

基于 YOLO11s 的无人机 航拍小目标检测技术报告

——受限算力下的高精度微调与断点续训工程实践

项目类型

个人独立项目

作者

杨子嘉

邮箱

zegarspace@163.com

个人网站

zegarspace.com

日期

2026 年 9 月 24 日

版本

v1.0

本报告为个人独立研究成果，仅供技术交流与展示。未经作者授权，禁止盗用图表及数据。

目 录

摘 要	1
1 项目背景与挑战	2
1.1 无人机航拍检测的价值	2
1.2 核心难点	2
1.3 硬件与算力限制	2
2 数据准备与预处理	3
2.1 数据集介绍与实验预处理	3
2.2 数据格式转换	3
2.3 类别映射策略	4
3 实验过程与踩坑复盘	4
3.1 尝试一：YOLOE 微调	4
3.2 尝试二：YOLO11s 成功版本	5
4 实验结果与分析	6
4.1 整体指标	6
4.2 训练过程曲线	6
4.3 混淆矩阵分析	8
4.4 可视化结果展示	9
5 结论与展望	11
5.1 结论	11
5.2 未来工作	12
参考文献	13

摘要

本项目针对无人机航拍场景中目标极小、极度密集、遮挡严重的痛点，在消费级硬件（NVIDIA RTX 5060 8GB）上，开展了基于 YOLO11s 的微调实验。实验经历了初期 YOLOE 架构的失败尝试，随后通过更换标准 YOLO11s 架构、提升输入分辨率至 1280，并在遭遇硬件断电限制时采用基于最优检查点的热重启策略（Warm Restart），最终在 VisDrone 验证集上取得了 $mAP50 = 0.619$ （6 类合并口径）的成绩；作为参考，同任务公开报道的 YOLO11s 640 分辨率基线（10 类口径）约为 0.38~0.40，结果表明高分辨率微调与断点续训策略在受限算力下具有明显的收益。

关键词：无人机航拍；目标检测；YOLO11s；小目标检测；高分辨率微调；断点续训

1 项目背景与挑战

1.1 无人机航拍检测的价值

无人机航拍凭借其极其广阔的视野和灵活的机动性，已在交通疏导、应急救援、农业估产、快递配送等领域发挥着不可替代的作用，然而无人机航拍图像的目标通常具有目标小、像素低、肉眼难以识别等技术挑战。本实验旨在探究在受限算力下，实现无人机航拍图像高精度检测的工程可行性。

1.2 核心难点

本实验选用的 VisDrone 数据集极具挑战性，被业界公认为值得作为目标检测训练的优秀数据集，其数据设置的难点主要体现在以下三个方面：

- **目标极小**：在 2000×1500 像素的高空航拍图像中，许多行人、自行车等目标仅有几个像素，考验模型识别肉眼难以分辨目标的能力
- **目标极度密集**：单张图片中最多包含 902 个待检测目标，目标类型不同且各目标之间互相遮挡严重，考验模型在互相遮挡条件下快速分辨不同目标的能力
- **类别不平衡**：该数据集中“汽车”和“行人”占据绝对多数，而“公交车”、“卡车”等类别样本极少，符合实际社会情况的同时也导致了模型的训练变得困难

1.3 硬件与算力限制

本实验在受限的实验环境中使用个人消费级游戏笔记本（搭载 NVIDIA RTX 5060 8GB 显存）进行。相比较于实验室服务器，消费级显卡面临显存小、散热压力大等现实约束。此外，校园夜间断电导致每晚训练被迫终止。如何在算力硬件受限与供电受限等现实条件下，通过合理的工程策略完成高精度的微调，是本次实验的核心挑战。

2 数据准备与预处理

2.1 数据集介绍与实验预处理

VisDrone2019 数据集包含 6471 张训练图、548 张验证图和 1610 张测试图。原始标注为 .txt 文本文件，每行代表一个目标的 8 个属性，完整格式为：
<bbox_left> , <bbox_top> , <bbox_width> , <bbox_height> , <score> ,
<object_category> , <truncation> , <occlusion>。该格式无法直接用于 YOLO 训练，因此需要编写脚本对标注进行预处理。

2.2 数据格式转换

原始标注文件中各字段含义如下：

<bbox_left><bbox_top>：目标边界框左上角的 x 和 y 像素坐标。

<bbox_width><bbox_height>：目标边界框的像素高度和宽度。

<score>：置信度分数。0 表示该区域为“忽略区域”，即通常用于标注遮挡或模糊导致无法准确识别的目标；1 表示有效目标。

<object_category>：目标类型编号。1-10 分别表示行人、人、自行车、小汽车、面包车、卡车、三轮车、带篷三轮车、公交车、摩托车。

<truncation>：截断程度。0-2 分别表示无截断、部分截断、严重截断。

<occlusion>：遮挡程度。0-2 分别表示无遮挡、部分遮挡、严重遮挡。

转换方法：

本实验通过使用 python 脚本，将上述格式转换为 YOLO 标准格式：

<class_id><x_center><y_center><width><height>。重点完成将像素坐标除以图像的宽高，归一化到[0,1]区间。

需要特别说明的是：本实验保留了 score=0 的“忽略区域”样本参与训练，仅过滤未知类别与无效坐标。这些区域目标极度模糊或被严重遮挡，增加了模型学习难度，但也使模型在实际部署时面对模糊小目标具备更强的特征提取能力。
mAP50 = 0.619 即为保留这些困难样本条件下的保守评估结果。

2.3 类别映射策略

原始数据集包含 10 个类别，本实验将其合并为以下 6 类：

person, bicycle, car, motorcycle, bus, truck

- **原因说明：**因航拍在实际应用中通常相较于识别目标具体是什么类型（例如三轮车和带篷三轮车），更关注找到这个目标。并且在航拍视角下，van 与 car、tricycle 与 motorcycle 视觉高度重叠，强行区分会增加模型的无意义困惑。合并后能显著缓解长尾效应，故将部分类似分类合并转化为同一类别，最终化为六类索引，从而使模型更专注于学习“人”与“车”的核心特征。

3 实验过程与踩坑复盘

3.1 尝试一：YOLOE 微调

- 1) **配置：**YOLOE-11s, lr0=1e-3, AdamW, imgsz=1280。
- 2) **现象：**第 1 轮 mAP50 达到 0.273，随后断崖式下跌至 0.12，cls_loss 停滞在 6.1 无法收敛。
- 3) **归因：**我们推测其主要原因是微调策略不当引发的灾难性遗忘（Catastrophic Forgetting）。YOLOE 的文本-图像对齐机制对优化超参数较为敏感，直接以 lr0=1e-3 微调可能破坏了预训练权重的分布，导致 cls_loss 无法收敛。需要说明的是，单次实验不足以排除其他因素（如学习率调度、提示词设计），此处归因仅为基于现象的合理推测。
- 4) **结论与处置：**考虑到时间与算力成本，本实验未继续做低学习率消融，而是果断止损，更换为标准 YOLO 检测架构。



图 3-1: YOLOE 微调 mAP 崩溃曲线

如图 3-1 所示，YOLOE 微调实验在第 1 轮达到了 0.273 的 mAP50 短暂峰值，但随后出现断崖式下跌。至第 30 轮时，mAP50 已跌至 0.12，跌幅超 50%。同时，其分类损失始终在 6.1 以上高位徘徊，无法正常收敛。

结合 YOLOE 的架构特性分析，其文本-图像对齐模块对学习率极其敏感。本实验初始设置的 $lr=1e-3$ 步长过大，导致预训练权重遭到强行破坏，模型发生了严重的灾难性遗忘（Catastrophic Forgetting）。这一结果为后续果断放弃 YOLOE，转向标准的 YOLO11s 微调提供了关键性决策依据。

3.2 尝试二：YOLO11s 成功版本

- 1) **模型选择**：YOLO11s（9.4M 参数，轻量级现代架构）。
- 2) **显存优化**：经历 batch=4 爆显存（9.43G），调整至 batch=2（稳定 4-5G），结合 AMP 混合精度训练。
- 3) **断点续训策略（热重启）**：由于断电中断，加载 best.pt 重置优化器状态并引入学习率热重启。首次重启（Epoch 9→10）后，mAP50 从 0.5243 回撤至 0.5232；第二次重启（Epoch 52→53）后从 0.6094 回撤至 0.6015。两次回撤均在数轮内恢复并继续爬升，说明热重启策略在该任务上稳定可行。**训练参数**：imgsz=1280，batch=2，AdamW，mosaic=1.0，scale=0.9，close_mosaic=10。

4 实验结果与分析

4.1 整体指标

经过 82 个 epoch 的训练，模型在验证集（6 类合并口径）上达到以下指标：

1. $mAP50 = 0.619$;
2. $mAP50-95 = 0.386$;
3. 作为参考，公开报道的 YOLO11s 在 VisDrone2019 验证集、640 分辨率、10 类口径下的 $mAP50$ 约为 0.38~0.40。需要指出，两者类别数与评估协议不同，该对比仅用于说明高分辨率微调的收益趋势，不构成严格的同口径比较。

4.2 训练过程曲线

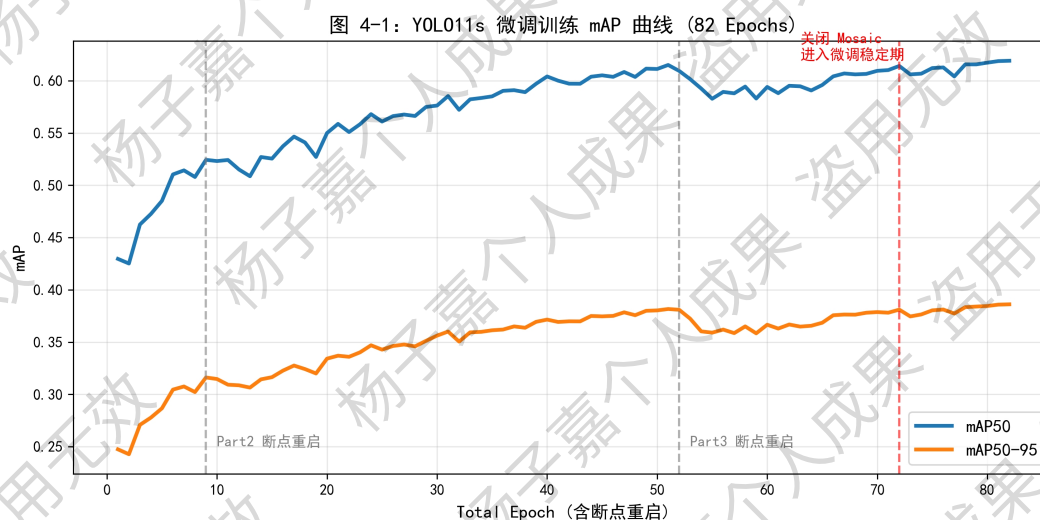


图 4-1: YOLO11s 微调训练 mAP 曲线

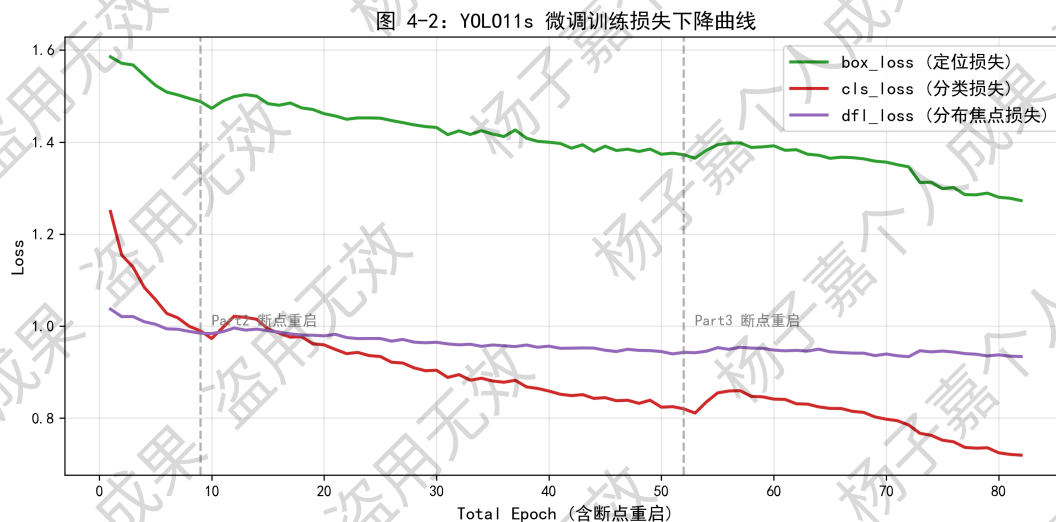


图 4-2：损失曲线

本次实验总共进行了 82 个 epoch 的训练。受外部供电限制，训练过程被断点续训为三个阶段（Epoch 1-9、Epoch 10-52、Epoch 53-82）。模型的 Loss 下降曲线与 mAP 曲线分别如图 4-1 和图 4-2 所示。

（1）mAP 变化趋势分析

从图 4-1 的 mAP 曲线可以看出，模型的收敛过程非常稳定。在训练初期（Epoch 1-20），mAP50 从 0.4296 快速跃升至 0.55 以上，展现出极快的学习速度。在经历两次断点重启（对应图中 $x=9$ 和 $x=52$ 处）时，mAP 曲线虽然出现了短暂的数据适应期，但均未发生崩溃，而是迅速恢复上升轨迹。至 Epoch 52 时，mAP50 已稳步达到 0.6094。

在训练的最后 10 个 epoch（Epoch 73-82），模型触发了 `close_mosaic=10` 策略，停止使用 Mosaic 畸变增强，转向高清原图进行微调。此时 mAP 曲线呈现极其平滑的收敛态势，最终在 Epoch 82 定格于 $\text{mAP50} = 0.6191$ 、 $\text{mAP50-95} = 0.3861$ 的峰值。这种平滑的末端曲线表明模型已完全拟合数据分布，未出现过拟合振荡。

（2）Loss 下降趋势分析

从图 4-2 的损失曲线来看，`box_loss`、`cls_loss` 和 `dfl_loss` 整体均保持平稳下降态势。

在断点重启处（ $x=9$ 和 $x=52$ ），由于优化器动量（AdamW）状态重置，三个损失函数均出现了符合预期的轻微跳动，但随后立即恢复下降。值得注意的是，在最后 10 轮关闭 Mosaic 后，`cls_loss` 从 0.7670 显著降至 0.7196，验证集的

val/box_loss 也降至 1.2386。整个训练过程中损失持续收敛，与 YOLOE 实验中的 cls_loss 停滞在 6.1 形成了鲜明对比，充分证明了 YOLO11s 架构搭配 AdamW 优化器在 VisDrone 数据集上的稳定性。

4.3 混淆矩阵分析

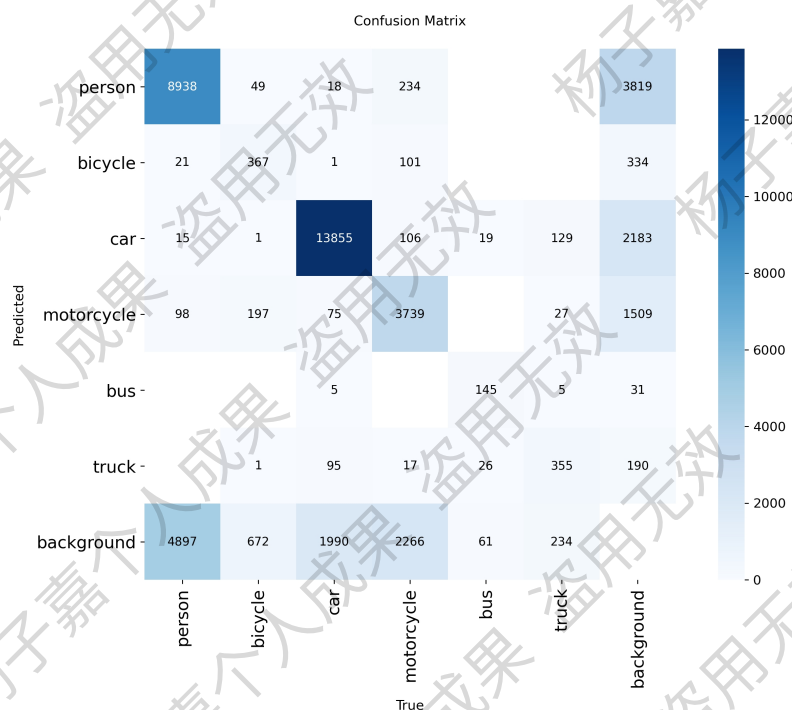


图 4-3：混淆矩阵

图 4-3 为置信度阈值 0.25 下的原始计数混淆矩阵，已做归一化。模型的混淆矩阵如图 4-3 所示。从图中可以看出，模型对主要类别（汽车和行人）具有极高的识别能力。其中，car 类别的对角线数值最高（13855），说明模型的预测结果与真实标签高度重合；person 类别也达到了 8938 的正确识别数。但在密集小目标场景下，模型仍存在一定的误检与漏检。具体表现为：

漏检（False Negative）：在真实为 person 的样本中，有 4897 个被预测为 background，漏检率相对较高。这主要是由于航拍图像中行人像素极少且遮挡严重，导致模型特征提取困难。以 person 类为例，漏检 4897 例，漏检率约为 $4897/(8938+4897) \approx 35\%$ ，是六个类别中最高的，与航拍行人类目标像素极少、遮挡严重的直观判断一致。

误检（False Positive）：在预测为 person 的结果中，有 3819 个实际为 background，说明模型容易将密集场景中类似于人的背景噪点误认为行人。

类间混淆：bicycle（真实）有 334 个被误判为 motorcycle，truck（真实）有 95 个被误判为 car。这主要是由于高空俯视视角下，这些类别之间的外观轮廓高度相似，且像素面积小，导致模型难以区分。

综合来看，尽管在极密集小目标下存在少量误检和漏检，但各类别的正确预测数（对角线）远高于错误预测数（非对角线），验证了模型在 VisDrone 数据集上的有效性。

4.4 可视化结果展示

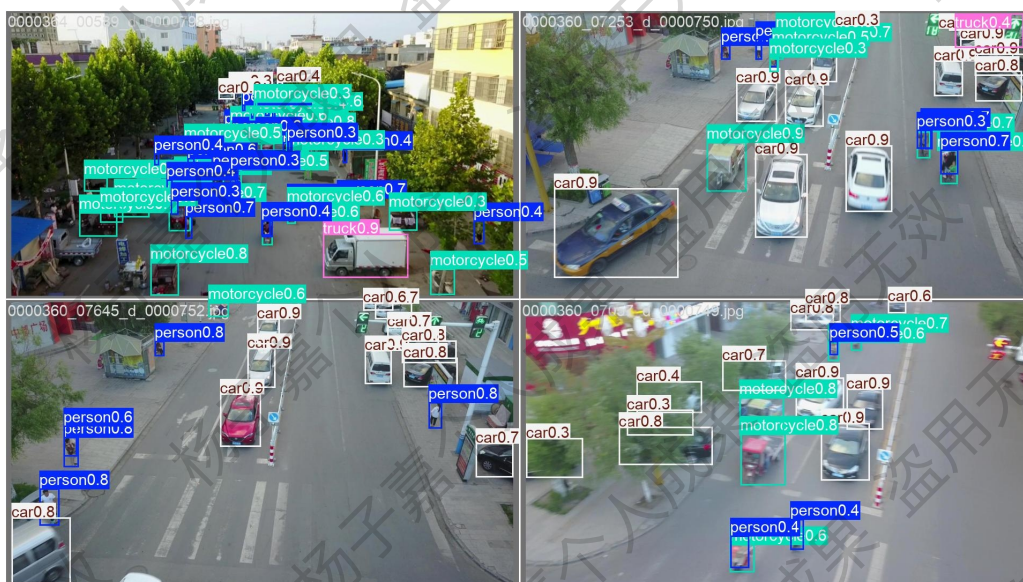


图 4-4-1：模型在航拍密集场景下的预测效果 1

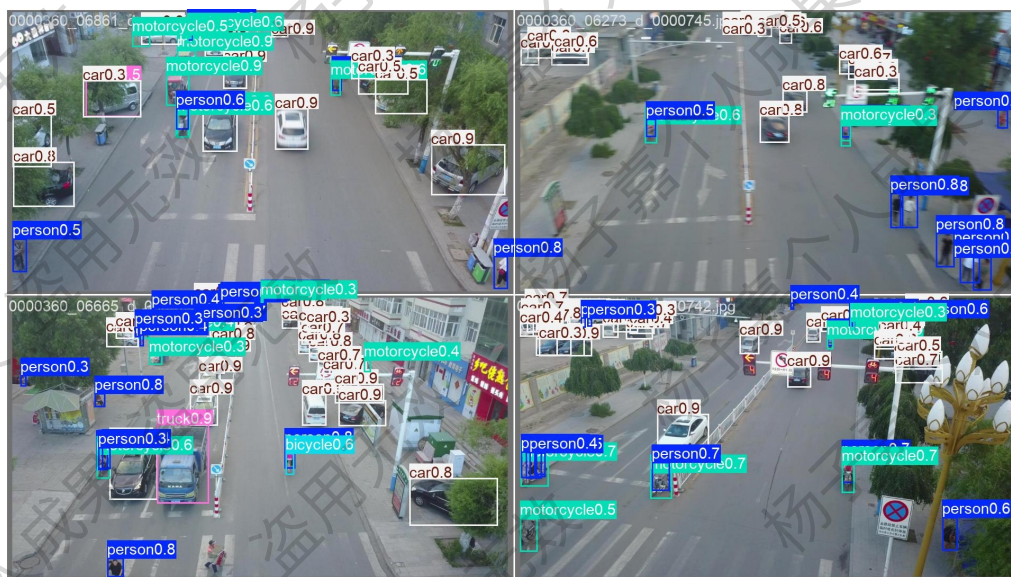


图 4-4-2：模型在航拍密集场景下的预测效果 2

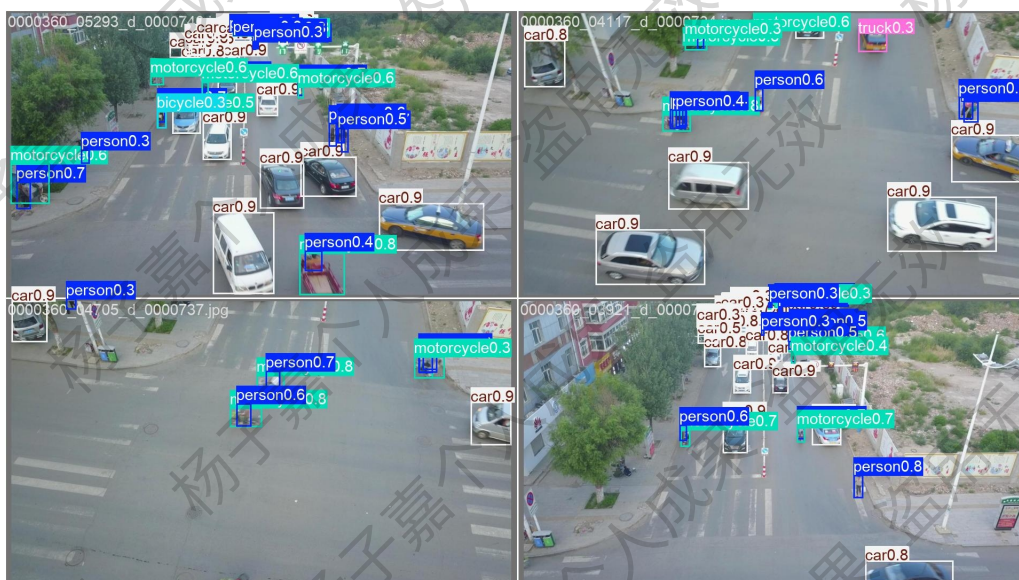


图 4-4-3：模型在航拍密集场景下的预测效果 3

为了直观评估模型在实际场景中的表现，本节选取了 VisDrone 验证集中具有代表性的预测结果进行可视化展示，如图 4-4 所示。

(1) 密集场景下的检测能力

从图 4-4-1 可以看出，模型在极其密集的场景下表现出色。画面中包含了大量相互遮挡的摩托车、行人和汽车，部分目标在原始图像中仅有十几个像素大小，人眼难以直接分辨。然而，模型不仅精准地定位了这些目标，还给出了较高的置信度分数（如多个 car 0.9、motorcycle 0.8）。这说明高分辨率微调（imgsz=1280）极大提升了模型对微小特征的提取能力。

(2) 不同视角的鲁棒性

图 4-4-2 和 4-4-3 展示了开阔路段和不同俯仰角度下的检测效果。模型能够准确识别出画面中远处的白色轿车、路边摩托车以及斑马线上的行人，展现了良好的透视变换适应能力。

(3) 局限性观察

尽管整体表现优秀，但在某些极端重叠区域（如左下角大量密集的摩托车群），模型仍出现了一些低置信度的预测（如 person 0.3），这反映了在极高遮挡情况下，局部特征丢失对模型分类造成的困难。这也是未来引入 SAHI 切片推理或设计更复杂的注意力模块来优化的方向。

5 结论与展望

5.1 结论

实验总结

本实验以无人机航拍小目标检测为任务，在个人消费级硬件（NVIDIA RTX 5060 8GB 显存）和外部供电中断的现实约束下，完成了从数据预处理到模型微调的完整工程实践。主要结论如下：

1. 受限算力下的高精度微调策略有效

通过将模型更换为轻量级的 YOLO11s，并将输入分辨率提升至 1280，结合 batch=2 与 AMP 混合精度训练，成功在 8GB 显存下完成了稳定训练，最终在 VisDrone 验证集上取得了 $mAP50 = 0.619$ 、 $mAP50-95 = 0.386$ 的成绩（6 类合并口径）；参考同任务公开的 640 分辨率基线（约 0.38~0.40，10 类口径），高分辨率微调带来了明显收益，但由于类别协议不同，该对比不构成严格同口径结论。

2. 断点续训展现了极强的工程鲁棒性

受供电中断影响，实验分三个阶段（总计 82 轮）完成。每次加载最优检查点（best.pt）重启时，虽然优化器动量被重置导致指标出现短暂回落，但模型均在数轮内迅速恢复并持续爬升，证明了 YOLO11s 架构的稳定性和断点续训策略的有效性。

3. YOLOE 架构在小目标任务上存在局限

YOLOE 架构在本任务上微调失败，提示文本对齐模型在极度密集的小目标数据集上对超参数较为敏感；该结果为选用标准 YOLO 架构提供了工程决策依据，其失败根因仍有待消融实验验证。

5.2 未来工作

尽管本实验取得了阶段性成果，但仍存在进一步提升的空间，未来可从以下方向继续推进：

引入 SAHI 切片推理：针对 VisDrone 中目标像素极小、极度密集的痛点，计划引入 Slicing Aided Hyper Inference (SAHI) 技术。将高分辨率原图切分为多个重叠子图分别检测后再合并，以降低边缘小目标的漏检率。

模型量化与边缘端部署：尝试采用 TensorRT 对 PyTorch 模型进行 FP16/INT8 量化，将推理速度提升数倍，以满足无人机机载嵌入式设备（如 NVIDIA Jetson）的实时性要求。

开发可视化 Web Demo：基于 Gradio 或 Streamlit 框架封装模型，制作一个支持图片/视频上传的交互式网页演示系统，提升项目的工程落地价值与展示效果。

参考文献

- [1] Jocher G, Qiu J. Ultralytics YOLO11[CP/OL]. (2024)[2026-09-24]. <https://github.com/ultralytics/ultralytics>.
- [2] Zhu P, Wen L, Du D, et al. Detection and tracking meet drones challenge[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 44(11): 7380-7399.
- [3] Tang L, et al. YOLOE: Real-Time Seeing Anything[J/OL]. arXiv preprint arXiv:2503.07465, 2025.
- [4] Akyon F C, Onur Altinuc S, Temizel A. Slicing aided hyper inference and fine-tuning for small object detection[C]//2022 IEEE International Conference on Image Processing (ICIP). IEEE, 2022: 966-970.
- [5] Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision[C]//International Conference on Machine Learning. PMLR, 2021: 8748-8763.