

基于 YOLOv4 与 KCF 的无人机飞行障碍 精准识别与实时跟踪研究

张继⁺¹

(1. 四川大学¹, 四川 成都 610064)

摘要: 低空飞行的无人机, 时常面临飞机、鸟类、行人、汽车、交通灯等障碍物的碰撞威胁。如何准确地检测出这些飞行障碍, 并在规划飞行路径 (Route Plan) 时躲避它们, 已经成为无人机研究领域中的一个重要课题。近年来, 基于深度学习 (Deep Learning) 的图像识别技术 (Image Recognition) 在生物特征识别、物体分类等领域展现了强大的分类检测效能, 这也为无人机飞行障碍检测提供了可能的解决思路及其实现途径。本文专注于无人机飞行障碍的检测与跟踪 (Detect and Track) 环节, 基于 YOLOv4 目标检测与 KCF 目标跟踪, 对低空常见的典型障碍物进行精准识别与实时跟踪 (Precise Recognition and Real-time Tracking); 探索了高准确率的 YOLOv4 目标检测 (Object Detection) 与高运行效率的 KCF 目标跟踪 (Object Tracking) 之间的最佳平衡点, 从而兼顾了无人机障碍物检测的准确率与实时性, 具有较强的理论探索意义和工程应用价值。

关键词: 飞行障碍、深度学习、图像识别、目标检测、目标跟踪

Precise Recognition and Real-time Tracking of UAV obstacle based on YOLOv4 and KCF

Zhang Ji⁺¹

(1. Sichuan University, Sichuan Province Chengdu 610064)

Abstract: Flying UAV (Unmanned Aerial Vehicle) often threatens by plane, birds, pedestrian, car, traffic light and so on. How to detect those obstacles, and avoid them while planning flight route, has become an important research topic of UAV. Image Recognition based on Deep-Learning has shown its ability to classify in domains like Biometrics Identification and Object Classification. Apply Image Recognition technology on UAV obstacle detection is also possible and achievable. This paper focuses on UAV obstacle detection and obstacle tracking. Based on YOLOv4 object detection and KCF object tracking, we carried out precise obstacle detection and real-time tracking on typical obstacles. We explored the optimal balance between high-accurate YOLOv4 object detection and high-efficient KCF object tracking, thus ensuring the accuracy and real-time performance of UAV obstacle detection simultaneously.

Keywords: Flying Obstacle, Deep Learning, Image Recognition, Object Detection, Object Tracking

0. 引言

随着技术的发展,无人机在军民两大领域逐渐得到广泛运用。近几年的阅兵式上,军用无人机愈发崭露头角,翼龙、暗剑、彩虹等无人机在情报侦察、军事打击、信息对抗、通信中继和后勤保障等领域发挥了重大作用;民用无人机最具代表性的当属大疆出品的系列无人机,其在农业、基建、电力巡检、摄影和公共安全等领域发挥了重大作用^[1]。

在城市中低空飞行的无人机,常常面临着与起降的民航飞机、行人、汽车、交通灯、飞鸟等发生相互碰撞的严峻挑战,这不仅会对无人机自身造成损伤,更有可能影响造成危机飞行安全、伤及重要建筑设施、碰撞车辆、伤害行人等严重事故。所以,无人机对低空飞行障碍物的实时检测跟踪就显得十分紧要 and 迫切。目前,无人机飞行障碍常用的检测手段有超声波回声定位(Ultrasonic Echo Location)、激光扫描定位(Laser Scan Location)、雷达搜索定位(Radar Search Location)和视觉图像检测(Visual Image Detection)^{[2][3]},但是障碍物的多样性、复杂性决定了它们中的任何一种方法都各有利弊,难以单独完成障碍物检测跟踪的任务,只有针对具体实用场景进行有机组合,才能达到较好的应用效果。为了更好地适应低空飞行无人机所处的复杂应用场景,本文拟从视觉图像检测(Visual Image Detection)的角度,将高准确率的YOLO目标检测与高运行效率的KCF目标跟踪相结合,对无人机飞行障碍的精准识别与实时跟踪(Precise Recognition and Real-time Tracking)进行深入研究。

1. 目标检测与 YOLOv4

目标检测的任务是找出视频图像中所有关注的目标障碍(物体),确定它们的位置和大小,这是现代机器视觉领域的核心问题之一。由于各类物体有不同的外观、形状、姿态,加上成像时光照、遮挡等因素的干扰,目标检测一直是机器视觉领域最具有挑战性的问题。随着计算机视觉技术的发展与进步,利用无人机搭载视频传感器进行目标检测和跟踪已广泛应用在地形勘测、交通监控、抗灾救援以及军事侦查等领域。因而,对无人机所获取的视频目标进行障碍精准识别与实时跟踪是其中重要环节。

目标检测算法主要有两大类:one-stage目标检测和two-stage目标检测。one-stage目标检测以SSD和YOLO为代表,它们使用一个卷积神经网络直接预测出目标的类别与位置;two-stage目标检测以R-CNN为代表,分为两步:首先产生大量的Proposal Bounding Boxes;然后通过卷积神经网络筛选出包含目标概率最高的Bounding Box,并判断出目标类别。

YOLO(You Only Look Once)代表了一系列优秀的目标检测模型,从YOLOv1^[4]、YOLOv2^[5]、YOLOv3^[6],如今进化到YOLOv4^[7]、YOLOv5,它们都是基于深度神经网络的目标检测模型:以整张图像作为输入,通过卷积神经网络的非线性处理,最终输出图像中目标的类别和其在图像中的位置(矩形包围框Bounding Box)。YOLOv4在检测精度、运行效率、训练难易程度等方面均有较优秀的表现。相比YOLOv3,YOLOv4在检测精度和运行效率上分别提升了10%和12%,并且与YOLOv5性能相当。YOLOv4支持单GPU训练,这让我们能够比较容易的添加自己的训练数据,从而扩展YOLOv4的检测能力。YOLOv4的模型结构(如图1)由以下几部分构成: CSPDarknet53 作为 backbone骨干网络、SPP作为Neck的附加模块、PANet作为Neck颈部网络的特征融合模块,以及使用YOLOv3作为head头部网络。

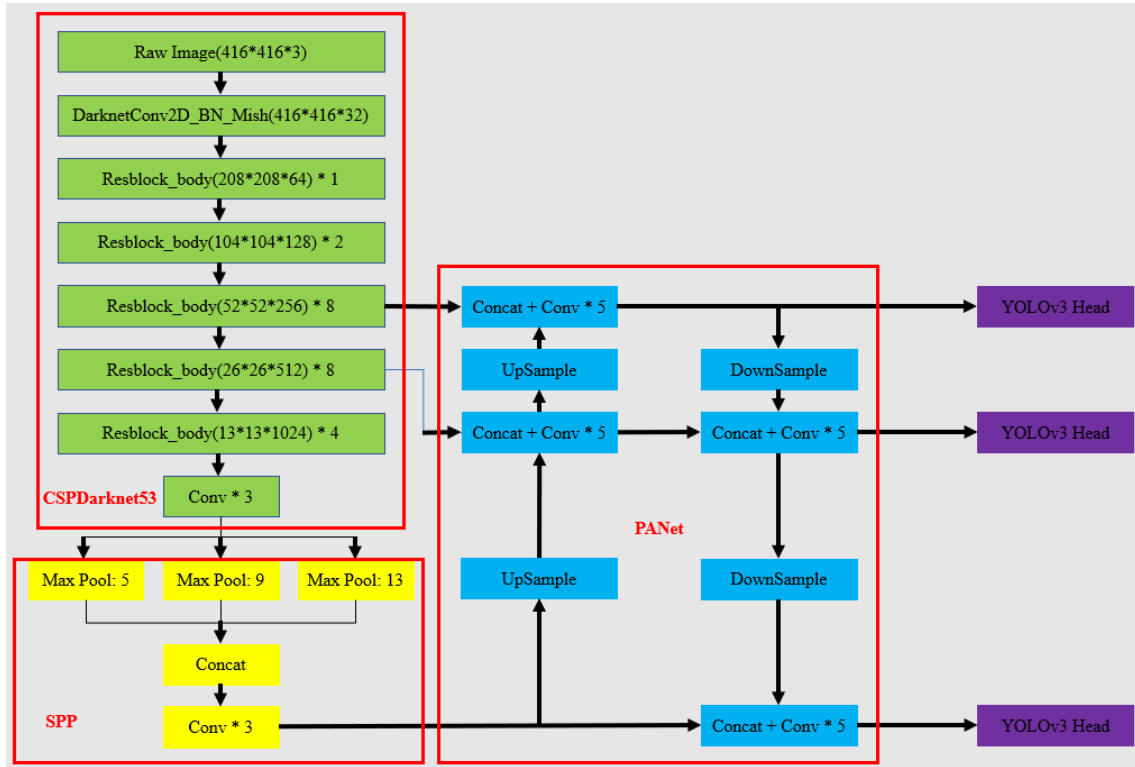


图1 YOLOv4网络结构

Fig.1 YOLOv4 Structure

常用的YOLOv4模型如表1所示，它们有着不同的准确度MAP（mean average precision）和检测速度FPS（Frames Per Second）。大体而言，准确度越高，则检测速度相对越慢。综合考虑准确度和检测速度的影响，我们选择YOLOv4-416作为本文的目标检测模型，它使用COCO^[8]（Common Objects in Context）作为训练数据集，具备飞机、行人、汽车、交通灯、鸟类等80个类别障碍物的检测识别能力（如图2）。

表1 常用YOLOv4模型

Table1. Common YOLOv4 Model

序号	模型名称	训练数据	MAP	FLOPS	FPS（RTX 2070）
1	YOLOv4-320	COCO	60%	35.5Bn	63
2	YOLOv4-416	COCO	62.8%	60.1 Bn	55
3	YOLOv4-512	COCO	64.9%	91.1Bn	45
4	YOLOv4-608	COCO	65.7%	128.5Bn	34



图 2 YOLOv4 飞行障碍检测效果

Fig.2 YOLOv4 Detection of Flying-Obstacle

2. 目标跟踪与 KCF

目标跟踪的任务是找出被关注目标在每帧视频图像中的位置。它与目标检测有着相似的功能：寻找目标位置，但“跟踪”二字对它的运行速度提出了较高的要求。与目标检测相比，目标跟踪运行速度大大加快、目标位置定位精度存在一定偏差，可以理解为牺牲少量定位精度以换取成倍的运行速度。

目标跟踪算法主要分为判别式（discriminative model）和生成式（generative model）。生成式方法采用特征模型描述目标的外观特征，再最小化跟踪目标与候选目标之间的误差来确认目标，例如 LK 光流法；判别式方法将目标跟踪视为二元分类问题，通过训练关于跟踪目标的分类器来从候选目标中找出正确的目标，例如 CSK、KCF 算法。此外，根据跟踪目标的数量还可以将目标跟踪算法分为单目标跟踪与多目标跟踪。相比单目标跟踪而言，多目标跟踪问题更加复杂和困难。多目标跟踪需要考虑视频图像帧中多个独立目标的不同类别、位置、大小、外观、运动方向、动态光照、相互遮挡等因素，这些因素是目标跟踪领域研究的难点。

KCF^[9]即 Kernel Correlation Filter（核相关滤波），它是一种判别式（discriminative model）的目标跟踪方法。通过物体在当前帧的位置，迅速地预测物体在下一帧的位置，其运行速度可达到 150fps。KCF 的运行速度得益于循环矩阵和相关滤波这两个特点：

2.1. 循环矩阵

KCF 算法将图像沿着上下、左右（如图 3）的方向进行平移，以产生更多的样本用于训练^[10]。同时这种平移可以通过循环矩阵来表示，循环矩阵可以对角化，计算时仅需关注对角线上的非零元素，因此能够大幅加快矩阵与矩阵的计算速度。



向左平移

基准图片

向右平移

图 3 左右平移

Fig.3 Shift Left/Right

2.2. 相关滤波

相关是衡量两个信号相似值的度量，相关值越高，表示这两个信号越相似^[1]。KCF 的输入往往是通过其他检测手段得到的物体在某一帧图像上的准确位置（例如 YOLOv4 检测得到的 bbox），该准确位置记为 *patch_0*。在后续的图像帧上，有 *patch_1~patch_n* 共计 *n* 个 patch，那么 KCF 预测的位置即为与 *patch_0* 相关值最高的 patch。

$$patch_predict = \max (correlation(patch_0, patch_k)), \quad (1 \leq k \leq n)$$

在计算 *patch_0* 与 *patch_k* ($1 \leq k \leq n$) 的相关值时，转换到了傅里叶域进行。因为两个 patch 的相关卷积相当于傅里叶域中的元素乘积（时域卷积 = 频域点积），而乘积计算大大快于卷积运算。

3. 研究场景

3.1. 研究场景

大疆无人机（御 Mavic Air 等型号）具有智能跟随功能，支持对行进车辆的识别跟随能力，无人机可以在顶部俯拍、车尾/侧面跟拍行进中的车辆。在顶部俯拍时，无人机的飞行高度保证了不会与被跟随车辆相撞。在车尾/侧面跟拍时，由于速度不同步，很有可能出现无人机加速接近汽车的情况。这种情况下，被跟随车辆就属于无人机飞行路线中有碰撞风险的障碍物，无人机需要识别、跟踪并规避车辆。本文以大疆无人机使用智能随车拍功能时，追随汽车录制的一段视频作为数据研究对象。该视频时长 54s，1080p（1920*1080），帧率 30fps，共计 $54 \times 30 = 1620$ 帧。

3.2. KCF 跟踪丢失

我们以 YOLOv4 目标检测的矩形包围框 bbox（Bounding Box）作为 KCF 跟踪算法的初始输入，并在随后所有视频帧使用 KCF 目标跟踪来找出物体在每帧图像中的位置。初始输入时无人机在汽车的正后方，YOLOv4 准确地找到了汽车的位置（如图 4，红色矩形框完整

包围了汽车)。随着时间的推移,无人机飞到了汽车的右侧面,视角相对初始位置发生较大变化,我们发现 KCF 逐渐无法跟踪到正确的目标(如图 5,红色矩形框仅能包围汽车的小部分车尾)。



图 4 准确检测

Fig.4 Accurate Detection



图 5 跟踪丢失

Fig.5 Loss of Object

跟踪丢失的原因在于视角相对初始位置发生较大变化,KCF 算法“记忆中的车尾”在侧飞时并未出现在图像中。为了矫正这种跟踪错误,每隔一定的帧数,使用 YOLOv4 目标检测模型重新获得准确的车辆位置 bbox,并更新 KCF 跟踪算法的输入,使 KCF 算法“记忆犹新”。

4. 最佳检测间隔

YOLOv4 目标检测有更高的准确率,但是检测耗时较长;KCF 目标跟踪有更快的运行速度,但是容易出现跟踪目标丢失的情况。因此,考虑将 YOLOv4 目标检测与 KCF 目标跟踪结合使用,从而兼顾检测准确率与实时性。

本节以 YOLOv4 检测间隔为 0(所有图像帧均使用 YOLOv4 目标检测)时得到的目标位置 bbox(Bounding Box)和运行时间(Elapse Time)作为基线,将 YOLOv4 检测间隔为 1~100(中间图像帧使用 KCF 目标跟踪)作为优化对照,研究不同 YOLOv4 检测间隔时的准确率和运行效率情况。基准与优化的检测间隔如图 6 所示。

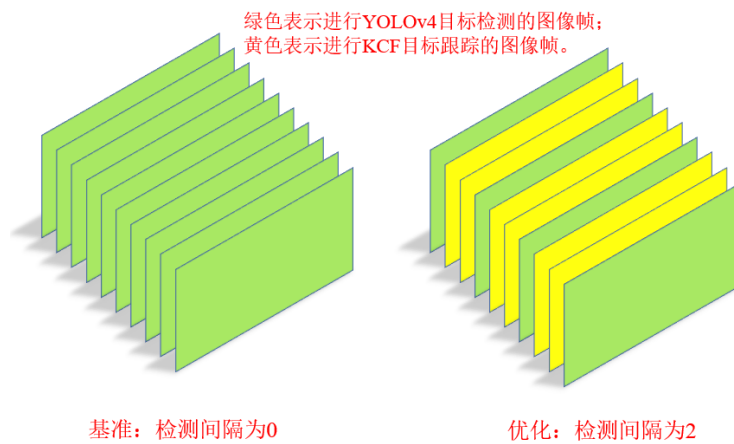


图 6 基准与优化

Fig.6 Benchmark and Optimization

4.1. 对照基准 benchmark

所有 1620 帧图像均使用 YOLOv4 目标检测，然后输出 bbox，并输出该段视频总共的检测时间，作为优化对照的基准。部分结果如表 2 所示：

表 2 基线检测结果

Table2. Benchmark Detection

<i>frameNo</i>	<i>x1_b</i>	<i>y1_b</i>	<i>width_b</i>	<i>height_b</i>
7	821	630	235	197
8	821	633	235	194
...
1608	913	682	87	70
<i>elapsedTime_b</i>	76.831s			

其中，(*x1_b*, *y1_b*)表示*benchmark bbox*的左上角点，*width_b*和*height_b*分别表示*benchmark bbox*的宽和高。*elapsedTime_b*为采用基准方法对整段视频进行跟踪检测的总耗时，76.831s，可计算得到基准方法的帧率为 $1620 / 76.831 = 21.08\text{fps}$ 。

由于实验条件限制，使用了台式机对无人机拍摄的视频进行处理，测试机配置如表 3 所示。

表 3 测试机配置

Table3. Configuration of Test Machine

系统	Windows 10 企业版 64 位操作系统
CPU	CPU: Intel(R) Core(TM) i5-4590 CPU @ 3.30GHz 8GB RAM
GPU	GPU: NVIDIA GeForce GTX 1060 6GB

4.2. 优化对照

对汽车视频每间隔 *interval* 帧抽取一帧进行 YOLOv4 目标检测，其余帧使用 KCF 输出 bbox 位置坐标和运行时间。*interval* 为 1 时的检测跟踪结果如表 4 所示：

表 4 interval 为 1 时的检测跟踪结果

Table4. Detection and Tracking (interval 1)

interval = 1				
<i>frameNo</i>	<i>x1_r</i>	<i>y1_r</i>	<i>width_r</i>	<i>height_r</i>
7	821	630	235	197
8	821	630	235	194
...
1608	913	682	87	72
<i>elapseTime_r</i>	46.145s			

其中, (*x1_r*, *y1_r*)表示*real bbox*的左上角点, *width_r*和*height_r*分别表示*real bbox*的宽和高。*elapseTime_r* 为采用间隔为 1 的优化方法对整段视频进行跟踪检测的总耗时, 46.145s, 可计算得到检测间隔为 1 时的帧率为 $1620 / 46.145 = 35.10\text{fps}$ 。间隔为 1 时相对基准方法速度提升 $\text{speed}_x = 76.831 / 46.145 = 1.66$ 倍。

4.2.1. 位置偏差

位置偏差采用交并比 IOU (Intersection over Union) 进行评价, 顾名思义, IOU 为两个集合的交集与并集之比 (如图 7), 即:

$$IOU = (A \cap B) / (A \cup B)$$

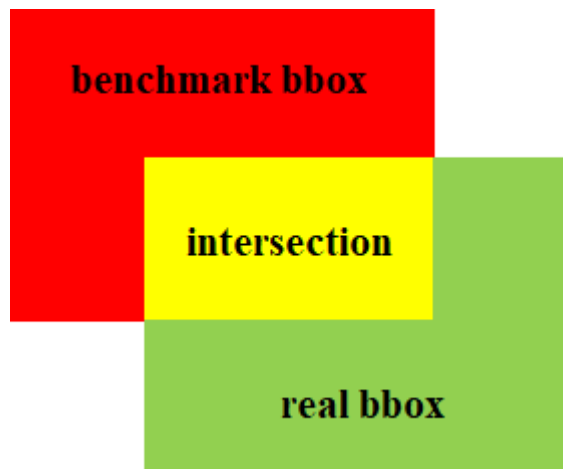


图 7 单帧图像的 IOU 计算

Fig.7 IOU Calculation of Single Image

IOU 在相同图像帧之间进行计算，使用基准方法得到 *benchmark bbox*，使用优化方法得到 *real bbox*，则该帧图像的 IOU 为：

$$IOU = \text{area}_{\text{intersection}} / (\text{area}_{\text{benchmark bbox}} + \text{area}_{\text{real bbox}} - \text{area}_{\text{intersection}})$$

计算所有 1620 帧图像的 IOU，最后得到该段视频的平均 IOU。

4.2.2. 速度提升

对不同的检测间隔，统计该视频所有 1620 帧图像的总共检测跟踪用时 *elapsedTime_r*。然后速度提升倍数为基准总用时与优化方法总用时之比，即：

$$\text{speed}_x = \text{elapsedTime}_b / \text{elapsedTime}_r$$

4.3. 优化结果分析

在实验中，我们将 interval 值依次设置为 1~100，并统计不同检测间隔的位置偏差和速度提升效果，数据整理后使用 matlab 绘制如图 8 所示：

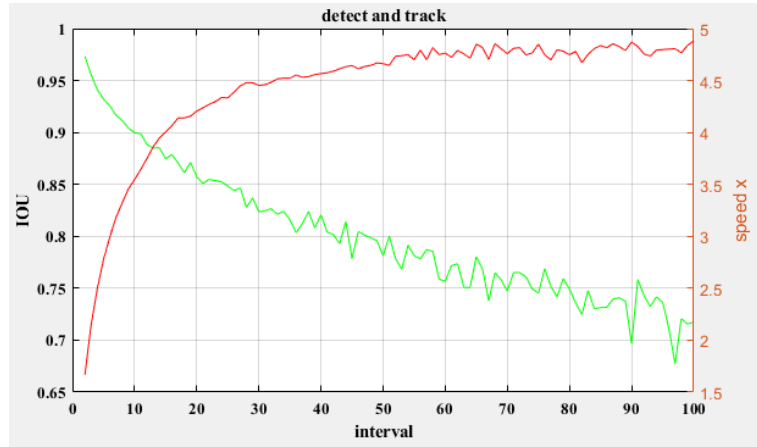


图 8 不同 Interval 的位置偏差和速度提升

Fig.8 Location Error and Speed Promotion

可以看到，检测间隔越大，IOU 越小（绿色曲线），表明位置偏差越大；速度提升效果（红色曲线）在检测间隔小于 30 时非常明显，随后逐渐趋于平缓。综合考虑检测位置准确性和运行效率，选择 20 作为最优检测间隔，该间隔下的平均 IOU、速度提升效果、总用时、帧率如表 5 所示：

表 5 最优检测间隔的各项参数

Table5. Parameters When Interval Equals 20

interval	20
avg_IoU	85.73%
speed_x	4.20577
elapseTime	18.268s
fps	88.68

观察检测跟踪结果，跟踪丢失的情况得到了很好的矫正。无人机飞行姿态任意变化，YOLOv4 与 KCF 均能够准确地找到汽车的位置（如图 10，红色矩形框完整包围了汽车）。



图 9 跟踪丢失

Fig.9 Loss of Object



图 10 跟踪矫正

Fig.10 Track of Object

5. 检测能力扩展

本文使用原生YOLOv4进行飞行障碍检测，原生YOLOv4模型基于COCO数据集训练得到，拥有飞机、行人、汽车、交通灯、鸟类等80个类别障碍物的检测识别能力。YOLOv4模型支持单GPU训练，使用单张1080 Ti 或2080 Ti GPU即可训练得到检测准确率高、实时性好的目标检测模型。这让我们能够比较容易的添加自己的训练数据，从而针对专用场景扩展YOLOv4的检测能力。在驱鸟无人机上，将机场附近经常出没的鸟类图片加入YOLOv4模型训练集，使其获得/提升特定鸟类的检测能力；在农业植保无人机上，将高大树木的图片加入YOLOv4模型训练集，使其获得识别树木并避开的能力^{[12][13]}；在航拍无人机上，将高楼大厦顶部的图片加入YOLOv4模型训练集，使其能够检测出屋顶并避免碰撞。

6. 研究结果分析

综上所述，我们以无人机低空作业场景中常见的典型障碍物作为研究对象，使用YOLOv4 目标检测（Object Detection）模型进行飞行障碍的精准识别（Precise Recognition），并结合 KCF 目标跟踪（Object Tracking）算法提升实时性，同时达到精准识别与实时跟踪

(Precise Recognition and Real-time Tracking) 兼顾的双重目的。我们测试了不同检测间隔 (interval) 对准确率与运行速率的影响, 找出了精准检测与快速跟踪之间的最优平衡点。

本文的研究成果主要体现在:

- 1、采用 YOLO 家族新锐的 YOLOv4 目标检测模型进行飞行障碍物的精准识别 (Precise Recognition), 保证了障碍物检测的准确性;
- 2、使用 KCF 目标跟踪算法提升运行实时性, 并分析出 KCF 目标跟踪丢失的原因及解决方案;
- 3、我们找出了最佳检测间隔: 在 1080p(1920*1080) 视频中, 每隔 20 帧使用一次 YOLOv4 目标检测, 其余帧使用 KCF 目标跟踪。能够在保证 85% 准确率的同时, 提升 4.2 倍的运行速度, 使运行帧率达到 88.68fps;
- 4、检测能力扩展研究结果: 将自定义类别数据 (如房屋、树木) 加入 YOLOv4 训练数据集, 使用单张 GPU (1080 Ti 或 2080 Ti) 即可重新训练 YOLOv4 目标检测模型, 可使无人机获得其他类别障碍物的检测避让能力。

7. 结束语

本文采用当前目标检测领域中准确度和实时性均有较好表现的 YOLOv4 模型, 搭配高效的 KCF 目标跟踪算法, 探索了无人机飞行障碍的精准识别与实时跟踪。我们发现, 随着时间推移、障碍物的移动、无人机飞行姿态发生变化等, KCF 目标跟踪算法得到的矩形包围框会逐渐偏离原有障碍物, 导致障碍物跟踪丢失。为此, 我们采用 YOLOv4 模型重新检测飞行障碍物的具体位置, 并更新 KCF 算法的输入, 能够帮助 KCF 继续准确地跟踪上目标障碍物。我们将 YOLOv4 检测间隔为 0 (所有 1620 帧图像均使用 YOLOv4 目标检测) 作为基线, 将 YOLOv4 检测间隔为 1~100 (中间图像帧使用 KCF 目标跟踪) 作为优化对照, 以找出飞行障碍物检测跟踪准确率与运行效率的最佳平衡点。经过实验对比发现, 检测间隔 interval 为 20 时, 相对于基线拥有 85.73% 的检测准确率, 运行效率提升 4.20577 倍, 检测帧率达到 88.68fps。这表明, 基于 YOLOv4 与 KCF 的无人机飞行障碍物精准识别与实时跟踪研究具有较强的理论探索意义和工程应用价值。

8. 参考文献

- [1] 孟凡琨. 无人机目标跟踪与避障研究[D]. 长安大学, 2018.
- [2] Xilin Yang, Luis Mejias Alvarez and Troy Bruggemann. A 3D Collision Avoidance Strategy for UAVs in a Non-cooperative Environment. *Journal of Intelligent and Robotic Systems*. 2012.
- [3] Roberto Sabatini, Alessandro Gardi and Subramanian Ramasamy. A Laser Obstacle Warning and Avoidance System for Unmanned Aircraft Sense-and-Avoid. *Applied Mechanics and Materials*, vol. 629, pp. 355-360, 2014.
- [4] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.
- [5] Joseph Redmon and Ali Farhadi. YOLO9000: better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7263–7271, 2017.

- [6] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.
- [7] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv preprint arXiv: 2004.10934, 2020.
- [8] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In Proceedings of the European Conference on Computer Vision (ECCV), pages 740–755, 2014.
- [9] João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-Speed Tracking with Kernelized Correlation Filters. arXiv:1404.7584v3, 2014.
- [10] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, “Multiple kernels for object detection,” in ICCV, 2009.
- [11] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, “Visual object tracking using adaptive correlation filters,” in CVPR, 2010, pp. 2544–255.
- [12] 明宇. 基于视觉的植保无人机避障研究[D].沈阳理工大学,2019.
- [13] 杨娟娟. 基于深度学习的农用无人机避障系统研究[D].甘肃农业大学,2019.