智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

融合视觉显著性再检测的孪生网络无人机目标跟踪算法

周士琪， 王耀南， 钟杭

引用本文:
周士琪, 王耀南, 钟杭. 融合视觉显著性再检测的孪生网络无人机目标跟踪算法[J]. 智能系统学报, 2021, 16(3): 584–594.
ZHOU Shiqi, WANG Yaonan, ZHONG Hang. Siamese network combined with visual saliency re-detection for UAV object tracking[J]. *CAAI Transactions on Intelligent Systems*, 2021, 16(3): 584–594.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202101035

DOI: 10.11992/tis.202101035

融合视觉显著性再检测的孪生网络无人机目标跟踪算法

周士琪[1,2], 王耀南[1,2], 钟杭[1,2]

(1. 湖南大学 电气与信息工程学院, 湖南 长沙 410082; 2. 湖南大学 机器人视觉感知与控制技术国家工程实验室, 湖南 长沙 410082)

摘 要: 针对旋翼飞行器在跟踪过程中目标尺度变化、快速运动、视角变化等问题, 本文提出了一种基于MobileNetV2的孪生网络目标跟踪算法, 可在无人机机载处理器上实时运行。该算法主要包含目标得分估计模块与目标尺度估计模块两个部分。结合多特征融合的策略, 可准确预测出目标位置与目标框IoU, 同时以目标框IoU为指导, 利用梯度上升法对目标框进行迭代修正, 进一步提升预测精度。针对完全遮挡而导致的目标跟丢问题, 本文设计了一个基于视觉显著性的目标再检测算法, 该算法可实时高效地预测出图像的显著性区域, 以指导对目标的再检测, 进而恢复跟踪。最后, 通过标准无人机跟踪数据集测试与实际无人机跟踪实验, 验证了算法的可行性。

关键词: 无人机; 计算机视觉; 目标跟踪; 轻量化网络; 孪生网络; 显著性检测; 目标遮挡; 特征融合

中图分类号: TP242 **文献标志码:** A **文章编号:** 1673−4785(2021)03−0584−11

中文引用格式: 周士琪, 王耀南, 钟杭. 融合视觉显著性再检测的孪生网络无人机目标跟踪算法 [J]. 智能系统学报, 2021, 16(3): 584–594.

英文引用格式: ZHOU Shiqi, WANG Yaonan, ZHONG Hang. Siamese network combined with visual saliency re-detection for UAV object tracking[J]. CAAI transactions on intelligent systems, 2021, 16(3): 584–594.

Siamese network combined with visual saliency re-detection for UAV object tracking

ZHOU Shiqi[1,2], WANG Yaonan[1,2], ZHONG Hang[1,2]

(1. College of Electrical and Information Engineering, Hunan University, Changsha 410082, China; 2. National Engineering Laboratory for Robot Vision Perception and Control Technology, Hunan University, Changsha 410082, China)

Abstract: Considering the problems associated with rotorcraft trackings, such as target scale variation, fast motion, and viewpoint change, this paper proposes a siamese network-based target tracking algorithm using MobileNetV2, which can run in real-time on an onboard UAV processor. The algorithm consists of target score and scale estimation modules. Combined with the multifeature fusion strategy, the target position and target box IoU were accurately predicted. At the same time, by employing the IoU, the gradient ascent method was used to iteratively modify the target bounding box to further improve the prediction accuracy. In addition, to solve the problem of target loss caused by full occlusion, a re-detection algorithm based on visual saliency detection was developed, which efficiently predicted the saliency map of the image in real-time to guide the re-detection of the target and resume tracking. Finally, the feasibility of the algorithm was verified by comparing the standard UAV tracking dataset and the actual UAV tracking experiment.

Keywords: drone; computer vision; object tracking; MobileNetV2; siamese network; saliency detection; target occlusion; feature fusion

无人机因其隐蔽性高、体积小、机动性强等优点, 已被广泛应用于智慧交通、抢险救灾、军事侦察、航拍等领域[1]。随着计算机视觉与人工智能技术的不断发展, 面向无人机的目标跟踪算法, 因能

收稿日期: 2021−01−28.

基金项目: 国家自然科学基金项目(61733004); 中国博士后科学基金项目(2020M682555).

通信作者: 钟杭. E-mail: zhonghang@hnu.edu.cn.

使无人机执行更多自主任务, 成为了当下研究的热点[2]。无人机在执行跟踪任务过程中, 由于平台的运动与图像的不稳定性, 往往会使拍摄的图像序列视角变化更大、目标尺度变化快、背景模糊、遮挡频繁等。因此, 研究一种高效而鲁棒的目标跟踪算法, 对无人机的应用具有重要的价值与意义。

近年来, 以 CSK(circulant structure of tracking-by-detection with kernels)[3]、KCF(kernelized correlation filters)[4] 为代表的传统相关滤波算法, 因其明显的速度优势在无人机目标跟踪领域中得到广泛关注。但是, 基于传统方法的目标跟踪算法利用手工设计的特征提取器, 往往对目标的变化不鲁棒, 在一些复杂条件下的无人机跟踪过程中, 效果并不理想。随着深度学习技术的发展, 深度网络因其良好的特征表达能力, 相较传统的特征提取方法已取得了明显的优势。如 C-COT(continuous convolution operators for visual tracking)[5]、ECO(efficient convolution operators for tracking)[6], 利用离线训练获取特征提取网络, 使跟踪精度得到了显著提升, 但是算法的运行速度难以满足无人机目标跟踪的实时性要求。2017 年, Tao 等[7] 提出 SINT(Siamese instance search tracking)算法, 首次将孪生网络应用于目标跟踪领域。之后, Bertinetto 等[8] 提出了 SiamFC(Siamese fully convolutional), 一种基于孪生网络全卷积目标跟踪算法, 采用端到端的方式直接训练整个网络,该算法在速度与精度上都取得了较好的效果, 这也使得孪生网络在跟踪领域受到了越来越多的重视。CFNet(correlation filter network)[9] 在孪生网络框架中引入了相关滤波层。SiamRPN(Siamese region proposal network)[10] 中, 开创性地引入了目标检测中的 RPN(region proposal network)[11] 网络到孪生网络跟踪中, 将相似度计算问题转化为目标分类与回归问题, 并在大规模的数据集上进行离线训练, 使算法在速度和精度上都取得了明显优势。但时, RPN 网络同时也存在着目标置信度不能准确表征目标框准确度、目标框无法迭代修正以及超参数较多等问题[12]。

本文针对图像序列的无人机目标跟踪问题, 提出了一种基于孪生网络的目标跟踪算法, 能在嵌入式无人机机载电脑上实时运行。该算法主要包括 3 部分: 为提升特征提取网络的特征表达能力同时保障算法的实时性能, 引入 MobileNetV2[13] 作为主干网络并采用一种基于偏移量学习的方式, 解决网络平移不变性的问题; 引入了一种新的目标尺度估计模块, 可预测出目标框与真实框的重叠率 IoU, 并以此为指导, 对目标框进行迭代修正; 最后, 利用网络多层特征, 采用残差融合的策略进一步提升网络性能。针对因完全遮挡导致目标跟丢的问题, 本文设计了一个基于 MoblieNetV2 的实时高效的显著性检测算法, 可轻易地嵌入目标跟踪框架中。在跟丢时, 切换到显著性检测算法对目标进行再检测以恢复跟踪。

1　目标跟踪算法设计

1.1　基于孪生网络的跟踪算法原理

传统的基于孪生网络的目标跟踪算法, 根据目标的相似度来对目标进行跟踪[8], 通过学习一个相似性函数 $\boldsymbol{f}(\boldsymbol{z},\boldsymbol{x})$, 将模板图像 $\boldsymbol{z}$ 与搜索图像 $\boldsymbol{x}$ 分别作为模板分支与搜索分支的输入, 在第一帧时, 通过卷积网络 $\boldsymbol{\varphi}$ 对模板图像 z 进行处理, 得到感兴趣目标的特征信息, 在后续帧中, 通过卷积网络 $\boldsymbol{\varphi}$ 对搜索图像 $\boldsymbol{x}$ 进行处理, 得到搜索区域特征, 并将其与第一帧得到的模板特征进行卷积匹配, 采用滑窗检测的方式, 得到输出的得分置信度特征相应图, 得分最大的点表示该位置与目标具有最大的相似度, 因此相似性函数可表示为

$$\boldsymbol{f}_{\theta}(z,\boldsymbol{x})=\boldsymbol{\varphi}_{\theta}(z)*\boldsymbol{\varphi}_{\theta}(\boldsymbol{x})+\boldsymbol{b}\cdot\boldsymbol{I}$$

式中: $\boldsymbol{\theta}$ 表示网络参数; $*$ 表示卷积操作; $\boldsymbol{b}\cdot\boldsymbol{I}$ 表示偏置项。

孪生网络对目标相似性的学习是基于全卷积网络的平移不变性基础上的, 令 L_{τ} 表示平移操作, 则 $(L_{\tau}\boldsymbol{x})[u]=\boldsymbol{x}[u-\tau]$; 这使得:

$$\boldsymbol{f}(z,L_{\tau}\boldsymbol{x})[u]=\boldsymbol{f}(z,\boldsymbol{x})[u-\tau] \tag{1}$$

式 (1) 表示将目标在搜索区域进行平移后, 其映射到得分置信度特征图相应位置的值与平移之前保持不变, 这让网络的训练与实际的跟踪具有意义。同时网络具有严格的结构对称 $\boldsymbol{f}(z,\boldsymbol{x})=\boldsymbol{f}(\boldsymbol{x},z)$, 这也有益于相似性的学习。

利用全卷积网络的这些特性, 模板图像与搜索图像可以使用不同分辨率的图片, 通过输入尺寸更大的搜索图像, 可以使模板图像在其不同的子窗口进行滑窗检测, 得到目标的得分置信度特征响应图。响应图中的每一个点与搜索图像中的每一个子窗口一一对应, 表示该子窗口区域与目标的相似度, 以此实现对目标的跟踪。

1.2　基于 MobileNetV2 的目标跟踪算法原理

目标跟踪模算法包含了 2 个模块, 分别是目标得分估计模块、目标尺度估计模块, 其网络结构图如图 1 所示。利用目标得分模块可得到目标在图像中的初始位置, 再利用目标尺度估计模块对目标位置及其目标框大小进行迭代修正, 得到目标最终的位置及尺度估计。

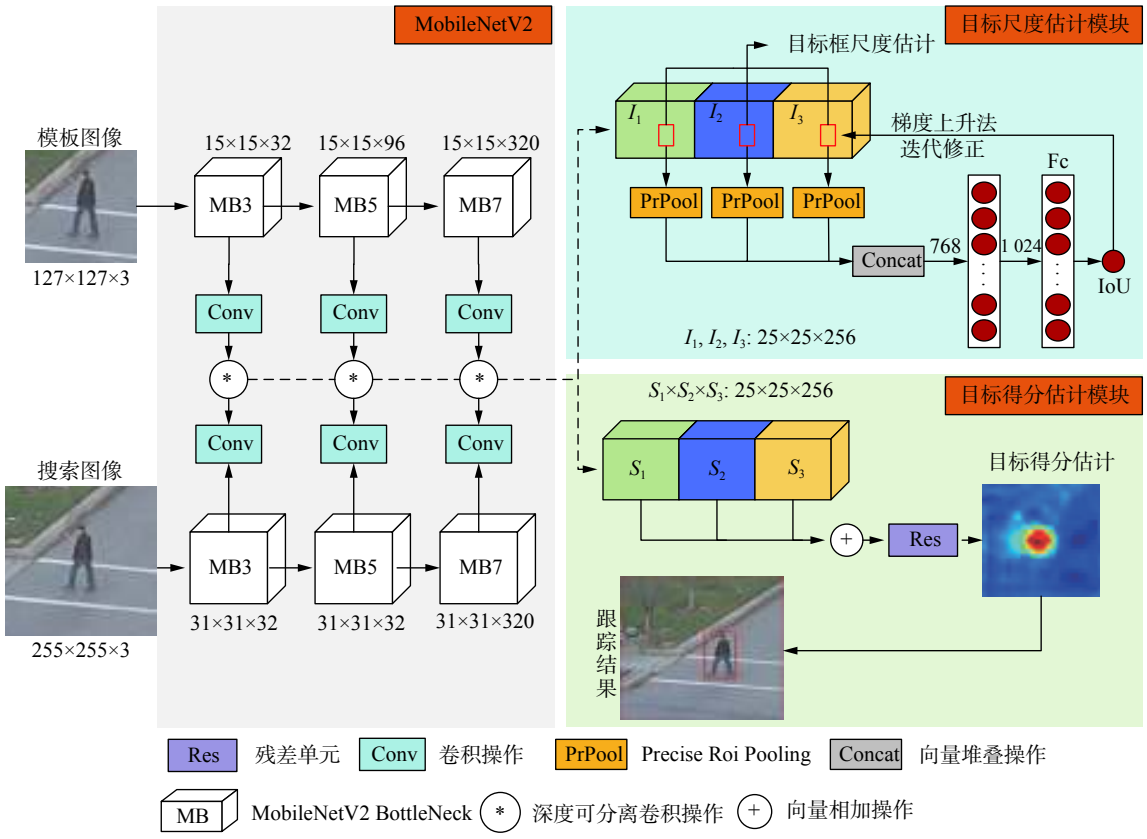


图 1　目标跟踪算法网络结构

Fig. 1　Object tracking network structure

1.2.1　目标得分估计模块设计

为了提升特征提取网络性能同时保障算法的实时性，本文引入 MobileNetV2 作为特征提取网络。对于图 1 中的目标得分估计模块，由于该网络中加入了 padding，使孪生网络的平移不变性遭到破坏[14]，导致网络学习遭到严重的中心偏置，造成跟踪失败。为解决这一问题，本文通过一种目标偏移量学习的方式，使网络能预测出目标在两帧图像之间的偏移量而不是相似度，这使得深度网络通过学习的方式获取到全卷积网络平移不变性的特性，成功引入 MobileNetV2 作为特征提取网络。同时，为了适应于目标跟踪任务，将 MobileNetV2 的第 4 个 bottleneck 块和第 6 个 bottleneck 块的下采样率设为 1，使网络的总下采样率由 32 调整为 8，并通过加入空调卷积层来增加网络的感受野[15]，最后在网络的输出部分增加一个 1×1 的卷积，使输出特征通道维度为 6。

在目标得分估计模块，采用偏移量学习的方式进行训练，通过引入一个偏移量 $t=(t_x,t_y)$，表示目标相对于得分置信度特征响应图中心点 c 的偏移量，其值等于 $(T_x/s, T_y/s)$，其中，T_x、T_y 表示训练时，通过数据增强的方式对图像目标区域随机产生的平移量，s 表示网络的总下采样率。将与目标偏移后的中心点的距离在一定范围内的点视为正样本，其余的点都为负样本，则目标得分估计模块的标签 $\boldsymbol{y}_s$ 表示为

$$\boldsymbol{y}_s=\begin{cases}+1, & s\|u-c-t\|\leqslant 16\\ -1, & \text{其他}\end{cases} \tag{2}$$

式中 u 表示标签 $\boldsymbol{y}_s$ 相应点的坐标。采用 logistics loss 作为目标得分置信度模块的损失函数，进行训练。通过这种训练方式，使网络能够主动地去学习目标产生的偏移量，解决了网络平移不变性被破坏的问题，使该模块能成功地估计出目标当前位置。

1.2.2　目标尺度估计模块设计

本文在孪生网络的基础上，新增了一个目标尺度估计模块，将 MobileNetv2 提取到的模板特征与搜索特征进行卷积，得到一个包含目标信息的特征图，再通过一个 1×1 的卷积层对其进行处理，得到适用于目标尺度估计模块的特征输入 $\boldsymbol{I}$。特征输入 $\boldsymbol{I}$ 包含了目标信息，因此可将目标在搜索图像上的边界框映射到特征输入 $\boldsymbol{I}$ 上，在特征图 $\boldsymbol{I}$ 上相应的区域通过 PrPool（Precise ROI Pooling）[12] 提取特征，PrPool 将目标区域划分为 $m×m$ 个 bin，对每个 bin 应用双线性插值，将离散的像素点区域插值为一个连续的区域后，对每个

bin 求积分, 该过程可表示为

$$f(x,y)=\sum_{i,j}\max(0,1-|x-i|)\times \max(0,1-|y-i|)\times w_{i,j}$$

$$P(\text{bin},\boldsymbol{I})=\int_{y_1}^{y_2}\int_{x_1}^{x_2}f(x,y)\mathrm{d}x\mathrm{d}y/S$$

式中: S 表示 bin 的面积; P 表示 PrPool 特征提取; $P(\text{bin},\boldsymbol{I})$ 表示利用 PrPoo 在特征图 $\boldsymbol{I}$ 上的 bin 区域提取特征, 得到该区域的一个标量特征值; (x_1,y_1)、(x_2,y_2) 分别表示该 bin 区域的左上角和右下角坐标。最后, 将所有 bin 区域提取到的特征通过 2 个全连接层可得到目标的 IoU 预测。

在实际跟踪时, 目标尺度估计模块根据目标得分模块得到目标 (h',w') 的最大响应值位置, 其相应的坐标表示目标在当前帧对于上一帧的偏移量 $(\Delta x,\Delta y)$。由此, 可得到目标在当前帧边界框的初始估计 (x,y,h',w'), 其中, (x,y) 表示目标框中心坐标, h'、w'分别表示目标框的高、宽, 其值与上一帧相同。对该目标框加入均匀随机噪声产生额外 9 个初始框, 将这 10 个框输入到目标尺度估计模块, 利用 PrPool 提取特征, 最后预测得到这 10 个边界框相对目标真实边界框的 IoU, 令 $(x_{1,j},y_{1,j}),(x_{2,j},y_{2,j})$ 表示边界框的左上角和右下角坐标, $j=1,2,\cdots,10$, 由于 $P(\text{bin},\boldsymbol{I})$ 相对于边界框坐标是连续可导的, $P(\text{bin},\boldsymbol{I})$ 相对于边界框左上角横坐标的导数可表示为

$$\frac{\partial P(\text{bin},\boldsymbol{I})}{\partial x_{1,j}}=\frac{P(\text{bin},\boldsymbol{I})}{x_{2,j}-x_{1,j}}-\frac{\int_{y_{1,j}}^{y_{2,j}}f(x_{1,j},y_j)\mathrm{d}y}{(x_{2,j}-x_{1,j})(y_{2,j}-y_{1,j})}$$

其他坐标导数同理可得, 因此通过梯度上升法可对目标边界框进行迭代修正, 使目标 IoU 最大:

$$\text{grad}=\nabla_{x_{1,j},y_{1,j},x_{2,j},y_{2,j}}F(P((x_{1,j},y_{1,j},x_{2,j},y_{2,j}),\boldsymbol{I}))$$

$$(x_{1,j},y_{1,j},x_{2,j},y_{2,j})=(x_{1,j},y_{1,j},x_{2,j},y_{2,j})+\lambda\cdot\text{grad}$$

式中: F 表示全连接层; P 表示 PrPool; λ 表示学习率。一共迭代 5 轮, 最后选取 IoU 最大的 3 个边界框, 对它们取平均值作为最后目标边界框的预测, 实现对目标的跟踪。

1.2.3 多层特征融合与网络训练

为了充分利用 MobileNetV2 网络的多层特征, 利用 MobileNetV2 的 bottleneck3、bottleneck5、bottleneck7 分别对模板图像与搜索图像提取特征进行卷积, 得到在 3 个不同层下的特征输出 $\boldsymbol{I}=(\boldsymbol{I}_1,\boldsymbol{I}_2,\boldsymbol{I}_3)$ 与 $\boldsymbol{S}=(\boldsymbol{S}_1,\boldsymbol{S}_2,\boldsymbol{S}_3)$。由于这 3 层输出具有相同的下采样率, 因此输出特征图分辨率相同, 但由于每层具有不同的空洞卷积率, 每层具有不同的感受野。感受野越大, 网络能捕获到更多的语义信息, 因此, 能更好地应对目标的外观变化、背景变化等。而感受野较小, 则能捕获到更多的细节信息, 如目标的轮廓及位置信息等。通过将高感受野的特征输出与低感受野的特征输出融合, 使网络的输出既具有高层的语义信息, 又具有低层的位置轮廓信息, 从而使网络对目标的检测更加鲁棒。

对于目标得分估计模块, 本文采用一种残差融合的策略, 将 $\boldsymbol{S}_1$、$\boldsymbol{S}_2$、$\boldsymbol{S}_3$ 特征直接相加后通过一个残差单元[16] 进行融合, 随后通过一个 1×1 卷积对融合后的特征进行降维, 最后得到目标得分估计相应图的预测。令 c_{res} 表示残差融合模块, 则目标得分估计响应图可表示为 $\text{score}=c_{\text{res}}(\boldsymbol{S}_1,\boldsymbol{S}_2,\boldsymbol{S}_3)$。该模块的损失函数可表为

$$L_s(\boldsymbol{y}_s,\text{score})=\frac{1}{|\text{score}|}\sum_{u\in\boldsymbol{M}}l(\boldsymbol{y}_s[u],\text{score}[u])$$

$$l(\boldsymbol{y}_s,\text{score})=\log(1+\exp(-\boldsymbol{y}_s\text{score}))$$

式中: u 表示 score 中相应的坐标; $|\text{score}|$ 表示响应图像素点总数量; $\boldsymbol{y}_s$ 为式 (2) 中相应的标签。

对于目标尺度估计模块, 在网络训练时, 将目标边界框的 groundtruth 加入一个高斯噪声, 生成 16 个不同的边界框, 并保证每个边界框与 groundtruth 的 IoU 大于 0.1, 将这 16 个边界框映射到 MobileNetV2 在目标尺度估计模块所提取到的特征 $\boldsymbol{I}=(\boldsymbol{I}_1,\boldsymbol{I}_2,\boldsymbol{I}_3)$ 上, 采用 PrPool 对这些区域提取特征, 最后通过全连接层 F, 计算得到 16 个目标边界框与 groundtruth 的 IoU 预测, 该过程可表示为

$$\text{IoU}=F(c(P(\text{bin},\boldsymbol{I}_1,\boldsymbol{I}_2,\boldsymbol{I}_3)))$$

式中: c 表示将 PrPool 提取到的 3 个 $m\times m\times 256$ 特征输出拼接成 1 个 $m\times m\times 768$ 的特征。最后将这 16 个边界框与 groundtruth 之间的 IoU 归一化到 [−1,1], 作为 IoU 的真实标签 $\boldsymbol{y}_{\text{iou}}$, 采用 MSEloss 作为目标尺度估计模块的损失函数, 则该模块的损失可表示为

$$L_{\text{iou}}=\frac{1}{2n}\sum_{i=1}^{n}\left\|y_{\text{iou},i}-\text{IoU}_i\right\|^2$$

式中: n 表示训练样本总数。采用多任务学习的方式对网络进行训练, 因此网络的总损失可表为

$$L=L_{\text{iou}}+\lambda\cdot L_s$$

式中$\lambda=2$。

在 ILSVRC2015、Lasot[17]、Coco、GOT-10k[18] 数据集中每次以相同的概率随机选取一个数据集, 并在数据集中随机选取一个视频帧中相隔一定距离的图像对 (Coco 数据集选取一对相同的图片作为数据扩充), 采用平移、缩放、翻转、模糊等数据增强手段对图像对进行处理, 最后将图像对输入到网络进行训练。利用随机梯度下降法来训练网络参数, 共训练 20 代, 前 5 代学习率从 10^{-3} 线性地增加到 5×10^{-3}, 后 15 代学习率衰减到 5×10^{-4}, 特征提取网络 MobileNetV2 使用预训练模型, 前 10 代不进行训练。

2　视觉显著性再检测算法设计

当目标发生完全遮挡时，由于无法检测到目标，导致目标丢失。当目标再次出现在视野中时，算法已经无法跟踪到目标。如果此时能知道目标可能存在的区域，并对这些区域进行检测，则能再次恢复对目标的跟踪。因此，可利用人眼视觉注意力机制，对图像的显著性区域进行检测，以恢复跟踪。

2.1　显著性检测算法设计

本文所设计的显著性检测算法包含 3 部分，分别是特征提取网络、编码网络、解码网络。首先将图像输入到特征提取网络得到图像各层次的通用特征，再利用编码网络得到图像的显著性特征，最后通过解码网络得到显著性图的预测，网络的整体结构如图 2 所示。

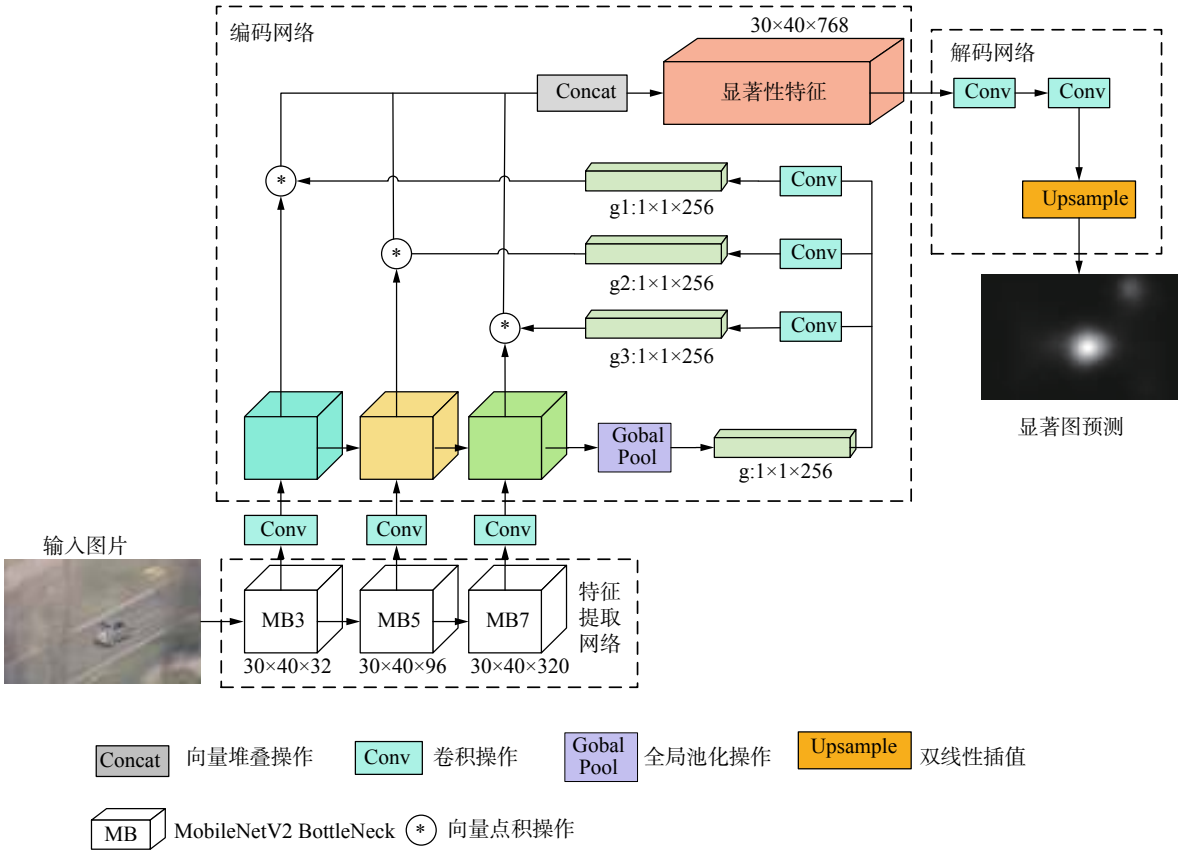


图 2　显著性检测算法网络结构

Fig. 2　Saliency detection network structure

2.1.1　特征提取网络

基于深度学习的显著性检测算法[19-22]多采用 VGG16[23]作为特征提取网络，并已取得不错的效果，但是却难以在无人机机载处理器上实时运行。因此本文同样采用 MobileNetV2 作为显著性检测算法的特征提取网络，同时为了将其预训练模型迁移到显著性检测任务中，需要对其结构进行一些调整。MobileNetV2 原本是适用于图像分类的网络，因此下采样率很高，且最后通过全连接层进行分类。为了应用于显著性检测，首先去掉了 MobileNetV2 网络最后的全连接层以及多余的卷积池化层。由于显著图是一种像素级的预测，太大的网络下采样率会损坏预测的精度。因此，将 MobileNetV2 的 bottleneck4 和 bottleneck6 的步长设为 1，这样使网络的总下采样率为 8，保障了输出显著性图的分辨率。同时，在减少下采样率后，为了保持调整后的网络和预训练模型中的网络具有相似的感受野，在最后 4 个 bottleneck 结构中加入空洞卷积。

2.1.2　编、解码网络

在语义分割任务中，同时利用网络不同层之间的特征，有利于分割的空间细节恢复[24]。而显著性检测同样是一种像素级的预测，虽然不需要准确地找出目标及其边界的像素，多层特征同样有利于显著性图的预测[22]。因此，对于编码网络，本文同时利用 MobileNetV2 的 bottleneck3、bottleneck5、bottleneck7 模块的输出，将这 3 层的输出表示为 f_1、f_2、f_3。同时，对 f_3 进行全局池化，得到

全局引导特征 $\boldsymbol{g}$, 将其通过 3 个不同的 1×1 卷积, 得到 3 个分别对应于 $\boldsymbol{f}_1$、$\boldsymbol{f}_2$、$\boldsymbol{f}_3$ 的全局引导特征 $\boldsymbol{g}_1$、$\boldsymbol{g}_2$、$\boldsymbol{g}_3$, 并通过一个 sigmoid 函数将它们的特征值映射到 [0,1]。利用通道注意力机制, 将 $\boldsymbol{g}_1$、$\boldsymbol{g}_2$、$\boldsymbol{g}_3$ 与 $\boldsymbol{f}_1$、$\boldsymbol{f}_2$、$\boldsymbol{f}_3$ 逐位相乘, 对其进行特征选择, 得到最后的特征输出。最后, 直接将它们拼接在一起得到显著性特征输出。通过这种方式, 网络提前获取了更高层的语义先验知识, 使网络的注意力能集中在更重要的部分。对于解码网络, 出于对算法实时性的考虑, 采用两层卷积网络对显著性特征进行解码。首先通过一个 3×3 卷积, 将显著性特征输出降维到 256, 再利用 1 个 1×1 卷积, 得到显著性图的预测, 最后通过双线性插值将其恢复到输入图像的尺度。

2.1.3　网络训练

在 SALICON 数据集[25] 上对网络进行训练,该数据集包含了 10 000 张训练图片, 5 000 张验证图片, 5 000 张测试图片, 笔者利用 CoCo 数据集中的图片, 通过鼠标运动代替眼动追踪系统, 对数据集进行标注, 经验证这 2 种方式标注产生的显著性图具有高度的相似性。

通过将网络输出的显著性图归一化到 [0,1], 并使显著性图的像素和为 1, 可以将显著性图的预测视为概率分布的预测[26]。为了使网络的训练更加鲁棒, 本文同时采用 KL-Div(kullback-leibler-divergence) 和 CC(linear correlation coefficient) 作为损失函数进行训练, 其中 KL-Div 用于衡量预测显著图的分布与标签显著图分布之间的差异, CC 用于衡量预测显著图的分布与标签显著图分布之间相关性, 则网络的总损失可表示为

$$L(\boldsymbol{y}^t,\boldsymbol{y}^p)=\sum_i y_i^t\log\left(\frac{\boldsymbol{y}_i^t}{\boldsymbol{y}_i^p+\varepsilon}\right)-\frac{\sigma(\boldsymbol{y}^p,\boldsymbol{y}^t)}{\sigma(\boldsymbol{y}^p)\cdot\sigma(\boldsymbol{y}^t)}$$

式中: $\boldsymbol{y}^t$ 表示标签; $\boldsymbol{y}^p$ 表示预测显著性图; ε 表示正则系数; i 表示显著性图$\boldsymbol{y}^t$、$\boldsymbol{y}^p$的像素坐标索引; σ 表示协方差。

利用 Adam[27] 优化算法, 对网络参数进行优化, 共训练 10 代, 每次同时训练 10 张图片, 学习率为 10^{-6} 。

2.2　目标再检测算法

为实现对目标的再检测, 将所设计的显著性检测算法嵌入到目标跟踪算法框架中。其具体的框架流程如图 3 所示。在跟踪过程中如果目标得分连续 T 帧小于设定的阈值 τ_l, 则认为目标已跟丢。这时应停止对目标的跟踪, 通过显著性检测算法提取当前帧的显著性图, 并滤除掉响应较小干扰区域, 得到当前响应最大的几个显著性区域。再单独利用目标跟踪算法中的得分估计模块得到当前显著性区域的目标最大得分响应值, 如果有连续 T 帧目标的得分大于设定的阈值 τ_r 则认为已经找到目标, 进而恢复跟踪。

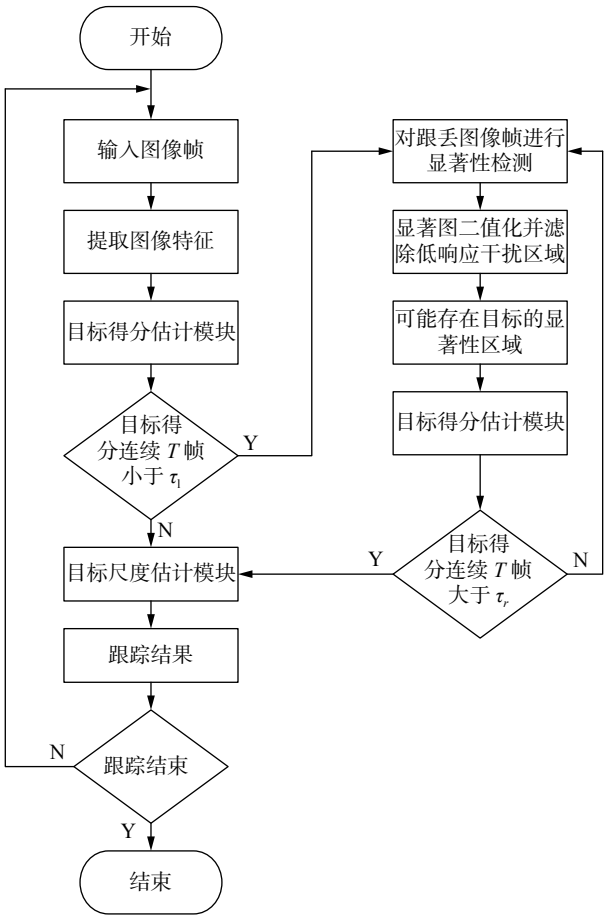


图 3　算法整体框架流程

Fig. 3　Overall framework structure of the algorithm

3　实验结果与分析

为了对无人机目标跟踪算法进行实验测试, 首先在 UAV123 数据集上对目标跟踪算法性能进行验证, 实验采用的平台为: Intel(R) CoreTM i9-9900k, Nvidia RTX2080ti。最后, 在六旋翼飞行器平台上对跟踪算法进行了测试验证, 该飞行器搭载 Jetson Xavier NX 作为机载处理器, 采用 Realsense D435 相机实时获取图像信息。

3.1　目标跟踪算法实验测试

为对本文目标跟踪算法进行测试, 选取 UAV123[28] 数据集作为测试序列, 该数据集包含了 123 个无人机拍摄的视频序列, 涵盖了目标遮挡、尺度变化、视角变化、背景复杂、相似目标、光照变化、相机运动等 12 种复杂情况。无人机在飞行过程中视角变化、运动幅度大, 因此该数据集具有较高挑战难度。将本文算法与 ECO[6]、SiamFC[8]、SiamRPN[10]、SAMF[29]、SRDCF[30]、STRUCK[31] 这 6 种跟踪算法进行实验效果对比,

采用覆盖率与中心位置误差 2 种评价指标进行定量分析。覆盖率表示预测目标框与真实目标框的重叠率，中心位置误差表示预测目标框中心与真实目标框中心的欧氏距离。最后利用成功率图与准确率图对测试结果进行表示，其中，成功率图表示在不同覆盖率阈值下，满足覆盖率阈值要求的视频图像帧与总视频图像帧的比率。准确率表示在不同中心位置误差阈值下，满足中心位置误差阈值要求的视频图像帧与总视频图像帧的比率。

将本文算法与这 6 种算法在 UAV123 数据集上的 12 种不同情况进行对比。本文算法的综合平均成功率达到了 0.602，平均准确率达到了 0.810，几乎在各种情况下的性能都优于其他算法，并且本文算法在完全遮挡、超出视野、背景复杂、快速运动、视觉变化、尺度变化等情况下的性能相较其他算法都有明显的优势。UAV123 数据集 12 种情况下的部分成功率与准确率结果如图 4 所示，其中 Ours_s 表示结合再检测模块的跟踪算法。

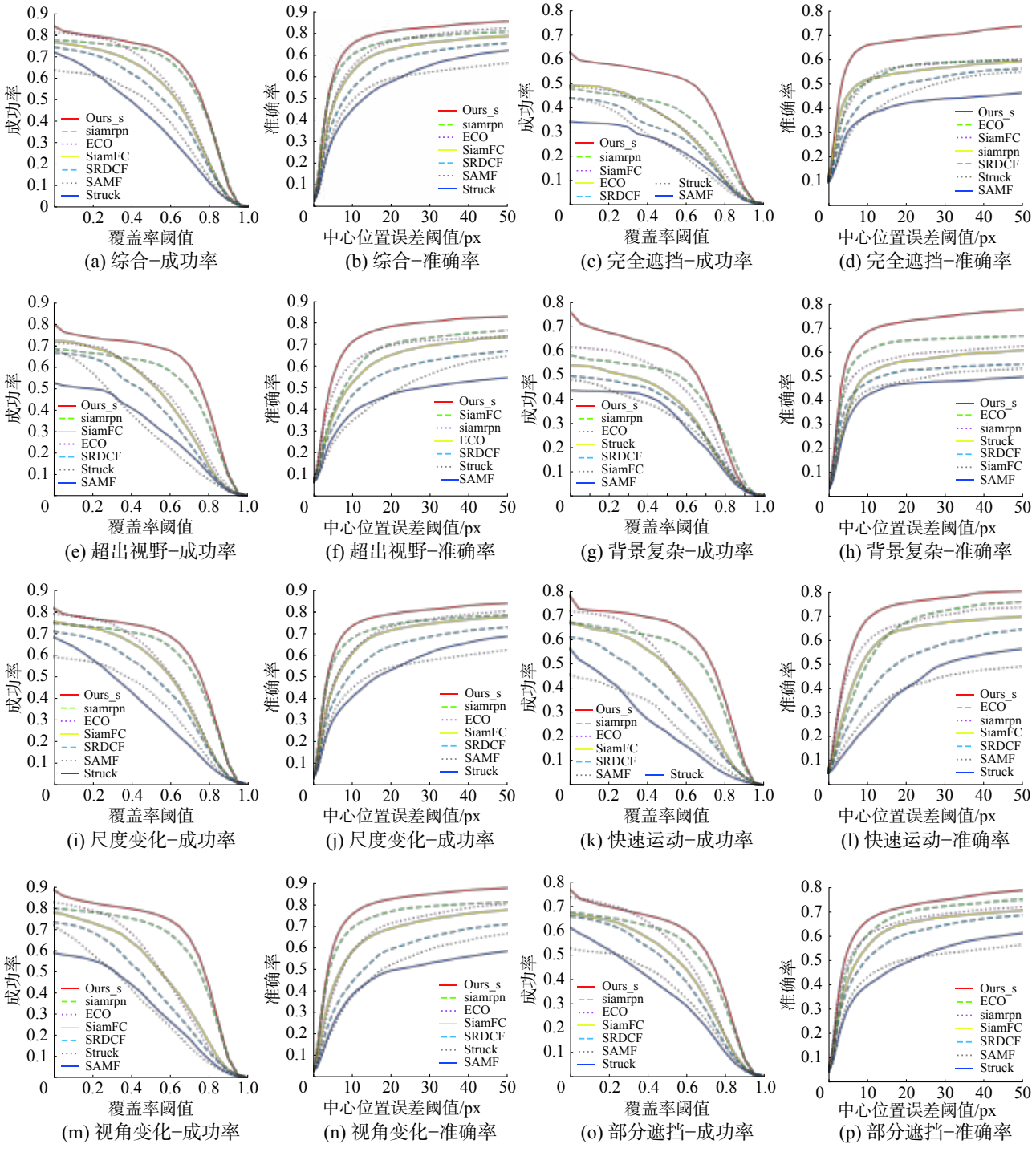


图 4　UAV123 数据集测试结果

Fig. 4　Tracking results of UAV123 dataset

为了进一步验证再检测模块对跟踪算法性能的提升, 对未加入再检测模块的跟踪算法进行测试, 并与加入再检测模块后的算法对比, 其测试的结果如表 1、2 所示。其中, Ours 表示仅利用本文图 1 中所提的跟踪算法, Ours_s 表示进一步结合再检测模块的跟踪算法。

表 1　再检测模块成功率性能对比

Table 1　Performance comparison of the success rate of re-detection module

方法	ALL	完全遮挡	超出视野	背景复杂	快速运动	视角变化	相机运动	尺度变化
Ours	0.590	0.389	0.574	0.406	0.543	0.620	0.611	0.574
Ours_s	0.602	0.435	0.581	0.477	0.548	0.632	0.632	0.588

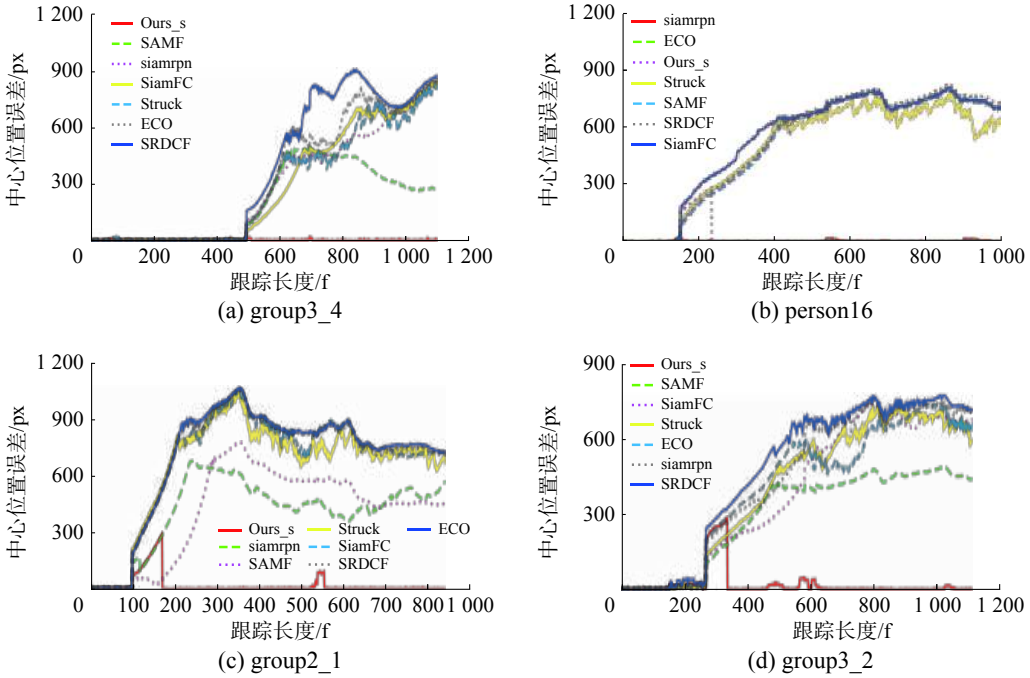
表 2　再检测模块准确率性能对比

Table 2　Performance comparison of the accuracy of re-detection module

方法	ALL	完全遮挡	超出视野	背景复杂	快速运动	视角变化	相机运动	尺度变化
Ours	0.794	0.621	0.768	0.650	0.745	0.807	0.818	0.769
Ours_s	0.810	0.680	0.777	0.731	0.755	0.824	0.846	0.786

在加入再检测模块后, 跟踪算法在完全遮挡情况下的性能具有明显提升, 对超出视野的情况, 由于数据集大部分目标超出视野再回到视野中时, 目标位置变化不大, 所以不需要再检测模块也能恢复对目标的跟踪, 这也说明了本文算法在没有加入再检测模块时, 相较于其他算法, 在超出视野的情况下也具有明显的优势, 这得益于基于偏移量学习的目标得分估计模块, 在目标回到视野时能准确识别出目标, 以及尺度估计模块, 对目标当前位置和尺度进行修正。对于背景复杂情况下性能的提升, 是由于再检测模块解决了部分因背景复杂而导致目标跟丢失的视频序列, 如 group2_1、group2_3、group3_2、group3_4 等。这使得当遮挡等问题解决时, 在背景复杂情况下的性能也得到了提升。而快速运动、尺度变化、视角变化等情况, 再检测模块只发挥了较小的作用。

进一步, 选取 group2_1、group3_2、group3_4 等遮挡情况下的视频序列, 将其每帧的中心位置误差进行分析, 结果如图 5 所示。本文算法在所有数据集测试时, 为了便于分析, 在目标跟丢失, 仍然用跟踪算法进行跟踪, 以获取跟丢失时的目标框位置, 进行误差分析, 同时利用再检测模块获取恢复跟踪后的目标位置。而在实际应用中, 当目标跟丢失时应停止运行跟踪算法, 只运行再检测模块, 直到重定位到目标。



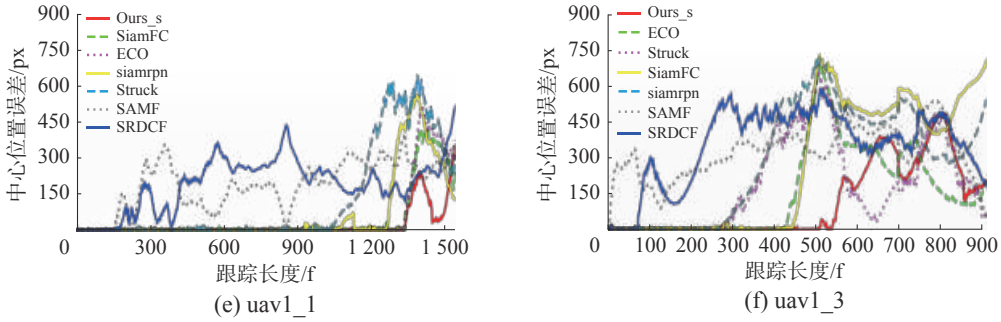


图 5　中心位置误差曲线

Fig. 5　Center position error curve

以 group3_2 为例，目标在 200 多帧时因遮挡跟丢，这时所有算法的中心位置误差都发生显著变化，并逐渐变高。在 300 多帧时，本文算法因再检测模块的作用，重新识别到目标，恢复跟踪，使中心位置误差又回到了较低的位置，明显改善了跟踪性能。对于 person16，由于 SiamRPN 和 ECO 算法在目标跟丢时，目标框位置刚好与目标再次出现在视野中的位置相近，使得目标能恢复跟踪，因此整体中心位置误差较小。而本文算法则通过再检测模块恢复了跟踪，由于有一段跟丢的图像序列，因此整体误差变大。对于 uav1_1 和 uav1_3，由于目标频繁地离开视野再回到视野，并且背景模糊，所有算法位置误差都产生了较大波动。相较于其他算法本文算法依然能使中心误差维持在一个更低的水平。

3.2　无人机跟踪实验

为了验证该算法在飞行器目标跟踪系统中的可行性，本文利用自主搭建的六旋翼飞行器进行实验。将 Realsense D435 相机水平安装在飞行器底部，以实时获取图像信息；通过搭载的 Nvidia Xavier NX 处理器实时地对输入图像运行跟踪算法；通过 ROS 系统实现地面站与机载处理器以及飞控之间的通信。最后用于飞行器对地面移动机器人进行自主跟踪，在跟踪过程中，使飞行器的飞行高度保持在 1.9 m，偏航角保持在 0°。飞行器结构如图 6 所示。跟踪结果如图 7 所示，当目标发生遮挡时，无人机悬停，停止跟踪，利用显著性检测，对目标进行检索，使得当目标再次出现在视野中时恢复跟踪。

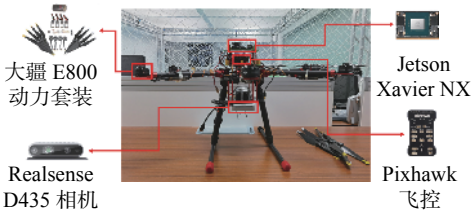
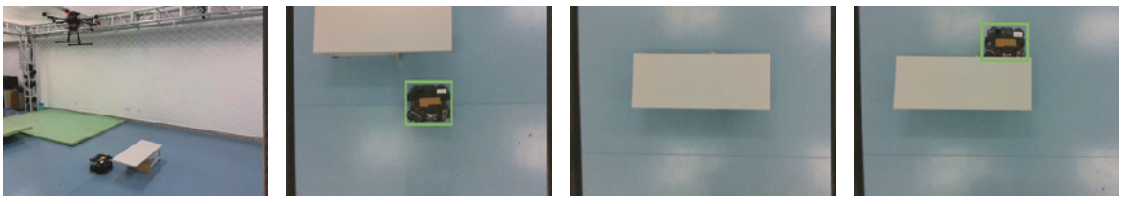


图 6　六旋翼飞行器结构

Fig. 6　Hexarotor structure diagram



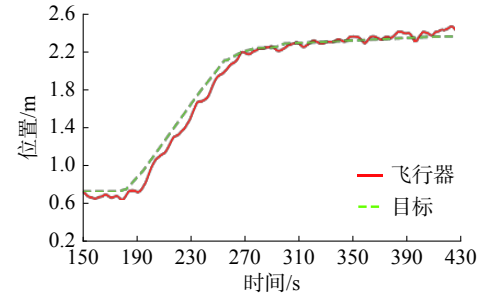
(a) 跟踪过程　(b) 遮挡前　(c) 遮挡时　(d) 遮挡后

图 7　跟踪结果

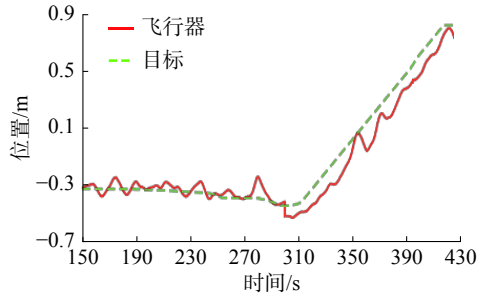
Fig. 7　Tracking results

为了对跟踪轨迹进行分析，利用定位系统，获取地面移动机器人坐标。将未发生遮挡时的地面移动机器人与飞行器的 X、Y 方向坐标以及飞行器当前高度与期望高度进行对比，结果如图 8 所示。从跟踪坐标曲线可以看出：在 150~280 s 时，目标 (地面移动机器人) 主要沿 X 方向运动，由于目标移动速度高于飞行器，这时飞行器 X 坐标值略微低于目标 X 坐标值，跟随在其后，而飞行器 Y 坐标围绕目标 Y 坐标上线波动，保持在目标 Y 坐标中心附近；在 280~400 s，目标主要沿 Y 方向移动，这时飞行器 Y 坐标值略微低于目标 Y 坐标值，跟随在其后，而飞行器 X 坐标围绕目标 X 坐标上下波动，保持在目标 X 坐标中心附近。总体来看，飞行器与目标在 X、Y 方向的坐标误差始终保持

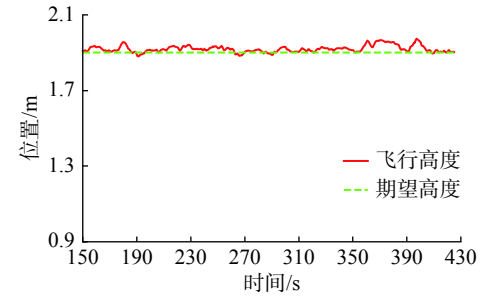
在 20 cm 以内, 飞行器的飞行高度与期望高度保持在 10 cm 以内, 保障了飞行器对目标的稳定跟踪。



(a) X 方向坐标对比



(b) Y 方向坐标对比



(c) 飞行高度对比

图 8　跟踪坐标曲线

Fig. 8　Tracking coordinate curve

4　结束语

本文针对旋翼飞行器目标跟踪问题提出了一种基于 MobileNetV2 的孪生网络目标跟踪算法, 该算法通过目标得分模块估计出目标初始位置, 在获取初始位置后, 利用目标尺度估计模块对目标位置及尺度进行迭代修正, 通过多层特征融合, 进一步提升算法性能。针对目标完全遮挡问题, 本文提出了一种基于显著性检测的目标再检测算法, 利用对显著性区域的再检测, 恢复对目标的跟踪, 并通过仿真实验证明了跟踪算法性能及再检测模块的作用, 最后, 通过实现飞行器对目标的跟踪, 证明了该算法的可行性。

参考文献:

[1] 徐怀宇, 黄伟, 董明超, 等. 无人机目标跟踪综述 [J]. 网络新媒体技术, 2019, 8(5): 11–20.
XU Huaiyu, HUANG Wei, DONG Mingchao, et al. Overview of UAV object tracking[J]. Journal of network new media, 2019, 8(5): 11–20.

[2] 刘芳, 孙亚楠, 王洪娟, 等. 基于残差学习的自适应无人机目标跟踪算法 [J]. 北京航空航天大学学报, 2020, 46(10): 1874–1882.
LIU Fang, SUN Yanan, WANG Hongjuan, et al. Adaptive UAV target tracking algorithm based on residual learning[J]. Journal of Beijing University of Aeronautics and Astronautics, 2020, 46(10): 1874–1882.

[3] HENRIQUES J F, CASEIRO R, MARTINS P, et al. Exploiting the circulant structure of tracking-by-detection with kernels[C]//12th European Conference on Computer Vision. Florence, Italy, 2012: 702−715.

[4] HENRIQUES J F, CASEIRO R, MARTINS P, et al. High-speed tracking with kernelized correlation filters[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(3): 583–596.

[5] DANELLJAN M, ROBINSON A, KHAN F S, et al. Beyond correlation filters: learning continuous convolution operators for visual tracking[C]//14th European Conference on Computer Vision. Amsterdam, The Netherlands, 2016: 472−488.

[6] DANELLJAN M, BHAT G, KHAN F S, et al. Eco: efficient convolution operators for tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, USA, 2017: 6638−6646.

[7] TAO Ren, GAVVES E, SMEULDERS A W M. Siamese instance search for tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 1420−1429.

[8] BERTINETTO L, VALMADRE J, HENRIQUES J F, et al. Fully-convolutional Siamese networks for object tracking[C]//European Conference on Computer Vision. Amsterdam, The Netherlands, 2016: 850−865.

[9] VALMADRE J, BERTINETTO L, HENRIQUES J, et al. End-to-end representation learning for correlation filter based tracking[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, USA, 2017: 2805−2813.

[10] LI Bo, YAN Junjie, WU Wei, et al. High performance visual tracking with Siamese region proposal network[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 8971−8980.

[11] REN Shaoqing, HE Kaiming, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 39(6): 1137–1149.

[12] JIANG Borui, LUO Ruixuan, MAO Jiayuan, et al. Acquisition of localization confidence for accurate object detection[C]//Proceedings of the European Conference on

Computer Vision (ECCV). Munich, Germany, 2018: 816−832.

[13] SANDLER M, HOWARD A, ZHU Menglong, et al. MobileNetv2: inverted residuals and linear bottlenecks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA, 2018: 4510−4520.

[14] LI Bo, WU Wei, WANG Qiang, 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. Long Beach, USA, 2019: 4277−4286.

[15] YU F, KOLTUN V. Multi-scale context aggregation by dilated convolutions[EB/OL]. (2020−12−29)[2021−01−10] https://arxiv.org/abs/1511.07122, 2016.

[16] HE Kaiming, ZHANG Xiangyu, REN Shaoqing, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 770−778.

[17] FAN Heng, LIN Liting, YANG Fan, et al. LaSOT: a high-quality benchmark for large-scale single object tracking[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, USA, 2019: 5369−5378.

[18] HUANG Lianghua, ZHAO Xin, HUANG Kaiqi. 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.

[19] CORNIA M, BARALDI L, SERRA G, et al. Predicting human eye fixations via an LSTM-based saliency attentive model[J]. IEEE transactions on image processing, 2018, 27(10): 5142–5154.

[20] KRONER A, SENDEN M, DRIESSENS K, et al. Contextual encoder-decoder network for visual saliency prediction[J]. Neural networks, 2020, 129: 261–270.

[21] WANG Wenguan, SHEN Jianbing. Deep visual attention prediction[J]. IEEE transactions on image processing, 2018, 27(5): 2368–2378.

[22] CORNIA M, BARALDI L, SERRA G, et al. A deep multi-level network for saliency prediction[C]//2016 23rd International Conference on Pattern Recognition (ICPR). Cancun, Mexico, 2016: 3488−3493.

[23] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[EB/OL]. (2020−12−19)[2021−01−10] https://arxiv.org/abs/1409.1556, 2014.

[24] LONG J, SHELHAMER E, DARRELL T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA, 2015: 3431−3440.

[25] JIANG Ming, HUANG Shengsheng, DUAN Juanyong, et al. SALICON: saliency in context[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA, 2015: 1072−1080.

[26] JETLEY S, MURRAY N, VIG E. End-to-end saliency mapping via probability distribution prediction[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA, 2016: 5753−5761.

[27] KINGMA D P, BA J. Adam: a method for stochastic optimization[EB/OL]. (2020−12−29)[2021−01−10] https://arxiv.org/abs/1412.6980, 2014.

[28] MUELLER M, SMITH N, GHANEM B. A benchmark and simulator for UAV tracking[C]//14th European Conference on Computer Vision. Amsterdam, The Netherlands, 2016: 445−461.

[29] LI Yang, ZHU Jianke. A scale adaptive kernel correlation filter tracker with feature integration[C]//European Conference on Computer Vision. Zurich, Switzerland, 2014: 254−265.

[30] DANELLJAN M, HÄGER G, KHAN F S, et al. Learning spatially regularized correlation filters for visual tracking[C]//Proceedings of the IEEE International Conference on Computer Vision. Santiago, Chile, 2015: 4310−4318.

[31] HARE S, GOLODETZ S, SAFFARI A, et al. Struck: structured output tracking with kernels[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 38(10): 2096–2109.

作者简介：

周士琪，硕士研究生，主要研究方向为计算机视觉、移动机器人。

王耀南，中国工程院院士，教授，博士生导师，主要研究方向为智能控制、计算机视觉、机器人控制。获国家技术发明奖二等奖、国家科学技术进步奖二等奖、教育部科学技术进步奖一等奖等。获授权发明专利 70 余项，发表学术论文 200 余篇，出版著作 8 部。

钟杭，博士研究生，主要研究方向为飞行器建模与控制、机器人技术。