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*，时宏伟*1*

(1.四川大学1，四川 成都 610064)

摘要：高低空飞行的无人机，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，并在规划飞行路径时躲避它们，已经成为无人机研究方向的一个重要课题。本文专注于无人机飞行障碍的检测与跟踪环节，基于YOLOv4目标检测与KCF目标跟踪，对高低空常见的典型障碍物进行精准识别与实时跟踪。探索了高准确率YOLOv4目标检测与高运行效率KCF目标跟踪之间的最佳平衡点，从而同时保证了无人机障碍物检测的准确率与实时性。

关键词：飞行障碍、目标检测、目标跟踪

中图分类号：TP3 文献标识码：A

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

Zhang Ji*+1*，Shi Hong-wei*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. 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, Object Detection, Object Tracking

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能发挥重大作用；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能发挥重大作用。

在高空飞行的无人机，时常会面临飞鸟以及其他飞机的威胁；在城市中低空飞行的无人机，与行人、汽车、交通灯、飞鸟相碰撞的几率也较大。这不仅会对无人机自身造成损伤，更有可能给其他飞机、行人带来严重伤害。

图1 典型飞行障碍

# 主要贡献

本文以无人机高低空作业场景中常见的典型障碍物作为研究对象，使用YOLOv4目标检测模型进行飞行障碍的精准识别，并结合KCF目标跟踪算法提升实时性，同时达到精准识别与实时跟踪的双重目的。我们测试了不同检测间隔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。

优点：

1. 采用YOLO家族最新锐的YOLOv4目标检测模型进行飞行障碍物的精准识别；
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. 检测能力可扩展：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。

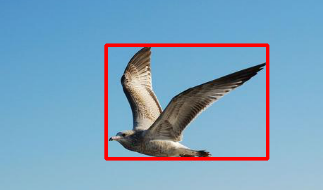
# YOLOv4目标检测

YOLO代表了一系列优秀的目标检测模型，从YOLOv1、YOLOv2、YOLOv3，如今进化到YOLOv4。YOLOv4是YOLO目标检测家族的新进成员，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是一种基于深度神经网络的目标检测模型，它的网络结构由几部分构成：CSPDarknet53 作为 backbone、SPP作为附加模块、PANet作为颈部网络，以及使用YOLOv3作为头部网络。常用的YOLOv4模型如表1所示，它们有着不同的准确度mAP和检测速度FPS。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。

表1 常用YOLOv4模型

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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 |

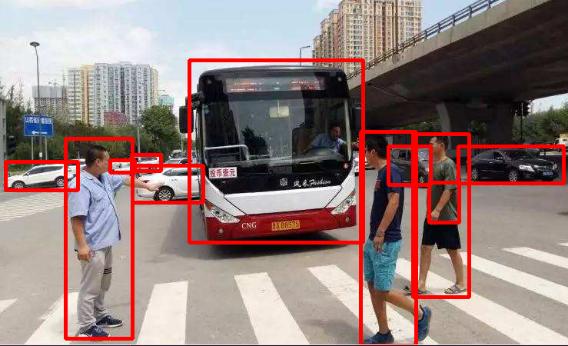
 

图5 YOLOv4飞行障碍检测效果

# KCF目标跟踪

KCF即Kernel Correlation Filter（核相关滤波），它通过物体在当前帧的位置，来预测物体在下一帧的位置，是一种快速的目标跟踪算法，运行速度可达到150fps。KCF主要有循环矩阵和相关滤波这两个特点：

## 循环矩阵

KCF算法将图像沿着上下、左右的方向进行平移，以产生更多的样本用于训练。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。

## 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。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))，（1≤k≤n）

在计算patch\_0与patch\_k（1≤k≤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。

# 研究场景与跟踪丢失

## 研究场景

大疆无人机具有智能随车拍的功能，无人机可以在顶部俯拍、车尾/侧面跟拍。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\*30=1620帧。

## KCF跟踪丢失

我们以YOLOv4目标检测的bbox作为KCF跟踪算法的初始输入，并在随后所有视频帧使用KCF替代YOLOv4检测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（见图3，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（见图3，红色矩形框仅能包围汽车的小部分车尾）。

图 3 图4

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”。

# 最佳检测间隔

本节以所有视频帧均进行YOLOv4检测得到的bbox和运行时间作为基准，将YOLOv4与KCF以不同间隔交替进行检测跟踪作为优化对照，研究不同YOLOv4检测间隔时的准确率和运行效率情况。

## 对照基准benchmark

对视频的每帧均进行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1所示：

表1 每帧均进行YOLOv4目标检测

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| framNo | x1\_b | y1\_b | width\_b | height\_b |
| 7 | 821 | 630 | 235 | 197 |
| 8 | 821 | 633 | 235 | 194 |
| ··· | ··· | ··· | ··· | ··· |
| 1608 | 913 | 682 | 87 | 70 |
| elapseTime\_b | 76.831s |  |  |  |

其中，elapseTime\_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=21.08fps。

(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。易知，benchmark bbox右上角点坐标为(x1\_b + width\_b, y1\_b)，benchmark bbox右下角点坐标为(x1\_b + width\_b, y1\_b + height\_b)，benchmark bbox左下角点坐标为(x1\_b, y1\_b + height\_b)。

由于实验条件限制，使用了实验室的台式机对无人机拍摄的视频进行处理，测试机配置如表2所示。

表 2 测试机配置

|  |  |
| --- | --- |
| 系统 | Windows 10 企业版 64位操作系统 |
| CPU | CPU: Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz 32GB |
| GPU | GPU: NVIDIA GeForce GTX 1060 6GB |

## 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3所示：

表3 interval为1时的检测跟踪结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| interval = 1 | | | | |
| framNo | x1\_r | y1\_r | width\_r | height\_r |
| 7 | 821 | 630 | 235 | 197 |
| 8 | 821 | 630 | 235 | 194 |
| ··· | ··· | ··· | ··· | ··· |
| 1608 | 913 | 682 | 87 | 72 |
| elapseTime\_r | 46.145s |  |  |  |

其中，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=35.10fps。间隔为1时相对基准方法速度提升speed = 76.831/46.145 = 1.66倍。

(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。易知，real bbox右上角点坐标为(x1\_r + width\_r, y1\_r)，real bbox右下角点坐标为(x1\_r + width\_r, y1\_r + height\_r)，real bbox左下角点坐标为(x1\_r, y1\_r + height\_r)。

### 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，即：。

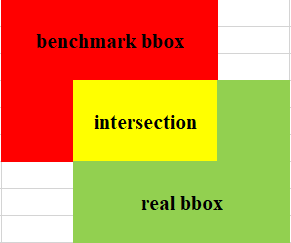


图4 单帧图像的IOU计算

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=area\_intersection / (area\_benchmark\_bbox + area\_real\_bbox – area\_intersection)。计算所有1620帧图像的IOU，最后得到该段视频的平均IOU。

### 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elapseTime\_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：speed\_x = elapseTime\_b / elapseTime\_r。

## 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，使用matlab绘制如下：

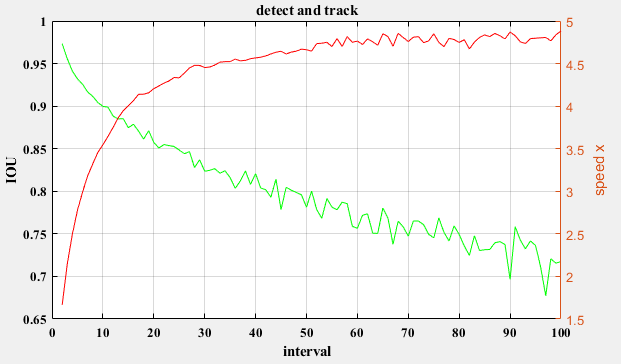


图4 不同Interval的位置偏差和速度提升

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2所示：

表2 最优检测间隔的各项参数

|  |  |
| --- | --- |
| Interval | 20 |
| Avg\_IOU | 85.73% |
| Speed\_x | 4.20577 |
| elapseTime | 18.268s |
| FPS | 88.68 |

# 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；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。

# 结束语

本文使用当前目标检测能力最强的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，KCF目标跟踪算法得到的矩形包围框逐渐偏离原有障碍物，导致障碍物跟踪丢失。使用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继续准确地跟上目标障碍物。我们以所有视频帧均进行YOLOv4目标检测为基线，设置了从1~100的不同检测间隔，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。

# 参考文献

[9]Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll´ar, 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]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.

[9]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.

[9]Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.

[9] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv preprint arXiv: 2004.10934, 2020.

[9] João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-Speed Tracking with Kernelized Correlation Filters. arXiv:1404.7584v3, 2014

[9] 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

[9] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, “Multiple kernels for object detection,” in ICCV, 2009.

[4]孟凡琨.无人机目标跟踪与避障研究[D].长安大学,2018.

[4]明宇. 基于视觉的植保无人机避障研究[D].沈阳理工大学,2019.

[4]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.

[9]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.

[9]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.