

实验报告

1. 实验环境

实验是在华为云服务器上进行训练，训练环境如下所示：

-操作系统：Ubuntu 22.04

-内存：24G

-Python 版本：3.8

-batchsize：16

-epochs：200

-优化器：SGD

2. 实验方法

2.1 数据集构建（模型训练集和测试集共 365 张图片）

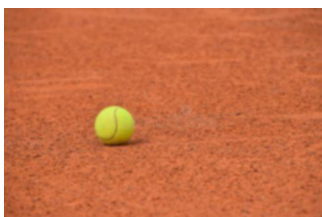
除官方提供的 30 张测试集图片外，实验还使用了部分飞桨社区网球数据集（[网球数据集](#) [数据集-飞桨 AI Studio 星河社区](#)）。为了提高模型对不同图像质量和光照条件下的鲁棒性（**针对户外条件的多样性**），我们对原始训练图像应用了两种数据增强策略：**光照增强**和**模糊增强**，具体实现如下：

光照增强包括随机亮度、对比度调整以及色调、饱和度、明度的变化，以模拟不同环境光照下的拍摄效果；模糊增强则采用高斯模糊和运动模糊，模拟图像抖动、运动等情况导致的图像质量下降。每张原始图像分别生成两张增强版本，对应的 YOLO 标签文件也同步复制，以保证图像与标注一致性。该策略有效扩充了数据集的多样性，有助于提升目标检测模型在复杂场景下的泛化能力。

示例图片如下：



原始图片



模糊增强



光照增强

2.2 模型选择

模型：YOLO11n（YOLO11 系列最轻量化的模型）

原因：选择 YOLOv11 进行网球识别，主要是因为其在精度与效率之间实现了极佳平衡。YOLOv11 引入了强化的特征提取结构和优化的训练流程，不仅提升了对小目标网球的检测能力，还在保持高精度的同时显著减少了模型参数，提高了计算效率。此外 YOLOv11 具备出色的跨平台适应性，能在边缘设备上流畅运行，非常适合部署在资源受限的实际应用场景中，为高效、实时的网球检测提供了可靠保障。

2.3 模型剪枝

约束训练：实验在剪枝前引入了约束训练策略，通过对 BatchNorm 层施加 L1 正则化，增强权重的稀疏性，使模型在保持准确性的同时更适合剪枝操作。相比直接剪枝，约束训练可降低因参数冗余性差、结构耦合度高和层间敏感性差异造成的性能下降，为后续高效、精度可控的模型压缩奠定基础。

模型剪枝：实验通过定义 PRUNE 类实现基于 BatchNorm 层权重稀疏性的结构剪枝，首先计算剪枝阈值以筛选出重要通道，然后依次对模型中关键卷积层和模块进行剪枝，调整相应卷积核和批归一化参数，确保剪枝后模型结构一致。剪枝过程涵盖主干网络和检测头，最终重新设置参数梯度并保存剪枝后的模型。该方法有效减少模型冗余，提升计算效率，同时尽量保持检测性能。

回调训练：剪枝过程会直接移除部分权重或通道，导致模型结构发生变化，可能引起性能下降甚至不稳定。通过回调训练，模型可以在剪枝后的新结构上重新调整参数，补偿剪枝带来的信息损失，恢复甚至提升检测准确度，确保剪枝既能减小模型规模和计算开销，又不显著影响最终的任务表现。

3. 实验结果与分析

3.1 实验结果

在第一章的实验环境下进行模型评估，模型剪枝前后指标如表 3.1 所示：

表 3.1 实验结果

模型	mAP50	mAP50-95	Precision	Recall	Parameters(M)	Speed(ms)
normal.pt	0.9849	0.9018	0.9938	0.9539	2.6	29.58
light.pt	0.9820	0.9040	0.9704	0.9724	2.1	28.73

3.2 结果分析

从实验结果来看，模型剪枝取得了一定成效。剪枝后的 `light.pt` 模型与原始 `normal.pt` 模型相比，在检测精度方面，`mAP50` 略有下降（从 0.9849 降至 0.9820），但 `mAP50-95` 有所提升（从 0.9018 升至 0.9040），且召回率显著提高（从 0.9539 升至 0.9724），表明模型对网球目标的整体检测能力和对不同置信度阈值下的适应性有一定优化。在模型规模与计算效率上，参数数量从 2.6M 减少至 2.1M，推理速度提升至 28.73ms，模型轻量化效果明显，在保持较高检测性能的同时，有效降低了计算开销，更适合在资源受限的场景中部署应用。

4. 未来优化方向

-优化数据增强策略：引入更多样化的增强方式（如几何变换、噪声添加等），以提升模型对极端天气、复杂背景等场景的鲁棒性。

-更精细的模型剪枝方法：结合通道重要性评估与结构重参数化技术，在压缩模型的同时尽可能减少精度损失。

-优化训练策略：采用动态学习率调整、难样本挖掘等方法，进一步提升模型收敛速度与检测精度，推动模型在实际场景中的高效部署与应用。