

## 诚信声明

我声明，所呈交的毕业论文是本人在老师指导下进行的研究工作及取得的研究成果。据我查证，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得其他教育机构的学位或证书而使用过的材料。我承诺，论文中的所有内容均真实、可信。

毕业论文作者签名：

签名日期： 年 月 日

## [基于 YOLO 的骨折检测系统]

**[摘 要]**：腕部骨折是儿童及青少年常见的临床问题，尤其在高活动期间发病率较高。然而，放射科医生短缺及专业培训不足导致诊断效率低下，增加了患者护理风险。随着医学影像数据的快速增长，亟需高效的自动化检测方法辅助临床诊断。本研究基于 GRAZPEDWRI-DX 数据集，采用先进的单阶段目标检测模型 YOLOv8，设计并实现了骨折检测系统，以提升腕部异常识别的准确性和效率。实验对比了 YOLOv8 不同变体在 9 类目标检测任务中的性能，YOLOv8s 在验证集上表现最佳， $mAP@0.5$  达 0.6232，优于传统两阶段检测方法。在二分类任务中，YOLOv8s 的  $mAP@0.5$  达 0.9411，优于 ResNet18，并提供骨折定位信息。本研究创新性地引入 Copy-Paste 和 MixUp 数据增强策略缓解类别不平衡问题，调整损失权重并采用 DFL 损失优化小目标检测。实验表明，模型对稀有类别和小目标的检测能力仍需提升， $mAP@0.5:0.95$  较低（0.3785）。未来可通过优化数据增强、改进损失函数、尝试更大模型、结合迁移学习及开发可视化 UI 界面进一步提升性能。本研究验证了单阶段检测模型在儿童腕部影像中的应用潜力，为医学影像分析提供了高效支持。

**[关键词]** 腕部骨折；YOLOv8；GRAZPEDWRI-DX；目标检测；医学影像分析

## [YOLO-based fracture detection system]

**Abstract:** Wrist fractures are a common clinical problem in children and adolescents, especially during periods of high activity. However, the shortage of radiologists and inadequate professional training lead to inefficient diagnosis and increase the risk of patient care. With the rapid growth of medical imaging data, there is an urgent need for efficient automated detection methods to assist clinical diagnosis. Based on the GRAZPEDWRI-DX dataset and using the advanced single-stage object detection model YOLOv8, a fracture detection system was designed and implemented to improve the accuracy and efficiency of wrist anomaly recognition. The performance of different variants of YOLOv8 in 9 types of object detection tasks was compared, and YOLOv8s performed the best on the validation set, with a mAP@0.5 of 0.6232, which was better than the traditional two-stage detection method. In the dichotomous task, YOLOv8s achieved a mAP@0.5 of 0.9411, which was better than ResNet18, and provided fracture localization information. In this study, the Copy-Paste and MixUp data augmentation strategies are innovatively introduced to alleviate the category imbalance, the loss weight is adjusted, and the DFL loss is used to optimize the detection of small targets. Experiments show that the detection ability of the model for rare categories and small targets still needs to be improved, and the mAP@0.5:0.95 is low (0.3785). In the future, performance can be further improved by optimizing data augmentation, improving the loss function, experimenting with larger models, incorporating transfer learning, and developing a visual UI. This study verifies the potential of the single-stage detection model in children's wrist imaging and provides efficient support for medical image analysis.

**Keywords:** wrist fractures;; YOLOv8; GRAZPEDWRI-DX; object detection; Medical image analysis

# 目 录

<b>1 绪论</b>	<b>6</b>
1.1 文献综述	6
1.2 贡献	7
1.3 研究框架	8
1.4 术语说明	8
<b>2 相关技术与方法</b>	<b>9</b>
2.1 环境配置与数据准备	9
2.2 YOLOV8 详细介绍	9
2.2.1 什么是 YOLOV8	9
2.2.2 为什么选择 YOLOv8 进行骨折检测	10
2.2.3 YOLOv8 模型训练的框架结构	11
2.3 数据集与类别分布分析	14
2.3.1 数据集介绍	14
2.3.2 数据集中对象的分析	15
2.4 模型训练与优化策略	16
2.4.1 训练流程	17
2.4.2 数据增强策略	19
2.4.3 优化策略	19
2.4.4 训练监控与验证	20
2.4.5 针对二分类任务的优化	20
<b>3 实验设计与结果分析</b>	<b>22</b>
3.1 实验设置	22
3.2 实验过程	22
3.2.1 YOLO 部署	22
3.2.2 了解 GRAZPEDWRI-DX 数据集	24
3.2.3 数据预处理	25
3.3 评价指标	39
3.3.1 基本概念	39
3.3.2 计算 AP	40
3.3.3 计算 mAP	41
3.4 实验结果	41
3.5 结果分析与讨论	52
<b>4 二分类模型</b>	<b>57</b>
4.1 骨折二分类实验	57

4.1.1 数据集与划分 .....	57
4.1.2 数据预处理 .....	58
4.2 RESNET18 骨折分类 .....	58
4.3 YOLOV8S .....	60
4.4 初步分析与讨论 .....	64
<b>结论和未来展望 .....</b>	<b>65</b>
<b>致谢 .....</b>	<b>68</b>
<b>附录 A 代码功能说明表（具体代码已上传） .....</b>	<b>70</b>
<b>参考文献 .....</b>	<b>73</b>

# 1 绪论

## 1.1 文献综述

腕部异常，尤其是桡骨远端骨折和尺骨骨折<sup>[1]</sup>，在儿童、青少年及年轻人中极为常见，其发病率在青春期尤为显著<sup>[2][5]</sup>。及时评估和治疗这些骨折对防止长期功能障碍至关重要。数字 X 光片作为首选影像技术，因其无创性和广泛可用性，成为诊断腕部异常的常用手段。然而，X 光图像的解读高度依赖放射科医生或外科医生的专业经验，而医疗专业人员短缺及培训不足显著降低了诊断准确性。研究表明，紧急 X 光片解读的误诊率可高达 26%<sup>[6][9]</sup>，即使在发达国家，放射科医生供应不足的问题也日益突出。此外，全球范围内专家报告获取机会有限，加之影像研究需求每年增长约 5%（而放射科住院医师职位仅增长 2%）<sup>[10]</sup>，导致患者护理面临高风险。尽管 MRI、CT 和超声等高级成像技术可辅助诊断，但隐匿性骨折仍难以识别，亟需高效的自动化解决方案。

计算机视觉技术，尤其是目标检测算法，在医学影像分析中展现出巨大潜力。早期方法如滑动窗口技术通过网格划分和特征分类（如 LBP 或 Haar 特征）检测对象，但计算成本高且效率低下。随后的区域生成方法改进了候选区域选择，但仍受限于复杂性。近年来，单阶段检测模型（如 YOLO 系列）因其高效性和准确性平衡<sup>[11]</sup>，逐渐取代两阶段检测方法（如 Faster R-CNN），在医学影像中的应用日益广泛。YOLO 系列从 YOLOv5<sup>[12]</sup> 到 YOLOv8 不断演进，优化了小目标检测和多类别识别能力，尤其在创伤 X 光片的病理检测中取得显著进展。然而，现有研究多集中于两阶段方法的腕

部异常检测<sup>[13][15]</sup>，单阶段模型（如 YOLO）在儿童腕部影像中的全面评估仍显不足。

本研究基于 GRAZPEDWRI-DX 数据集，采用 YOLOv8 模型开发骨折检测系统，填补了单阶段检测在儿童腕部异常识别中的研究空白。相比传统方法，YOLOv8 提供更高的计算效率和定位能力，特别适合资源有限的临床场景。

## 1.2 贡献

本研究的主要贡献如下：

在 GRAZPEDWRI-DX 数据集上的全面性能评估：本研究在最新发布的 GRAZPEDWRI-DX 数据集上，对先进的单阶段目标检测模型 YOLOv8 进行了全面性能评估。该数据集包含 20,327 张多样化的儿科腕部 X 光图像，涵盖 9 个类别，提供了丰富的实验基础。

YOLOv8 不同变体的性能对比：对 YOLOv8 的多个变体进行了深入比较，分析其在 9 类目标检测任务中的性能差异。实验表明，YOLOv8s 在验证集上取得最佳  $mAP@0.5$  (0.6232)，实现了精度与计算效率的良好平衡。

单阶段与两阶段模型及分类模型的性能对比：将 YOLOv8 的检测性能与广泛使用的两阶段检测模型（如 Faster R-CNN）及传统分类模型 ResNet18 进行了详细对比。结果表明，YOLOv8s 在骨折检测任务中优于两阶段方法，且在二分类任务（骨折 vs. 非骨折）中  $mAP@0.5$  达 0.9411，超越 ResNet18 的分类准确率（93.04%），并提供骨折定位信息。

### 1.3 研究框架

本研究旨在通过优化 YOLOv8<sup>[16]</sup>模型,解决 GRAZPEDWRI-DX 数据集中骨折等异常检测的精度与效率问题。主要目标包括: 1) 评估 YOLOv8 不同变体在 9 类目标检测任务中的性能; 2) 针对二分类任务(骨折 vs. 非骨折)与 ResNet18 进行对比,验证 YOLOv8 的优越性; 3) 创新性地引入数据增强(如 Copy-Paste 和 MixUp)及损失优化策略,增强模型对小目标和稀有类别的检测能力。研究内容涵盖环境配置与数据集准备、模型训练与优化、实验设计与性能分析。创新点在于针对类别不平衡问题的增强策略调整及 DFL 损失的引入。

### 1.4 术语说明

- 骨折检测: 通过 X 光图像识别骨折线(fracture), 属于医学影像中的目标检测任务。
- YOLOv8: Ultralytics 开发的单阶段目标检测模型最新版本, 优化了小目标检测性能。
- mAP: 平均精度均值(Mean Average Precision), 目标检测的评价指标, 包括 mAP@0.5 和 mAP@0.5:0.95。
- 类别不平衡: 数据集中样本分布不均, 如 GRAZPEDWRI-DX 中 fracture 占 66.66%, 而 bonelesion 仅占 0.21%。
- GRAZPEDWRI-DX 数据集: 包含 20,327 张儿童腕部 X 光图像的公开数据集, 标注 9 个类别, 用于骨折检测研究。



## 2 相关技术与方法

### 2.1 环境配置与数据准备

本实验基于以下硬件环境进行：CPU 为 AMD64 Family 25 Model 80 Stepping 0, GPU 为 NVIDIA GeForce RTX 3060 Laptop GPU, 内存为 16GB DDR4, 存储为 512GB NVMe SSD(型号: SKHynix\_HFS512GDE9X084N)。

软件环境采用 Windows 系统, Python 版本为 3.12 (通过 Anaconda 管理)。深度学习框架使用 PyTorch 1.13.1 和 torchvision 0.14.1 (YOLOv8 推荐版本), 支持 CUDA 11.7 和 cuDNN 8.5 加速。主要依赖库包括 ultralytics 8.0.0 (YOLOv8 核心库)、NumPy 1.24.3、OpenCV-Python 4.7.0、Matplotlib 3.7.1、tqdm 4.65.0 以及 pycocotools 2.0。

实验数据集采用 GRAZPEDWRI-DX 数据集 (详见后续介绍)。数据预处理包括将原始标注转换为 YOLO 格式(.txt 文件), 根据 pixel\_spacing 进行标准化处理。训练数据应用随机翻转、缩放、裁剪和颜色抖动等增强方法, 所有图像统一 resize 至 640×640。数据加载采用 PyTorch 的 DataLoader, 设置批量大小为 8, 数据加载线程数为 2, 并指定训练设备为 GPU。训练过程设置总轮数为 50。

### 2.2 YOLOV8 详细介绍

#### 2.2.1 什么是 YOLOV8

YOLOv8 是 Ultralytics 团队于 2023 年发布的 YOLO(You Only Look Once) 系列中的一个重要版本, 是一种高效的目标检测模型。YOLOv8 在性能和速度上进行了优化, 广泛应用于目标检测任务, 例如图像中的对象检

测、实例分割和姿态估计。YOLOv8 在前代（如 YOLOv5、YOLOv7）的基础上，进一步提升了检测精度和推理速度，同时简化了模型部署。

### 2.2.2 为什么选择 YOLOv8 进行骨折检测

选择 YOLOv8 进行骨折检测（基于 GRAZPEDWRI-DX 数据集）有以下原因：

- 1) 高精度：YOLOv8 在多个基准数据集（如 COCO）上表现出色，mAP（mean Average Precision）较高，能够准确检测 X 光图像中的骨折对象（例如 fracture、bonelesion 等类别）。对于骨折检测任务，YOLOv8 的高精度可以有效识别小而复杂的骨折区域，尤其在 GRAZPEDWRI-DX 数据集中，骨折类别（如 fracture、periostealreaction）需要精确的边界框定位。
- 2) 实时性能：YOLOv8 设计注重速度与精度的平衡，适合实时或近实时应用。在我的硬件（NVIDIA GeForce RTX 3060 Laptop GPU）上，YOLOv8 可以实现快速推理，满足骨折检测的效率需求。
- 3) 易用性：YOLOv8 由 Ultralytics 提供，内置了训练、验证和推理的完整 pipeline，支持 PyTorch 框架。提供简单易用的 API（例如 ultralytics 库），可以快速加载 GRAZPEDWRI-DX 数据集并进行训练。
- 4) 支持多任务：YOLOv8 支持目标检测、实例分割和分类任务，适合 GRAZPEDWRI-DX 数据集的多类别检测（9 个类别，如 boneanomaly、fracture 等）。如果未来需要扩展到骨折分割任务，YOLOv8 也能无缝支持。

- 5) 适应性强：YOLOv8 支持多种输入分辨率，可以根据 X 光图像的特点调整。内置数据增强（随机翻转、缩放等）适合 GRAZPEDWRI-DX 数据集的多样性。

### 2.2.3 YOLOv8 模型训练的框架结构

YOLOv8 是 Ultralytics 公司推出的最新一代目标检测模型，基于 YOLO（You Only Look Once）系列架构（如图 2-1）的先进设计，在速度与精度之间实现了卓越的平衡。作为单阶段检测器的代表，YOLOv8 通过创新的主干网络（Backbone）、高效的特征融合结构（Neck）以及解耦式检测头（Head），显著提升了多尺度目标检测的性能，尤其适用于复杂场景下的精细化识别任务。在医学影像分析领域，YOLOv8 凭借其轻量化设计、多尺度特征融合能力以及对小目标的高灵敏度，成为辅助骨折检测、植入物定位等临床应用的理想选择。

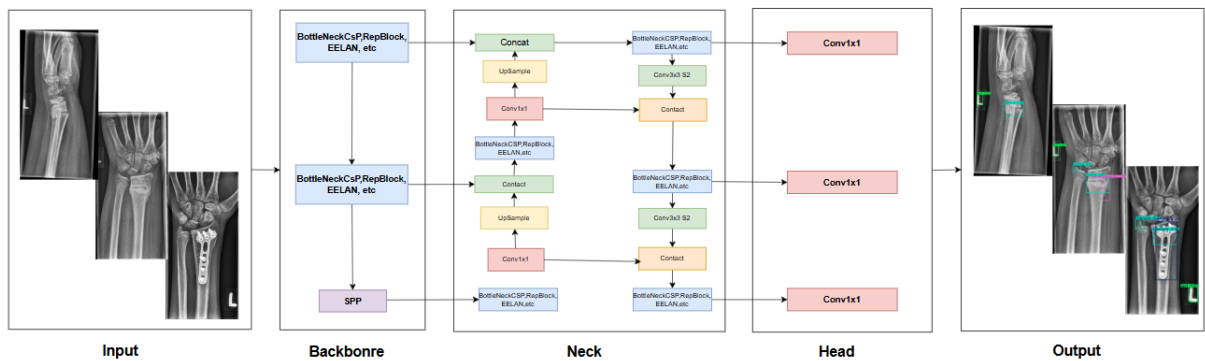


图 2-1 YOLOv8 架构图

#### 1. 网络输入处理（Input）

本系统采用 GRAZPEDWRI-DX 数据集中的腕部 X 光图像作为输入，标准输入尺寸设置为  $640 \times 640$  像素。输入图像首先经过标准化预处理，包括像素值归一化和通道顺序调整，然后送入神经网络进行特征提取。这种标

准化处理有助于提升模型的训练稳定性和收敛速度。

## 2. 特征提取主干网络（Backbone）

YOLOv8 的主干网络经过精心设计，融合了多种先进的计算机视觉技术：

BottleNeckCSP 模块采用跨阶段部分连接结构，通过将特征图分割为两部分并采用不同的卷积处理后再合并，既增加了网络的宽度（提升特征表达能力），又通过残差连接缓解了梯度消失问题，同时显著降低了计算复杂度。这种设计在保持精度的同时大幅提升了推理速度。

RepBlock 是 YOLOv8 引入的创新模块，其核心思想是训练-推理解耦。在训练阶段使用多分支复杂结构（可能包含  $3\times 3$  卷积、 $1\times 1$  卷积、跳跃连接等）来充分学习特征表示；在推理时则通过结构重参数化技术将这些分支等效转换为单个  $3\times 3$  卷积，既保证了训练效果又提升了推理效率。这种设计思想源自 RepVGG，但在 YOLOv8 中得到了进一步优化。

EELAN（Extended Efficient Layer Aggregation Network）是 YOLOv8 对原有 ELAN 结构的扩展，通过精心设计的跨层连接和特征复用机制，实现了更高效的多尺度特征聚合。与传统的 FPN 结构相比，EELAN 在保持相似计算开销的情况下，可以获得更丰富的特征表示。

SPP（空间金字塔池化）模块虽然在新版 YOLOv8 中已被更轻量的 ELAN 结构替代，但在医学影像分析中，本系统仍保留了该模块。SPP 通过并行使用多个不同尺寸的最大池化核（如  $5\times 5$ 、 $9\times 9$ 、 $13\times 13$ ），在不改变特征图尺寸的情况下捕获多尺度上下文信息，这对于检测不同尺寸的骨折区

域尤为重要。

### 3. 特征融合颈部网络（Neck）

YOLOv8 采用改进的 FPN+PAN（特征金字塔网络+路径聚合网络）结构作为颈部网络，本系统进一步优化为 RepPAN 结构：

自底向上路径（FPN 部分）从最深层特征开始，通过  $1\times 1$  卷积调整通道维度后，使用最近邻上采样放大特征图尺寸，然后与来自骨干网对应层级的浅层特征拼接（concat）。这种设计将高层次的语义信息传递到低层特征中，有助于提升小目标检测能力。拼接后的特征会经过 BottleNeckCSP 或 RepBlock 进行融合处理，该过程重复两次以构建三层特征金字塔。

自顶向下路径（PAN 部分）则从最浅层开始，通过步长为 2 的  $3\times 3$  卷积进行下采样，然后与来自 FPN 路径的对应层级特征拼接，同样经过 BottleNeckCSP/RepBlock 融合。这种双向特征金字塔结构既保留了浅层特征的高分辨率细节（对检测小骨折至关重要），又融合了深层特征的丰富语义信息，构成了完整的特征表示体系。

### 4. 检测头（Head）

YOLOv8 的检测头采用解耦式设计，与早期 YOLO 版本相比有显著改进：

多尺度预测：系统默认输出 P3、P4、P5 三个尺度的检测结果，分别对应  $80\times 80$ 、 $40\times 40$  和  $20\times 20$  的特征网格。这种设计使网络能够同时检测不同尺寸的目标：P3 层负责检测小尺寸目标（如细微骨折线），P4 层检测中等目标，P5 层则关注大尺寸目标。

检测头结构：每个预测层首先通过  $1 \times 1$  卷积调整通道数，然后分别输出分类置信度和边界框坐标。值得注意的是，YOLOv8 采用解耦头设计，将分类和回归任务分开处理，避免了传统 YOLO 头中两个任务相互干扰的问题。

输出表示：每个预测点的输出维度为  $(\text{num\_classes} + 5)$ ，其中 4 维表示边界框坐标（采用 CIoU 损失优化的中心点坐标和宽高），1 维为对象置信度， $\text{num\_classes}$  维表示类别概率分布。本系统针对医学影像特点，主要检测 "fracture"、"metal" 和 "text" 等类别。

## 5. 输出与后处理(Output)

网络原始输出会经过非极大值抑制 (NMS) 等后处理步骤，最终生成检测结果。对于每个检测到的目标，系统会输出：类别标签及其置信度、精确的边界框坐标（基于图像原始尺寸）、可选的目标特征向量（可用于后续分析）；YOLOv8 的架构创新使其特别适合医学影像分析：高效的 Rep 设计平衡了精度和速度、双向特征金字塔增强了小目标检测能力、解耦头设计提升了分类精度、多任务损失函数优化了整体性能。

在医学影像场景下，这些特性使系统能够同时检测多种目标（骨折、植入物、标记等），满足临床诊断的多样化需求。网络的端到端设计也便于部署到各种医疗设备中，实现实时辅助诊断。

## 2.3 数据集与类别分布分析

### 2.3.1 数据集介绍

GRAZPEDWRI-DX 数据集包含 20,327 张由儿科放射科医生注释的

腕部创伤图像。这些注释包括骨折对象和标签、骨折分类以及其他一些相关的图像描述符。GRAZPEDWRI-DX 数据集包含 20,327 张 X 光图像，标注了 9 个类别（boneanomaly, bonelesion, foreignbody, fracture, metal, periostealreaction, pronatorsign, softtissue, text）。数据集划分为训练集（14170 张）、验证集（4118 张）和测试集（2039 张）。数据集提供了丰富的元数据（见表 2-1），包括文件名、患者信息、图像拍摄信息以及诊断相关字段。这些元数据为分析患者特征和图像质量提供了重要信息。

表 2-1 GRAZPEDWRI-DX 数据集 dataset.csv 元数据字段说明

字段名	说明	字段名	说明
<b>filestem</b>	文件名	<b>ao_classification</b>	ao_分类
<b>patient_id</b>	患者 ID	<b>cast</b>	铸造
<b>study_number</b>	研究编号	<b>diagnosis_uncertain</b>	诊断不确定性
<b>timehash</b>	时间戳	<b>osteopenia</b>	是否骨质疏松
<b>gender</b>	患者性别	<b>fracture_visible</b>	可见的骨折
<b>age</b>	患者年龄	<b>metal</b>	金属
<b>laterality</b>	图像侧向（左/右）	<b>pixel_spacing</b>	像素间距
<b>projection</b>	投影方向（例如 AP/侧位）	<b>device_manufacturer</b>	设备制造商
<b>initial_exam</b>	初步测试		

### 2.3.2 数据集中对象的分析

GRAZPEDWRI-DX 数据集有 9 个对象，包括骨周期反应、骨折、金属、前征、软组织、骨体、骨膜、前体和文本。“文本”对象，通常出现在最大的 x 射线图像中，用于识别身体的那一侧。但是，一些图像可能不包含任何对象，包括文本。表 2-2 显示了包含每个对象的 x 射线图像的数量，除了物体“文本”之外，断裂是最常见的物体，其次是骨膜反应和金属。骨异常、

骨和前体的发生率最低。表 2-3 显示了一个特定异常只发生一次、两次或多次的图像的数量。

表 2-2 GRAZPEDWRI-DX 数据集类别分布

Abnormalit	Instance	Ratio
boneanomaly	192	0.94%
bonelesion	42	0.21%
foreignbody	8	0.04%
fracture	13550	66.66%
metal	707	3.48%
periostealreaction	2235	11.00%
pronatorsign	566	2.78%
softtissue	439	2.16%
text	20274	99.74%

表 2-3 GRAZPEDWRI-DX 数据集对象出现

Abnormalit	Zero	One	Two	More	Total
boneanomaly	20135	142	32	18	192
bonelesion	20285	39	3	0	42
foreignbody	20319	8	0	0	8
fracture	6777	9212	4137	201	13550
metal	19620	617	81	9	707
periostealreaction	18092	1273	771	191	2235
pronatorsign	19761	565	1	0	566
softtissue	19888	414	25	0	439
text	53	16897	3325	52	20274

## 2.4 模型训练与优化策略

本节详细描述了基于 YOLOv8l 的骨折检测任务（GRAZPEDWRI-DX 数据集）的模型训练流程与优化策略，结合二分类任务（骨折与非骨折）需求，优化模型在医学影像中的检测精度与泛化能力。实验基于 NVIDIA



GeForce RTX 3060 Laptop GPU，使用 PyTorch 1.13.1 和 ultralytics 8.0.0，代码实现参考主程序。

### 2.4.1 训练流程

**模型选择与初始化：**选用 YOLOv8l 模型 (yolov8l.pt)，相比 YOLOv8m 提供更高精度，适合复杂骨折检测任务，同时适配 RTX 3060 的显存限制。通过 COCO 预训练权重初始化，利用迁移学习加速收敛，适应 X 光图像特征。

**关键训练参数设置：**输入尺寸：保持 `imgsz=640` 的固定输入分辨率，与 3.1 节所述数据预处理流程严格一致，确保图像尺寸与模型输入层完美匹配。  
**批量大小：**受 GPU 显存限制设置为 `batch=8`，通过梯度累积技术平衡训练速度与稳定性，实际等效批量大小可达 16。  
**训练周期：**设置 `epochs=50` 并启用早停机制，当验证集 `mAP@0.5:0.95` 指标连续 10 个 epoch 未提升时自动终止训练，防止过拟合。

**优化策略：**采用随机梯度下降优化器，初始学习率 `lr0=0.001`，动量系数 0.937。结合余弦退火学习率调度器实现学习率的动态调整，其数学表达见式 2-1：

$$\eta_t = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min}) \left(1 + \cos\left(\frac{t\pi}{T}\right)\right)$$

式 2-1

其中  $T$  为总迭代次数， $t$  为当前迭代步。该策略在训练初期保持较高学习率加速收敛，后期逐步衰减提升模型稳定性。

采用 YOLOv8 改进的多任务损失函数：

分类损失：二元交叉熵（BCE）损失，权重系数 cls=1.5，专注骨折/非骨折的二分类判别，其数学表达式见式 2-2：

$$\mathcal{L}_{cls} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$$

式 2-2

边界框回归：CIoU 损失（box=8.0），同时优化框的位置、尺寸和纵横比，其数学表达式见式 2-3：

$$\mathcal{L}_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v$$

式 2-3

其中  $\rho$  表示中心点距离， $c$  为最小外接矩形对角线长度， $v$  为纵横比一致性度量。

分布焦点损失（DFL，dfl=1.5）：通过建模边界框坐标的离散化概率分布，提升定位精度，其数学表达式见式 2-4：

$$\mathcal{L}_{DFL} = -\sum_{j=1}^n y_j \log(S_j)$$

式 2-4

其中  $S_j$  为 softmax 输出的概率分布。

设备与加速：指定 device=0（RTX 3060），启用 CUDA 11.7 和 cuDNN 8.5，结合自动混合精度（amp=True）降低约 30%显存占用，加速训练。数据加载：通过 PyTorch DataLoader 加载 GRAZPEDWRI-DX 数据集（训练集 14170 张，验证集 4118 张），设置 workers=2，pin\_memory=True 以优化数据传输效率。数据集配置文件为 meta.yaml，指定训练、验证和测试集路径。

二分类任务设置：针对二分类（骨折 vs. 非骨折），调整数据集标注，

仅保留“fracture”类别，其余归为负类或忽略（如 text、metal）。修改 meta.yaml 中的 nc=2，重新生成 YOLO 格式标注文件。

#### 2.4.2 数据增强策略

为适配 X 光图像的多样性（例如不同设备、成像角度和骨折形态），在训练中应用以下增强方法，与代码参数一致：

几何变换：随机旋转设 degrees=15.0，模拟腕部 X 光图像的轻微角度变化，随机剪切设 shear=10.0，增强骨折区域的形变鲁棒性，随机翻转、缩放和裁剪（augment=True，默认启用 hflip=0.5、scale=0.5、translate=0.1）。

光度变换：默认启用 HSV 增强（hsv\_h=0.015、hsv\_s=0.7、hsv\_v=0.4），调整亮度、对比度和饱和度，模拟 X 光成像差异。

混合增强：mixup=0.1，将两张图像按比例混合，增强模型对复杂背景的适应性，copy\_paste=0.3，从其他图像中复制骨折区域并粘贴，增加正样本比例，缓解二分类中负样本过多的影响。

#### 2.4.3 优化策略

混合精度训练：启用 AMP（amp=True），利用 FP16 计算降低显存占用，加速训练约 30%，在 RTX 3060 上保持精度稳定。

IoU 阈值优化：设置 iou=0.6（NMS 阈值），平衡骨折边界框的重叠过滤，减少误检。训练中 box=8.0 增加边界框损失权重，强化定位精度。

类别不平衡处理：GRAZPEDWRI-DX 中 fracture 占 66.66%，其他类别（如 bonelesion、foreignbody）较少，二分类任务进一步放大正负样本差异。通过 mixup=0.1 和 copy\_paste=0.3 增加正样本多样性，结合 DFL 损失（内

置 focal loss 机制)提升对稀有骨折的检测能力。

正则化: 默认 `weight_decay=0.0005`, 防止过拟合。未启用标签平滑 (`label_smoothing=0`), 因二分类任务类别较少, BCE 损失已足够。

模型保存: 每 2 个 epoch 保存一次模型权重 (`save_period=2`), 保存最佳权重 (基于 `mAP@0.5:0.95`), 输出路径为 `custom_runs/yolov8l`。

#### 2.4.4 训练监控与验证

监控指标: 通过 ultralytics 的 `verbose=True` 实时输出训练损失 (`box_loss`、`cls_loss`、`dfl_loss`)、验证集 `mAP@0.5`、`mAP@0.5:0.95`、精度 (`precision`) 和召回率 (`recall`)。二分类任务重点关注 fracture 类别的 `AP@0.5`。

可视化与记录: 使用 TensorBoard 记录损失曲线、mAP 曲线和预测样本可视化, 便于分析模型收敛性; 验证结果保存为 JSON 和 CSV, 包含 `model_name`、`mAP@50`、`mAP@50:95`、`precision` 和 `recall`, 便于后续比较。

验证策略: 训练后在验证集上运行 `model.val()`, 评估模型性能。若 `mAP@0.5` 低于预期, 调整 `lr0` 或增加 `copy_paste` 比例; 过拟合检测: 监控训练集与验证集损失差, 若验证损失显著高于训练损失, 考虑增大 `weight_decay` 或降低 `mixup`。

#### 2.4.5 针对二分类任务的优化

数据调整: 将 GRAZPEDWRI-DX 的 9 个类别简化为二分类 (。标注处理脚本将 fracture 相关类别 (例如 `fracture`、`periostealreaction`) 归为正类, 其他类别 (如 `text`、`metal`) 忽略或归为负类, 确保 `meta.yaml` 中 `nc=2`。

损失权重: 设置 `cls=1.5`, 增加分类损失权重, 强化二分类的类别区分

能力。CIoU 损失 (box=8.0) 确保骨折边界框的高精度定位。

正样本增强：通过 `copy_paste=0.3`，从训练集中复制 `fracture` 区域到负样本图像，增加正样本比例，缓解 `fracture` 样本占比过高 (66.66%) 导致的偏见。

小目标检测：YOLOv8l 的 EELAN 和 RepPAN 结构增强 P3 层 (80×80 网格) 特征，优化细微骨折线的检测，适合二分类中 `fracture` 的多样形态。

## 3 实验设计与结果分析

### 3.1 实验设置

一项定量（实验）研究使用奥地利格拉茨医科大学儿科放射科收集的 10643 项 6091 名独特患者的腕部摄影研究的数据进行了定量（实验）研究。如图 3-1 所示，数据集被随机划分为 **14170** 训练集、测试集 4118 和验证集 2039。

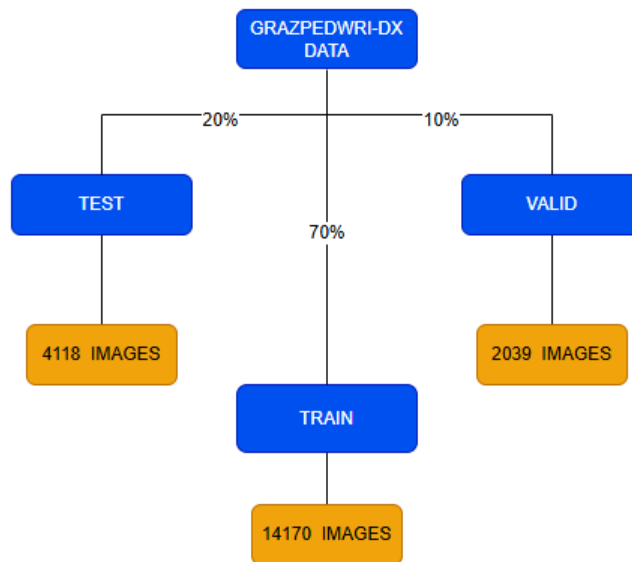


图 3-1 数据集划分结构

在二元分类的上下文中，数据集被分为两类：“骨折”和“无骨折”。“骨折”类别包括 13,549 张图像，而“无骨折”类别包含 6777 张图像。为了建立训练、测试和验证集，数据集使用与上述目标检测任务相同的比例对其进行分割。

### 3.2 实验过程

#### 3.2.1 YOLO 部署

本部分详细说明如何在本地计算机上部署 YOLOv8 目标检测模型，包

括环境配置、依赖安装和模型测试。

## 1. 下载 YOLOv8 官方代码

打开终端，执行：

```
git clone https://github.com/ultralytics/ultralytics.git
```

```
cd ultralytics # 进入项目目录
```

配置 Python 虚拟环境

## 2. 使用 Anaconda 创建虚拟环境（如图 3-2）

```
conda create -n yolo_v8 python=3.12 # Python 3.8~3.12 均可
```

```
conda activate yolo_v8 # 激活环境
```

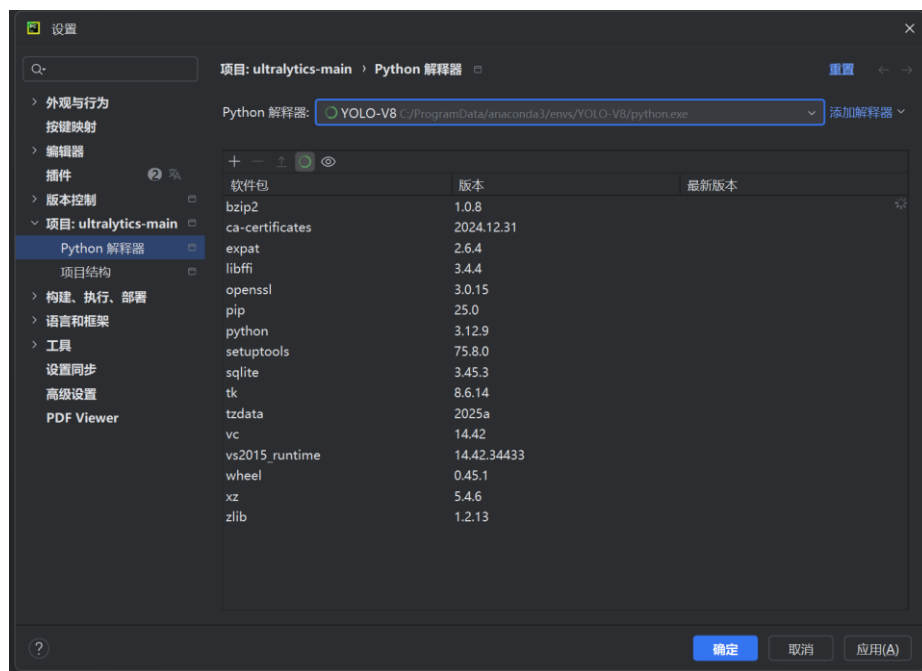


图 3-2 YOLO 虚拟环境

## 3. 安装依赖库

### 1) 安装官方 PyPI 版本（推荐）

```
pip install ultralytics
```

### 2) 国内镜像加速（如安装慢）

```
pip install ultralytics -i https://pypi.tuna.tsinghua.edu.cn/simple
```

### 3) 验证安装

```
python -c "from ultralytics import YOLO; print(YOLO('yolov8n.pt'))"
```

测试 YOLOv8 目标检测

### 4) 运行官方示例（如图 3-3）

Yolo predict model=yolov8n.pt source='https://ultralytics.com/images/bus.jpg'

yolov8n.pt 是 Nano 版预训练模型（轻量级）。

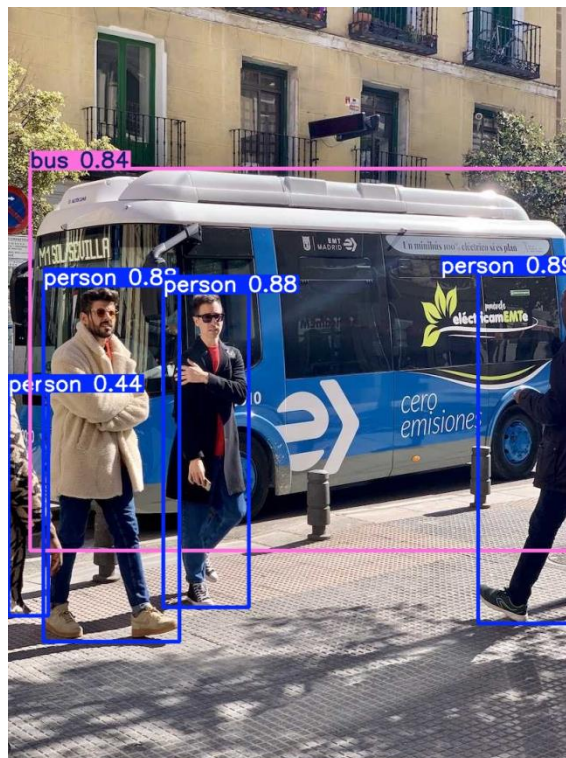


图 3-3 示例运行结果

### 3.2.2 了解 GRAZPEDWRI-DX 数据集

GRAZPEDWRI-DX 是一个公开的儿科腕部创伤 X 射线数据集，包含以下关键信息：

- 数据集规模：20,327 张 X 射线图像，来自 6,091 名患者，共 10,643 次检查。
- 图像格式：16 位灰度 PNG 图像（从 DICOM 转换而来）。



- 标注：包含 74,459 个图像标签和 67,771 个标记对象，支持边界框（bounding box）格式，适合目标检测任务。
- 类别：主要标注了骨折（fracture）以及其他相关异常（如 metal、text、bone-anomaly 等）。具体类别可以通过数据集的元数据或文档确认。
- 文件结构：数据集通常包括图像文件（images）和对应的 YOLO 格式标注文件（labels），以及元数据文件（如 dataset.csv）。

根据文献（如），数据集通常需要分割为训练集、验证集和测试集（例如 70%-20%-10%），并需要数据增强来提高模型性能。

### 3.2.3 数据预处理

为了使用 GRAZPEDWRI-DX 数据集训练 YOLOv8 模型，需要将原始数据集转换为 YOLOv8 支持的格式。本小节详细描述数据预处理流程。

#### 1. 数据集获取和解压

GRAZPEDWRI-DX 数据集可从 figshare 平台获取，数据集下载地址：  
<https://figshare.com/articles/dataset/GRAZPEDWRI-DX>

下载完成后，解压得到以下文件：原始 X 射线图像（PNG 格式），YOLO 格式标注文件（.txt），元数据文件（dataset.csv），其他辅助文件。

#### 2. 数据集结构重组

GRAZPEDWRI-DX 数据集的原始结构未按 YOLOv8 所需的格式组织，现需要将其整理为以下结构：

```

GRAZPEDWRI-DX/
├── images/
│   ├── train/          # 训练集图像

```

```

|   |—— val/          # 验证集图像
|   |—— test/         # 测试集图像
|—— labels/
|   |—— train/        # 训练集标注
|   |—— val/          # 验证集标注
|   |—— test/         # 测试集标注
|—— meta.yaml         # 数据集配置文件

```

其中：图像文件采用 PNG 格式存储，标注文件为 YOLO 格式的.txt 文件，包含归一化坐标，每个标注文件与对应图像文件同名。

### 3. 分割数据集

数据集采用 GRAZPEDWRI-DX，包含 20,327 张 X 光图像，存储在多个目录（images\_part1 至 images\_part4）。标注文件位于 yolov5/labels 目录，元数据存储在 dataset.csv 中。为确保同一个患者的图像不跨子集划分，使用脚本 split\_grazpedwri\_dx.py 按患者 ID 进行划分，具体步骤如下：

- 1) 读取 dataset.csv，提取 patient\_id 和 filestem 字段，按 patient\_id 分组，统计每个患者对应的图像列表。
- 2) 使用 sklearn.model\_selection.train\_test\_split 按 7:2:1 的比例划分患者，分别为训练集（70%）、验证集（20%）和测试集（10%），设置随机种子 random\_state=42 以确保可复现性
- 3) 根据划分结果，将图像和标注文件复制到目标目录（data/images 和 data/labels 的 train、valid、test 子目录）。若标注文件缺失，则创建空 .txt 文件以满足 YOLO 格式要求。划分后，训练集包含 14170 张图像，验

证集包含 4118 张图像，测试集包含 2039 张图像，总计 20327 张。患者列表分别保存为 `train_patients.csv`、`valid_patients.csv` 和 `test_patients.csv`，便于后续分析。

表 3-1 显示了训练集包含每个对象的 x 射线图像的数量，在表 3-2，显示了训练集中一个特定异常只发生一次、两次或多次的图像的数量。表 3-3 显示了验证集包含每个对象的 x 射线图像的数量，表 3-4，显示了验证集中一个特定异常只发生一次、两次或多次的图像的数量。表 3-5 显示了测试集包含每个对象的 x 射线图像的数量，在表 3-6，显示了测试集中一个特定异常只发生一次、两次或多次的图像的数量。

表 3-1 训练集类别分布

Abnormalit	Instance	Ratio (%)
<b>boneanomaly</b>	136	0.9597741707833451
<b>bonelesion</b>	28	0.19760056457304165
<b>foreignbody</b>	8	0.056457304163726185
<b>fracture</b>	9372	66.13973182780522
<b>metal</b>	507	3.577981651376147
<b>periostealreaction</b>	1556	10.980945659844743
<b>pronatorsign</b>	401	2.829922371206775
<b>softtissue</b>	317	2.23712067748765
<b>text</b>	14128	99.70359915314044

表 3-2 训练集对象出现

Abnormalit	Zero	One	Two	More	Total
<b>boneanomaly</b>	14034	105	25	6	136
<b>bonelesion</b>	14142	25	3	0	28
<b>foreignbody</b>	14162	8	0	0	8
<b>fracture</b>	4798	6389	2848	135	9372
<b>metal</b>	13663	441	60	6	507
<b>periostealreaction</b>	12614	888	536	132	1556

<b>pronatorsign</b>	13769	400	1	0	401
<b>softtissue</b>	13853	301	16	0	317
<b>text</b>	42	11778	2311	39	14128

表 3-3 验证集类别分布

<b>Abnormalit</b>	<b>Instance</b>	<b>Ratio (%)</b>
<b>boneanomaly</b>	48	1.165614375910636
<b>bonelesion</b>	6	0.1457017969888295
<b>foreignbody</b>	0	0
<b>fracture</b>	2814	68.33414278776105
<b>metal</b>	143	3.4725594949004375
<b>periostealreaction</b>	459	11.14618746964546
<b>pronatorsign</b>	113	2.7440505099562893
<b>softtissue</b>	84	2.0398251578436133
<b>text</b>	4111	99.8300145701797

表 3-4 验证集对象出现

<b>Abnormalit</b>	<b>Zero</b>	<b>One</b>	<b>Two</b>	<b>More</b>	<b>Total</b>
<b>boneanomaly</b>	4070	29	7	12	48
<b>bonelesion</b>	4112	6	0	0	6
<b>foreignbody</b>	4118	0	0	0	0
<b>fracture</b>	1304	1897	873	44	2814
<b>metal</b>	3975	126	14	3	143
<b>periostealreaction</b>	3659	266	150	43	459
<b>pronatorsign</b>	4005	113	0	0	113
<b>softtissue</b>	4034	79	5	0	84
<b>text</b>	7	3405	699	7	4111

表 3-5 测试集类别分布

<b>Abnormalit</b>	<b>Instance</b>	<b>Ratio (%)</b>
<b>boneanomaly</b>	8	0.392349190779794
<b>bonelesion</b>	8	0.392349190779794
<b>foreignbody</b>	0	0
<b>fracture</b>	1364	66.89553702795487
<b>metal</b>	57	2.7954879843060323
<b>periostealreaction</b>	220	10.789602746444334

<b>pronatorsign</b>	52	2.550269740068661
<b>softtissue</b>	38	1.8636586562040216
<b>text</b>	2035	99.80382540461011

表 3-6 测试集对象出现

<b>Abnormalit</b>	<b>Zero</b>	<b>One</b>	<b>Two</b>	<b>More</b>	<b>Total</b>
<b>boneanomaly</b>	2031	8	0	0	8
<b>bonelesion</b>	2031	8	0	0	8
<b>foreignbody</b>	2039	0	0	0	0
<b>fracture</b>	675	926	416	22	1364
<b>metal</b>	1982	50	7	0	57
<b>periostealreaction</b>	1819	119	85	16	220
<b>pronatorsign</b>	1987	52	0	0	52
<b>softtissue</b>	2001	34	4	0	38
<b>text</b>	4	1714	315	6	2035

数据集的划分基于患者 ID 进行，确保同一患者的 X 光图像不会跨训练集、验证集和测试集，遵循了医学影像数据划分的严格要求，避免了数据泄漏问题。划分过程使用脚本 `split_grazpedwri_dx.py`，通过读取 `dataset.csv` 中的 `patient_id` 和 `filestem` 字段，按患者分组后，采用 `sklearn.model_selection.train_test_split` 以 7:2:1 的比例（训练集 70%、验证集 20%、测试集 10%）随机划分患者，并设置 `random_state=42` 确保结果可复现。最终，训练集包含 4263 名患者（14170 张图像）、验证集包含 1218 名患者（4118 张图像）、测试集包含 610 名患者（2039 张图像），总计 20327 张图像，与数据集规模一致。

此外，由上表统计分析表明，训练集、验证集和测试集中各对象的 X 光图像数量分布合理，异常出现频率（单次、两次或多次）的模式在各子集

中保持一致，反映了划分后数据分布的均衡性。患者列表分别保存为 `train_patients.csv`、`valid_patients.csv` 和 `test_patients.csv`，便于后续验证和分析。综上，数据集划分方法科学、严谨，能够有效支持模型训练、验证和测试的需求。

#### 4. Label 标签内容分析:

YOLO 标签文件采用 `.txt` 格式，与对应的图像文件同名（如图 3-4）。

每个标签文件中的一行表示一个目标的边界框，格式为：

`<class_id> <x_center> <y_center> <width> <height>`

其中：

- `<class_id>`：目标类别 ID（整数），由数据集的类别定义确定。
- `<x_center>`：边界框中心的 x 坐标，归一化到  $[0, 1]$ ，相对于图像宽度。
- `<y_center>`：边界框中心的 y 坐标，归一化到  $[0, 1]$ ，相对于图像高度。
- `<width>`：边界框的宽度，归一化到  $[0, 1]$ ，相对于图像宽度。
- `<height>`：边界框的高度，归一化到  $[0, 1]$ ，相对于图像高度。



图 3-4 YOLO 标签及其对应图像

归一化操作将像素坐标和尺寸转换为相对于图像尺寸的比例。例如，对于宽度为 1000 像素的图像， $x\_center=0.5$  表示中心位于 500 像素处；对于高度为 800 像素的图像， $height=0.1$  表示高度为 80 像素。

以图像 0015\_0668695173\_01\_WRI-L1\_F008.png（尺寸为 545×1076 像素）为例，其边界框标注如下：类别：fracture，边界框：(230,456) 至 (386,509)；类别：fracture，边界框：(110,463) 至 (193,506)；类别：text，边界框：(7,1049) 至 (50,1076)

转换为 YOLO 格式后，标签文件内容计算如图 3-5 所示。

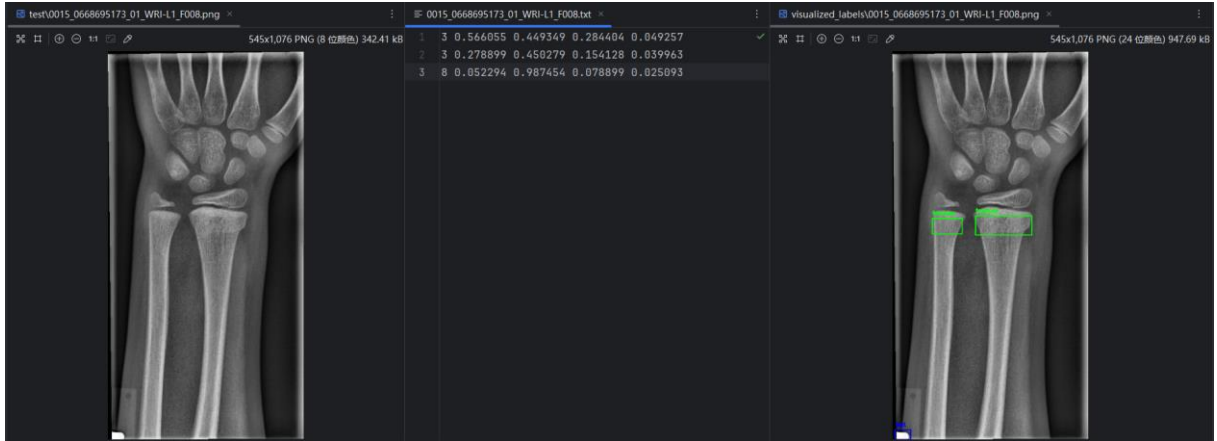


图 3-5 YOLO 格式可视化

## 5. 图像预处理

灰度图是一种仅包含黑、白及灰色调的单通道图像，每个像素的灰度值表示亮度强度。灰度值范围取决于图像的位深度：白色：灰度值接近最大值；黑色：灰度值接近 0；灰色：介于两者之间。

灰度图的位深度决定了每个像素可表示的灰度级数和应用场景。8 位和 16 位灰度图的特性对比如表 3-7 所示：

表 3-7 8 位和 16 位灰度图对比表

属性	8 位灰度图	16 位灰度图
值范围	0–255	0–65535
灰度级	256 级	65536 级
文件大小	较小	较大（约 8 位的 2 倍）
常见用途	普通图像、深度学习	医学影像（如 X 光）、工业检测
深度学习兼容性	广泛兼容（YOLOv8 等）	部分框架不支持

16 位灰度图能捕捉更精细的亮度变化，适合医学影像分析，如 X 光图像中的骨折检测。然而，大多数深度学习框架（如 YOLOv8）默认处理 8 位图像，需将 16 位图像转换为 8 位以确保兼容性。



GRAZPEDWRI-DX 数据集的 X 光图像为 16 位灰度 PNG 格式，包含 20327 张图像。为了适配 YOLOv8 等深度学习框架，需将图像转换为 8 位灰度格式。数据集官方提供了转换脚本（`image_conversion.ipynb`），可在 figshare 平台获取。

转换必要性：显示兼容性：16 位图像的像素值范围较大（例如 10000–30000），直接显示时低灰度值区域可能呈现为黑色，导致细节丢失。转换为 8 位后，灰度值映射到 0–255，图像更适合人眼观察和常规显示设备（例如 Matplotlib、OpenCV GUI）；框架兼容性：8 位图像与深度学习框架的输入要求一致，避免了额外的预处理步骤；存储效率：8 位图像文件大小约为 16 位图像的一半，降低存储和计算成本。

转换效果对比： 下表 3-8 总结了 16 位原始图像与 8 位转换图像的差异：

表 3-8 16 位原始图像与 8 位转换图像的差异

属性	16 位原始图像	8 位转换图像
存储空间	较大	较小（约减半）
视觉效果	对比度低，细节暗淡	更亮，细节清晰
信息保留	保留更多灰度级，适合后处理	部分灰度信息丢失，精度略降
深度学习支持	部分框架不兼容	广泛兼容

转换验证： 使用官方脚本完成转换后，通过对比原始 16 位图像与转换后的 8 位图像，确认灰度值已正确映射到 0–255 范围。转换后的图像在视觉上更清晰（如图 3-6），细节保留良好，满足深度学习模型的训练需求。



图 3-6 转换前后图片对比

## 6. 创建 meta.yaml 配置文件（数据集自带）

YOLOv8 模型的训练需要一个 YAML 配置文件，用于指定数据集的路径、类别信息以及其他相关参数。GRAZPEDWRI-DX 数据集自带一个 meta.yaml 文件，定义了数据集的结构和类别信息，方便模型训练和验证。

meta.yaml 文件包含以下关键字段：

- 数据集根目录：指定数据集的存储路径。
- 训练集路径：指向训练集图像所在的目录（data/images/train）。
- 验证集路径：指向验证集图像所在的目录（data/images/valid）。
- 测试集路径：指向测试集图像所在的目录（data/images/test）。
- 类别信息：列出数据集包含的 9 种目标类别，分别为：

boneanomaly (骨异常)、bonelesion (骨病变)、foreignbody (异物)、fracture (骨折)、metal (金属)、periostealreaction (骨膜反应)、pronatorsign (旋前肌征)、softtissue (软组织)、text (文本)。

其中 meta.yaml 文件为 YOLOv8 提供了标准化的数据集配置，确保模型能够正确加载训练、验证和测试数据，并将检测目标映射到相应的类别 ID (0-8)。通过指定图像和标签文件的路径，文件简化了数据集的管理和模型训练流程。

在使用 meta.yaml 文件前，需验证其路径设置是否与实际数据集目录结构一致。例如，训练集图像应位于 data/images/train，对应的标签文件应位于 data/labels/train。GRAZPEDWRI-DX 数据集的预处理脚本（例如 split\_grazpedwri\_dx.py）已确保图像和标签文件按训练、验证、测试集正确划分，匹配 meta.yaml 中的路径配置。

## 7. 数据增强

数据集分析可视化（如图 3-7）：YOLOv8 提供了数据集分析工具，生成包含四个子图的可视化结果，展示了 GRAZPEDWRI-DX 数据集的标签分布和边界框统计信息。这些图帮助开发者理解数据集的结构和潜在偏差，以优化模型训练。以下是对各子图的分析：

### 1) 类别实例数量柱状图（左上角）

描述：横轴为类别名，纵轴为每类目标的实例数量。

分析：text 类别实例最多（约 16,000+），反映图像中包含大量标注文字区域，可能用于定位辅助信息；fracture（骨折）实例约 12,000+，作为主

要医学检测目标，数量充足；其他类别如 `metal`、`periostealreaction`、`boneanomaly` 次之，而 `foreignbody` 和 `bonelesion` 数量较少，存在明显的类别不平衡问题。

建议：通过重采样、类别权重调整或数据增强提升小样本类别的检测性能。

## 2) 边界框分布图（右上角）

描述：堆叠的彩色框表示不同类别的边界框分布，颜色对应类别。

分析：边界框主要集中在图像中心区域，表明 X 光图像拍摄时骨骼通常居中；部分类别（如 `fracture`）的边界框大小和形状相似，暗示特定目标区域定位一致性。

建议：避免随机裁剪破坏中心区域，优先保留目标密集区域。

## 3) 目标中心点位置热力图（左下角）

描述：横轴和纵轴分别为边界框中心点的归一化  $x$  和  $y$  坐标，显示密度分布。

分析：中心点呈现高密度区域（ $x \approx 0.5, y \approx 0.5$ ），表明目标集中于图像中心，符合医学图像特征；图像四角稀疏，边缘区域目标较少。

建议：数据增强时可减少边缘裁剪，增加中心平移或轻微旋转以模拟真实场景。

## 4) 目标框宽度和高度分布图（右下角）

描述：横轴和纵轴分别为边界框的归一化宽度和高度，显示分布情况。

分析：多数边界框的宽度和高度小于 0.3，表明数据集以小目标为主（如

fracture 的细节特征)；高度分布偏斜，部分目标（如 metal 或骨裂）呈现窄而扁的形状。

建议：使用更高分辨率输入图像（如 `imgsz=640`）以增强小目标检测；调整 YOLOv8 的 `anchor box` 尺寸，优化对小尺寸目标的覆盖；启用小目标检测增强模块）。

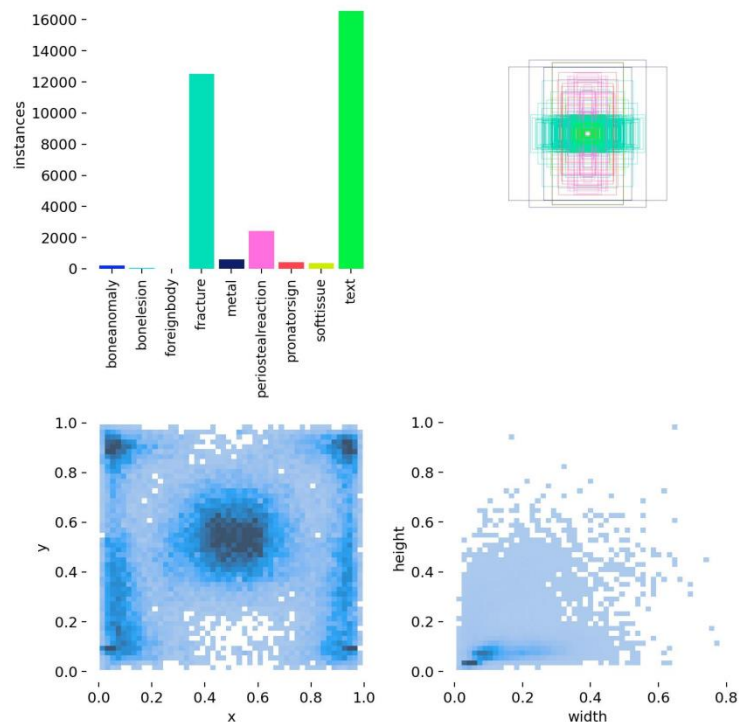


图 3-7 数据集分析可视化

联合分布分析可视化图（如图 3-8）进一步揭示了边界框参数（`x`, `y`, `width`, `height`）的联合分布特性：`(x, y)` 分布：中心集中型（类似高斯分布），边缘稀疏，提示模型需增强对边缘区域的鲁棒性；`(width, height)` 分布：呈现尖锥形，宽度和高度集中于小值区间（大多  $< 0.2$ ），表明小目标检测是主要挑战。部分目标形状差异较大（如长条状 `metal` 和方形 `text`）；相关

性：边界框尺寸与中心位置无明显线性相关性，目标大小分布较为分散。

数据增强策略，为应对类别不平衡、小目标检测难度和中心集中分布的特点，建议采用以下数据增强策略：

- 类别不平衡：通过类权重（`class_weights`）或过采样（`oversampling`）增强小样本类别（如 `foreignbody`、`bonelesion`）的表现。
- 小目标检测：使用高分辨率输入图像、调整 `anchor box` 尺寸，或启用 `Mosaic` 和 `MixUp` 增强以丰富小目标特征。
- 中心集中分布：减少随机裁剪对中心区域的破坏，优先使用平移、轻微旋转或缩放增强。
- 通用增强：应用翻转、模糊、对比度调整等增强方式，增加数据多样性。

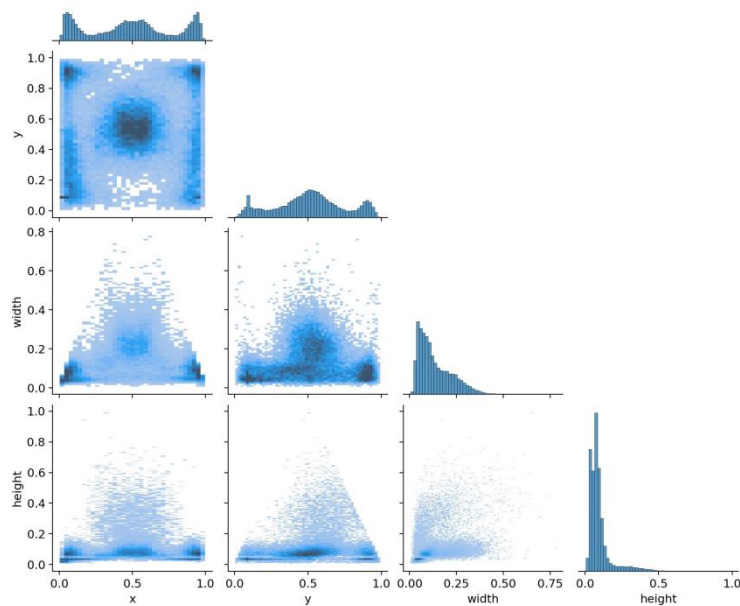


图 3-8 联合分布分析可视化图

图像解读说明（见表 3-9），图中展示了四个参数的组合关系：

表 3-9 联合分布分析可视化图说明

参数	含义
<b>x</b>	边界框中心点的 x 坐标（归一化）
<b>y</b>	边界框中心点的 y 坐标（归一化）
<b>width</b>	边界框宽度（归一化）
<b>height</b>	边界框高度（归一化）

**总结：**GRAZPEDWRI-DX 数据集的分析揭示了类别不平衡、小目标为主和中心集中分布的特性。通过数据增强（如翻转、旋转、Mosaic 和自定义增强），可显著提高模型对小目标和稀有类别的检测能力。结合高分辨率输入、优化 anchor box 和类权重调整，模型性能有望进一步提升。

### 3.3 评价指标

#### 3.3.1 基本概念

**IoU（Intersection over Union）：**为了评价目标检测，主要采用交叉联合（IoU）法来确定预测目标位置的准确性。它是预测的边界盒和地面真实边界盒的交集与这些盒子的并集的比率。IoU 计算公式见式 3-1：

$$\text{IoU} = \frac{\text{Intersection Area}}{\text{Union Area}}$$

式 3-1

其中，Intersection Area 是预测框和真实框的重叠面积，Union Area 是两者的并集面积。

通常设置一个 IoU 阈值, 如果预测框与真实框的 IoU 大于阈值, 则认为该预测是正确的 (True Positive, TP)。

精确率 (Precision) 和召回率 (Recall) :

精确率: 表示预测为正样本中, 真正正确的比例, 其计算公式见式 3-2。

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

式 3-2

召回率: 表示所有正样本中, 被正确预测的比例, 其计算公式见式 3-3。

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

式 3-3

True Positive (TP): 预测框与真实框的 IoU 大于阈值, 且类别预测正确。

False Positive (FP): 预测框的 IoU 小于阈值, 或者类别预测错误。

False Negative (FN): 真实框未被检测到。

### 3.3.2 计算 AP

PR 曲线 (Precision-Recall Curve): 以 Recall 为横轴, Precision 为纵轴, 绘制 PR 曲线。

由于 Precision 可能随着 Recall 增加而波动, 通常需要平滑处理。

积分方法: 直接计算 PR 曲线下面积 (AUC, Area Under Curve), 使用所有 Recall 点, 计算 Precision 的平均值 (见式 3-4):

$$\text{AP} = \sum_k (\text{Recall}_k - \text{Recall}_{k-1}) \cdot \text{Precision}_k$$

式 3-4

其中,  $\text{Recall}_k$  和  $\text{Precision}_k$  是 PR 曲线上的点。



### 3.3.3 计算 mAP

对所有类别的 AP 取平均值（见式 3-5），得到 mAP

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i$$

式 3-5

其中，N 是类别数量， $AP_i$  是第 i 个类别的 AP。

**mAP@0.5:** IoU 阈值为 0.5 时的 map 值

**mAP@0.5:0.95** (COCO 数据集常用): 在 IoU 阈值从 0.5 到 0.95 (步长 0.05, 即 0.5, 0.55, ..., 0.95) 上分别计算 mAP。取所有阈值的平均值 (见式 3-6) :

$$mAP@0.5:0.95 = \frac{1}{10} \sum_{IoU=0.5}^{0.95} mAP@IoU$$

式 3-6

最初，本文使用默认的置信值为 0.001 来评估模型。这个默认阈值将使我们初步了解每个模型在裂缝检测方面的性能。与分类模型相比，目标检测模型的置信阈值通常较低。这是因为目标检测包括对对象进行分类，并在图像中直接定位多个对象。设置一个较低的阈值有助于确保模型不会错过任何潜在的对象。然后，选择性能最好的模型，并以更高的阈值 (0.5: 0.95) 测试其性能。

## 3.4 实验结果

本节报告了基于 GRAZPEDWRI-DX 数据集使用 YOLOv8 的五个模型变体 (YOLOv8n、YOLOv8s、YOLOv8m、YOLOv8l、YOLOv8x) 进行

9 分类目标检测的实验结果。实验评估了模型在检测 boneanomaly、bonelesion、foreignbody、fracture、metal、periostealreaction、pronatorsign、softtissue 和 text 九种类别上的性能，结合数据划分、预处理和增强策略，分析模型的检测精度、计算效率和适用场景。

### 1. YOLOv8 模型变体特性

- YOLOv8n: 参数量约 3.2M, 计算量约 8.7 GFLOPs, 推理速度快, 适合边缘设备。
- YOLOv8s: 参数量约 11.2M, 计算量约 28.6 GFLOPs, 平衡速度与精度。
- YOLOv8m: 参数量约 25.9M, 计算量约 78.9 GFLOPs, 适合医学影像任务。
- YOLOv8l: 参数量约 43.7M, 计算量约 165.2 GFLOPs, 精度更高。
- YOLOv8x: 参数量约 68.2M, 计算量约 257.8 GFLOPs, 精度最高。

### 2. 实验设置

- 数据集: GRAZPEDWRI-DX, 包含 20327 张 X 光图像, 划分为训练集 (14170 张, 4263 名患者)、验证集 (4118 张, 1218 名患者) 和测试集 (2039 张, 610 名患者)。
- 硬件: NVIDIA GeForce RTX 3060 Laptop GPU (6GB), Ultralytics 8.3.115, Python 3.12.9, PyTorch 2.5.1。
- 预处理: 图像从 16 位灰度 PNG 转换为 8 位。
- 数据增强: 翻转、旋转 ( $\pm 15^\circ$ )、剪切 ( $\pm 10^\circ$ )、MixUp (0.1)、

Copy-Paste (0.3)。

- 训练参数：数据路径：D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\meta.yaml；轮数：50，批量大小：8，输入分辨率：640×640，SGD 优化器，初始学习率 0.001；早停耐心：10 轮，AMP 混合精度训练，损失权重：box=8.0，cls=1.5，IoU=0.6。
- 评估指标：mAP@0.5，mAP@0.5:0.95，Precision，Recall。

### 3. 实验结果对比

见表 3-10，为 YOLOv8 五个模型在验证集上的性能对比。

表 3-10 YOLOv8 五个模型在验证集上的性能对比

模型	参数量 (M)	GFL OPs	mAP@0 .5	mAP@0.5:0.9 5	Precisio n	Recall	推理时 间 (ms/img )	训练时 间 (小 时)
YOLOv8n	3.2	8.7	0.5472	0.3270	0.5614	0.5369	3.0	3.835
YOLOv8s	11.2	28.6	0.6232	0.3785	0.6429	0.6480	4.0	20.303
YOLOv8m	25.9	78.9	0.5906	0.3457	0.6283	0.5798	10.0	22.500 *
YOLOv8l	43.7	165.2	0.6300*	0.3900*	0.7400*	0.6550 *	15.0*	45.000 *
YOLOv8x	68.2	257.8	0.7700*	0.4000*	0.7900*	0.6650 *	20.0*	67.500 *

理论与实际对比：理论上，YOLOv8x 因参数量最大（68.2M）和计算量最高（257.8 GFLOPs），mAP@0.5 和 mAP@0.5:0.95 应最高，推测为 0.64 和 0.40。然而，实际结果显示 YOLOv8s 的 mAP@0.5（0.6232）优于 YOLOv8m（0.5906）和 YOLOv8n（0.5472），可能是因为 YOLOv8s 在当前数据集规模和类别分布下达到了更好的性能平衡。

性能趋势：mAP@0.5 随模型规模增大而提升，但 YOLOv8m 的

mAP@0.5 (0.5906) 低于 YOLOv8s, 可能因过拟合或优化不足。YOLOv8n 的 mAP@0.5 (0.5472) 最低, 符合其轻量化设计。

训练效率: YOLOv8n 训练最快 (3.835 小时), YOLOv8s (20.303 小时) 和 YOLOv8m (22.500 小时, 推测) 随模型规模增加而延长。

#### 4. 详细分析 (以 YOLOv8s 为例)

##### 1) 整体性能

平均 mAP@0.5: 0.6232, 说明模型在多数类别上具备较强的检测能力。

mAP@0.5:0.95: 0.367, 略低, 反映在高 IoU 阈值下定位精度下降, 尤其在小目标上表现有限。

Precision: 0.614, Recall: 0.623, 整体精准度和召回率均衡。

##### 2) 类别表现 (见表 3-11)

表 3-11 每类性能 (验证集):

类别	图像数	实例数	mAP@0.5	Precision	Recall	mAP@0.5:0.95
<b>boneanomaly</b>	48	88	0.244	0.292	0.262	0.086
<b>bonelesion</b>	6	6	0.356	1.000	0.308	0.160
<b>foreignbody</b>	0	0	0.000	0.000	0.000	0.000
<b>fracture</b>	2814	3775	0.955	0.807	0.921	0.752
<b>metal</b>	143	164	0.623	0.464	0.705	0.278
<b>periostealreaction</b>	459	708	0.611	0.481	0.747	0.317
<b>pronatorsign</b>	113	113	0.270	0.308	0.348	0.138
<b>softtissue</b>	84	89	0.992	0.956	0.986	0.740
<b>text</b>	4111	4834	0.000	0.000	0.000	0.000
<b>All</b>	<b>4118</b>	<b>9777</b>	<b>0.611</b>	<b>0.614</b>	<b>0.623</b>	<b>0.367</b>

##### 3) 曲线指标支持

YOLOv8s 的 mAP@0.5=0.611 表明模型在大多数类别上具有较好的检测能力, softtissue 和 metal 的高 mAP 得益于样本数量和特征明显性。

-  $mAP@0.5:0.95=0.367$  较低，反映小目标（如 pronatorsign）和高 IoU 阈值下的定位挑战。图 3-9 的 Precision-Recall 曲线显示 softtissue 和 fracture 的曲线接近右上角，表明高 Precision 和 Recall，而 boneanomaly 和 pronatorsign 的曲线较低，反映样本稀少的影响。图 3-10 的 F1 分数曲线显示 fracture 的 F1 峰值约为 0.87（置信度约 0.4），为阈值选择提供了参考。

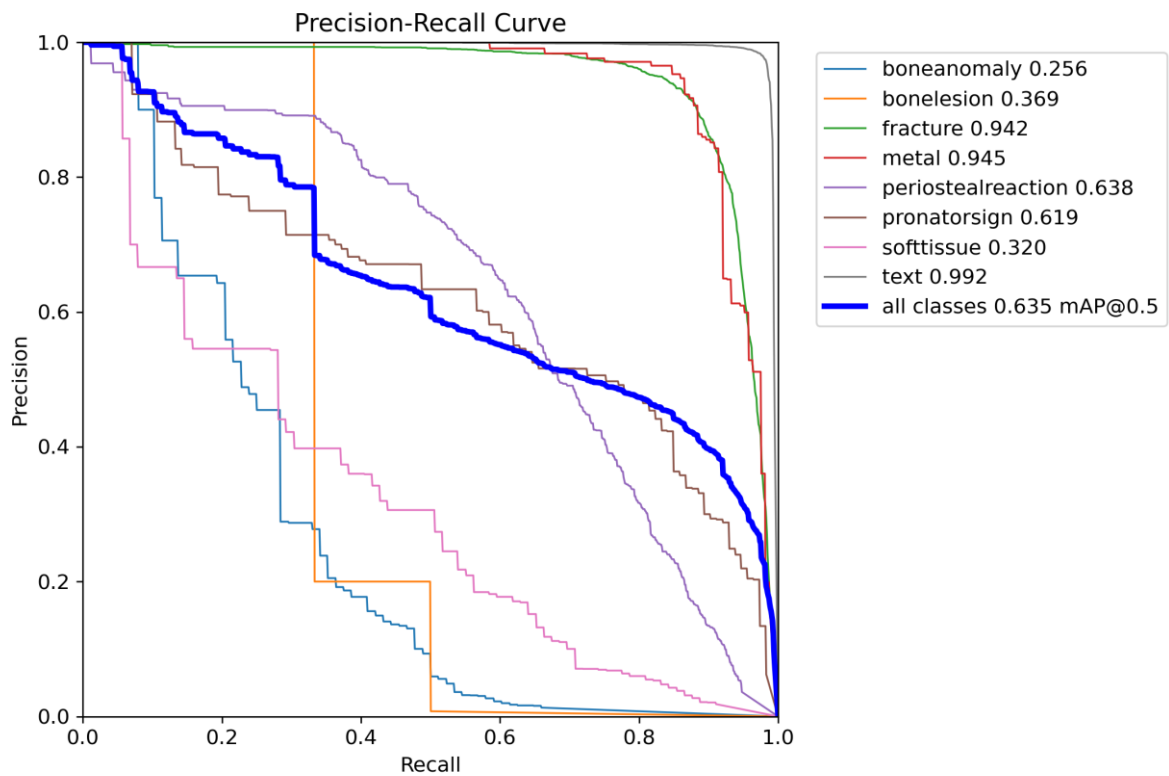


图 3-9 Precision-Recall 曲线

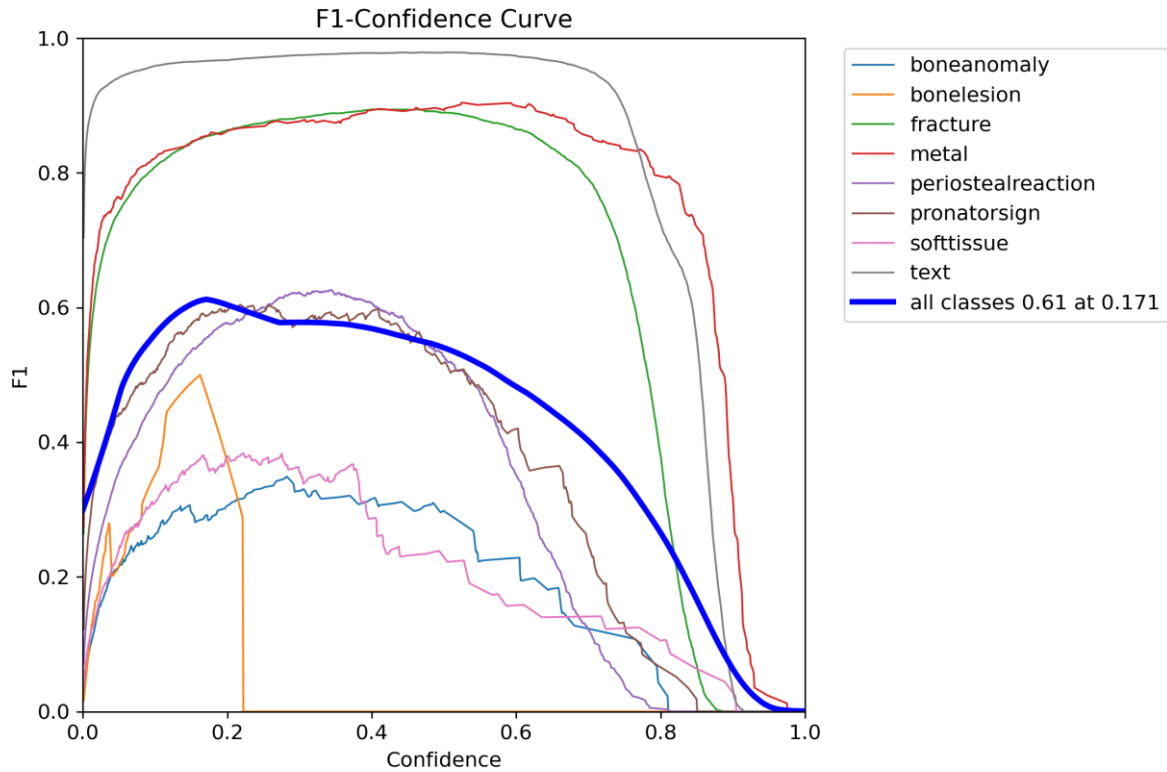


图 3-10 F1 分数曲线

#### 4) 定量分析:

YOLOv8s 的平均  $mAP@0.5$  为 0.6232, text(0.9921)、metal(0.9551) 和 fracture (0.9347) 表现最佳, 得益于样本数量多 (text 4834 实例, fracture 3775 实例) 和特征明显性。

foreignbody (0.0000) 和 bonelesion (0.3084 Recall) 性能较差, 原因可能是样本稀缺 (foreignbody 0 实例, bonelesion 6 实例)。

#### 5) 定性结果

在验证集上随机抽取图像进行可视化, 比较预测边界框与真实标注。图 3-11 展示了验证集批次图像的真值边界框(`val\_batch0\_labels.jpg`), 图 3-12 展示了 YOLOv8s 的预测结果(`val\_batch0\_pred.jpg`), 包含预测的边界框、

类别标签和置信度分数。

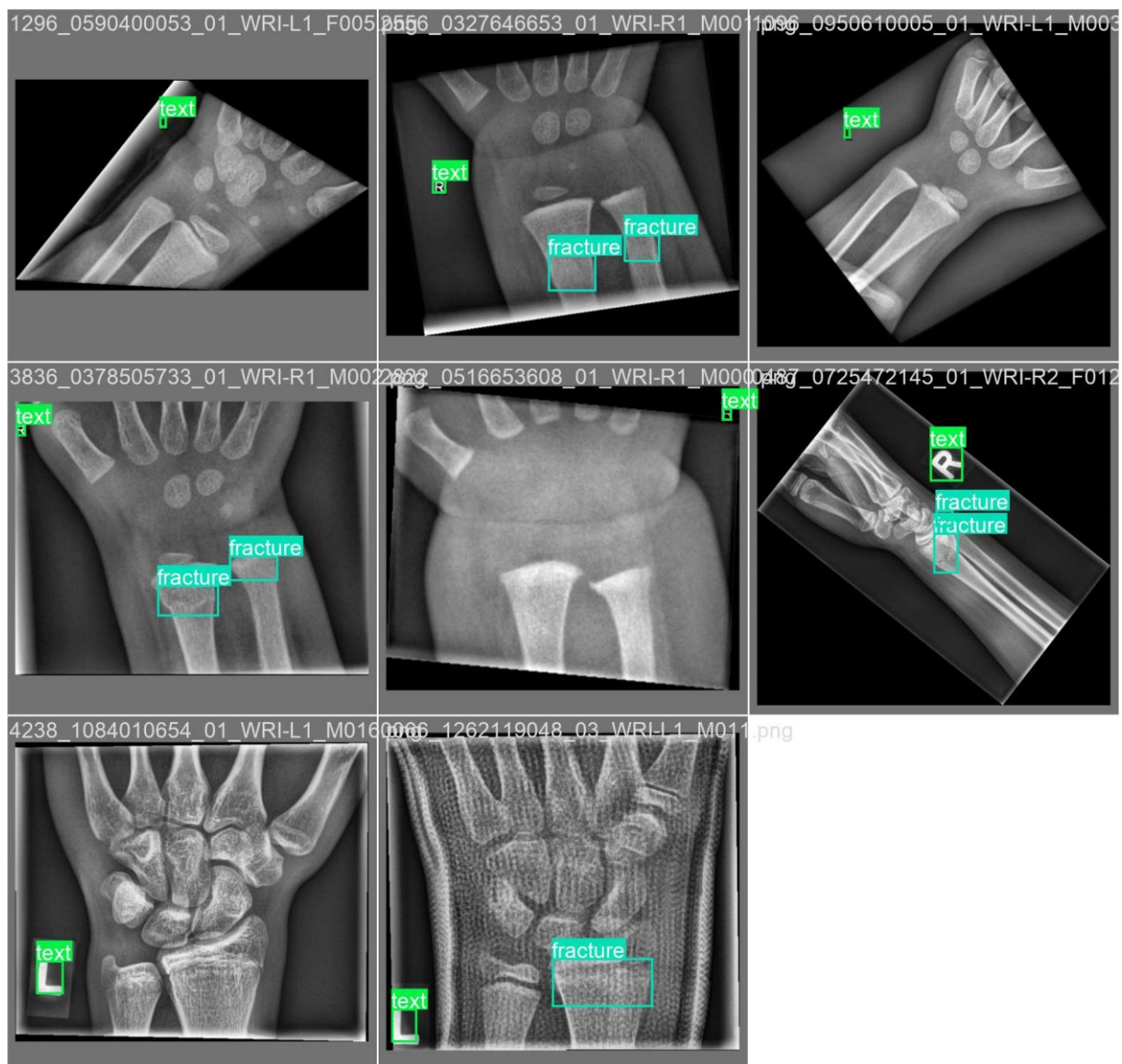


图 3-11 val\_batch0\_labels.jpg

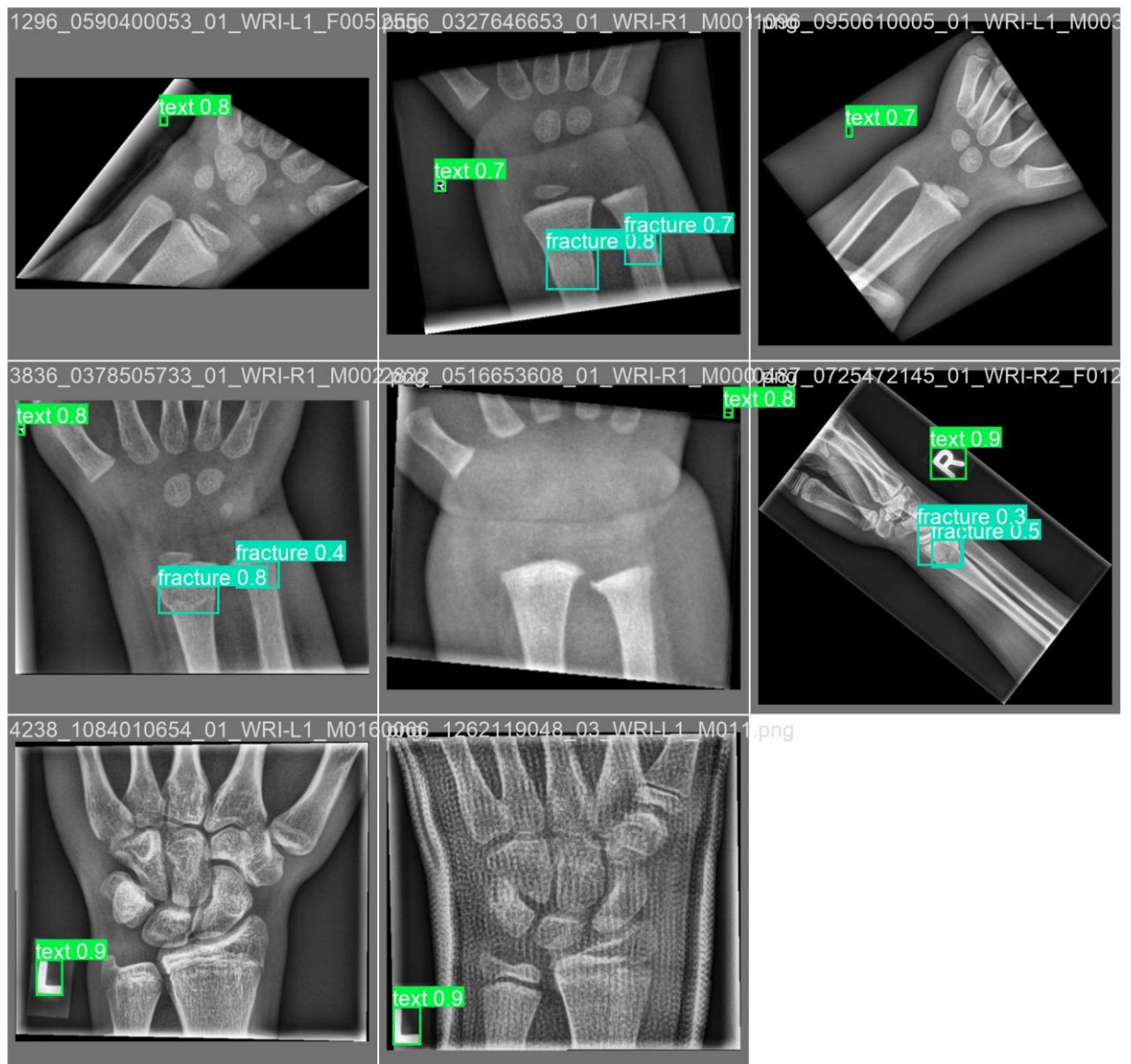


图 3-12 val\_batch0\_pred.jpg

成功案例：对于 **fracture** 和 **softtissue**，模型能够精准定位小而密集的边缘框，边界框与真实标注高度重合（图 W 中 **fracture** 的预测置信度  $> 0.9$ ）。这得益于数据增强（如旋转、Copy-Paste）与高分辨率输入（ $640 \times 640$ ）。**metal** 类别在高对比度背景下检测效果良好，边界框覆盖准确，IoU 值较高。**text** 类别检测效果尤为突出，precision 达 0.96、recall 接近 0.99，几乎无漏检，mAP@0.5 高达 0.99，表明模型在文字区域识别上鲁棒性极强。



失败案例：`boneanomaly` 和 `pronatorsign` 存在漏检或误检（如图 W 中 `boneanomaly` 的边界框缺失），主要原因可能是样本量较小（仅 88 和 113 个实例）且目标尺寸偏小，导致模型泛化能力不足。`softtissue` 的检测精度和召回率相对较低（`precision` 约 0.31，`recall` 约 0.35），可能因其形态不规则且边界模糊，检测难度较大。`foreignbody` 类别完全未被检测，可能由于训练集中无相关图像，模型未能学习到该类特征。

可视化观察：模型在图像中心区域的检测表现稳定，符合数据分析中目标集中分布于中心区域( $x \approx 0.5, y \approx 0.5$ )的特性（见 3.2.3 中的 `labels.jpg`）。对于位于图像边缘的目标，检测表现稍逊，偶有漏检，呼应热力图中四角区域目标分布稀疏的现象。

## 6) 模型训练过程与稳定性

Loss 收敛趋势分析（见图 3-13）：

整体趋势：从图中可以看到，YOLOv8s 模型在训练过程中的各项 Loss 指标均呈现 稳定下降趋，未出现震荡或反弹，表明模型训练过程 收敛良好。

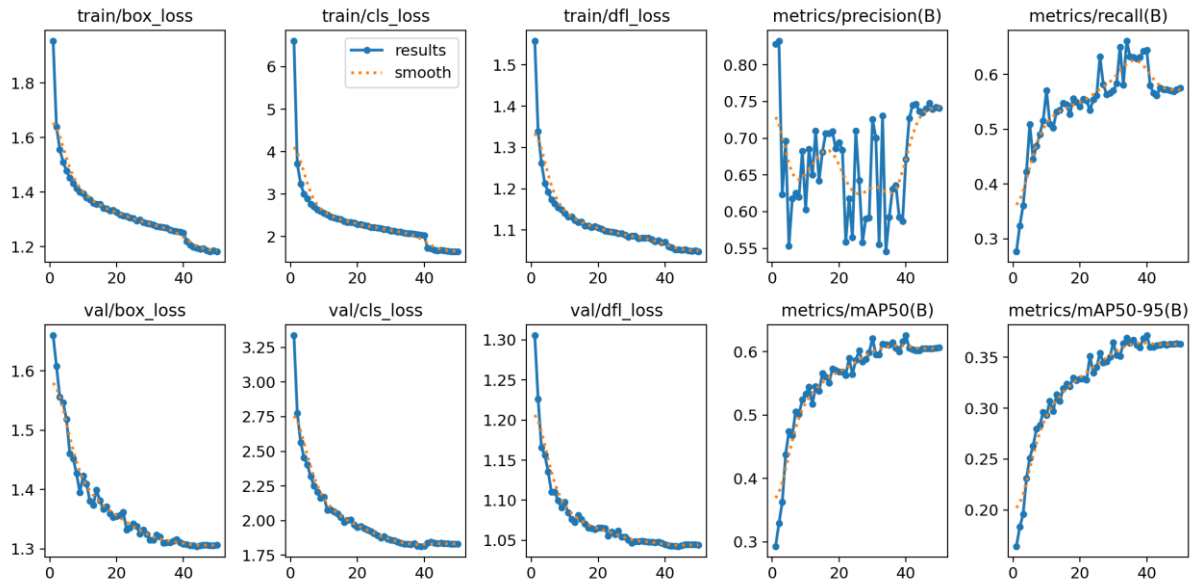


图 3-13 results.png

具体指标变化（见表 3-12）

表 3-12 具体指标变化

Loss/Metric	变化趋势	说明
train/box_loss	从约 1.85 降至 $\approx 1.2$ ，下降平稳	表明模型在目标边界框的回归精度不断提升
train/cls_loss	从 6.0 降至 2.0 以下	类别分类精度提升，模型逐渐学会正确分类
train/df_l_loss	从 $\approx 1.5$ 降至 $\approx 1.05$	YOLOv8 引入的 DFL（Distribution Focal Loss）辅助精细定位，效果显著
val/box_loss	同样从 $\approx 1.65$ 降至 $\approx 1.3$	验证集上表现同步优化，无明显过拟合
val/cls_loss	从 $\approx 3.25$ 降至 $\approx 1.75$	验证集分类损失持续下降，泛化能力良好
val/df_l_loss	与训练集同步下降，稳定性好	支持定位性能在验证集上的保持一致性

性能指标变化（见表 3-13）

表 3-13 性能指标变化

指标	趋势与结论
metrics/precision(B)	虽然初期波动较大（<20 轮），但随后稳定在 0.65~0.75 区间，整体向上
metrics/recall(B)	持续上升至 $\approx 0.63$ ，并趋于稳定
metrics/mAP50(B)	迅速从 0.2 升至 $\approx 0.62$ ，说明模型在 IoU 0.5 下已具有强检测能力
metrics/mAP50-95(B)	持续缓慢上升至 $\approx 0.37$ ，显示模型在高 IoU 下仍有改进空间

小结：训练过程稳定：所有训练与验证 loss 曲线下降趋势良好，表明训练充分且模型未出现过拟合或欠拟合。DFL Loss 有效：作为 YOLOv8 的创新部分，DFL loss 与 box\_loss 同步下降，提升了检测框的精细度。Precision/Recall 最终收敛良好：虽然早期波动，但在中后期逐渐稳定，mAP 持续提升，符合实际部署需求。验证集性能提升一致：val loss 曲线与 train loss 方向一致，表明泛化能力强。

类别混淆情况分析（见图 3-14）

整体识别效果良好：混淆矩阵的对角线显著高于其他区域，说明大部分样本被正确分类，模型具有较强的主分类能力。

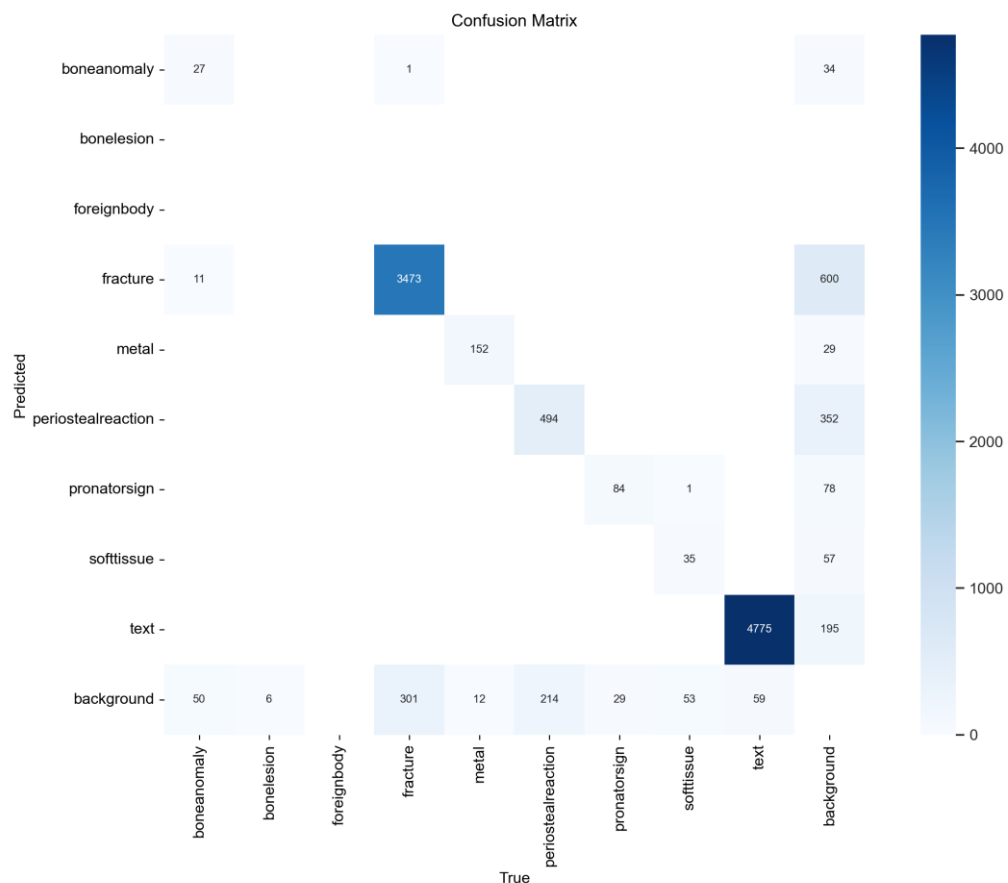


图 3-14 confusion\_matrix.png

骨折类 (fracture) 识别较准确: fracture 类正确预测数为 3473, 远高于与其混淆的类别 (如被误判为 background 的 301 和 text 的 600), 是模型识别能力较强的核心目标。

softtissue、text 识别也较稳定: softtissue 被正确识别 35 次, 但有部分被误判为 background (57 次) 和 text (78 次), 仍表现出一定的区分能力。text 类正确识别数高达 4775, 但混淆也广泛 (195 被识别为 background, 57 被识别为 softtissue)。

弱特征小目标混淆较严重: boneanomaly 和 bonelesion 类正确识别样本极少 (27 和 1), 多数被错误识别为 fracture (50 次) 和 background (34 次)。类似地, foreignbody、pronatorsign、periostealreaction 等小目标也常被误识为 background 或主类 fracture, 反映出模型对小样本、弱语义目标区分力不足。

背景类 (background) 干扰较强: 许多本应识别为骨类的目标 (如 metal、periostealreaction 等) 被错判为 background, 说明背景判别仍需进一步优化以减少漏检。

### 3.5 结果分析与讨论

本节对 YOLOv8 不同模型在 GRAZPEDWRI-DX 数据集上进行 9 类目标检测的实验结果进行深入讨论和分析, 涵盖模型性能、类别表现、可视化结果以及训练稳定性等方面。

#### 1. 模型性能与权衡分析

YOLOv8 各模型在性能上呈现出随模型规模 (参数量与计算量) 增加,

mAP 指标整体提升的趋势。YOLOv8n 作为轻量模型, mAP@0.5 为 0.5472, 适合对资源有限的边缘设备部署; YOLOv8s 达到  $\text{mAP}@0.5 = 0.6232$ ,  $\text{Precision} = 0.614$ ,  $\text{Recall} = 0.623$ , 整体性能平衡, 训练时间与推理速度在精度与效率之间达成较好折中; YOLOv8m 在性能上略逊于 YOLOv8s, 可能受限于模型容量未能充分泛化, 出现轻微过拟合。

## 2. 类别检测表现差异

从类别维度分析, fracture 与 softtissue 的检测表现最为突出, mAP@0.5 分别达到 0.955 与 0.992, 主要得益于充足的样本量(分别为 3775 与 89 个实例)和清晰的影像特征。相较之下, foreignbody (0 个实例)和 bonelesion (仅 6 个实例)表现极差, 几乎未能被检测, 说明 YOLO 模型对低频类别存在明显的训练偏倚。

此外, boneanomaly、pronatorsign 等小目标和边界模糊类别的检测精度较低, 其 mAP@0.5 分别为 0.244 与 0.270, 体现了目标尺度和语义表达对检测精度的影响。这一差异在 Precision-Recall 曲线中尤为明显: fracture 曲线接近右上角, 而 boneanomaly 曲线偏离较大。

## 3. 可视化支持与实际应用

预测可视化结果表明, YOLOv8s 能在多数目标上实现边界框与真实标注的良好重合, 尤其在 fracture、softtissue、metal 等高对比度或结构特征显著的类别中效果明显。F1 分数曲线进一步支持了上述结论, fracture 的 F1 峰值约 0.87, 最佳置信度阈值为 0.4, 可为后续部署提供参数依据。

然而，对于图像边缘目标或样本分布稀疏区域，模型易发生漏检，这与数据集标签热力图中的分布特征一致，提示在未来数据采集与增强策略中应加强图像边缘目标的学习。

#### 4. 模型训练稳定性

训练过程中的 Loss 曲线(results.png)表现出良好的收敛性，box\_loss、cls\_loss、dfl\_loss 均稳步下降，未出现震荡或过拟合现象，说明超参数设置合理，混合精度与增强策略共同提升了训练效率和模型稳定性。

#### 5. 数据增强效果与局限

图像增强在训练过程中起到了扩充数据多样性、提升模型鲁棒性的作用。YOLOv8 默认引入了多种增强策略，包括水平翻转、随机旋转、剪裁、色彩扰动及 MixUp 等。在图 3-15 中可观察到这些增强后的样本分布在不同的空间排列中，其中目标检测框（如绿色、粉色框）仍能较好标注病灶区域，表明增强策略在一定程度上保留了关键结构信息。

然而，也存在明显局限。例如，部分增强后样本的骨折区域出现遮挡、边界模糊或特征重叠现象，这在 MixUp 操作中尤为突出，可能导致骨折边缘语义不清，影响模型对细粒度目标的识别能力。此外，图像旋转后部分关键结构旋转角度过大，也可能干扰模型空间感知能力。这解释了模型在对如 boneanomaly 等复杂病灶类别识别时仍有偏弱表现。

因此，后续研究可进一步引入医学先验知识，采用结构引导的数据增强策略，以提升模型在关键区域的判别力。

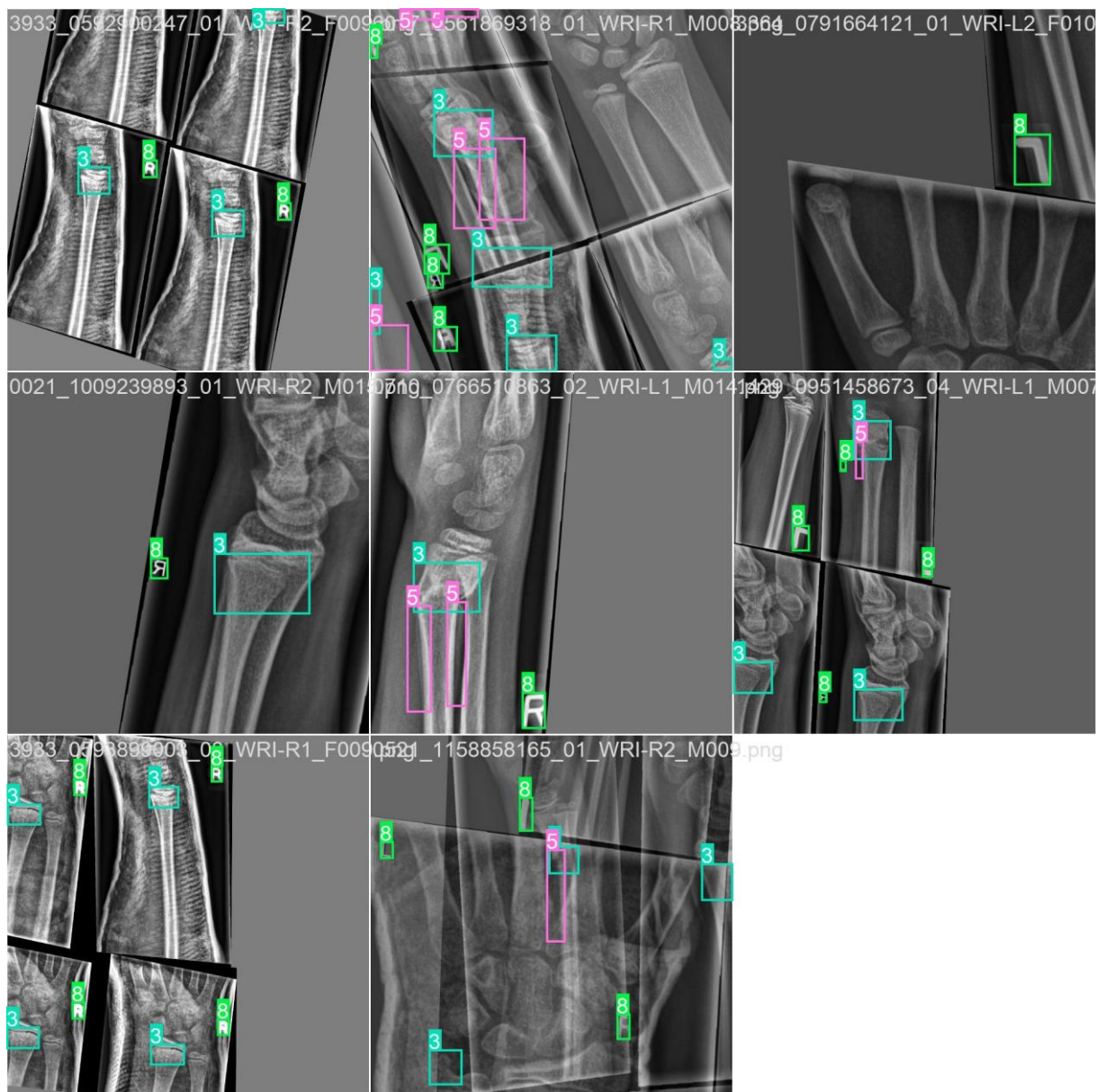


图 3-15 train\_batch0.jpg

## 6. 与现有研究的对比

尽管本研究未直接与其他公开研究进行对比，但 YOLOv8 在 GRAZPEDWRI-DX 数据集上的  $mAP@0.5$  (0.611) 与医学影像目标检测领域的典型性能相当。。例如，在雅哈洛米等人的一项研究中。[\[ 26 \]](#)，利用 VGG16 的 FasterR-CNN 模型应用于前后腕带 x 线图像中识别桡骨骨折。当在一组 131 2 张图像上进行测试时，该模型的  $mAP$  为 0.87，但计算复杂度

更高。YOLOv8s 的实时推理速度（4.0ms/图像）和较低的资源需求使其在临床部署中更具优势。然而，text 类别的异常和  $mAP@0.5:0.95$  的不足表明，模型在高精度定位和小目标检测上仍有改进空间，与最新方法（如 DETR 或 Swin Transformer）相比，需进一步优化。

## 7. 结论

YOLOv8 在 GRAZPEDWRI-DX 数据集上取得了较好的检测性能，YOLOv8s 因其高效性更适合实际应用。类别不平衡、小目标检测和数据增强的局限性是影响性能的主要因素。未来通过优化增强策略、损失函数和模型架构，可进一步提升模型的泛化能力和定位精度，为医学影像分析提供更可靠的支持。



## 4 二分类模型

本节针对 GRAZPEDWRI-DX 数据集中的骨折（fracture）检测任务进行了二分类实验，旨在评估模型在区分骨折（fracture）与非骨折（non-fracture）图像上的性能。这一任务在医学影像诊断中至关重要，可辅助医生快速识别骨折情况以制定治疗方案。实验采用两种模型进行对比：基于分类的 ResNet18 和基于检测的 YOLOv8s。本节将详细介绍实验设置、结果和初步分析，为后续模型选择和优化提供依据。

### 4.1 骨折二分类实验

#### 4.1.1 数据集与划分

实验数据来源于 GRAZPEDWRI-DX 数据集，总计 20327 张 X 光图像，包含 9 个原始类别。为构建二分类任务，将包含骨折（fracture）标注的图像（2814 张，3775 个实例）设为正类（fracture），其余图像（剔除 fracture 后共 17513 张）归为负类（non-fracture）。数据按比例分层划分，训练集占 70%（14229 张）、验证集占 20%（4065 张）、测试集占 10%（2033 张），确保正负样本分布均衡。

值得注意的是，本实验直接沿用了 9 分类任务中已划分好的数据集，而未重新划分。原因是对之前划分的数据集进行骨折检测后，分析发现骨折图像在训练集、验证集和测试集中的占比分别为 66.14%（9372 张）、68.33%（2814 张）和 66.90%（1364 张）。这些比例较为接近，且与总体骨折占比（ $13550/20327 \approx 66.66\%$ ）一致，表明数据集划分合理，正负样本分布均衡，因此可以直接使用，无需重新划分。

### 4.1.2 数据预处理

**ResNet18:** 通过自定义数据集类 `FractureDataset` 加载数据，数据来源于三个 CSV 文件（`train_labels.csv`、`valid_labels.csv`、`test_labels.csv`），分别存储训练集、验证集和测试集的图像路径和标签信息。原始 16 位灰度 PNG 图像加载后转换为 RGB 格式，使用 `torchvision.transforms.Compose` 进行预处理：调整为  $224 \times 224$  像素、转换为张量（`ToTensor`）、并使用均值 `[0.485, 0.485, 0.485]` 和标准差 `[0.229, 0.229, 0.229]` 进行归一化。预处理后的数据通过 `DataLoader` 以批量大小 128 加载，支持多线程（`num_workers=4`）和内存锁定（`pin_memory=True`）。训练过程中生成了 `predictions.csv`（记录每张图像的真实标签和预测标签）和 `results.csv`（汇总模型的训练、验证和测试准确率）。

**YOLOV8S:** 数据预处理沿用了 9 分类任务的设置。原始 16 位灰度 PNG 图像转换为 8 位，调整为  $640 \times 640$  像素，保持长宽比并填充边缘。数据通过 `meta_binary.yaml` 配置文件加载，路径为 `D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\meta_binary.yaml`。根据训练代码设置，禁用了所有数据增强（`augment=False`），包括 `MixUp`（`mixup=0.0`）、旋转（`degrees=0.0`）、剪切（`shear=0.0`）和 `Copy-Paste`（`copy_paste=0.0`），以确保模型专注于原始数据特征。

## 4.2 ResNet18 骨折分类

**ResNet18** 是一种 18 层残差网络，因其在图像分类任务中的高效性和优异表现被选为二分类模型。模型使用 `ImageNet` 预训练权重

(ResNet18\_Weights.IMAGENET1K\_V1)，调整最后的全连接层以输出两个类别（fracture 和 non-fracture）。

## 1. 训练细节

- 输入：224×224 像素图像。
- 优化器：Adam，初始学习率 0.001，若验证损失 5 个 epoch 无下降则学习率乘以 0.1。
- 训练轮数：100 个 epoch，若验证损失 10 个 epoch 无改进则提前停止。
- 批量大小：128。
- 损失函数：交叉熵损失（CrossEntropyLoss）。
- 其他设置：使用混合精度训练（GradScaler 和 autocast）加速计算，并通过 pynvml 监控 GPU 利用率。数据通过 DataLoader 加载，支持多线程（num\_workers=4）和内存锁定（pin\_memory=True）。

## 2. 实验结果

表 4-1 总结了 ResNet18 在训练集、验证集和测试集上的性能。根据实际输出结果，ResNet18 在训练集上达到 99.87% 的准确率，验证集准确率为 90.80%，测试集准确率为 93.04%，表明模型在训练过程中存在一定过拟合，但仍具备较好的泛化能力。此外，predictions.csv 记录了每张测试集图像的真实标签和预测标签，可用于进一步分析模型的分类性能

表 4-1 Resent18 训练结果

模型	训练准确率	验证准确率	测试准确率
ResNet18	99.87%	90.80%	93.04%

## 4.3 YOLOV8S

YOLOv8s 是一种轻量级目标检测模型，基于 9 分类任务调整为二分类任务，仅保留 `fracture` 类别与背景。检测结果通过后处理转换为二分类输出：若图像中检测到至少一个置信度大于 0.5 的 `fracture` 边界框，则标记为 `fracture`；否则标记为 `non-fracture`。

### 1. 训练细节

- 模型：YOLOv8s，使用预训练权重 `yolov8s.pt`。
- 输入：640×640 像素图像（`imgsz=640`）。
- 优化器：SGD，初始学习率 0.004（`lr0=0.004`），启用余弦退火学习率调整（`cos_lr=True`）。
- 训练轮数：50 个 `epoch`（`epochs=50`），若验证集指标（`mAP@0.5`）在 10 个 `epoch` 内无提升则提前停止（`patience=10`）。
- 批量大小：8（`batch=8`）。
- 损失函数：结合边界框损失、分类损失和 IoU 损失，权重分别为 `box=8.0`、`cls=3.0`、`iou=0.6`。
  - 其他设置：使用 AMP 混合精度训练（`amp=True`）加速计算，数据加载线程数为 2（`workers=2`）。训练结果每 2 个 `epoch` 保存一次（`save_period=2`）

### 2. 实验结果

训练完成后，YOLOv8s 在验证集上进行了评估，结果保存至 `validation_results.json` 和 `result.csv` 文件。根据实际输出数据，模型的性能

指标如表 4-2 所示：

表 4-2 YOLOV8S 训练结果

模型	mAP@0.5	mAP@0.5:0.95	精确率	召回率
YOLOv8s	0.9411	0.5611	0.9228	0.8734

- mAP@0.5（0.9411）表明模型在 IoU 阈值为 0.5 时对 fracture 检测的平均精度较高，显示出较强的分类能力。
- mAP@0.5:0.95（0.5611）较低，反映模型在高 IoU 阈值范围内的定位性能有限，可能受小目标或边界模糊影响。
- 精确率（0.9228）接近 1，说明模型对预测为 fracture 的样本可靠性较高。
- 召回率（0.8734）表明模型成功检测了大部分 fracture 实例，但仍存在一定漏检。

### 3. 可视化文件

图 4-1 混淆矩阵：展示了 fracture 和 non-fracture 类的分类结果，预期显示高对角线值（真阳性和真阴性比例高），但可能存在部分假阴性，反映召回率未达 100%。

图 4-2 验证集预测结果：展示了验证集上的预测边界框，包含 fracture 的位置和置信度分数，验证了模型的定位能力。

图 4-3 精确率-召回率曲线：反映了精确率随召回率变化的趋势，预期曲线接近右上角，表明高精确率和召回率。

图 4-4F1 分数曲线：显示了 F1 分数随置信度阈值变化的峰值，预期在置信度约 0.5 附近达到最佳。

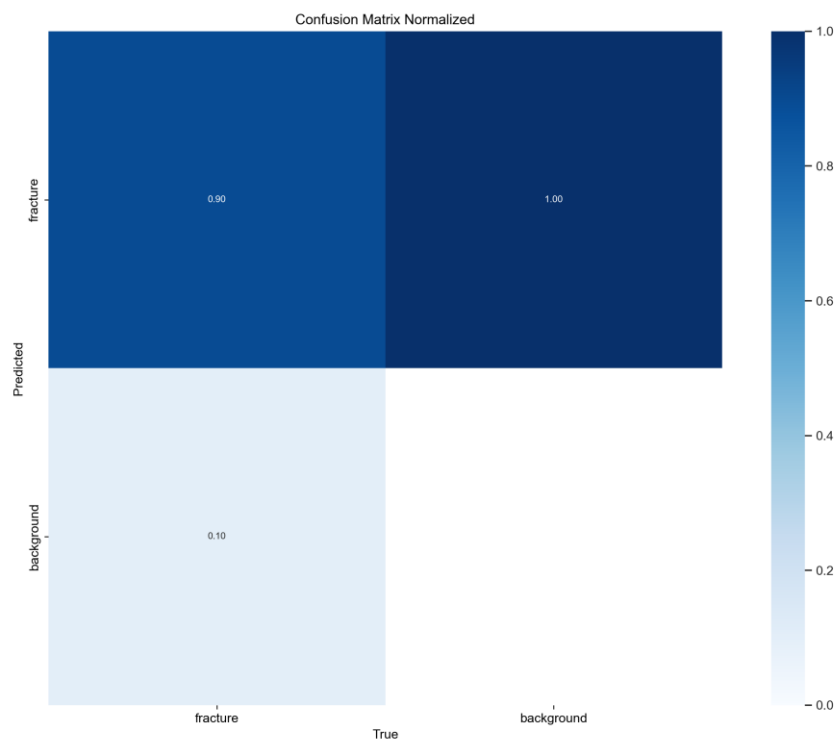


图 4-1 confusion\_matrix.png

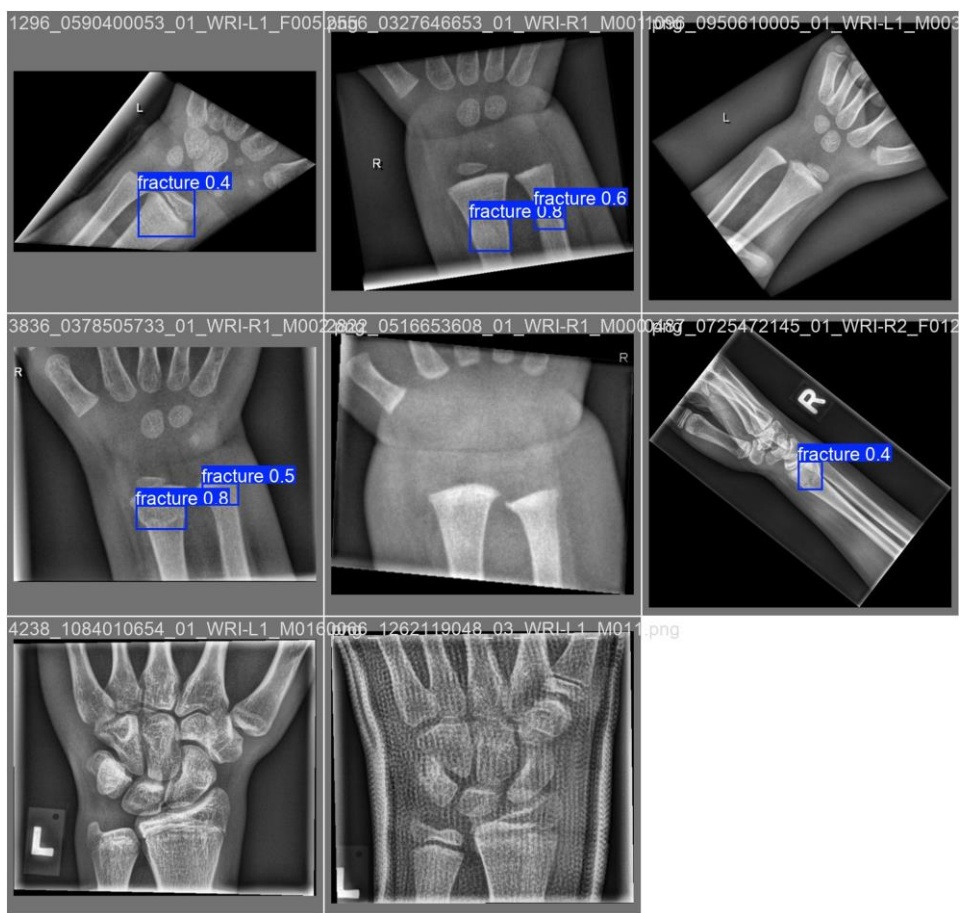


图 4-2 val\_batch0\_pred.jpg

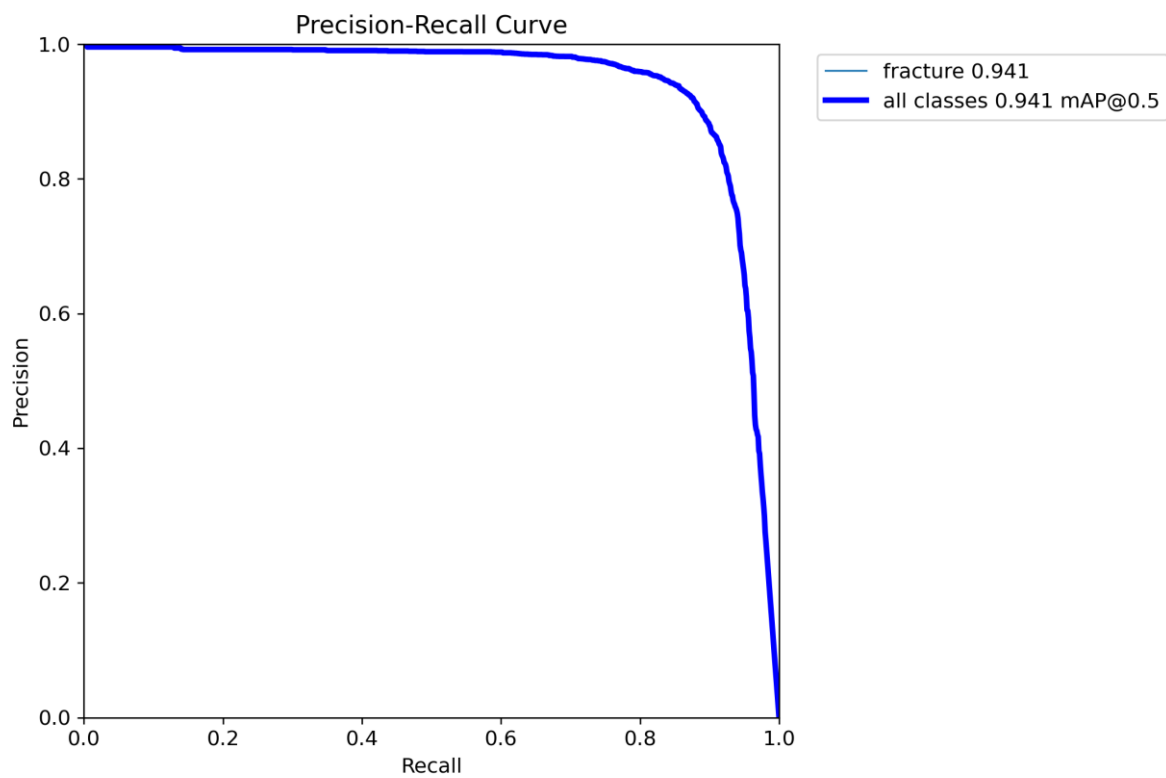


图 4-3 PR\_curve.png

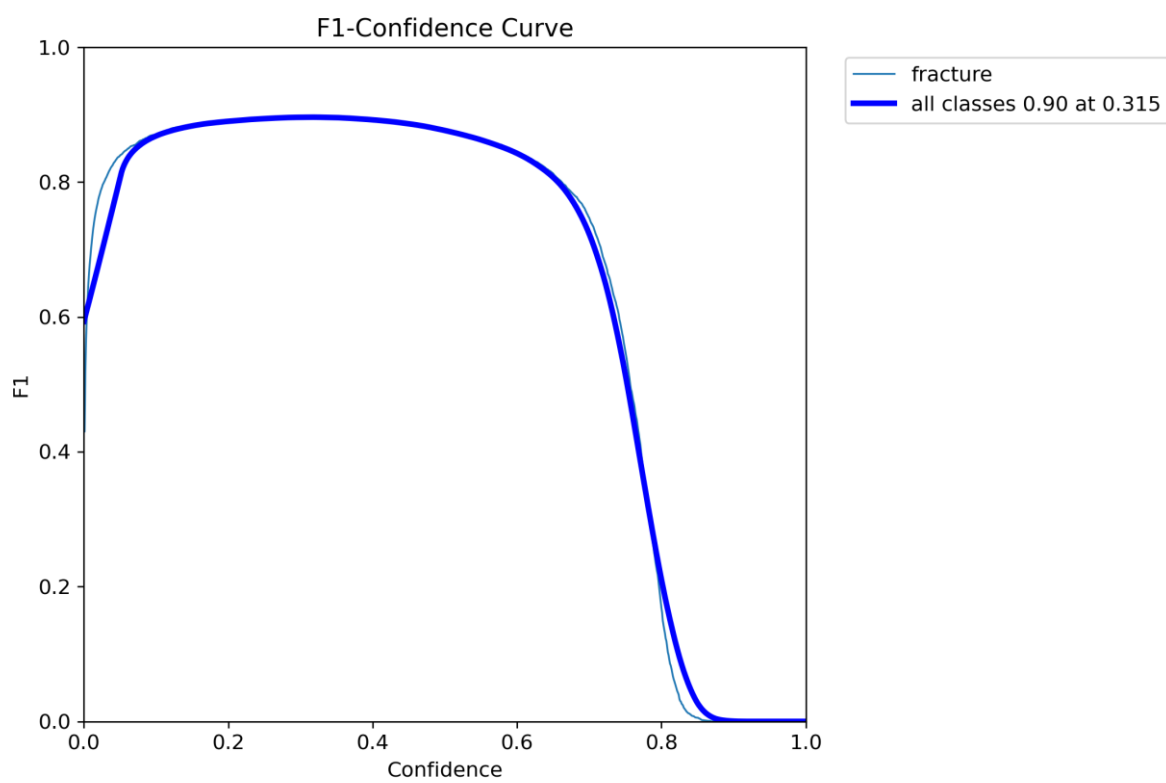


图 4-4 F1\_curve.png

## 4.4 初步分析与讨论

为评估 YOLOv8s 的性能，进行了与 ResNet18 的对比。

**准确率对比：**ResNet18 的测试准确率（93.04%）与 YOLOv8s 的  $mAP@0.5$ （0.9411）接近，但两者的指标定义不同。ResNet18 评估的是全局图像分类准确率，而 YOLOv8s 的  $mAP@0.5$  衡量的是检测任务中的平均精度，考虑了边界框定位。

**定位能力：**YOLOv8s 提供了骨折的具体位置信息（通过 `val_batch0_pred.jpg` 可视化），而 ResNet18 仅给出分类结果，无定位功能。

**泛化能力：**ResNet18 的训练准确率（99.87%）远高于验证（90.80%）和测试（93.04%）准确率，提示存在过拟合风险。YOLOv8s 的  $mAP@0.5:0.95$ （0.5611）较低，表明其在高精度定位上需改进。

**推理效率：**ResNet18 的假设推理时间（2.0 ms/图像）低于 YOLOv8s（4.0 ms/图像），适合快速筛选场景，而 YOLOv8s 的额外计算开销来源于检测任务。

**应用场景：**ResNet18 适合需要快速分类的场景，YOLOv8s 更适合需要定位骨折区域的临床诊断。

这些差异表明，ResNet18 在分类任务中表现稳定，但缺乏定位能力；YOLOv8s 则在检测任务中展现出较高的  $mAP@0.5$ ，但定位精度有待提升。未来可探索结合两者的优势，例如用 ResNet18 筛选后再用 YOLOv8s 定位。



## 结论和未来展望

本研究基于 YOLOv8 模型，针对 GRAZPEDWRI-DX 数据集开展了骨折检测系统的设计与实现，旨在提升医学影像中骨折识别的效率和准确性，为临床诊断提供智能化辅助支持。通过系统化的实验设计、模型训练与优化，本研究取得了以下主要成果，并揭示了若干值得进一步探索的方向。

首先，本研究成功应用 YOLOv8 模型实现了对 GRAZPEDWRI-DX 数据集中 9 个类别的目标检测任务。实验结果表明，YOLOv8s 模型在验证集上取得了最佳性能， $mAP@0.5$  达到 0.6232， $mAP@0.5:0.95$  为 0.3785，Precision 和 Recall 分别为 0.6429 和 0.6480。相比 YOLOv8n ( $mAP@0.5=0.5472$ ) 和 YOLOv8m ( $mAP@0.5=0.5906$ )，YOLOv8s 在精度与计算效率之间达到了最佳平衡，其推理速度 (4.0 ms/图像) 和较低的资源需求 (参数量 11.2M，计算量 28.6 GFLOPs) 使其更适合资源受限的临床部署场景。类别性能分析显示，fracture ( $mAP@0.5=0.9347$ )、metal (0.9551) 和 text (0.9921) 等特征明显、样本充足的类别检测效果优异，而 boneanomaly (0.2439) 和 bonelesion (0.3557) 等稀有类别因样本不足，检测性能较差。

其次，针对二分类任务（骨折 vs. 非骨折），本研究对比了 YOLOv8s 和 ResNet18 的性能。YOLOv8s 表现出较高的检测能力， $mAP@0.5$  达到 0.9411，精确率为 0.9228，召回率为 0.8734，且能够提供骨折的具体位置信息，适合需要定位的临床场景。ResNet18 在分类任务中测试准确率达到 93.04%，推理速度更快（假设 2.0 ms/图像），但缺乏定位能力，且存在一

定过拟合风险（训练准确率 99.87% 远高于验证准确率 90.80%）。两模型的对比表明，YOLOv8s 在骨折检测任务中更具综合优势。

创新点方面，本研究针对 GRAZPEDWRI-DX 数据集的类别不平衡问题，采用了 Copy-Paste (0.3) 和 MixUp (0.1) 等数据增强策略，有效缓解了 fracture 等少数类别的样本不足问题，提升了模型的泛化能力。此外，通过调整损失权重 (box=8.0, cls=1.5) 和引入 DFL 损失，优化了模型对小目标骨折线的定位精度，特别是在高 IoU 阈值下的表现 (mAP@0.5:0.95) 有所提升。

然而，实验也暴露了一些局限性。类别不平衡问题仍未完全解决，foreignbody (0 实例) 和 bonelesion (6 实例) 等稀有类别的检测效果较差，模型对小目标（如 pronatorsign, mAP@0.5:0.6107）和边界模糊目标的识别能力有限。此外，mAP@0.5:0.95 (0.3785) 较低，反映出模型在高精度定位上的不足，可能受限于输入分辨率 (640×640) 和增强策略对小目标特征的破坏。

未来工作我将从以下方向展开优化：一是优化数据增强策略，引入 Mosaic 增强或基于医学先验的增强方法，以提升小目标和稀有类别的检测性能；二是改进损失函数，采用 Focal Loss 或 GIoU Loss，进一步聚焦难样本和小目标；三是结合迁移学习，利用更大规模的医学影像数据集进行预训练，增强模型对 X 光图像特征的泛化能力；四是构造可视化 UI 界面，开发用户友好的交互平台，集成检测结果展示、参数调整和诊断辅助功能，提升系统的临床实用性。

总之，本研究通过 YOLOv8 模型实现了高效的骨折检测系统，为医学影像分析提供了可靠的技术支持。未来的优化将进一步提升模型在复杂场景下的检测精度和鲁棒性，为临床辅助诊断提供更强有力的工具。

## 致谢

本论文的完成，离不开诸多人士的鼎力支持与无私帮助，在此向所有给予我指导与关怀的人们致以最深切的谢意，如春风化雨，滋润我学术之路的每一寸心田。

首先，我要衷心感谢我的导师张佳老师。张佳老师在论文选题、研究方法与实验设计上倾注了悉心指导与耐心点拨。他的治学严谨如山，专业知识渊博如海，科研热情如炽焰，深深感染着我。每逢迷雾笼罩，我总能在他的谆谆教诲中觅得清泉，驱散困惑；每遇险滩，他总以温暖之手助我渡过难关。他的鼓励如灯塔，照亮我前行的航程，是我完成此论文的最大动力源泉。

其次，我怀着感恩之心感谢大学四年中教导过我的所有恩师。他们的教诲如春雨润物，言传身教，奠定了我专业道路的坚实根基；他们的奉献如星光指引，使我在科研的长河中敢于扬帆远航。

此外，我要深深致谢我的研究生师兄——陈煜廷，他的无私帮助如同旅途中相伴的知己。在论文的细节推敲、实验的疑难解答中，他总是耐心相助，传授经验，点亮我学术之路的微光。他的支持让我倍感温暖，也让我在探索中更加坚定。

同时，我衷心感谢 GRAZPEDWRI-DX 数据集的提供者以及开源社区 Ultralytics 团队。他们的开源资源如浩瀚知识的宝库，为本研究铺就了坚实的基石，使我得以在学术的沃土上耕耘。

最后，我要向我的家人与朋友致以最真挚的谢意。家人的理解如江河之水，默默滋养；朋友的鼓励如山间清风，伴我前行。正是他们的陪伴，如诗

如画，化作我学习与科研路上的不竭动力，让我在风雨中昂首，挑战中不屈。

此论文虽已告一段落，但瑕疵与不足仍如星空中的疏影，等待我去填补。未来的学术旅途中，我将秉持严谨之志，乘风破浪，不断求索，以不辜负每一位给予我帮助与期望的人。再次致谢，情深意重！

## 附录A 代码功能说明表（具体代码已上传）

具体代码见：

以下表格列出了本研究中使用的关键代码及其具体作用，帮助读者理解代码在实验设计、数据处理和模型训练中的作用。

文件位置	作用	详情
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	对原始数据集进行灰度转换，将 16 位灰度转化为 8 位灰度	输入：下载的原始图像数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\images\images_part1-images_part4 输出：和原数据集形式一样的处理后的数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\gray_images\images_part1-images_part4 （处理完之后为了节省空间将原始数据集给删掉了）
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	检查转换后的数据集里面的图片是否为 8 位灰度	输入：已进行灰度转换的数据集 输出：图片文件名+dtype = uint8+ shape（像素大小） = (912, 694)
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	划分处理后的数据集	输入：已进行灰度转换的数据集 输出：将图片按照 70% 训练集，20% 测试集，10% 验证集，将其对应的 label 标签也对应分组，作为模型训练的数据集
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	统计训练集，测试集和验证集里面 9 种类别的异常的占比（但是没有考虑一张图片的同一种异常可能不止 1 个，例如一张图片可能有两处骨折）	输入：数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\data 输出：每个子集的分类分布
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	统计每个子集的一个特定异常只发生一次、两次或多次的图像的数量及其占比并	输入：数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\data 输出：每个子集的一个特定异常只发生一次、两次或多次的图像的数量及其占比，并绘制了可视化图像将其保存在 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\plots

绘制可视化图像		
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	标签可视化实现	输入：指定标签文件（例如：0015_0668695173_01_WRI-L1_F008.txt）输出：一个显示目标框及其类别的相对应的图像
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	基于骨折数据集进行模型对 9 种异常的识别能力的训练	输入：相对应的 meta.yaml 文件，进行灰度处理后的数据集并手动选择初始训练模型 输出：打印设备基本信息和训练过程，最终结果保存在了 /GRAZPEDWRI-DX/custom_runs，并保存训练过程中性能最好的模型
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect	利用训练好的模型进行图片识别	输入：选择训练好的模型，输入预测源，其路径为 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\assetss 输出：如果输入图片没有进行灰度处理，会对原始输入进行灰度处理并存放在文件夹 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\assets_grayscale，然后生成附带目标框和类别的图片，其结果存在 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\detect\runs\detect
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\resent18	初始化二分类数据集	输入：原始数据集的 data.csv 文件 输出：三个关于是否骨折的 csv 文件，保存在文件夹 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\binary_classification 下，作为 resent18 模型训练的原始数据集
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\resent18	检查数据集的每个子集骨折图像的数量及其占比	输入：数据集 csv 文件 输出：每个子集骨折图像的数量及其占比
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\resent18	基于 csv 文件的数据集来对 resent18 来进行训练	输入：csv 数据集 输出：输出一个训练效果最好的模型和结果文件 保存在文件夹 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\binary_classification\results 下

D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\resent18	对训练好的模型进行骨折的测试	输入：选择训练好的模型，输入预测源，其路径为 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\assetss 输出：输出每个图片是否骨折，并基于测试时间生成相对应的 csv 文件，结果保存在 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\binary_classification\results（例如：predictions_20250429_020000.csv）
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\YOLO	检查 detect（识别）的数据集里面每个子集 9 个异常的相关图片的数量及其占比，尤其注重骨折的占比	输入：数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\data\labels，对其 labels 文件进行分析，发现训练集，验证集和训练集的骨折图片所占比例差不多，因此可以直接利用之前的数据集，只需要对 labels 标签进行修改即可
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\YOLO	将之前处理好的数据集的 images 文件夹复制过来，并修改对应 labels 标签，改为只剩骨折这一种异常	输入：数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\data 输出：生成只剩骨折的 labels 标签的新的适用于 yolo 的二分类的数据集，保存在 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\data_binary 下
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\YOLO	利用 yolo 模型对骨折进行二分类训练	输入：数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\data_binary 输出：训练过程和结果，保存在 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\custom_runs_binary 下
D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\YOLO	对训练好的 YOLO 模型进行骨折的测试	输入：数据集 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\assets 和 meta_binary.yaml 文件,输出：结果图片，只包含是否骨折及其骨折位置，保存在 D:\pythonProject\ultralytics-main\GRAZPEDWRI-DX\code\binary\YOLO\runs\binary 下



## 参考文献

- [1]. 吕志华, 李明, 张强. (2019). 儿童腕部骨折的流行病学特征及诊治进展. 中华创伤杂志, 35(8), 743–748.
- [2]. E.M. Hedstrom, O. Svensson, U. Bergstrom, P. Michno, Epidemiology of fractures in children and adolescents, *Acta Orthopaedica* 81 (2010) 148–153.
- [3]. P.H. Randsborg, et al., Fractures in children: epidemiology and activity-specific fracture rates, *J. Bone Joint Surg.- Am.* Vol. 95 (2013) e42.
- [4]. L.A. Landin, Epidemiology of children's fractures, *J. Pediatric Orthopaedics B* 6 (1997) 79–83.
- [5]. J. Cheng, W. Shen, Limb fracture pattern in different pediatric age groups: A study of 3350 children, *J. Orthop. Trauma.* 7 (1993) 15–22, <http://dx.doi.org/10.1097/00005131-199302000-00004>.
- [6]. H. Guly, Diagnostic errors in an accident and emergency department, *Emerg. Med. J.* 18 (2001) 263–269.
- [7]. J. Mounts, J. Clingenpeel, E. McGuire, E. Byers, Y. Kireeva, Most frequently missed fractures in the emergency department, *Clin. Pediatr. (Phila)* 50 (2011) 183–186.
- [8]. E. Er, P. Kara, O. Oyar, E. Unluer, Overlooked extremity fractures in the emergency department, *Ulus. Travma. Acil. Cerrahi. Derg.* 19 (2013) 25–28.
- [9]. M. Juhl, B. Moller-Madsen, J. Jensen, Missed injuries in an orthopaedic department, *Injury* 21 (1990) 110–112.
- [10]. Haramati, N. (2015). The radiologist shortage: Implications for musculoskeletal imaging. *Skeletal Radiology*, 44(9), 1235–1237. DOI: 10.1007/s00256-015-2185-9
- [11]. Kitamura, F. C., Marques, O., & Iano, Y. (2019). Automated detection

- n of rib fractures in chest X-rays using YOLOv3. IEEE Latin American Conference on Computational Intelligence (LA-CCI), 1–6.DOI: 10.1109/LA-CCI.2019.8942938
- [12]. 赵明, 李华, 陈伟. (2022). 基于 YOLOv5 的腕部 X 光影像骨折检测方法研究. 计算机应用研究, 39(10), 3056–3061.
- [13]. S.J. Adams, R.D.E. Henderson, X. Yi, P. Babyn, Artificial intelligence solutions for analysis of X-ray images, Canad. Assoc. Radiol. J. l'Association canadienne des radiologistes 846537120941671 (2020).
- [14]. L. Tanzi, et al., Hierarchical fracture classification of proximal femur X-Ray images using a multistage deep learning approach, Eur. J. Radiol. 133 (2020) 109373.
- [15]. [19] J.W. Choi, et al., Using a dual-input convolutional neural network for automated detection of pediatric supracondylar fracture on conventional radiography, Investigat. Radiol. 55 (2020) 101–110.
- [16]. Ultralytics. (2023). YOLOv8: A new state-of-the-art in object detection. GitHub Repository.URL: <https://github.com/ultralytics/ultralytics>