



·前沿技术与交叉科学·

# 复杂背景下的低空无人机检测与跟踪算法\*

汪建伟<sup>1,2</sup>, 游 疆<sup>1,2</sup>, 万 敏<sup>1</sup>, 顾静良<sup>1</sup>

(1. 中国工程物理研究院 应用电子学研究所, 四川 绵阳 621900; 2. 中国工程物理研究院 研究生院, 北京 100088)

**摘 要:** 提出一种基于 YOLOv5 与 CSRT 算法优化的实时长跟踪方法, 实现了对无人机在净空、城市、森林等场景的稳定跟踪。针对跟踪的不同阶段建立不同分辨率的两个捕获网络, 分别对两个网络进行小目标检测优化和性能优化, 并根据无人机数据集特点对其进行正负样本的添加以实现数据增强。然后, 对 CSRT 算法使用 GPU 进行优化并结合特征点提取构建了低空无人机检测与跟踪模型。最后, 将算法使用 Tensorrt 部署后在自建数据集上进行实验, 实验结果表明, 所提方法在 RTX 2080Ti 上实现了 400FPS 的跟踪性能, 在 NVIDIA Jetson NX 上实现了 70FPS 的性能。在实际外场实验中也实现了稳定的长时间跟踪。

**关键词:** 无人机检测; 实时跟踪; 复杂背景; 机动目标; 无人机反制

中图分类号: TP391.4

文献标志码: A

doi: 10.11884/HPLPB202335.230026

## Low-altitude UAV detection and tracking algorithms in complex backgrounds

Wang Jianwei<sup>1,2</sup>, You Jiang<sup>1,2</sup>, Wan Min<sup>1</sup>, Gu Jingliang<sup>1</sup>

(1. Institute of Applied Electronics, CAEP, Mianyang 621900, China;

2. Graduate School of China Academy of Engineering Physics, Beijing 100088, China)

**Abstract:** With the frequent appearance of UAVs in several recent local wars and armed conflicts, the study of UAV detection and tracking technology has become a research hotspot in imagery and other fields. Due to the characteristics of low altitude UAV targets such as large mobility, small size, low contrast and complex background, their capture and tracking is a major challenge in the field of photoelectric detection. To address these difficulties, this paper proposes a real-time long tracking method based on YOLOv5 and CSRT algorithm optimization to achieve stable tracking of UAVs in clear sky, urban and forest scenes. First, two capture networks with different resolutions are established for different stages of tracking, and the two networks are optimized for small target detection and performance optimization respectively, and positive and negative samples are added to the UAV data set according to its characteristics to achieve data enhancement. Then, the CSRT algorithm is optimized using GPU and combined with feature point extraction to construct a low-altitude UAV detection and tracking model. Finally, the algorithm is deployed using Tensorrt and experimented on a self-built dataset. The experimental results show that the proposed method achieves a tracking performance of 400FPS on RTX 2080Ti and 70FPS on NVIDIA Jetson NX. Stable long-time tracking is also achieved in real field experiments.

**Key words:** drone detection, real-time tracking, complex background, maneuvering target, drone countermeasures

随着以大疆为代表的消费级无人机市场的快速增长, 以及小型飞行器在大众生活中的快速普及, 无人机 (UAV) 逐渐被应用于各种领域, 如航拍、农业、植保、快递运输、灾难救援、观察野生动物、监控传染病、测绘、新闻报道、电力巡检和军事领域。以四旋翼无人机为代表的“低慢小”目标带来的低空安全威胁越来越大。一方面, 在繁华的城市飞行严重威胁了公众的生命财产安全, 城市中的无人机伤人事件常有发生; 另一方面, 小型无人机隐

\* 收稿日期: 2023-02-15; 修订日期: 2023-04-11

联系方式: 汪建伟, [daozhixs@gmail.com](mailto:daozhixs@gmail.com);

游 疆, [youjiang09@163.com](mailto:youjiang09@163.com)。

通信作者: 万 敏, [wanmin@caep.cn](mailto:wanmin@caep.cn);

顾静良, [20332079@qq.com](mailto:20332079@qq.com)。

身特性好、突防能力强、获取成本低廉、使用灵活多样,可执行多种军事任务,这种非传统的攻击方式对国家安全会产生巨大的负面影响。因此亟需一种有效的低空无人机反制系统,而在无人机反制系统中,检测与跟踪是最重要的部分,是进行反制的基础。目前,常见的无人机探测技术手段包括雷达探测<sup>[1]</sup>、无线电频谱探测<sup>[2]</sup>、声波探测<sup>[3]</sup>等。但由于设备成本高昂、体积大、探测过程中容易受干扰等局限性,以上探测设备的实用化较少。光电探测系统具有价格更低、看得更清晰、打得更准、反应更快和生存能力更强的优势,并且随着人工智能技术的飞速发展,基于计算机视觉的信息感知技术取得了长足进步,因此,针对低空无人机,基于图像的检测与跟踪技术成为了最常用的技术之一。

低空无人机具有体积小、机动性高、背景复杂等特点,而且无人机反制系统需要一定的实时性要求,为了提高无人机目标的检测速度与跟踪稳定性,本文提出了一种基于YOLOv5<sup>[4]</sup>和CSRT<sup>[5]</sup>算法优化的目标检测与跟踪方法。首先,对自建数据集进行数据增强等预处理。其次,根据不同的跟踪阶段在YOLOv5的基础上设计两个不同分辨率的检测网络:UavYOLO-S和UavYOLO-B,其中UavYOLO-B网络的输入图像尺寸为1024×1024,用于从大视场中捕获无人机;根据自动学习不同特征通道重要程度的思想,在主干网络的空间金字塔池化(SPP)前加入SE注意力模块<sup>[6]</sup>。由于大视场中无人机的尺寸较小,在预测层增加了160×160的检测特征图,用于强化对小目标的检测能力。UavYOLO-S网络的输入图像尺寸为320×320,用于目标跟踪过程中目标丢失时重新捕获,为了保证目标丢失时能快速再次捕获,该网络主要进行性能优化。使用轻量化的Ghostnet<sup>[7]</sup>代替了特征混合层中的C3结构。在该视场中无人机的尺寸普遍较大,为了提升性能,去除了预测层中的P3结构。然后,使用GPU优化了CSRT算法中的矩阵运算,本论文采用的输入图像为灰度图,故去除了CSRT算法中的CN特征<sup>[8]</sup>。最后,将优化的两个YOLOv5网络作为目标检测模型,结合CSRT目标跟踪算法和特征点检测算法设计了一整套跟踪与检测策略并进行算法部署,在保证检测与跟踪准确率的同时满足了实时跟踪。实验结果表明,所提方法对于复杂背景下低空无人机目标的检测与跟踪取得了良好的效果。

## 1 目标检测与跟踪算法

目标检测与识别是指从一幅场景(图片)中找出目标,包括检测和识别两个过程。传统的目标检测技术大多基于HOG<sup>[9]</sup>、SIFT<sup>[10]</sup>、Haar<sup>[11]</sup>等特征,其泛化能力弱,所以占用计算资源大且效果差。近年来,随着深度学习技术的快速发展,基于深度学习的目标检测技术取得了很多成果,其中基于卷积神经网络(CNN)的目标检测算法可以有效地提取图像深层次特征,因而取得了优异的效果,并且具有很强的泛化能力。目前基于CNN的目标检测算法主要分为两类:一类是以R-CNN系列<sup>[11]</sup>为代表的两阶段目标检测算法,采用枚举方法验证预设的候选区域,然后再逐步微调检测框,检测速度较慢但是精度比较高;另一类是以YOLO系列<sup>[12]</sup>以及SSD<sup>[13]</sup>模型为代表的单阶段目标检测算法,此类算法通过应用不同的卷积核,直接预测特征图中检测框的位置和类别,不仅能保证一定的精度,检测速度也能满足实时性要求。并且随着YOLO不同版本的迭代,近几年在目标检测领域展现出很大的优势<sup>[14-15]</sup>。

目标跟踪技术是最受关注的计算机视觉问题之一,由于运动目标的尺度变化、遮挡和背景杂波等,导致目标跟踪仍然是一项极具挑战性的任务。视觉目标跟踪方法主要分为生成类跟踪方法<sup>[16-17]</sup>和判别类跟踪方法<sup>[18-20]</sup>。生成类跟踪方法通常忽略背景信息的影响且假设目标外观在一定时间内保持不变,故该方法无法处理和适应复杂的跟踪变化。判别类跟踪算法出现之后,经典的生成类跟踪算法逐渐淡出。2012年,以AlexNet为代表的深度学习方法在图像处理领域的成功应用,使目标跟踪开始迈入深度学习时代。近年来目标检测中的CNN技术也被应用于目标跟踪技术,不仅提高了目标跟踪的精度,还提高了跟踪的可靠性。2016年,冉涛等人将孪生网络应用在目标跟踪上提出了SINT算法<sup>[21]</sup>,开创性地将目标跟踪问题转换为一个图像patch块匹配问题。宋奕兵等人在MDNet的基础上加入生成对抗网络(GAN),在2018年提出了VITAL算法<sup>[22]</sup>,解决了目标正样本数量少、重叠率高的问题。基于深度学习的目标跟踪技术虽然精度较高,但计算量过大,无法部署在边缘计算设备上,所以实际应用还很少。

### 1.1 YOLOv5 算法

YOLO系列目标检测将目标检测看作一个回归问题,使用一个神经网络实现了在一次推理中直接从图像预测出边界框和类别概率。YOLOv5是在YOLOv4算法的基础上做了进一步改进,检测性能得到进一步提升,其网络结构如图1所示。YOLOv5与YOLOv4有着相似的网络结构,都使用了CSPDarknet53作为主干特征提取网络,并且使用了PANET和SPP作为Neck网络,而且都使用YOLOv3的结果预测网络。YOLOv5在输入端增加了

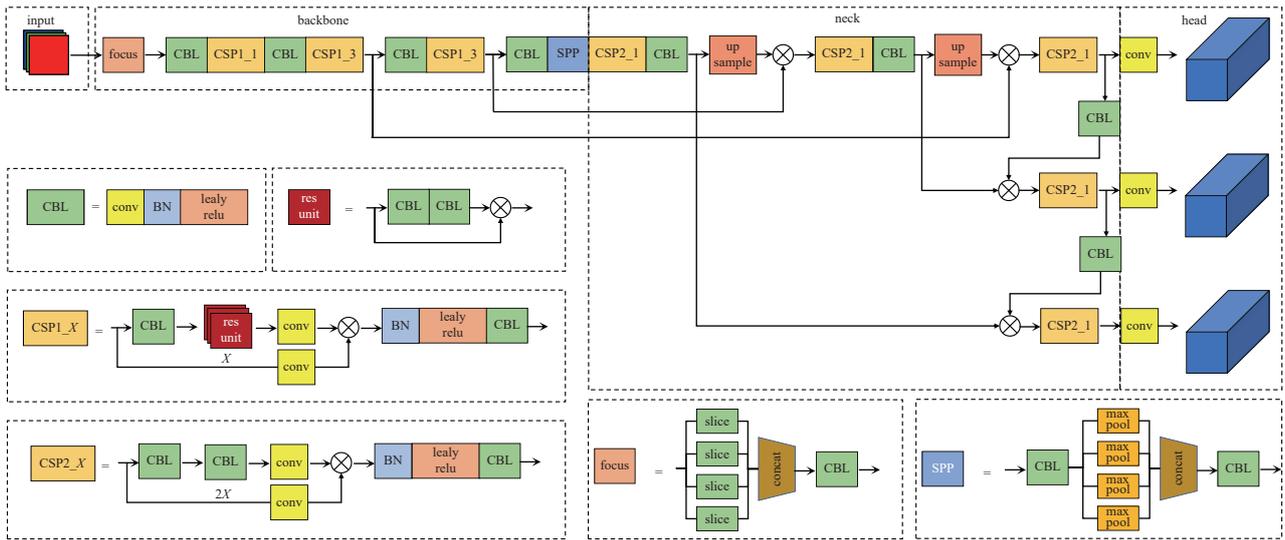


Fig. 1 Schematic diagram of YOLOv5 structure

图 1 YOLOv5 结构示意图

Mosaic 数据增强和自适应锚框计算, 在主干网络中引入了 Focus 结构, 在 Neck 网络中采用了借鉴 CSPNet 设计的 CSP2\_X 结构, 并且在预测网络中改进了边界框的损失函数。

YOLOv5 进行了三种数据增强: 缩放、色彩空间调整和马赛克增强。其中马赛克增强是通过将四张图像进行随机缩放、随机裁剪、随机分布方式进行拼接, 使得小目标的检测效果得到提升。YOLOv4 中的锚框沿用了 YOLOv3 的策略, 即预先使用 k-means 聚类算法<sup>[23]</sup> 计算, YOLOv5 中的锚框根据训练数据自动学习。YOLOv4 和 YOLOv5 都使用 CSPDarknet 作为主干特征提取网络, 从输入图像中提取特征信息, 而 YOLOv4 只有主干网络中使用了 CSP 结构, YOLOv5 中设计了两种 CSP 结构, 如图 2 所示, CSP1\_X 应用于主干网络, 另一种 CSP2\_X 结构则应用于 Neck 网络中, 既保证了推理速度和准确率, 又减小了模型尺寸。YOLOv4 使用 Ciou Loss 作为预测边界框的损失函数, 而 YOLOv5 使用 GIOU Loss 代替 Ciou Loss 来提高 YOLOv5 的训练速度, Ciou Loss 损失函数的表达式为

$$L_{\text{GIOU}} = \frac{|A \cap B|}{|A \cup B|} - \frac{|V|(A \cup B)}{|V|} \quad (1)$$

式中:  $A$  是真值,  $B$  是预测框,  $V$  是这两个区域的闭包。YOLOv5 新引入的 Focus 结构是在图片进入主干特征提取网络之前对图片进行切片操作, 具体操作是在一张图片中每隔一个像素拿到一个值组成新的图片, 这样得到的结果是输入通道扩充了 4 倍, 最后将得到的新图片再经过卷积操作, 最终得到了没有信息丢失情况下的二倍下采样特征图, Focus 结构可以降低参数量以达到增加检测速度的目的。

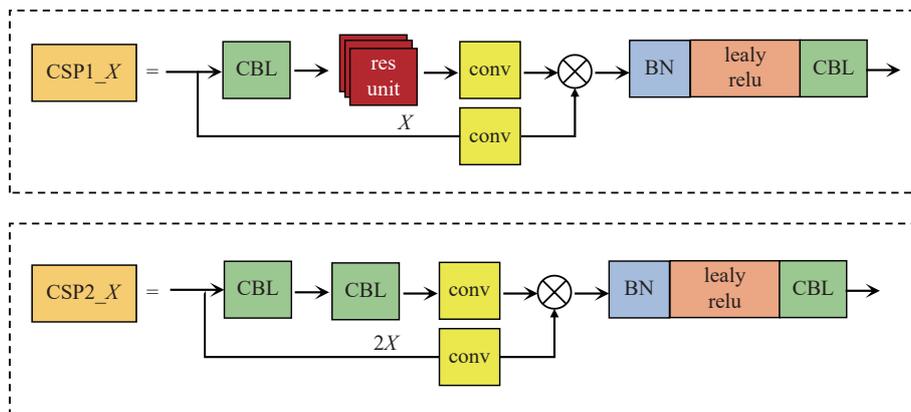


Fig. 2 Two CSP structures in YOLOv5

图 2 YOLOv5 中的两种 CSP 结构

YOLOv5 还包含了一些其他的创新, 例如隐藏层使用了 Leaky ReLU 激活函数, 最后的检测层使用了 Sigmoid 形激活函数, 从而优化了检测的速度。在 MS COCO 数据集上与 YOLOv4 相比较, 在保证检测精度不变的情况下,

检测速度上提升了10%，并且模型的大小缩小为YOLOv4的11%，这非常有利于在边缘计算模型上部署。YOLOv5实现了检测精度与速度的平衡，综合性能优于SDD、YOLOv4等算法。

## 1.2 CSRT 算法

CSRT算法即具有通道和空间可靠性的判别相关滤波器，是一种判别类跟踪算法，CSRT使用空间可靠性图将相关滤波器调整为从帧中的目标框选择区域的一部分以进行跟踪。这确保了所选区域的放大和定位，并改善了对非矩形区域或对象的跟踪，其框架如图3所示。它仅使用2个特征(HoG特征和ColorName特征)进行相关滤波。CSRT提出之前的基于相关滤波的跟踪算法，主要存在3方面的问题：(1)由于使用了快速傅里叶变换，导致滤波器尺寸与图像块尺寸必须一致，这样就限制了检测范围；(2)由于引入了循环矩阵，导致滤波器的训练数据和真实数据并不一样；(3)只能支持矩形目标，与真实世界不符合。CSRT为解决上述问题，提出了“空间置信”，其主要思想是利用图像分割方法产生适应能力更好的蒙版，这个空间置信图就是通过下面后验概率的求解得到的。其公式为

$$p_p = \frac{p_o h}{p_o h + p_b h} \quad (2)$$

式中： $p_p$ 是后验概率； $p_o$ 和 $p_b$ 是先验概率，可以认为是固定的数值； $h$ 为前景的似然，与采样数据中的前景/背景区域有关，而 $h$ 相当于似然，可以理解为“在给定的前景图像中，该像素属于目标的概率(疑似目标)”，最后求解后验概率 $p_p$ 可以理解为“在给定的目标图像条件下，该像素属于前景蒙版的概率”。CSRT在计算似然时，用到了反向投影方法。就是利用反向投影中的颜色信息，在采样范围内捕捉与目标颜色相近的部分，作为似然(疑似目标)。根据贝叶斯后验概率公式，在已知先验概率和似然的前提下，可以求出后验概率，因此能够得出最终需要的蒙版的概率图，求解公式为

$$p(m=1|y, x) \propto p(y|m=1, x) \cdot p(m=1) \quad (3)$$

式中： $m$ 为每个像素对应的学习可靠性。求解得到蒙版的概率图后就可以得到新的蒙版，即被跟踪目标新的位置。

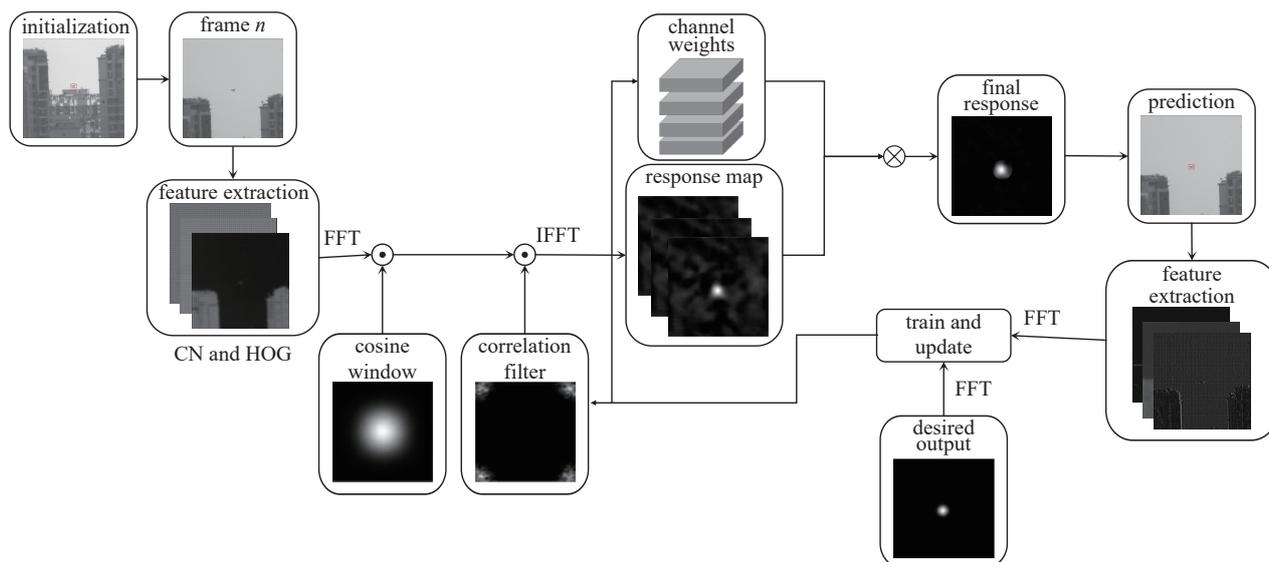


Fig. 3 Block diagram of the CSRT algorithm

图3 CSRT算法的原理框图

从物理意义上分析，这种思路可以理解为用先验概率图与似然概率图进行的某种图像卷积而形成的滤波过程，最终实现了对不规则形状的物体的自适应，这种创新思路使得目标跟踪的精度得到了提高，并且可以保证一定的跟踪可靠性，但是该算法的性能较低，原始算法的跟踪速度只能达到25帧/s，需要对算法进行优化才能满足无人机目标跟踪的实时性需求。

## 2 优化的目标检测与跟踪算法

所提方法首先在YOLOv5算法的基础上提出两种适应于不同跟踪阶段的目标检测模型，提高了无人机目标检



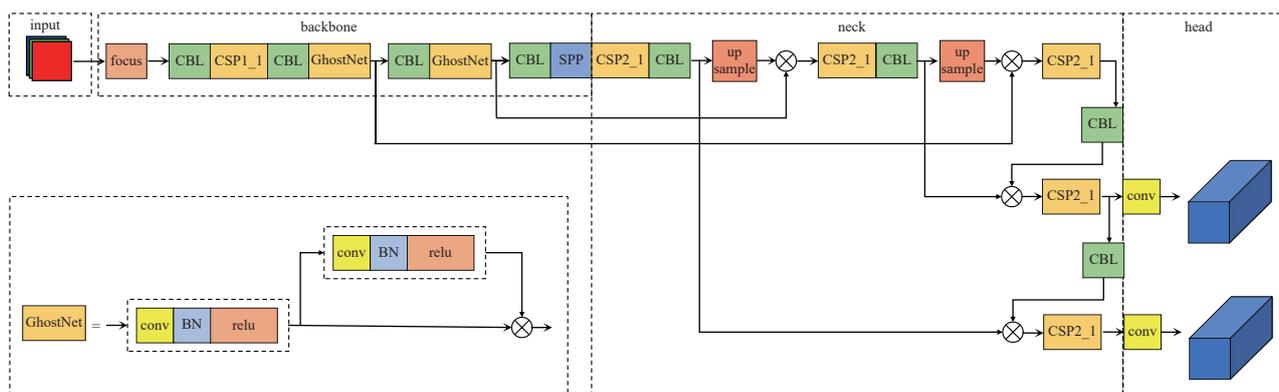


Fig. 5 Schematic diagram of UavYOLO-S structure

图 5 UavYOLO-S 结构示意图

掉原来结构中的 3×3 标准卷积减少了计算量。当输入图像为 320×320 时,原 YOLOv5 网络的第一个 Head 网络特征图大小为 40×40,此预测网络用于预测尺寸大于 8×8 的无人机目标。经过对跟踪视场中无人机尺寸的统计发现在跟踪视场中无人机均大于 16×16,40×40 特征图对应的 Head 网络实际上没有起到预测作用,所以在 UavYOLO-S 中去除了该预测层,进一步减少了计算量,提升了检测速度。

## 2.2 优化的 CSRT 跟踪算法

CSRT 算法运行速度较慢的主要原因是算法中使用了大量的矩阵运算,而 CPU 并不擅长矩阵运算,所以本文对 CSRT 算法中的部分矩阵运算使用 GPU 进行了优化。本文使用的无人机图像均为灰度图像,CSRT 算法中使用的 ColorName 特征并没有起到作用,所以直接去除了 ColorName 特征部分的计算,只使用 HOG 特征进行目标跟踪。计算 HOG 特征时设计的矩阵运算主要有图像梯度求取和图像金字塔求取,本文对这两种操作分别做了 GPU 优化。

图像梯度的求解是通过卷积运算实现的,使用 GPU 进行卷积运算的性能远优于 CPU,图像梯度的求取分为水平梯度的求取与垂直梯度的求取,两个方向的梯度求解使用的卷积运算均满足公式

$$f(t) * k(t) = \sum_{j=-r}^r f(j+r)k(t+j), \quad (4)$$

式中:  $f(t)$  代表图像,  $k(t)$  代表卷积核,  $r$  代表卷积核的半径。使用 GPU 计算卷积的核心是将图像合理分块, GPU 中的多个线程并行处理,每个线程可以计算出图像卷积的部分结果,将所有线程的运算结果结合起来就得到了图像的卷积结果。本文采用共享内存的处理方式,进行图像分块的策略为每两个图像块相距 32 个像素,等于 NVIDIA GPU 编程模型中一个 warp 的大小,这样同一线程在处理图像块中的像素点时可以高效地利用内存(线程块内部的线程是以 32 个线程组成为一个 warp 进行执行的),图像分块策略如图 6 所示。卷积核长度为  $k$  时进行卷积操作,每个线程需要访问以当前像素为中心与之水平或垂直相邻的  $k-1$  个像素,所以如果每一个线程将当前像素用到的数据拷贝到共享内存中,那么处理每个图像块的边缘像素时部分线程会处于空闲状态,空闲线程数目会随着卷积核长度的增长而增加。为了减少空闲线程,每个线程必须多载入相邻的  $k-1$  个像素到共享内存中,使每个线程块处理多个目标像素的像素值。具体到水平和垂直梯度的求解,对于水平梯度求解,增加每个线程处理的像素等同于增大处理的图形块的宽度,对于垂直梯度的求解,则需要增大图形块的处理高度。按照以上的思路操作,每个线程块一次处理图像中的一个部分,使用多个线程块即可完成整个图像块的处理,在这个过程中多个线程对图像并行处理,充分利用了共享存储器和 GPU 运算器,从而提高了程序的性能。

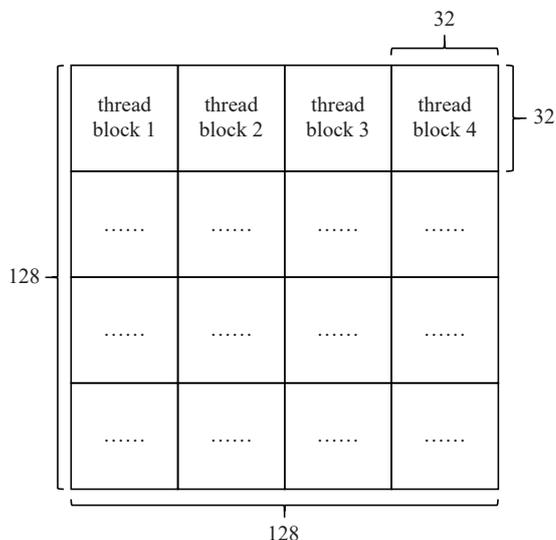


Fig. 6 Schematic diagram of image gradient block calculation

图 6 图像梯度分块计算示意图

CSRT 算法中用到的图像金字塔是使用双线性插值实现的,最底层图像数据与原始数据相同,之后的每一层图像均是在它的下一层图像的基础上通过双线性插值操作得到。图像金字塔的最底层结果直接从主机端拷贝到设备端,然后将每一层图像金字塔在设备端分配好内存进行计算。由于当前层图像的每个像素点的值都是由上一层图像的 4 个像素值经过双线性插值得到的,因此在计算该层的每个像素时都需要访问上层图像的 4 个像素值,图像金字塔相邻的两层操作如图 7 所示。本文采用共享内存的存取方式时,线程通过白色区域的 4 个像素值计算出灰色区域的 9 个像素值,将白色区域的像素拷贝到线程块的共享内存中,在后续的计算中线程块中的每个线程都需要访问白色区域的像素值。一个线程块内的线程利用 4 个像素完成了对 9 个像素值的求解,发挥了共享内存的优势。本文设计的线程处理方式如图 7 所示,其中每个红色箭头代表一个线程,每个线程需要计算出当前层的 8 个像素值,每 32 个线程作为一个 warp 执行。在以上计算过程中,每一层的图像都是由上一层的图像插值得到的,此时在计算过程中还需要存储上一层的计算结果,浪费了存储资源。本文的改进方式是每层图像都从原始图像直接插值计算得来,不存储上一层的计算结果,在性能不变的前提下节约了宝贵的存储资源,提高了存储资源的利用率。

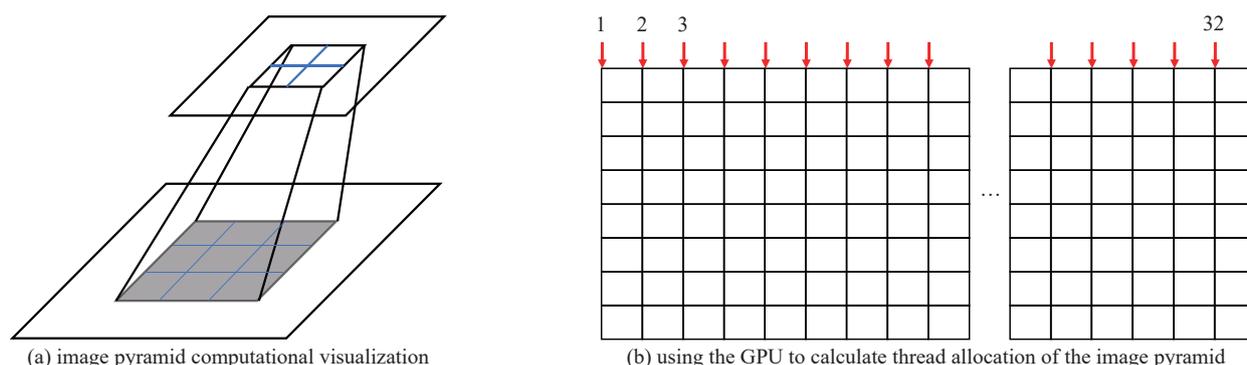


Fig. 7 Schematic diagram of image pyramid calculation

图 7 图像金字塔计算示意图

CSRT 算法中 HOG 特征的计算基于图像金字塔操作,在金字塔的各个层级均需计算 HOG 描述子,在每次 HOG 描述子的计算过程中需要求取梯度信息,从而构造梯度方向直方图,梯度求解与 HOG 特征的计算是 CSRT 算法中使用最频繁的两种操作,将上述使用 GPU 优化的图像梯度求取和图像金字塔求取应用于 CSRT 算法中使得目标跟踪性能得到了显著提升。

### 2.3 特征点提取算法

由于无人机在图像上跟踪的结果为一个边界框,所以需要确定出一个点作为无人机反制任务中的打击点。打击点的提取可以基于边缘、分割或者特征点检测,即提取无人机的轮廓、线条和特征点等信息计算出用于打击的目标点。本文针对不同的跟踪场景分别使用质心特征点提取和基于分割的特征点提取。由于伺服系统存在抖动并且无人机在飞行过程中也会出现一定的抖动,所以还需要对跟踪点做滤波处理,本文使用卡尔曼滤波对得到的目标点进行处理,使用卡尔曼滤波的另一个好处是在目标丢失时可以向前继续预测,如果目标是由于短暂遮挡丢失,在向前预测一段时间后还可以继续对目标进行跟踪,提高了算法的实用性。

特征点提取算法分为两个部分实现:一部分是针对无人机跟踪的结果进行提取,从而确定出打击点;另一部分为打击起火后对起火区域质心的提取,用于起火后的目标跟踪。对于无人机起火后的场景,使用简单的图像质心提取就获得了良好的效果,对于无人机跟踪结果的特征点提取,本文经过多次尝试提出了一种基于阈值分割后再次提取质心的特征点提取方法,能够满足打击点提取的需求。

图像的质心,也称为图像的重心,即对于  $x$  方向的质心,图像在质心左右两边像素和相等;对于  $y$  方向的质心,图像在质心上下两边像素和相等。以  $x$  方向质心计算为例,公式为

$$x = \frac{\sum_{i=1}^n p_i x_i}{\sum_{i=1}^n p_i}, \quad (5)$$

式中:  $x_i$  为每一像素在  $x$  方向上的坐标,  $p_i$  为该像素的像素值。打击起火后的无人机被光斑淹没,所以使用质心提取可以有效获得无人机的跟踪点,如图 8 所示。对于跟踪过程中的无人机需要先确定出无人机所在区域,然后再对

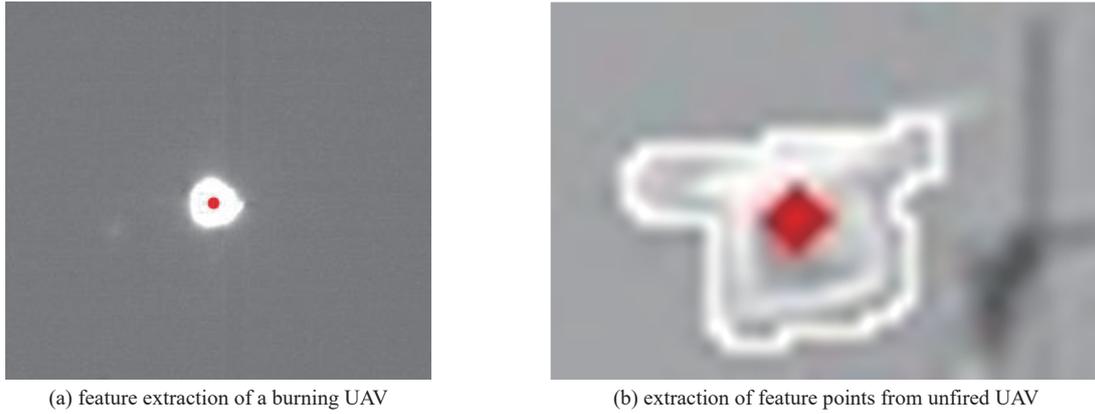


Fig. 8 Sketch map of UAV feature point extraction

图 8 无人机特征点提取示意图

该区域求质心得到无人机的打击点。首先对无人机跟踪结果使用大津阈值<sup>[24]</sup>进行分割得到二值化图像, 然后对二值化后的图像先做开运算再做闭运算去除分割后图像中的噪声和感染, 最后使用 Laplacian 算子对图像进行边缘检测就得到了无人机的轮廓, 无人机打击点就是通过求解该轮廓的质心得到, 如图 8 所示。

#### 2.4 结合检测跟踪算法与特征点提取的无人机检测跟踪模型

针对无人机反制系统的跟踪算法需要包括捕获、跟踪和跟踪点提取等部分, 整个跟踪策略的框图如图 9 所示。在无人机目标进入视场时, 捕获算法首先识别到目标, 并给出精确的目标框, 将当前帧送入跟踪器对跟踪模块进行初始化; 跟踪器初始化之后对每一帧进行持续跟踪, 跟踪器输出的结果是目标框; 要实现对该目标的打击还需要提取出打击点, 使用跟踪器输出的目标框在原始图像上取 ROI 后进行跟踪点提取, 为了使跟踪系统进行闭环控制, 需要根据当前帧计算的打击点和设定的跟踪点计算出当前的跟踪脱靶量发送给伺服控制系统进行闭环跟踪; 当跟踪过程中目标丢失的时候需要再次启动检测模块对目标进行再次识别, 为了加快检测速度, 先使用 UavYOLO-S

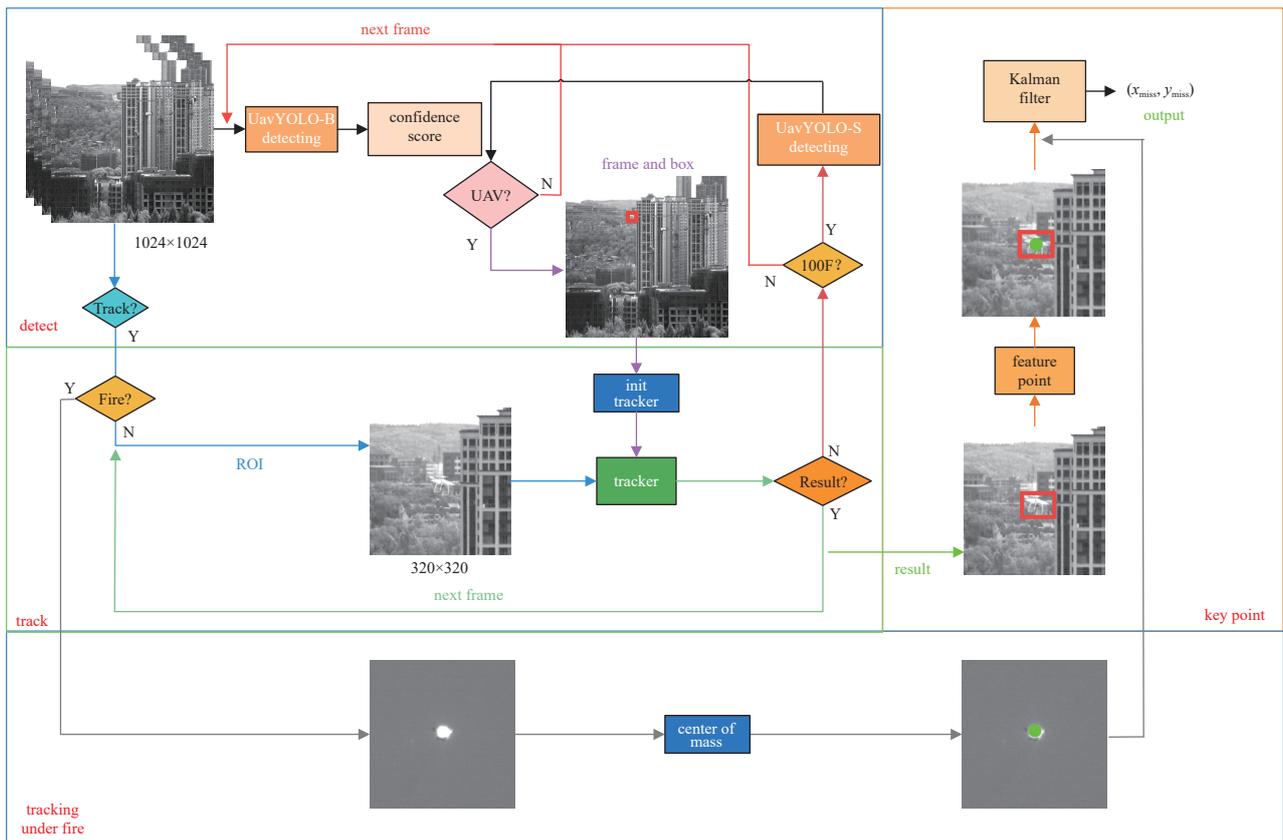


Fig. 9 Block diagram of UAV detection and tracking model combining detection and tracking algorithm and feature point extraction

图 9 结合检测跟踪算法与特征点提取的无人机检测跟踪模型框图

网络在目标丢失位置 320×320 区域内进行检测,如果检测不到目标再使用 UavYOLO-B 网络进行全靶面检测。

### 3 结果与分析

优化的 YOLOv5 算法提高了对无人机目标检测的精度与速度,与优化的 CSRT 算法和特征点提取算法结合可以实现对无人机的实时检测与跟踪。为了验证所提方法的整体性能,将优化后的检测算法与原始 YOLOv5 算法进行对比实验,然后将所提的整套算法应用于无人机目标反制系统中进行实验以检测算法的整体效果。在研究过程中,本文自行采集并构建了一个数据集用于目标检测算法的训练与评估,该数据集由大疆精灵 3、大疆精良 4、大疆 Mavic Air、穿越机等各种型号的旋翼无人机构成,包含了净空、森林、城市等背景,包括无人机的平飞、机动、快速升降等姿态,共有原始图片 10000 张。采用 labelme 图片标注工具按照 VOC 数据集格式对原始图片进行人工标注。为了提升研究方法的检测精度,采取水平镜像、小幅度旋转、模糊、随机添加飞鸟等方式对原始无人机图片进行数据增强。将所有算法部署在 NVIDIA Jetson Xavier NX 平台上进行实验,CPU 为 NVIDIA Carmel ARM@v8.2 64-bit,内存为 8 GB, GPU 为 NVIDIA Volta™ GPU,运行库版本为 Cuda10.1、cudnn8.3.1 与 TensorRT7.0.0。

#### 3.1 目标检测对比实验

为验证优化后 YOLOv5 对复杂背景下低空无人机目标的检测能力,分别对原 YOLOv5 算法和本文所提的两种改进模型在自建数据集上进行对比实验,主要对比不同算法的检测准确率和速度,所有模型均进行 INT8 量化后使用 TensorRT 进行部署。为保证对比的有效性,原算法与改进后的算法均采用 PyTorch 深度学习框架并设定相同的训练参数迭代至损失函数收敛。在实验中,分别采用平均精度与检测速度进行定量分析,表 1 为 3 种算法的对比结果。其中 AP50 与 AP75 分别为交并比阈值为 50% 和 75% 时的检测精度,表 1 的结果表明,所提 UavYOLO-B 模型的 AP50 和检测速度分别为 89.1% 和 25 帧/s,相比于原 YOLOv5 算法速度有所下降,但是在检测精度上有大幅提升,适合作为大视场中无人机目标的捕获;所提 UavYOLO-B 模型的 AP50 和检测速度分别为 88.3% 和 200 帧/s,相比于原 YOLOv5 算法精度有明显提升,速度有非常大的提升,可以满足跟踪丢失时的快速检测需求。在 IOU 为 0.5 的情况下,优化后的 UavYOLO-B 模型检测精度较原 YOLOv5 算法提升了 2.9%, UavYOLO-S 模型检测精度较原 YOLOv5 算法提升了 2.1%,检测速度提升了 165 帧/s。

表 1 不同检测算法性能对比

Table 1 Performance comparison of different detection algorithms

algorithm	size of input image/pixel	$\delta_{AP50}/\%$	$\delta_{AP75}/\%$	detection speed/(frames·s <sup>-1</sup> )
YOLOv5	1024×1024	86.2	57.8	35
UavYOLO-B	1024×1024	89.1	59.2	25
UavYOLO-S	320×320	88.3	58.5	200

选取两组低空无人机图片,一组为大视场捕获场景,一组为跟踪丢失时重新检测场景,分别利用上述两种模型进行目标检测,可视化检测效果如图 10 所示,从图 10 可以看出,改进后的算法能够准确检测出无人机并且准确度较高。

#### 3.2 复杂背景下低空无人机检测与跟踪实验

将本文提出的整套目标检测与跟踪算法应用于无人机反制系统中进行实验,以大疆精灵 4 为测试无人机,在距离无人机 1 km 处进行实验,相机分辨率为 1024×1024,采集频率为 85 帧/s,使用跟踪脱靶量来衡量整体跟踪效果,公式为

$$M = \sqrt{\frac{\sum_c (x_c - x_0)^2}{N}} \quad (6)$$

式中:  $M$  为跟踪脱靶量,  $x_i$  为当前帧目标位置,  $x_0$  为目标闭环位置,  $N$  为总跟踪帧数,实验结果如表 2 所示。

测试结果表明,本文改进的目标检测与跟踪算法具有较高的精度和稳定性,也可以满足实时性要求,可以应对复杂背景下的低空无人机目标。在外场实验中,该算法不仅可以准确地捕获 1 km 处的无人机目标,并且可以稳定地持续跟踪,最终脱靶量控制在反制系统打击要求之内,成功击落目标无人机。

#### 3.3 结果分析

本文提出的整套无人机目标检测与跟踪算法在测试平台上达到了 75 帧/s 地跟踪速度,所提算法地捕获成功



Fig. 10 Visual detection effects of small targets

图 10 可视化小目标检测效果

表 2 不同检测算法性能对比

Table 2 Performance comparison of different detection algorithms

algorithm	capture accuracy/%	miss distance/pixel	tracking speed/(frames·s <sup>-1</sup> )
YOLOv5	87.2	10.6	35
UavYOLO-B+UavYOLO-S	93.2	5.4	25
UavYOLO-B+UavYOLO-S+CSRT	97.8	4.8	75
UavYOLO-B+UavYOLO-S+CSRT+ KeyPoints	97.8	1.5	75

率达到了 97.8%，较优化之前提升了 10.6%，并且可以有效检测出 1 km 处地小目标，避免了复杂背景下小目标的漏检和误检情况，通过实际实验证明该算法的跟踪脱靶量达到 1.5 pixel 以内，可以满足无人机反制系统的打击需求，成功实现了复杂场景下无人机目标地实时检测与跟踪，对其中一段跟踪视频第 102、103、104 三帧连续帧和 140、160、180 三帧离散帧的跟踪效果进行可视化，效果如图 11 所示。

#### 4 结 论

本文提出了一种基于 YOLOv5 与 CSRT 算法优化的目标检测与跟踪算法，并将算法应用于复杂背景下低空无人机的检测与跟踪。针对不同的跟踪阶段设计了两个不同大小的目标检测网络，在大视场捕获网络中引入了注意力机制并增加了小目标检测层，使得对小目标的检测能力得到提升；在跟踪丢失重检测网络中使用 GhostNet 代替了 YOLOv5 算法中的 C3 结构并去除了部分预测网络，使得计算量明显降低，重检测性能得到了较大提升。在 CSRT 算法的基础上，对算法中的部分操作进行了 GPU 优化，使得无人机目标的跟踪可以满足实时性要求。在此基础上，将优化的目标检测和跟踪算法与特征点提取算法相结合，提出了一套用于无人机检测和跟踪的方法，实现了对复杂背景下低空无人机的稳定实时跟踪。将所提算法在自建数据集和外场实验中分别进行验证，实验结果表明，所提方法具有较高的准确性和实时性，有较强的泛化能力，可以满足对无人机目标的反制任务，有较高的应用价值。虽然所提方法在实验中取得了良好的效果，但是也碰到了无法应对的极端复杂场景，如何结合其他技术提高算法的适应能力是未来研究的重点。

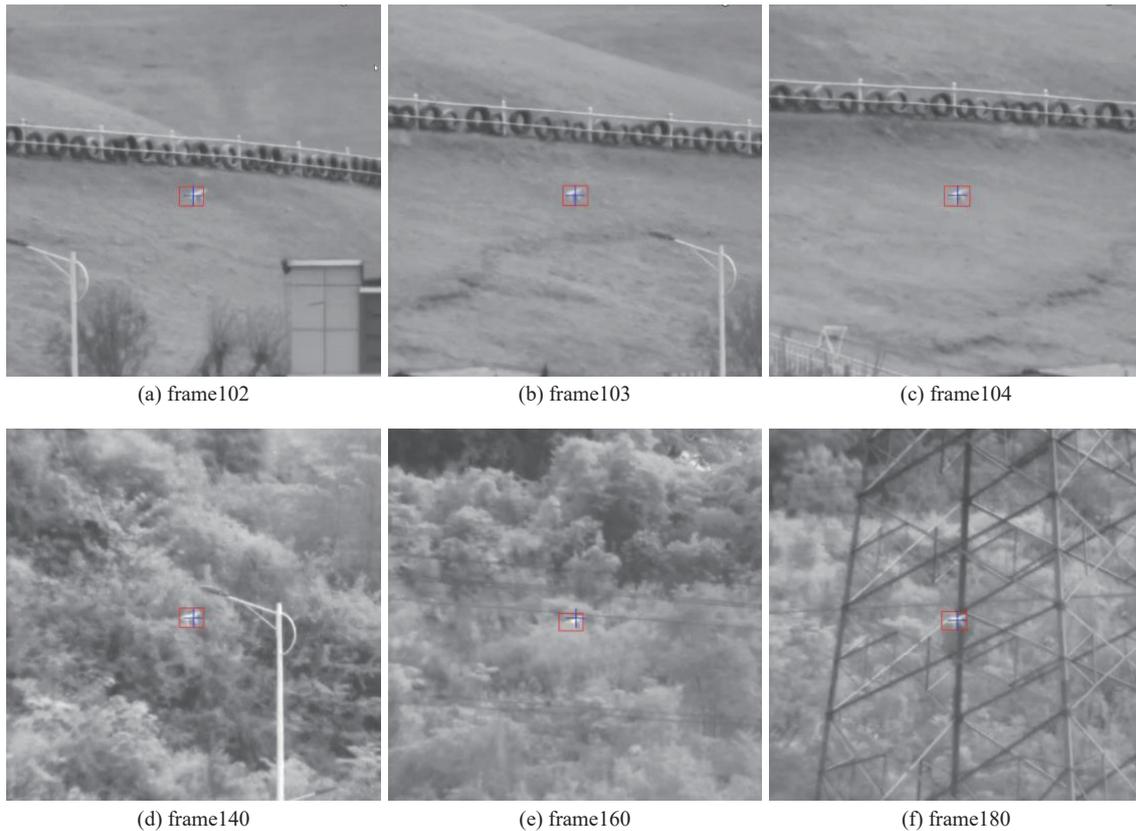


Fig. 11 Visualization of algorithm tracking effect after optimization

图 11 优化后算法跟踪效果可视化

虽然本文针对复杂背景下的低空无人机检测与跟踪提出了一个具有实际意义的解决方案并且通过实验验证了可行性,但同时通过本文的研究也意识到还有一些问题有价值进一步研究:(1)本文只开展了对低空无人机单目标的研究,针对多个无人机目标的情况还有待研究;(2)本文的无人机背景涉及了城市、森林等情况,但是对于一些对比度极低、湍流严重且成像模糊的极端情况仍然无法应对,可以引入主动照明、主动光学(AO)等技术做进一步研究。

#### 参考文献:

- [1] Klare J, Biallawons O, Cerutti-Maori D. UAV detection with MIMO radar[C]//Proceedings of the 18th International Radar Symposium (IRS). 2017: 1-8.
- [2] Xiao Yue, Zhang Xuejun. Micro-UAV detection and identification based on radio frequency signature[C]//Proceedings of the 6th International Conference on Systems and Informatics (ICSAI). 2019: 1056-1062.
- [3] Kim J, Park C, Ahn J, et al. Real-time UAV sound detection and analysis system[C]//Proceedings of 2017 IEEE Sensors Applications Symposium (SAS). 2017: 1-5.
- [4] Jocher G. v5.0 - YOLOv5-P6 1280 models, AWS, Supervise. ly and YouTube integrations[CP/OL]. [2021-03-12]. <https://gitcode.net/mirrors/ultralytics/yolov5/-/releases/v5.0>.
- [5] Lukežić A, Vojir T, Zajc L C, et al. Discriminative correlation filter with channel and spatial reliability[C]//Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. 2017: 6309-6318.
- [6] Hu Jie, Shen Li, Sun Gang. Squeeze-and-excitation networks[C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 7132-7141.
- [7] Han Kai, Wang Yunhe, Tian Qi, et al. GhostNet: more features from cheap operations[C]//Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 1580-1589.
- [8] van de Weijer J, Schmid C, Verbeek J, et al. Learning color names for real-world applications[J]. *IEEE Transactions on Image Processing*, 2009, 18(7): 1512-1523.
- [9] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2005: 886-893.
- [10] Lowe D G. Object recognition from local scale-invariant features[C]//Proceedings of the Seventh IEEE International Conference on Computer Vision. 1999: 1150-1157.

- [11] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features[C]//Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2001.
- [12] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real-time object detection[C]//Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. 2016: 779-788.
- [13] Liu Wei, Anguelov D, Erhan D, et al. SSD: single shot MultiBox detector[C]//Proceedings of the 14th European Conference on Computer Vision. 2016: 21-37.
- [14] Santad T, Silapasupphakornwong P, Choensawat W, et al. Application of YOLO deep learning model for real time abandoned baggage detection[C]//Proceedings of 2018 IEEE 7th Global Conference on Consumer Electronics. 2018: 157-158.
- [15] Dos Reis D H, Welfer D, De Souza Leite Cuadros M A, et al. Mobile robot navigation using an object recognition software with RGBD images and the YOLO algorithm[J]. *Applied Artificial Intelligence*, 2019, 33(14): 1290-1305.
- [16] Mei Xue, Ling Haibin. Robust visual tracking and vehicle classification via sparse representation[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, 33(11): 2259-2272.
- [17] Wang Naiyan, Wang Jingdong, Yeung D Y. Online robust non-negative dictionary learning for visual tracking[C]//Proceedings of 2013 IEEE International Conference on Computer Vision. 2013: 657-664.
- [18] Held D, Thrun S, Savarese S. Learning to track at 100 FPS with deep regression networks[C]//Proceedings of the 14th European Conference on Computer Vision. 2016: 749-765.
- [19] 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.
- [20] 葛宝义, 左宪章, 胡永江. 视觉目标跟踪方法研究综述[J]. *中国图象图形学报*, 2018, 23(8): 1091-1107. (Ge Baoyi, Zuo Xianzhang, Hu Yongjiang. Review of visual object tracking technology[J]. *Journal of Image and Graphics*, 2018, 23(8): 1091-1107)
- [21] Tao Ran, Gavves E, Smeulders A W M. Siamese instance search for tracking[C]//Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1420-1429.
- [22] Song Yibing, Ma Chao, Wu Xiaohe, et al. VITAL: visual tracking via adversarial learning[C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 8990-8999.
- [23] MacQueen J B. Some methods for classification and analysis of multivariate observations[C]//Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability. 1967: 281-297.
- [24] Otsu N. A threshold selection method from gray-level histograms[J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 1979, 9(1): 62-66.