feature fusion network

1.1 基于结构相似性判别的模板更新算法

跟踪过程中，目标开始变化的前几帧依赖跟踪网络的泛化能力，仍然可以正确跟踪，随着目标变化越来越大，会逐渐跟踪失败，不同情况下目标的变化过程如图 3 所示。

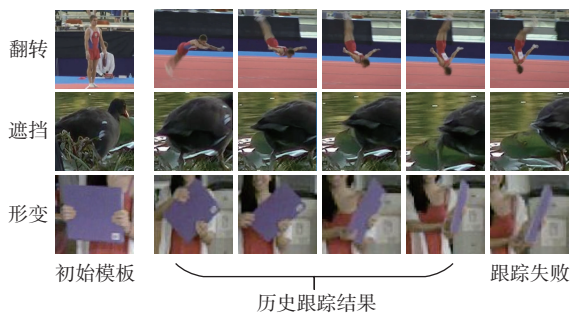

图 3 目标在跟踪失败前的变化过程

Fig. 3 Process of object changes before tracking failure

图 3 中，运动员在经历翻转后跟踪失败，失败帧形状和初始模板相似度很低，而和跟踪失败的前 2 帧更接近；鸟被遮挡和书在形变时同样符合该规律，利用该特性，本文算法从历史跟踪结果中选取模板图像。同时，为了防止目标恢复为初始形状时，更新后的模板无法恢复的问题，设计了模板初始化，达到条件时，将当前模板恢复为初始模板。具体实现过程如下：

1）判别条件。计算当前帧跟踪网络的预测结果图像和模板图像的 SSIM 得分，若当前得分 S_n 比上一帧预测结果图像和模板图像的 SSIM 得分 S_{n-1} 小，且差值大于阈值 Δ_1，则认为当前帧中的目标发生了明显变化，需要更新模板；同时，由于对不同场景的跟踪，SSIM 得分分布情况不同，正确跟踪目标时，跟踪结果和模板得分总是在历史 SSIM 得分的平均值附近保持平稳。因此，增加约束条件，当前 SSIM 得分 S_n 小于均值 S_{m} 且差值大于阈值 Δ_2 时，认为当前跟踪出现了偏差，需要更新模板，从而得到模板更新的判别条件：

$$\begin{cases} S_{\mathrm{m}} - S_n > \Delta_1 \\ S_{n-1} - S_n > \Delta_2 \end{cases} \tag{1}$$

2）算法流程。模板更新流程如算法 1 所示。

算法 1 基于 SSIM 判别的模板更新算法。

定义： 队列 Q，保存历史跟踪图像，长度为 N；score 为当前预测结果图像和模板图像 SSIM 得分；列表 S，保存历史 SSIM 得分；S_{m} 为列表 S 中得分均值；T 表示模板；T_0 为初始帧模板；f 为每帧视频序列；$P(\,)$ 为跟踪过程，P 为跟踪预测的结果。

输入： 视频序列 f，初始目标图像和位置（在 T_0 中）。

输出： 分割和跟踪结果（保存在 P 中）。

// 初始化
$T \leftarrow T_0$
$P \leftarrow P(T)$ # 用初始帧模板进行跟踪
// 跟踪过程
for f in video do
    $P \leftarrow P(T)$ # 通过循环对视频序列持续跟踪
    score $\leftarrow$ SSIM(T_0,P) # 计算 SSIM 得分并存入 score
    $Q \leftarrow P$ # 跟踪结果存入队列 Q
    if $f > N$ and S_{m} − score $< \Delta_1$ and $S[f-2]$ − score $> \Delta_2$ then
        for i in Q do # 遍历队列 Q
            best_score $\leftarrow$ score # 假设当前得分为最优得分
            $P \leftarrow P(Q[i])$ # 重新跟踪
            if (score $\leftarrow$ SSIM($Q[i]$,P)) > best_score then
                best_score $\leftarrow$ score # 选出最优模板
                $T \leftarrow Q[i]$
            end
        end
        if best_score < $S[f-1]$ then
            $T \leftarrow T_0$ # 模板初始化
        end
    end
end

使用初始模板对视频序列进行跟踪，当触发

式 (1) 中的更新条件时, 将队列中的图像分别作为新的模板对当前帧进行重新跟踪, 再计算出新的 SSIM 得分, 选择 SSIM 得分最高所对应的图像作为新模板进行后续跟踪, 同时将更新模板后重新跟踪得到的 SSIM 得分和未更新前的得分进行比较, 若得分变高, 则表明更新模板后重新跟踪起到了正面作用, 得分变低则表明起到了负面作用, 此时, 应将模板更换为初始模板, 通过以上流程实现模板更新。

1.2　跟踪特征增强模块

特征提取时, 随着网络层数变深, 特征图的尺寸将会逐渐变小, 难以捕捉到全局信息, 当目标发生遮挡和形变时, 目标的形状等浅层特征丢失严重, 难以进行精准定位。同时, 考虑到在跟踪过程中, 目标的位置和运动不仅存在于局部区域, 而是涉及整个图像, 不同视频帧之间的目标往往具有连续的运动特征, 这些特征需要与周围环境和上下文进行交互, 如图 4 所示。

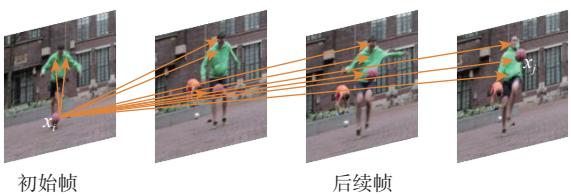

图 4　目标在不同视频帧之间的关联性

Fig. 4　Object correlation between different video frames

本文通过引入非局部操作来建立输入图像中不同像素之间长距离依赖关系, 通过卷积下采样调整特征图大小, 将增强后的特征图与 Adjust 的结果相融合, 从而获取更多的细节信息, 为最终的跟踪过程提供更准确、更稳定的特征表达。所设计的跟踪特征增强模块如图 5 所示。

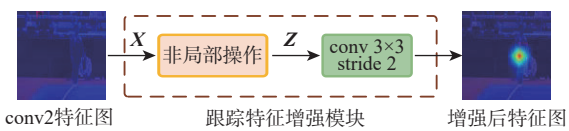

图 5　跟踪特征增强模块

Fig. 5　Tracking feature enhancement module

非局部操作在网络中的传播过程如图 6 所示, 其通过计算输入特征图中每个位置与其余所有位置之间加权和来确定关联性。计算方法如下:

$$\boldsymbol{y}_i = \frac{1}{C(\boldsymbol{x})}\sum_j f(\boldsymbol{x}_i,\boldsymbol{x}_j)g(\boldsymbol{x}_j) \tag{2}$$

式中: $\boldsymbol{x}$为输入特征图; $\boldsymbol{y}_i$为输出i位置的特征值; $\boldsymbol{x}_i$为i位置上与$\boldsymbol{x}$相同维度的向量; $\boldsymbol{x}_j$为j位置上与$\boldsymbol{x}$相同维度的向量; $f(\boldsymbol{x}_i,\boldsymbol{x}_j)$为相似性计算函数, 用于计算$\boldsymbol{x}_i$和$\boldsymbol{x}_j$这 2 个向量之间的相似性; $g(\ )$为特征值计算函数; $C(\boldsymbol{x})$为归一化项, 用于对函数

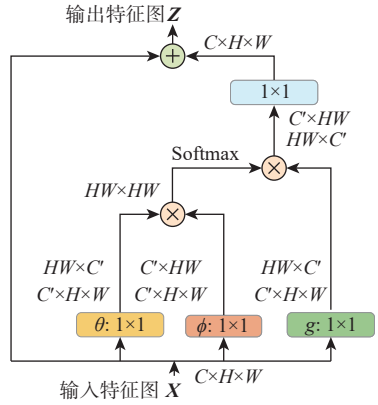

图 6　非局部操作

Fig. 6　Non-local operation

$f(\boldsymbol{x}_i,\boldsymbol{x}_j)$所有值求和, 即

$$C(\boldsymbol{x}) = \sum_j f(\boldsymbol{x}_i,\boldsymbol{x}_j) \tag{3}$$

本文采用嵌入高斯函数来计算 2 个向量的相似性, 公式如下:

$$f(\boldsymbol{x}_i,\boldsymbol{x}_j) = \mathrm{e}^{\theta(\boldsymbol{x}_i)^{\mathrm{T}}\phi(\boldsymbol{x}_j)} \tag{4}$$

为了在卷积网络中易于实现, 特征值计算函数$g(\boldsymbol{x})$、$\theta(\boldsymbol{x})$和$\phi(\boldsymbol{x})$分别由 1×1 卷积映射得出, 即输入特征图$\boldsymbol{x}$乘上对应权重$\boldsymbol{W}$, 公式为

$$\begin{cases} g(\boldsymbol{x}) = W_g\boldsymbol{x} \\ \theta(\boldsymbol{x}) = W_\theta\boldsymbol{x} \\ \phi(\boldsymbol{x}) = W_\phi\boldsymbol{x} \end{cases} \tag{5}$$

1.3　分割特征增强模块

特征融合网络是为了产生更准确的目标掩码, 在图 2 中, RoW 通过反卷积变换后通道数减少, 尺度变大, 这意味着特征图在空间维度上需要更多信息, 但由于通道信息有限, 无法表示更多细节, 在目标分割时, 会形成模糊的分割边界。因此, 本文设计了分割特征增强模块, 旨在将对目标分割最有利的低层特征提取出来, 得到更完整的掩码信息。该模块由 CBAM 模块和 DCN 网络组成, 网络结构如图 7 所示。

图 7　分割特征增强模块

Fig. 7　Segmentation feature enhancement module

CBAM 模块由通道注意力模块和空间注意力模块串联而成, 如图 8 所示。对输入的特征图, 先分别经过平均池化和最大池化, 再通过共享多层感知机和 Sigmoid 激活函数来学习通道权重。将结果分别经过平均池化和最大池化, 通过一个 7×7 的空洞卷积和 Sigmoid 激活函数学习空间各点权重, 最

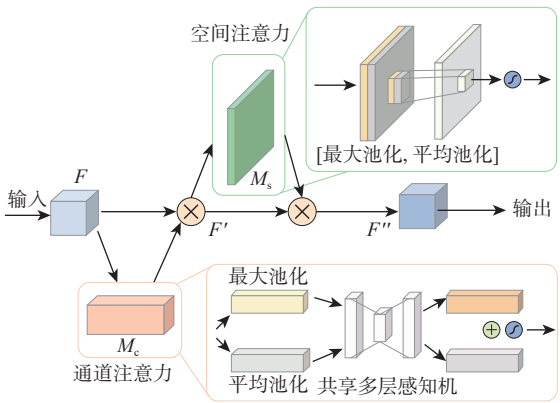

图 8 CBAM 模块

Fig. 8 Convolutional block attention module

终生成权重优化的特征图。

DCN 网络通过学习偏移量使卷积核的形状和大小能够自适应地调整，如图 9 所示。输入特征图通过普通卷积来学习每个卷积核的偏移量，对特征图进行非均匀采样，以适应感受野内特征的空间变化，输出新的特征图。实现过程表示为

$$\boldsymbol{y}(p)=\sum_{k=1}^{K} w_k \boldsymbol{x}(p+p_k+\Delta p_k)\Delta m_k \tag{6}$$

式中：$\boldsymbol{y}(p)$为输出位置p的特征值；$\boldsymbol{x}(p)$为输入特征$\boldsymbol{x}$映射在p位置的特征；K为卷积核个数，本文使用3×3卷积，K的值取9；w_k为k位置的权重；p_k为预先设定偏移量；Δp_k为可学习的偏移量；Δm_k为调节系数，范围为[0,1]，用来约束卷积核学习偏移量的范围。

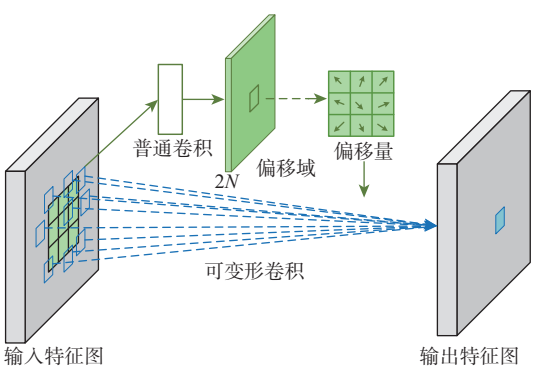

图 9 可变形卷积网络

Fig. 9 Deformable convolution network

2 实验结果及分析

2.1 实验设置

实验平台采用 Ubuntu18.04 操作系统，处理器为 Intel(R) Core(TM) i7-10700K CPU@ 3.80 GHz，显卡为 NVIDIA GTX 3070(8 GB)，采用 Pytorch 深度学习框架，相关依赖为：cuda11.3，cudnn8.3.2，python3.8，pytorch1.12.1。

训练数据集由 COCO、YouTube-VOS、ImageNet-VID 和 ImageNet-Det 组成，训练时使用 ResNet-50 预训练权重，采用 SGD 优化器，动量为 0.9，权重衰减为 10^{-5}，batchsize 为 8，训练 20 个 epoch，前 5 个 epoch 为预热阶段，学习率从 10^{-3} 增长到 5×10^{-3}，后 15 个 epoch 按对数下降到 2.5×10^{-3}。

2.2 测试数据集与评价指标

1）VOT 数据集。跟踪时使用VOT2016、VOT2018 和 VOT2019 这 3 个数据集，3 个数据集均由 60 个视频序列组成，采用旋转框标注，包含尺度变化、遮挡、光照变化、相机运动、运动变化和无定义 6 个视觉属性。使用准确率、鲁棒性和 EAO 作为评价指标。准确率衡量跟踪过程预测框和真实框的交叠程度，值越大，表明准确率越高。鲁棒性衡量跟踪算法的稳定性，值越小，表明稳定性越强。EAO 结合了准确率和鲁棒性，是衡量跟踪算法最重要的指标，值越大，性能越好。

2）DAVIS 数据集。分割时使用 DAVIS2016 和 DAVIS2017 这 2 个数据集，数据集由二进制掩码进行稠密标注，具有像素级别的分割精度。使用不同阈值下的平均交并比(mean IoU，mIoU)作为分割结果的评价指标，值越大，表明分割效果越好。

2.3 对比实验

2.3.1 VOT2016 数据集上的实验结果

1) 不同视觉属性下的结果对比。在 VOT2016 数据集上测试了本文算法和其他跟踪算法在不同视觉属性下的 EAO 得分，测试结果如表 1 和图 10

表 1 VOT2016 数据集上不同视觉属性的实验结果

Table 1 Experimental results of different visual attributes on VOT2016 dataset

跟踪算法	总体EAO得分	EAO得分					
		遮挡	相机运动	尺度变化	光照变化	运动变化	无定义
SiamFC[2]	0.234	0.161	0.191	0.242	0.180	0.231	0.059
MDNet[17]	0.257	0.218	0.238	0.312	0.313	0.252	0.030
C-COT[18]	0.331	0.246	0.249	0.327	0.402	0.354	0.154
SiamRPN[3]	0.344	0.117	0.205	0.280	0.270	0.176	0.065
DaSiamRPN[4]	0.411	0.241	0.280	0.422	0.233	0.294	0.106
SiamMask[6]	0.433	0.325	0.394	0.444	0.463	0.409	0.109
本文	0.485	0.470	0.472	0.527	0.617	0.470	0.104

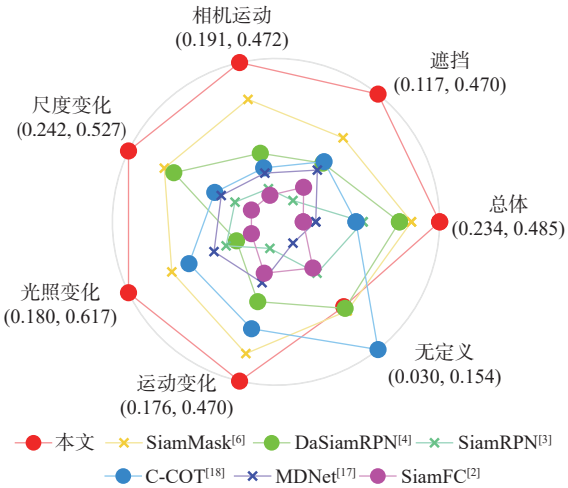

图 10　VOT2016 数据集上视觉属性对比

Fig. 10　Comparison of visual attributes on VOT2016 dataset

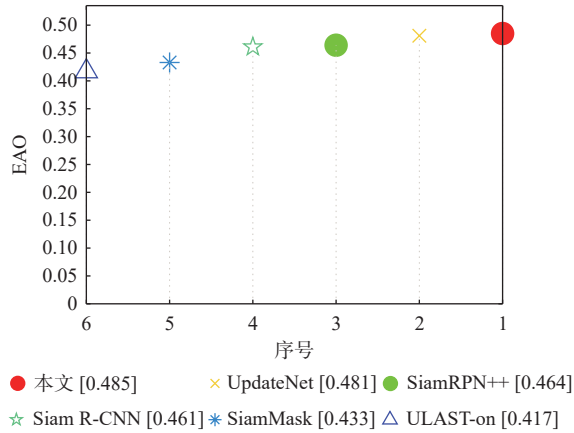

图 11　VOT2016 数据集上 EAO 排名

Fig. 11　EAO rankings on VOT2016 dataset

所示。对比 SiamMask, 在遮挡场景下, 本文算法的 EAO 提升了 0.145, 运动变化场景下提升了 0.078, 尺度变化场景下提升了 0.083, 光照变化场景下提升了 0.154, 相机运动场景下提升了 0.061, 同时也超过了其他算法, 表明本文算法可以有效应对目标形变、翻转和遮挡等复杂场景。

2) 与其他算法总体性能的对比。表 2 显示了本文算法和其他跟踪算法在 VOT2016 数据集上的比较结果。EAO 排名如图 11 所示。本文算法具有最高的 EAO 和最好的鲁棒性, 相比于 UpdateNet, 准确率高出 0.02, 鲁棒性高出 0.056, EAO 高出 0.004; 相比于 Siam R-CNN, 鲁棒性高出 0.019, EAO 高出 0.024; 相比于 SiamRPN++, 鲁棒性高出 0.046, EAO 高出 0.021。可以表明, 本文算法不仅准确跟踪了目标的位置, 而且保持了良好的稳定性。

表 2　不同跟踪算法在 VOT2016 数据集上的结果

Table 2　Results of different tracking algorithms on VOT2016 dataset

跟踪算法	准确率↑	鲁棒性↓	EAO↑
SiamMask[6]	0.622	0.214	0.433
SiamRPN++[5]	0.640	0.200	0.464
UpdateNet[9]	0.610	0.210	0.481
Siam R-CNN[19]	0.645	0.173	0.461
ULAST-on[20]	0.603	0.214	0.417
本文	0.630	0.154	0.485

图 12 展示了部分复杂场景下的跟踪可视化结果。在目标快速运动时, SiamFC 存在跟丢和跟错的情况。SiamMask 的跟踪框有时无法精准锁定目标, 与目标真实形状误差较大, 受背景干扰容易跟踪失败。本文算法在目标发生形变、翻转和遮挡时, 仍能正确跟踪目标, 所产生的跟踪框与目标实际形状可以保持很好的贴合, 与真值框十分接近, 实现了更好的跟踪效果。

2.3.2　VOT2018 数据集上的实验结果

将本文算法在 VOT2018 数据集上与其他跟踪算法进行比较, 结果如表 3 所示, EAO 排名如图 13 所示。鲁棒性上, 本文算法为 0.197, 超越了大多数跟踪算法, 仅次于 SiamFC++。EAO 上, 本文算法达到了 0.433 的最佳分数, 比 SiamFC++高出 0.007, 比 Siam R-CNN 高出 0.025, 且优于其他算法, 可以表明, 本文算法整体性能表现较好。

2.3.3　VOT2019 数据集上的实验结果

在更具有挑战性的 VOT2019 数据集上进行测试, 实验结果如表 4 所示, EAO 排名如图 14 所示。本文算法在准确率、鲁棒性和 EAO 上都取得了最好的结果。相比 SiamRPN++, 本文算法准确率高出 0.002, EAO 高出 0.014, 鲁棒性高出 0.066。

2.4　消融实验

2.4.1　不同模板更新参数的实验结果对比

为验证模板更新算法中不同队列长度 N、阈值 Δ_1 和 Δ_2 对跟踪结果和速度的影响, 在 VOT2018 数据集上进行了实验, 实验结果如表 5 所示。

通过表 5 可以发现, 当队列长度为 10、Δ_1 和 Δ_2 分别为 0.25 和 0.2 时, EAO 取得最好结果 0.396, 分割速度为 82 帧/s; 当队列长度为 5、Δ_1 和 Δ_2 分别为 0.25 和 0.2 时, EAO 为 0.394, 分割速度为 105 帧/s; 相比之下, 二者 EAO 相差仅为 0.002, 但分割速度相差 23 帧/s。综合考虑, 队列长度取 5 时, 分割速度最快且对最终结果提升有利, 因此, 本文在其他实验中使用模板更新算法时, 队列长度均设置为 5, Δ_1 和 Δ_2 设置为 0.25 和 0.2。

2.4.2　跟踪数据集上的消融实验

在 VOT2016、VOT2018 和 VOT2019 这 3 个跟踪数据集上进行消融实验, 用来验证本文所设计模块的有效性, 实验结果如表 6～表 8 所示。其中,

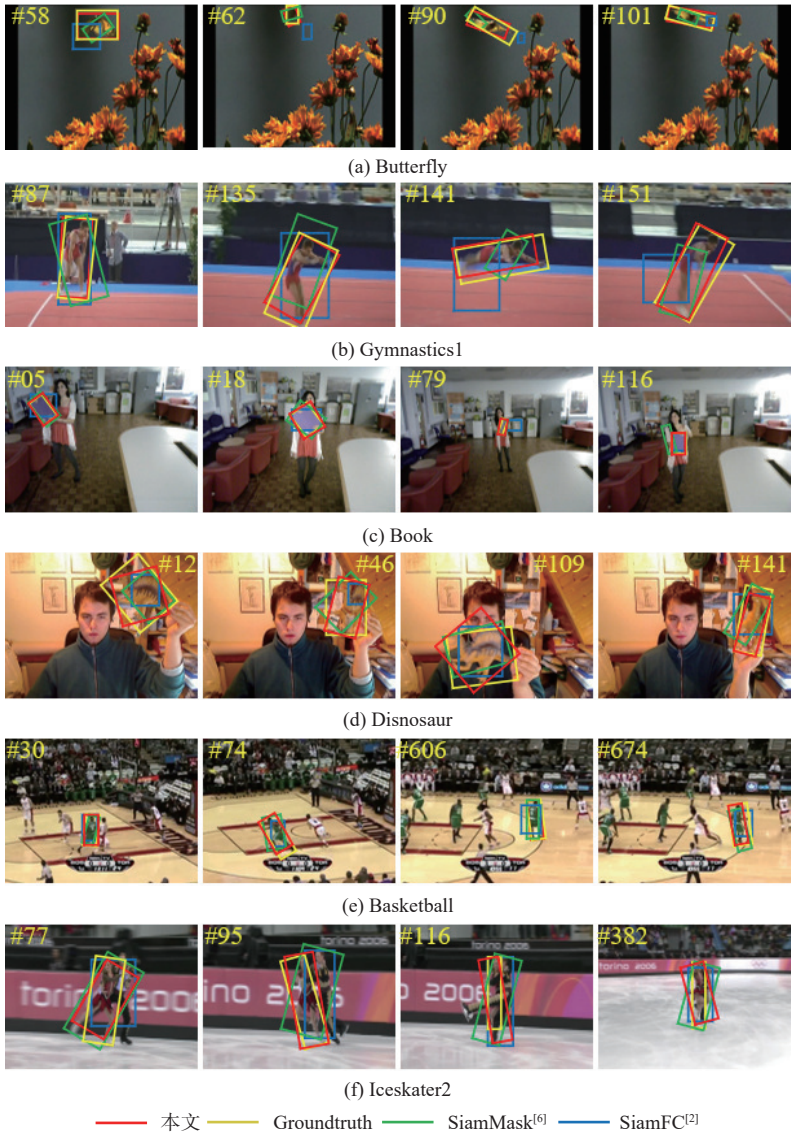

图 12 跟踪效果对比

Fig. 12 Comparison of tracking results

表 3 不同跟踪算法在 VOT2018 数据集上的结果

Table 3 Results of different tracking algorithms on VOT2018 dataset

跟踪算法	准确率↑	鲁棒性↓	EAO↑
SiamMask[6]	0.609	0.276	0.380
SiamRPN++[5]	0.600	0.230	0.415
Siam R-CNN[19]	0.609	0.220	0.408
SiamFC++[10]	0.587	0.183	0.426
ULAST-on[20]	0.571	0.286	0.355
本文	0.603	0.197	0.433

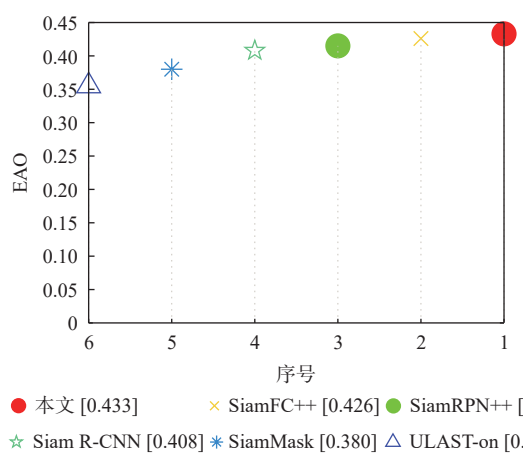

图 13 VOT2018 数据集上 EAO 排名

Fig. 13 EAO rankings on VOT2018 dataset

ΔEAO 表示相比于基础模型 SiamMask 在 EAO 上的提升值。

在分别加入某个模块后，EAO 有了一定的提升。将模板更新算法、跟踪特征增强模块和分割特征增强模块全部加到模型中后，实现了最好的效果，在 VOT2016、VOT2018 和 VOT2019 这 3 个数据集上，EAO 分别提升了 0.052、0.053 和 0.025，鲁棒性分别提升了 0.06、0.079 和 0.156，表明加入的模块是有效且兼容的，能显著提升跟踪算法的跟踪准确率和鲁棒性。同时也注意到，在加入 3 个模块

表 4　不同跟踪算法在 VOT2019 数据集上的结果

Table 4　Results of different tracking algorithms on VOT2019 dataset

跟踪算法	准确率↑	鲁棒性↓	EAO↑
SiamFC[2]	0.511	0.923	0.183
SiamRPN[3]	0.582	0.527	0.272
SiamMask[6]	0.594	0.572	0.274
SPM[21]	0.577	0.507	0.275
SiamRPN++[5]	0.599	0.482	0.285
本文	0.601	0.416	0.299

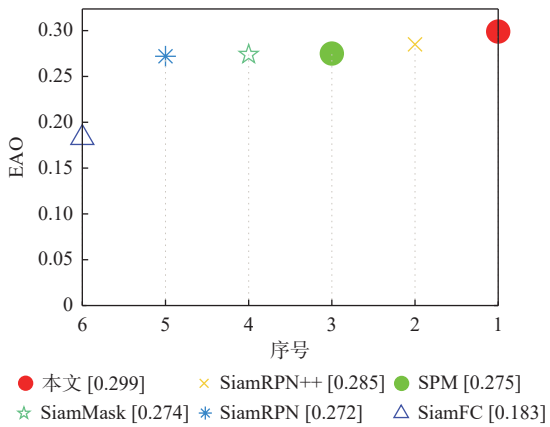

图 14　VOT2019 数据集上 EAO 排名

Fig. 14　EAO rankings on VOT2019 dataset

后, 算法的分割速度有了小幅度下降, 这是由于添加 3 个模块额外消耗了一定的计算资源, 但仍可以保持 90 帧/s 以上的实时速度, 说明加入的模块并未带来特别巨大的计算开销。

2.4.3　分割数据集上的消融实验

在 DAVIS2016 和 DAVIS2017 这 2 个分割数据集上进行消融实验, 用于验证模板更新算法、跟踪特征增强模块和分割特征增强模块对分割结果的影响, 实验结果如表 9 和表 10 所示。

表 5　不同模板更新参数在 VOT2018 数据集上的结果

Table 5　Results of different template update parameters on VOT2018 dataset

队列长度N	Δ_1	Δ_2	EAO	分割速度/(帧·s^{-1})
5	0.2	0.15	0.362	92
		0.2	0.375	96
		0.25	0.366	98
	0.25	0.15	0.381	102
		0.2	0.394	105
		0.25	0.389	105
	0.30	0.15	0.381	104
		0.2	0.379	105
		0.25	0.380	106
10	0.2	0.15	0.373	74
		0.2	0.379	77
		0.25	0.385	80
	0.25	0.15	0.384	79
		0.2	0.396	82
		0.25	0.388	84
	0.30	0.15	0.376	88
		0.2	0.378	89
		0.25	0.380	91

可以发现, 在 SiamMask 基础上单独加入某个模块或组合加入某些模块, 在 2 个分割数据集上, 不同阈值下的 mIoU 都有一定提升, 在 mIoU(0.45)指标下, 同时加入模板更新算法、跟踪特征增强模块和分割特征增强模块后, 在 DAVIS2016 数据集上提升了 0.064, 在 DAVIS2017 数据集上提升了 0.038, 达到了最好的结果, 可以验证加入的模块对于目标分割同样是有效的。

图 15 为部分数据集上分割可视化结果。可以看出, SiamMask 对目标细节部分分割不够准确, 容

表 6　VOT2016 数据集上的消融实验结果

Table 6　Results of ablation experiments on VOT2016 dataset

SiamMask	模板更新算法	跟踪特征增强模块	分割特征增强模块	准确率↑	鲁棒性↓	EAO↑	分割速度/(帧·s^{-1})↑	ΔEAO↑
√				0.622	0.214	0.433	108	
√	√			0.623	0.210	0.448	106	0.015↑
√		√		0.631	0.210	0.447	107	0.014↑
√			√	0.616	0.228	0.440	98	0.007↑
√		√	√	0.637	0.182	0.470	93	0.037↑
√	√	√	√	0.630	0.154	0.485	91	0.052↑

表 7　VOT2018 数据集上的消融实验结果

Table 7　Results of ablation experiments on VOT2018 dataset

SiamMask	模板更新算法	跟踪特征增强模块	分割特征增强模块	准确率↑	鲁棒性↓	EAO↑	分割速度/(帧·s^{-1})↑	ΔEAO↑
√				0.609	0.276	0.380	107	
√	√			0.601	0.239	0.394	105	0.014↑
√		√		0.607	0.267	0.395	107	0.015↑
√			√	0.606	0.276	0.403	98	0.023↑
√		√	√	0.612	0.234	0.420	93	0.04↑
√	√	√	√	0.603	0.197	0.433	91	0.053↑

表 8 VOT2019 数据集上的消融实验结果

Table 8 Results of ablation experiments on VOT2019 dataset

SiamMask	模板更新算法	跟踪特征增强模块	分割特征增强模块	准确率↑	鲁棒性↓	EAO↑	分割速度/(帧·s^{-1})↑	ΔEAO↑
√				0.594	0.572	0.274	109	
√	√			0.596	0.511	0.285	106	0.011↑
√		√		0.606	0.507	0.280	108	0.006↑
√			√	0.606	0.492	0.286	98	0.012↑
√		√	√	0.611	0.477	0.287	95	0.013↑
√	√	√	√	0.601	0.416	0.299	92	0.025↑

表 9 DAVIS2016 数据集上的消融实验结果

Table 9 Results of ablation experiments on DAVIS2016 dataset

SiamMask	模板更新算法	跟踪特征增强模块	分割特征增强模块	mIoU (0.30)	mIoU (0.35)	mIoU (0.40)	mIoU (0.45)	分割速度/(帧·s^{-1})↑
√				0.637	0.637	0.633	0.626	79
√	√			0.670	0.675	0.674	0.673	71
√		√		0.674	0.670	0.662	0.649	78
√			√	0.681	0.674	0.664	0.650	73
√		√	√	0.675	0.677	0.677	0.673	71
√	√	√	√	0.686	0.690	0.692	0.690	67

表 10 DAVIS2017 数据集上的消融实验结果

Table 10 Results of ablation experiments on DAVIS2017 dataset

SiamMask	模板更新算法	跟踪特征增强模块	分割特征增强模块	mIoU (0.30)	mIoU (0.35)	mIoU (0.40)	mIoU (0.45)	分割速度/(帧·s^{-1})↑
√				0.499	0.498	0.495	0.490	84
√	√			0.505	0.508	0.509	0.507	75
√		√		0.525	0.525	0.522	0.517	83
√			√	0.514	0.511	0.505	0.496	77
√		√	√	0.526	0.527	0.526	0.522	75
√	√	√	√	0.525	0.529	0.530	0.528	70

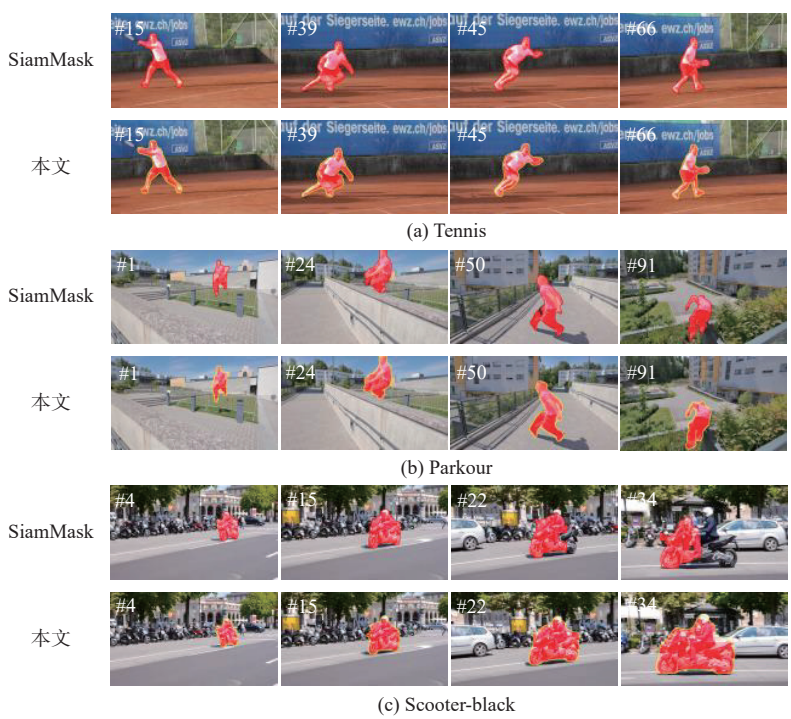

图 15 分割效果对比

Fig. 15 Comparison of segmentation effect

易受背景干扰等情况而导致分割不完整, 相比之下, 本文算法的分割结果更光滑自然, 很少有分割不完整和过度分割的情况, 整体分割结果更接近目标真实的轮廓。

3 结　论

1) 本文算法在 VOT2016、VOT2018 和 VOT2019 这 3 个数据集上取得了优异的跟踪结果, 超过了多数跟踪算法, 所设计的基于图像结构相似度判别的模板更新算法、跟踪特征增强模块和分割特征增强模块有效提升了跟踪网络在复杂场景下的准确率和鲁棒性。

2) 本文算法同时提升了目标分割的精度, 在 DAVIS2016 和 DAVIS2017 这 2 个分割数据集上, 不同阈值下的 mIoU 指标均有提升, mIoU(0.45)分别提升了 0.064 和 0.038。

然而, 本文算法牺牲了少量运行速度, 因此, 网络模型轻量化, 提升算法的实时性是下一步的研究重点。

参考文献 (References)

[1] XIAO H, LIU X. Robust target tracking based on spatio-temporal context learning[J]. Journal of Information Hiding and Multimedia Signal Processing, 2019, 10(1): 212-220.

[2] BERTINETTO L, VALMADRE J, HENRIQUES J F, et al. Fully-convolutional Siamese networks for object tracking[C]//Proceedings of the European Conference on Computer Vision. Berlin: Springer, 2016: 850-865.

[3] LI B, YAN J J, WU W, et al. High performance visual tracking with Siamese region proposal network[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2018: 8971-8980.

[4] 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. Berlin: Springer, 2018: 103-119.

[5] LI B, WU W, WANG Q, et al. SiamRPN: evolution of Siamese visual tracking with very deep networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2019: 4277-4286.

[6] HU W M, WANG Q, ZHANG L, et al. SiamMask: a framework for fast online object tracking and segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(3): 3072-3089.

[7] PARK E, BERG A C. Meta-Tracker: fast and robust online adaptation for visual object trackers[C]//Proceedings of the European Conference on Computer Vision. Berlin: Springer, 2018: 587-604.

[8] GUO Q, FENG W, ZHOU C, et al. Learning dynamic Siamese network for visual object tracking[C]//Proceedings of the IEEE International Conference on Computer Vision. Piscataway: IEEE Press, 2017: 1781-1789.

[9] ZHANG L C, GONZALEZ-GARCIA A, VAN DE WEIJER J, et al. Learning the model update for Siamese trackers[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. Piscataway: IEEE Press, 2019: 4009-4018.

[10] 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.

[11] CHEN Z D, ZHONG B N, LI G R, et al. Siamese box adaptive network for visual tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2020: 6667-6676.

[12] GUO D Y, WANG J, CUI Y, et al. SiamCAR: Siamese fully convolutional classification and regression for visual tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2020: 6268-6276.

[13] WANG Z, BOVIK A C, SHEIKH H R, et al. Image quality assessment: from error visibility to structural similarity[J]. IEEE Transactions on Image Processing, 2004, 13(4): 600-612.

[14] WANG X L, GIRSHICK R, GUPTA A, et al. Non-local neural networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2018: 7794-7803.

[15] WOO S, PARK J, LEE J Y, et al. CBAM: convolutional block attention module[C]//Proceedings of the European Conference on Computer Vision. Berlin: Springer, 2018: 3-19.

[16] ZHU X Z, HU H, LIN S, et al. Deformable ConvNets V2: more deformable, better results[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2019: 9300-9308.

[17] NAM H, HAN B. Learning multi-domain convolutional neural networks for visual tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2016: 4293-4302.

[18] DANELLJAN M, ROBINSON A, KHAN F S, et al. Beyond correlation filters: learning continuous convolution operators for visual tracking[C]//Proceedings of the European Conference on Computer Vision. Berlin: Springer, 2016: 472-488.

[19] VOIGTLAENDER P, LUITEN J, TORR P H S, et al. Siam R-CNN: visual tracking by re-detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2020: 6577-6587.

[20] SHEN Q H, QIAO L, GUO J Y, et al. Unsupervised learning of accurate Siamese tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2022: 8091-8100.

[21] WANG G T, LUO C, XIONG Z W, et al. SPM-Tracker: series-parallel matching for real-time visual object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2019: 3638-3647.

Visual tracking algorithm based on template updating and dual feature enhancement

DING Qishuai[1, 2, 3], LEI Bangjun[1, 2, 3, *], MOU Qianxi[1, 2, 3], WU Zhengping[1, 2, 3]

(1. Hubei Key Laboratory of Intelligent Visual Monitoring for Hydropower Engineering, Yichang 443002, China;
2. College of Computer and Information Technology, China Three Gorges University, Yichang 443002, China;
3. Yichang Key Laboratory of Hydropower Engineering Vision Supervision, Yichang 443002, China)

Abstract: Aiming at the problem of tracking failure due to target deformation, flipping and occlusion in visual tracking, a template updating algorithm based on image structural similarity is proposed by dynamically updating the template to adapt to the changes of the target during tracking. The tracking feature enhancement module and segmentation feature enhancement module are also designed based on the SiamMask network. The tracking feature enhancement module consists of non-local operations and convolutional downsampling, which is used to establish contextual correlation, enhance the target features, suppress the background interference, improve the tracking robustness, and solve the feature attenuation problem due to the occlusion of the target. The segmentation feature enhancement module introduces the convolutional block attention module and deformable convolution to improve the network's ability to capture channel and spatial features, adaptively learn the shape and contour information of the target, and enhance the network's segmentation accuracy of the tracked target, which in turn improves the tracking accuracy. In comparison to the baseline SiamMask, experiments demonstrate that the proposed algorithm performs well and steadily in solving the aforementioned problems, improving the expected average overlap rate by 0.052, 0.053, and 0.025 and the robustness by 0.06, 0.079, and 0.156 on the VOT2016, VOT2018, and VOT2019 datasets, respectively. It also achieves a real-time speed of 91 frames per second on average.

Keywords: object tracking; image segmentation; SiamMask; template update; feature enhancement

Received: 2024-01-11; **Accepted**: 2024-01-28; **Published Online**: 2024-02-27 13:29

URL: link.cnki.net/urlid/11.2625.V.20240226.1633.001

Foundation items: National Natural Science Foundation of China (61871258); Hubei Key Laboratory of Intelligent Visual Monitoring for Hydropower Engineering Project (2019ZYYD007); Yichang Science and Technology Research and Development Progect (A201130225)

* **Corresponding author**. E-mail: bangjun.lei@ieee.org