# 《YOLOv10: Real-Time End-to-End Object Detection》

**摘要**

本研究报告探讨了 YOLOv10 模型在实时端到端目标检测领域的创新与应用。通过引入 NMS-free 训练策略和全面的效率-准确性驱动的模型设计，YOLOv10 解决了传统 YOLO 模型依赖非极大值抑制（NMS）的问题，显著降低了推理延迟。该模型采用了轻量级分类头、空间-通道解耦下采样、等级引导的块设计、大核卷积以及部分自注意力模块（PSA）等技术，提升了全局建模能力和特征提取能力，增强了检测精度。实验结果表明，YOLOv10 在多个标准基准测试中展示了卓越性能，并且在计算资源有限的情况下表现出色。未来的研究将致力于优化 NMS-free 训练策略、硬件适应性优化、模型压缩及多任务学习，以进一步提升模型的效率和适用性。

**关键词：**实时目标检测；NMS-free 训练；效率-准确性驱动设计；模型压缩；多任务学习

## 1 引言

### 1.1 研究背景

在计算机视觉领域，实时目标检测技术对于多种应用至关重要，包括自动驾驶 [1]、机器人导航 [2]、目标跟踪 [3] 等。这些应用要求目标检测系统能够在保持高准确性的同时，快速地处理和响应图像数据。YOLO（You Only Look Once）系列模型因其速度快、易于部署而受到广泛关注，成为实时目标检测的主流方法之一。然而，传统的 YOLO 模型依赖于非极大值抑制（NMS）来进行后处理，这一步骤不仅增加了计算负担，也限制了模型的端到端优化和部署。

### 1.2 研究目的和重要性

本研究旨在解决 YOLO 模型在实时目标检测中的效率和准确性问题。通过提出新的 NMS-free 训练策略和全面的效率-准确性驱动的模型设计策略，本研究的目标是减少计算冗余，提高模型能力，并实现实时端到端目标检测。这对于推动实时目标检测技术的发展，以及在资源受限的环境中部署高效目标检测模型具有重要意义。

## 2 相关工作

### 2.1 实时目标检测器

YOLO 系列模型自 YOLOv1 起，通过不断的迭代和改进，已经成为实时目标检测领域的标杆。YOLOv1 [4] 以其简洁的网络结构和快速的检测速度获得关注，随后的 YOLOv2 [5] 引入了卷积神经网络（CNN）的改进，YOLOv3 [6] 进一步优化了检测流程，YOLOv4 [7] 和 YOLOv5 [8] 则通过改进网络结构和数据增强策略提升了性能。YOLOv6 至 YOLOv9 [9–12] 在模型架构、训练策略等方面进行了更多的探索和优化，使得 YOLO 系列模型在实时目标检测领域始终保持领先地位。

### 2.2 端到端目标检测器

端到端目标检测器，如 DETR（Detection Transformer）[13]，通过引入变换器架构和匈牙利损失来实现一对一匹配预测，消除了手工制作的组件和后处理。DETR 的出现为目标检测领域带来了新的思路，但其训练难度大、推理速度慢等问题限制了其在实时场景下的应用。后续的变体如 Deformable DETR [14]、DINO [15] 等通过不同的方式改进了 DETR 的性能和效率，但与 YOLO 系列相比，仍存在一定的差距。

## 3 本文方法

### 3.1 用于无 NMS 训练的一致的双重分配策略

在 YOLO 模型中，传统的训练方法采用 TAL（目标分配学习），通常通过一对多分配为每个实例分配多个正样本，以增强监督信号并提高模型性能。然而，这种方法需要依赖 NMS 后处理来去除冗余的预测，导致在推理阶段的效率较低。为了克服这一问题，本文提出了一种 NMS-free 的训练策略，通过一致的双重标签分配来提高 YOLO 模型的训练效率和推理性能。

如图 1所示是该策略的结构及原理图，主要由以下 4 部分构成：

一、Backbone(主干网络): 输入图像通过 Backbone 进行特征提取。

二、PAN (路径聚合网络): 提取的特征通过 PAN 进一步处理，增强特征表示。

三、双重标签分配：与传统的“一对一”或“一对多”标签分配策略不同，本研究引入了另一个“一对一”的头部网络，它与原始的“一对多”分支共享相同的结构和优化目标，但在标签分配时采用“一对一”匹配。这样，在训练期间，两个头部可以共同优化模型，而在推理期间，本研究只使用“一对一”头部进行预测，从而实现端到端的部署。

四、一致的匹配度量：为了使“一对一”头部与“一对多”头部在训练时保持一致的监督，本研究提出了一致的匹配度量方法。该方法通过调整匹配度量中的参数，使得两个头部在选择最佳预测时保持一致，从而减少了理论监督间隙，并提高了性能。

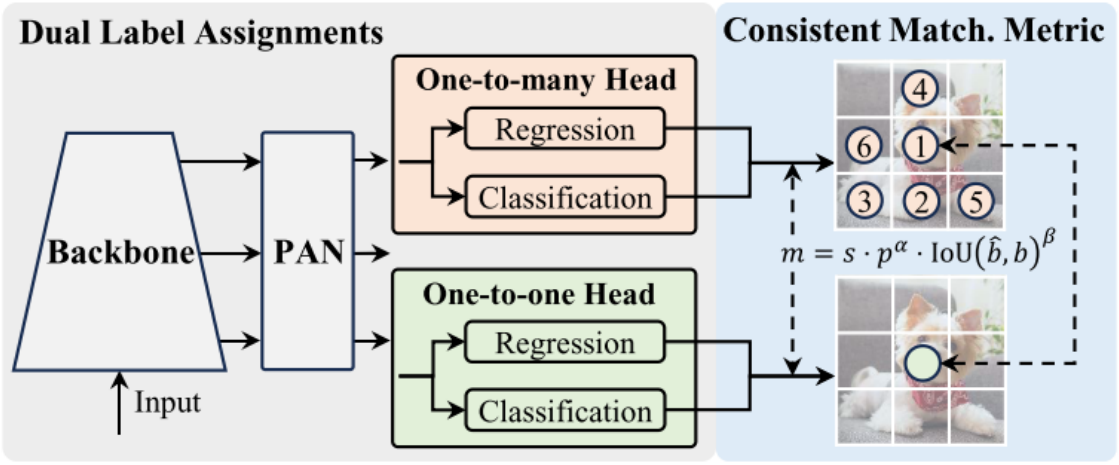


图 1. 无 NMS 训练的一致双重分配

### 3.2 整体效率-精度驱动的模型设计

“整体效率-精度驱动的模型设计” 是 YOLOv10 中一个核心概念，它旨在优化模型的计算效率同时不牺牲检测精度。这种设计方法考虑了实时目标检测应用对于快速响应和高准确性的双重需求，并且通过一系列创新的技术手段来实现。以下是该设计策略的主要部分：

一、轻量级分类头：为了减少计算负担，YOLOv10 采用了轻量级架构作为分类头部。这意味着在不影响性能的前提下，尽可能地减少参数数量和计算开销。通过简化分类头的设计，可以在保持竞争力的检测精度的同时降低对硬件资源的需求。

二、空间-通道解耦下采样：传统的 YOLO 模型在同一操作中同时进行空间维度（即图像大小）和通道数目的变换。而 YOLOv10 提出将这两个过程分开：首先使用逐点卷积调整通道数量，然后通过深度卷积来进行空间维度的缩减。这种方法不仅减少了计算成本，还保留了更多的特征信息，有助于提高模型的表达能力。

三、等级引导的块设计：通过对每个阶段的信息冗余度进行分析，YOLOv10 提出了等级引导的块设计策略。这意味着根据特定层或阶段的重要性来选择适合的网络组件。例如，在那些被认为具有较低内在重要性的部分采用更高效的构建模块如 CIB（紧凑型逆置块），以减少不必要的计算复杂性而不影响整体性能。

四、大核卷积：在深层网络中引入大内核卷积可以扩大感受野，从而增强模型对全局上下文的理解。这对于小规模模型特别有用，因为它们通常有较小的感受野，限制了其捕捉较大范围内的物体的能力。此外，结构重参数化技术被用来缓解因使用较大卷积核所带来的优化问题。

五、部分自注意力（PSA）模块：PSA 模块允许仅对特征图的一部分执行自注意力机制，而不是像常规 transformer 那样对整个特征图进行处理。这样做的好处是在维持低计算成本的情况下增加了模型的全局建模能力。实验表明，当 NPSA（PSA 头的数量）设置为 1 时，可以在不显著增加延迟的情况下获得最佳性能提升。

综上所述，该设计确保了 YOLOv10 能够在不同应用场景中高效运行，尤其是在资源受限环境下，如移动设备或边缘计算平台，同时也保证了高水平的目标检测准确性。这些改进使得 YOLOv10 成为当前最先进的实时端到端目标检测器之一。

# 4 复现细节

## 4.1 与已有开源代码对比

参考代码链接yolov10, 主要应用及修改如下:

### 4.1.1 数据集选择

选择的数据集为 NEU-DET, 本数据集由东北大学宋克臣团队制作, 专注于钢材表面缺陷的检测与识别。数据集包含了 1800 张图片，涵盖了六种常见的钢材表面缺陷类型。如图 2所示，展示了数据集的一张图片。如图 3所示，展示了数据集的存放位置。

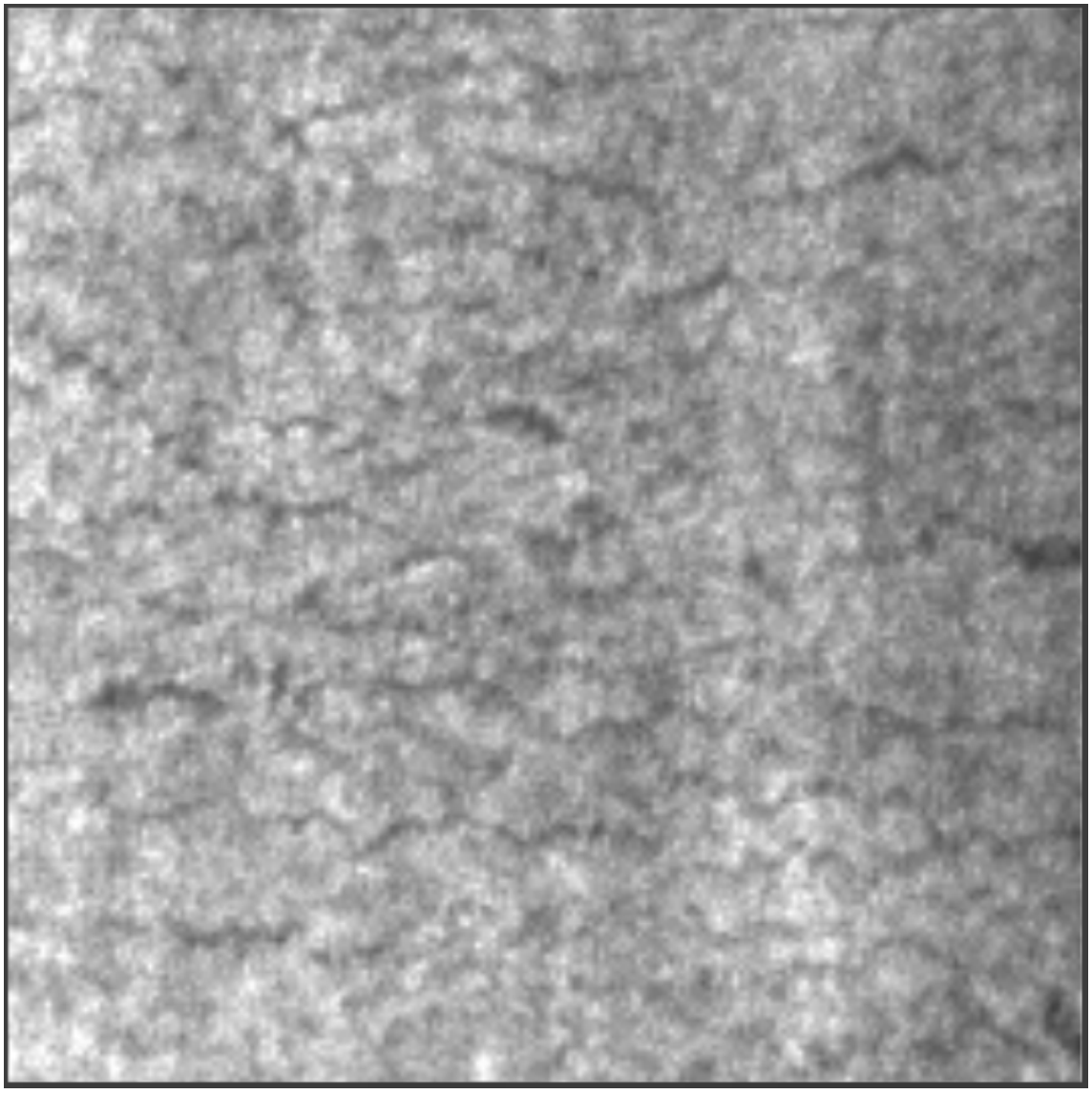

图 2. 数据集的部分图片

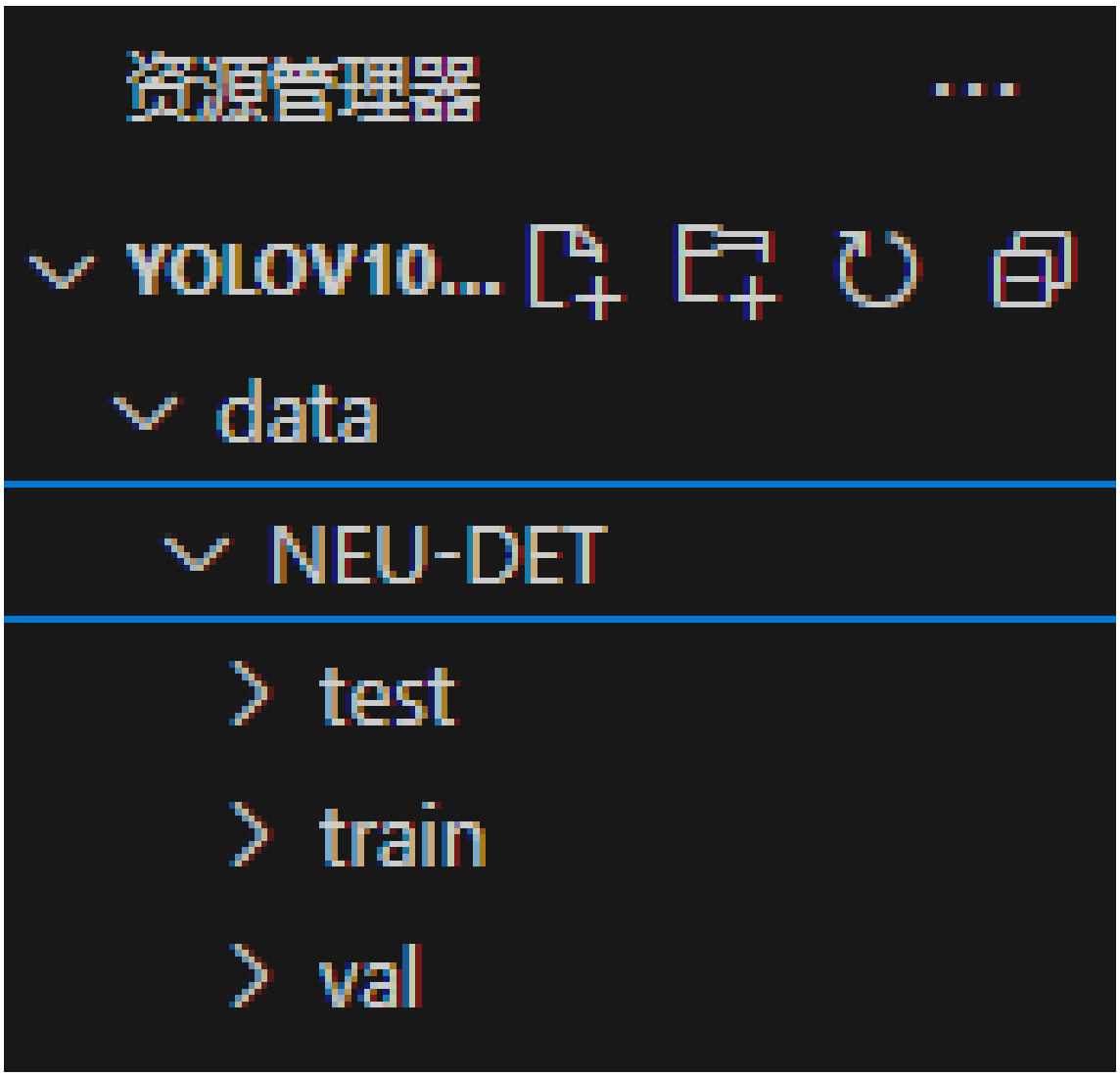


图 3. 数据集存放位置

#### 4.1.2 创建数据集的配置文件

在文件夹中新建了一个 NEU-DET.yaml 文件，如图 4所示。

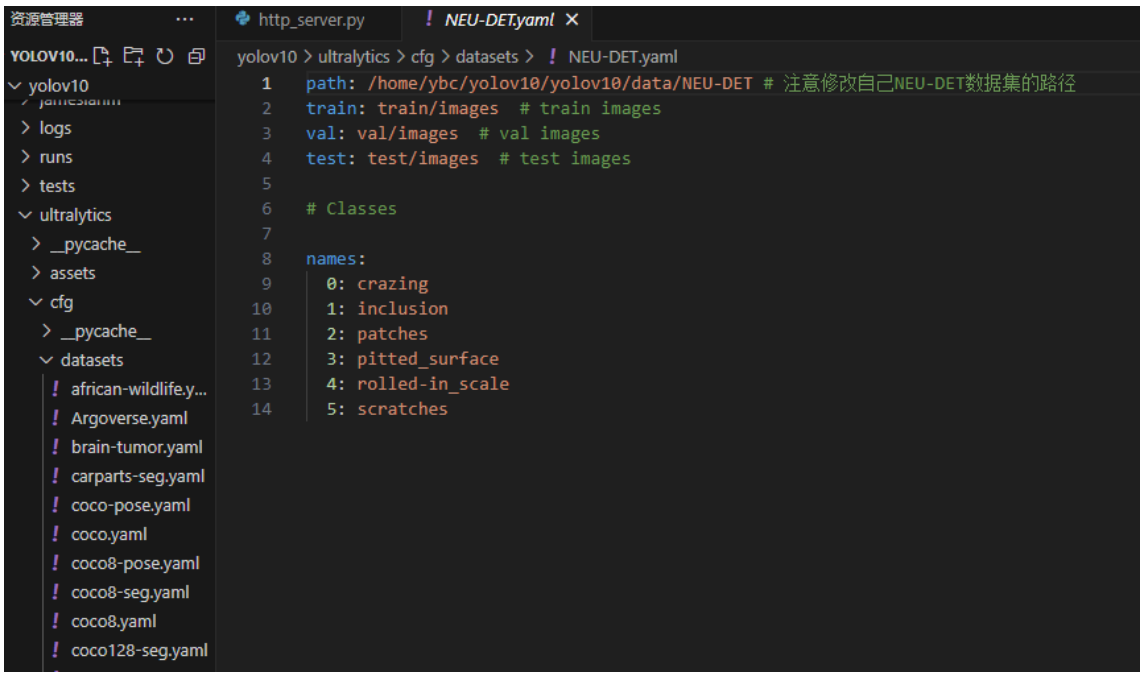


图 4. 数据集的配置文件

### 4.1.3 创建模型对象

在文件夹中新建一个 yolov10-neu-det.yaml 文件，如图 5所示。

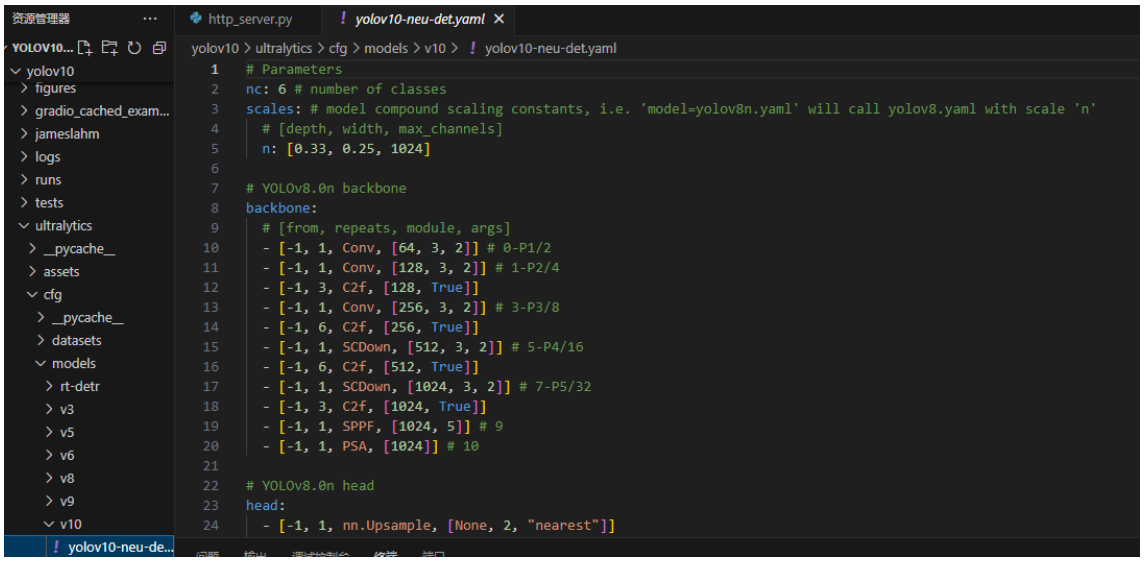


图 5. 模型对象文件

### 4.1.4 创建训练文件

在文件夹中创建一个 train.py 文件方便后续的训练，如图 6所示。


图 6. 训练文件

### 4.2 实验环境搭建

1. 包管理器选择 anaconda，虚拟环境配置 python 的版本官方是 3.9，命令如下：

```
conda create -n yolov10 python=3.9
```

2. 激活虚拟环境后（conda activate yolov10），安装 gpu 版本的 torch，官方的建议 torch 版本为 2.0.1。

3. 安装其他的项目依赖，命令如下：

```
pip install -r requirements.txt
```

4.cd 到 yolov10 目录后，运行下面代码即可配置好基本的环境：

```
pip install -e .
```

### 4.3 界面分析与使用说明

如图 7所示，在模型训练完毕后，运行 app.py 文件得到一个 web 链接。

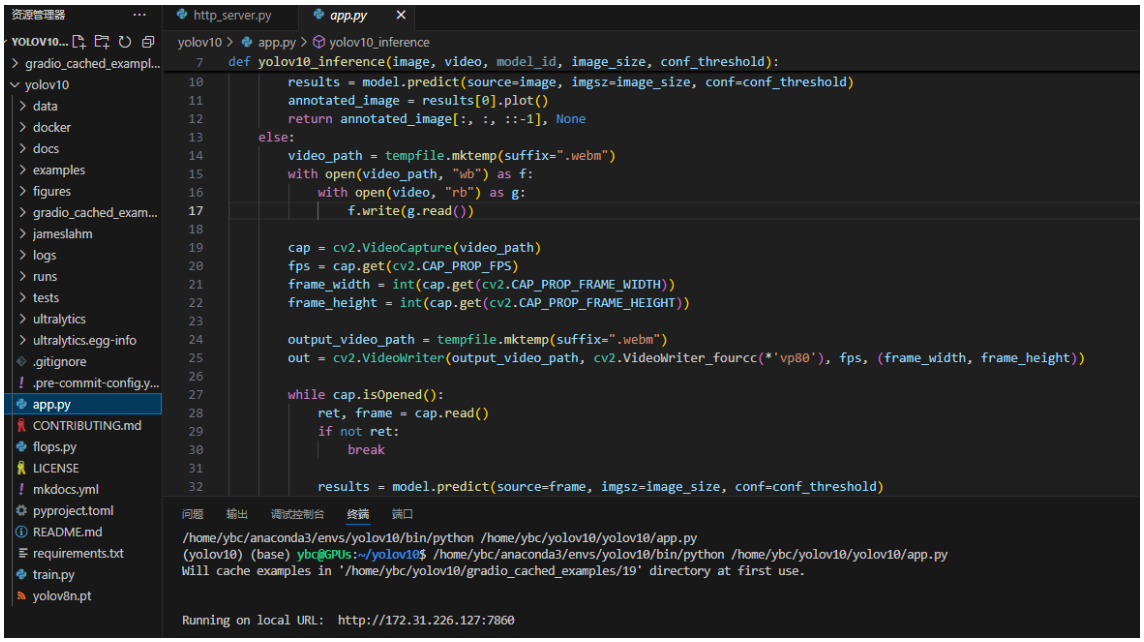


图 7. 运行 app.py 文件结果

如图 8所示，展示了通过链接打开的可视化图片检测页面，可在此上传图片进行检测。图 9为一个展示图片检测效果的例子。

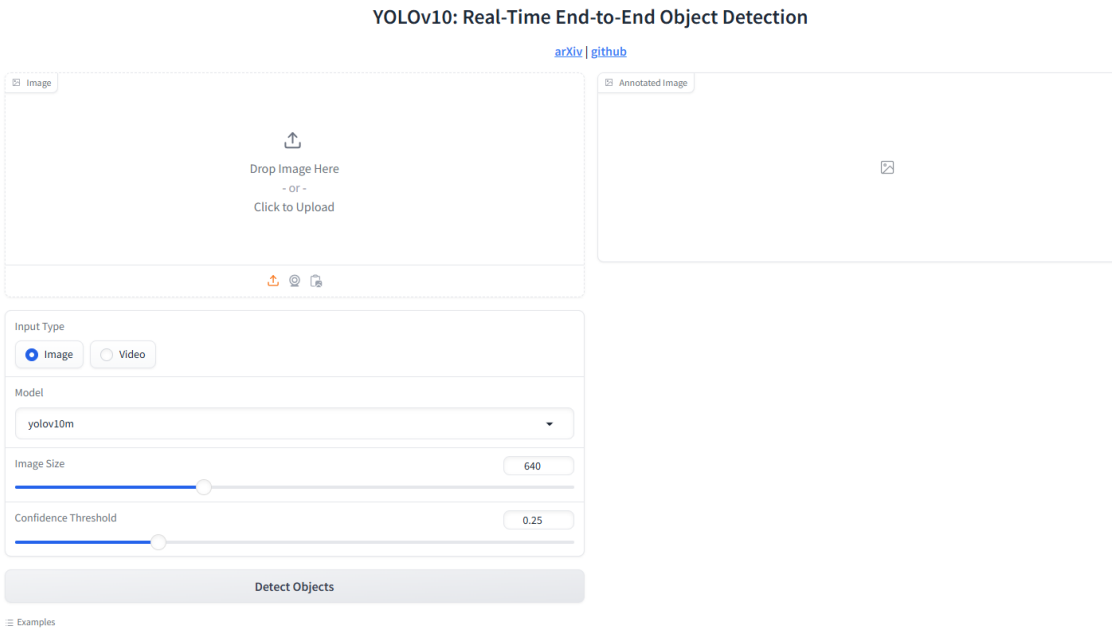


图 8. 可视化图片检测页面

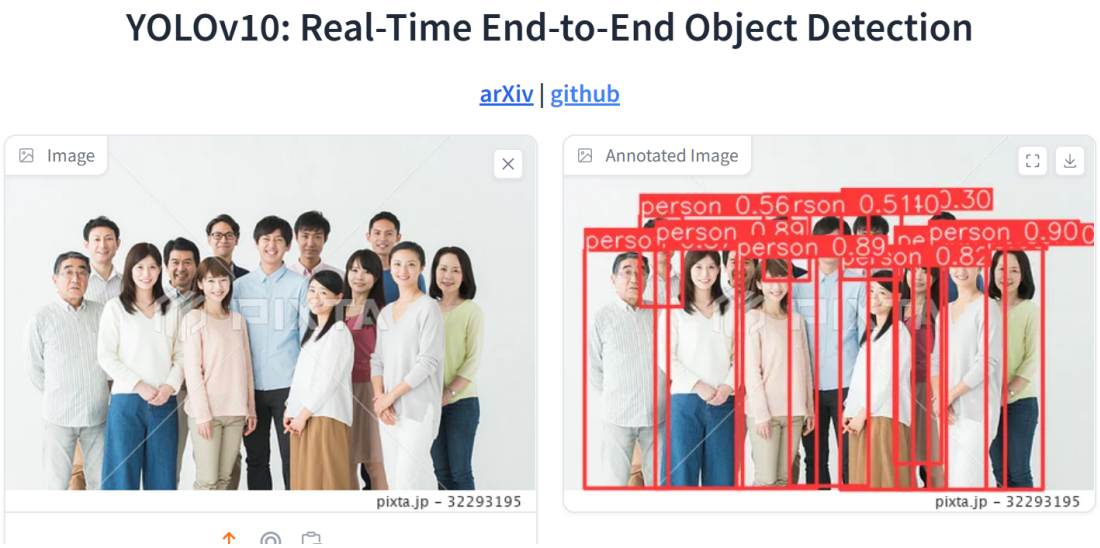


图 9. 图片检测效果

## 4.4 创新点

### 4.4.1 NMS-free 训练策略

YOLOv10 引入了一种新颖的一致的双重分配策略，以解决传统 YOLO 模型在实时目标检测中对 NMS 后处理步骤的依赖。该策略包含两个关键元素：一是双重标签分配，在训练阶段同时利用一对多和一对一头部网络进行预测，确保模型能够从丰富的监督信号中学习；二是一致的匹配度量方法，通过调整匹配度量中的参数，如缩放因子、预测概率以及 IoU 等，使得两个头部在网络训练时能够保持一致的选择标准，从而减少了理论上的监督差距，并提升

了模型的整体性能。这种方法允许模型在推理期间消除 NMS 的需求，实现了真正的端到端部署，极大地提高了效率并简化了系统架构。

#### 4.4.2 全面的效率-精度驱动模型设计

为了提高 YOLO 模型的效率和准确性，YOLOv10 提出了全面的效率-精度驱动的模型设计策略。首先，采用了轻量级分类头来减少计算负担，同时维持高水平的检测性能。其次，空间-通道解耦下采样的技术被用来优化下采样过程，即先通过逐点卷积调整通道维度，然后使用深度卷积进行空间下采样，这不仅降低了计算成本，还保留了更多特征信息。此外，等级引导的块设计根据每个阶段的信息冗余度分析，自适应地整合紧凑的块设计，有效提升模型效率。再者，大核卷积的应用扩大了感受野，增强了模型的全局建模能力，特别是在小规模模型中表现尤为突出。最后，部分自注意力（PSA）模块的设计允许仅对特征的一部分进行自注意力计算，减少了计算复杂度的同时提升了模型性能，确保了高效率与高性能之间的良好平衡。

#### 4.4.3 改进的数据增强与预处理

YOLOv10 改进了数据增强技术，采用了 Mosaic 和 MixUp 两种方法来增强数据集的多样性。Mosaic 数据增强通过将四张图像拼接成一张输入图像，为模型提供了更多的上下文信息，有助于改善对不同尺度目标的检测能力。而 MixUp 则通过对两张图像及其对应的标签进行线性插值，生成新的训练样本，增加了数据的多样性，促进了模型泛化能力的提升。这些数据增强手段共同作用，使得 YOLOv10 在面对复杂场景时具有更好的鲁棒性和更高的检测精度。

#### 4.4.4 更高效的损失函数

在损失函数方面，YOLOv10 选择了 CIoU（Complete Intersection over Union）损失和 Focal Loss 相结合的方式。CIoU 损失不仅仅考虑了边界框之间的重叠区域，还包括中心点距离和长宽比差异等因素，这样可以更精确地指导模型回归正确的边界框。Focal Loss 用于解决类别不平衡的问题，特别是对于稀有类别的检测任务，它能更好地聚焦于困难样本，保证模型不会忽略那些出现频率较低的目标。这两种损失函数的组合使用，显著提高了 YOLOv10 的检测准确性和稳定性。

#### 4.4.5 硬件友好型设计

考虑到实际应用中的硬件限制，YOLOv10 专注于开发更紧凑高效的模型版本，以满足不同应用场景的需求。例如，针对移动设备和边缘计算环境下的实时检测任务，研究团队探索了模型压缩与量化的方法，包括剪枝和量化技术，进一步减小了模型大小，缩短了推理时间。这样的硬件友好型设计，使得 YOLOv10 能够在资源受限的环境中高效运行，同时也为未来的模型部署提供了更大的灵活性。

#### 4.4.6 扩展的应用场景支持

为了增加模型的适用性，YOLOv10 还支持多任务学习框架，允许同时进行多个相关任务

（如分类、分割等）的联合训练。这种多任务学习的能力提高了模型的灵活性和实用性，使其能够应对更加复杂的现实世界问题。无论是自动驾驶汽车需要识别多种交通标志，还是智能监控系统需要同时进行人员检测和行为分析，YOLOv10 都能够提供一个强大且灵活的解决方案。

#### 4.4.7 改进的训练流程

YOLOv10 优化了训练流程，采用了渐进式训练策略和动态锚点调整机制。渐进式训练策略让模型从简单任务逐渐过渡到复杂任务，稳定了训练过程并提高了最终性能。动态锚点调整则是根据训练数据自动优化锚点配置，提高了模型对不同类型目标的适应性。这些改进措施确保了 YOLOv10 在训练过程中能够获得最佳的学习效果，从而为实际应用提供了更加可靠的技术保障。

## 5 实验结果分析

### 5.1 性能指标图表

如图 10所示，是一系列训练过程中的性能指标图表。整体来看，随着训练的进行，损失值逐渐下降，召回率和精确率逐渐上升，mAP 值也在不断提高，表明模型的性能在不断优化。平滑后的曲线（虚线）有助于观察趋势，减少噪声的影响。

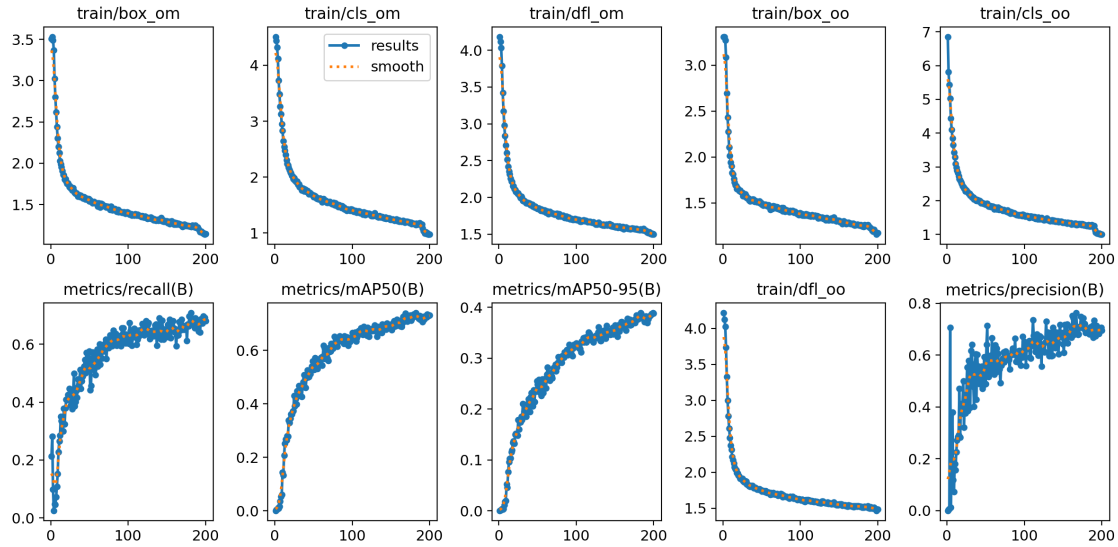


图 10. 一系列训练过程中的性能指标图表

### 5.2 F1-Confidence 曲线

如图 11所示，是 F1-Confidence 曲线。从图中可以看出，随着置信度阈值的提升，F1 分数通常会先增加到一个峰值然后开始下降。具体而言，对于所有类别综合来看，当置信度设置为 0.253 时，能够获得最佳的整体性能，此时平均 F1 分数达到 0.69。值得注意的是，不同类别的最佳置信度和对应的 F1 分数可能会有所差异，反映出各类别之间的特性及检测难度的不同。这表明，在实际应用中针对每个类别调整置信度阈值可以进一步优化检测效果。

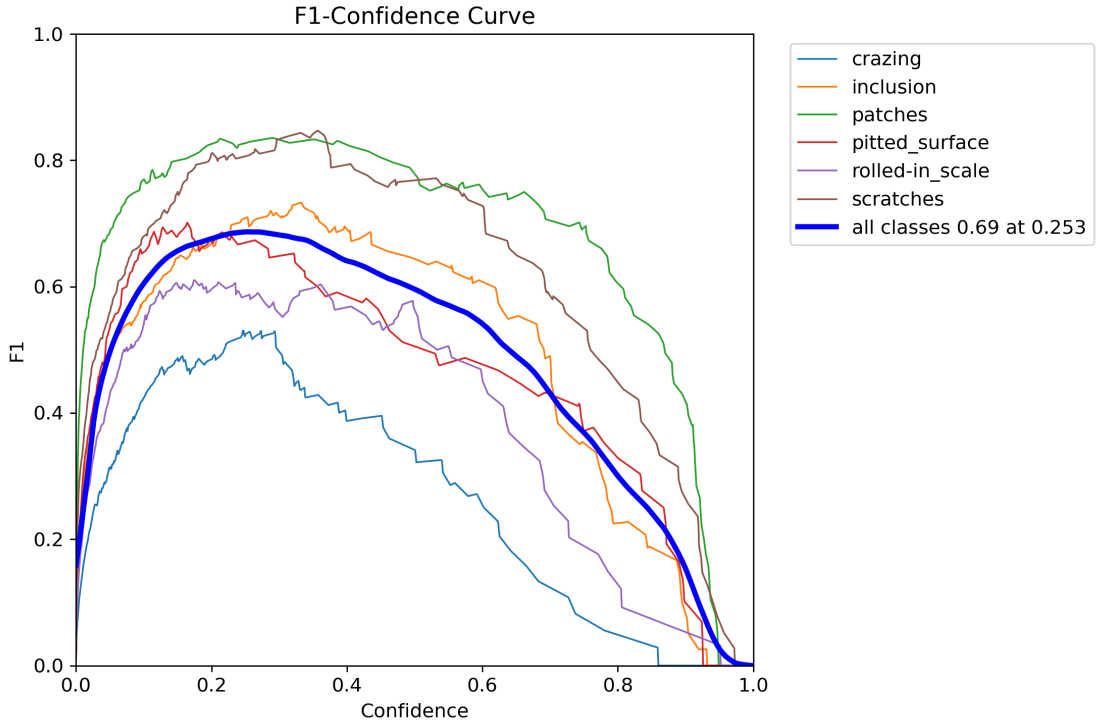


图 11. F1-Confidence 曲线

### 5.3 Precision-Confidence 曲线

如图 12所示，是 Precision-Confidence 曲线。从图中可以看出，随着置信度的增加，精度通常会先上升，然后趋于稳定或略有波动。对于所有类别而言，最佳置信度为 0.957，此时能够达到最高的平均精度，达到 1.00。然而，不同类别的最佳置信度和对应的精度可能有所不同，体现了各类别之间的差异性。

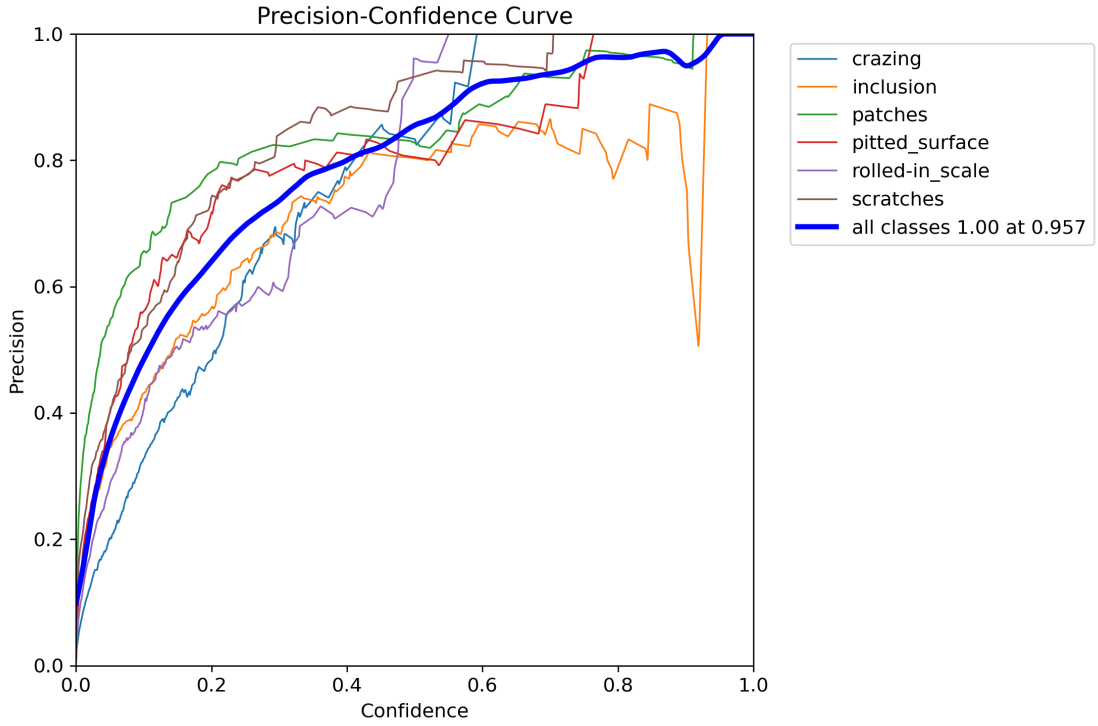


图 12. Precision-Confidence 曲线

### 5.4 Precision-Recall 曲线

如图 13所示，是 Precision-Recall 曲线。从图中可以看出，随着召回率的增加，精确率往往会下降。对于所有类别而言，最佳的平衡点出现在置信度阈值为 0.5 时，此时平均精度均值（mAP）达到 0.731。然而，不同类别的最佳召回率和精确率可能有所不同，反映了各类别在这两者之间的平衡差异。

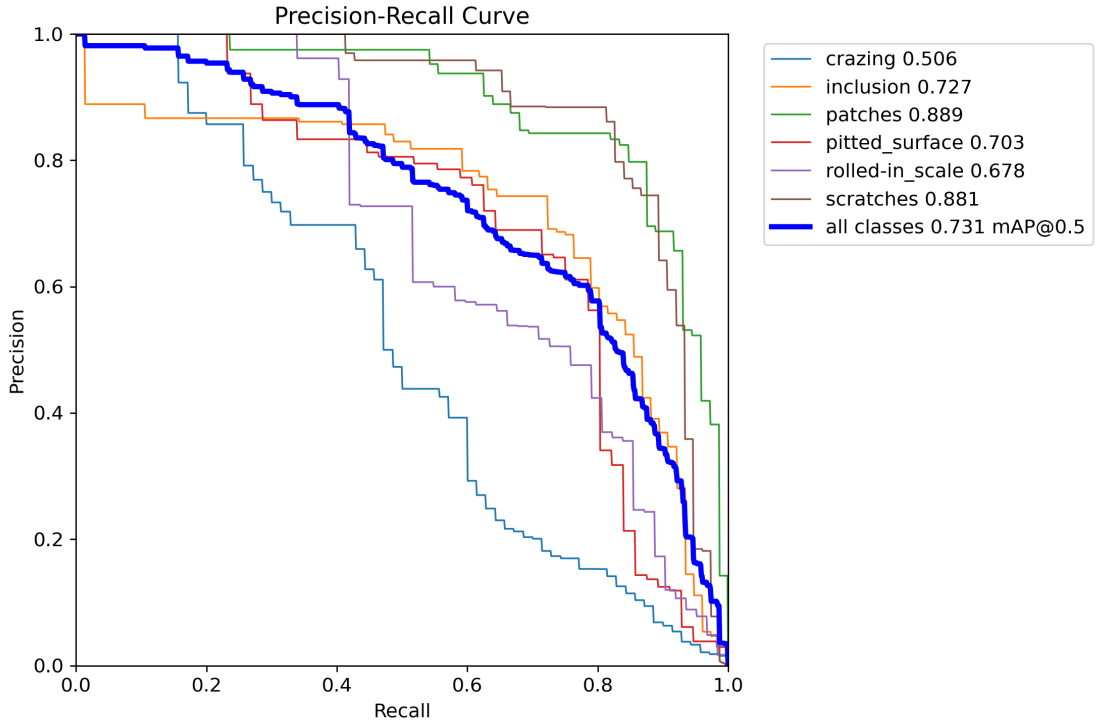


图 13. Precision-Recall 曲线

### 5.5 Recall-Confidence 曲线

如图 14所示，是 Recall-Confidence 曲线。从图中可以看出，随着置信度的增加，召回率往往会下降。对于所有类别而言，最佳的置信度为 0.000，此时能够达到最高的平均召回率，达 0.96。然而，不同类别的最佳置信度和召回率可能有所不同，显示出各类别在这一关系上的差异性。

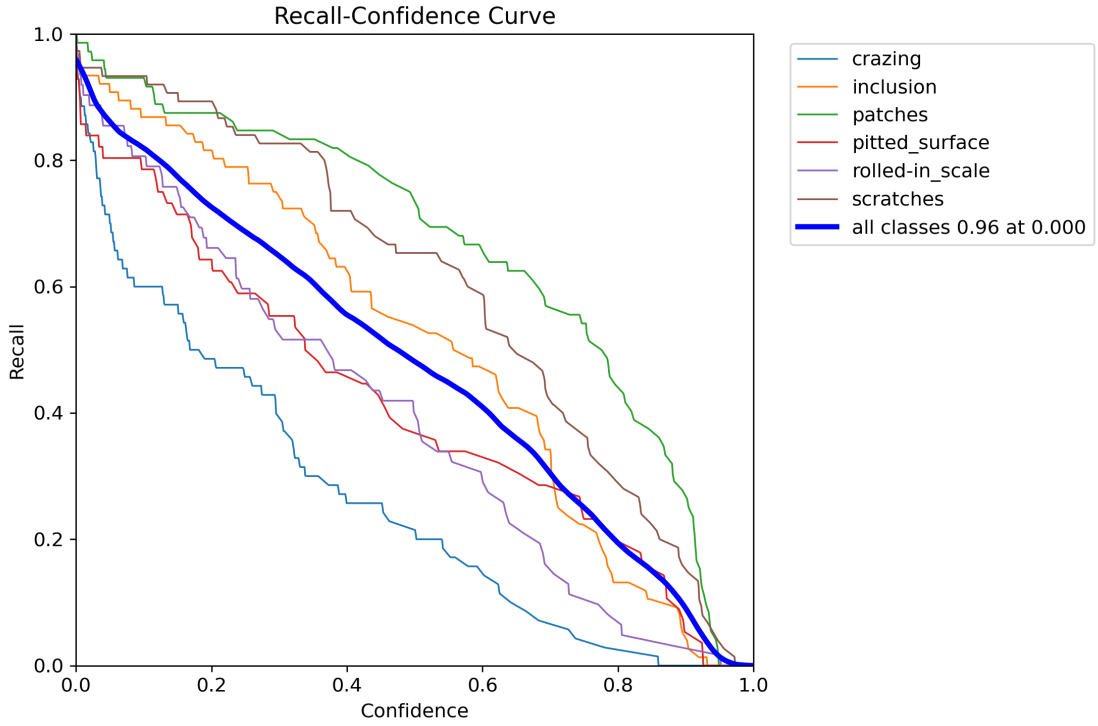


图 14. Recall-Confidence

# 6 总结与展望

## 6.1 研究总结

本研究深入探讨了 YOLOv10 模型在实时端到端目标检测领域的创新与应用。通过引入一系列先进的技术和方法，YOLOv10 成功解决了传统 YOLO 模型依赖 NMS 后处理的问题，并实现了无 NMS 训练策略，显著降低了推理延迟。该模型不仅在多个标准基准测试中展示了卓越的性能，如 COCO 数据集上的表现，还在计算资源有限的情况下表现出色，参数数量和计算量大幅减少，延迟显著降低。此外，通过采用轻量级分类头、空间-通道解耦下采样、等级引导的块设计、大核卷积以及部分自注意力模块（PSA）等创新技术，YOLOv10 在保持高效的同时提升了模型的全局建模能力和特征提取能力，从而增强了检测精度。

## 6.2 贡献与展望

YOLOv10 的主要贡献在于其提出了一种全新的 NMS-free 训练策略和全面的效率-准确性驱动的模型设计策略。这些改进使得 YOLOv10 能够在不牺牲检测精度的前提下，大幅提高计算效率，为实时目标检测提供了强有力的技术支持。此外，模型的设计充分考虑到了实际应用场景的需求，特别是移动设备和边缘计算环境下的实时检测任务，确保了模型在不同计算资源环境下都能高效运行。未来，YOLOv10 将继续优化 NMS-free 训练策略，以期接近或超越传统方法的检测效果，进一步缩小小模型中的性能差距，同时探索更高效的硬件适应性优化方案，提升模型性能以适应实际应用中的硬件限制。

### 6.3 未来的研究方向

尽管 YOLOv10 已经在许多方面取得了显著进展，但仍有几个关键领域值得进一步研究：

一、优化 NMS-free 训练策略：虽然 YOLOv10 在 NMS-free 训练方面取得了突破，但在小模型中仍存在一定的性能差距。未来的工作将致力于优化这一策略，以实现更好的检测效果，尽可能地接近甚至超过传统的 NMS 方法。

二、硬件适应性优化：研究人员将继续寻找高效的方法来提升模型性能，确保 YOLOv10 能够在不同类型的硬件上高效运行，尤其是在资源受限的环境中，如移动设备和边缘计算平台。

三、模型压缩与多样化应用：为了满足更多样化的应用场景需求，未来工作将专注于开发更加紧凑高效的模型版本，以更好地服务于不同的应用领域，尤其是那些对实时性和计算资源有严格要求的任务。

四、多任务学习和跨域泛化：探索如何让 YOLOv10 支持更多的任务类型，如同时进行目标检测和语义分割，以及如何增强模型在不同领域间的泛化能力，使其能够更好地应对未知环境中的挑战。

## 参考文献


[1] Daniel Bogdoll, Maximilian Nitsche, and J Marius Zöllner. Anomaly detection in autonomous driving: A survey. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4488–4499, 2022.

[2] Douglas Henke Dos Reis, Daniel Welfer, Marco Antonio De Souza Leite Cuadros, and Daniel Fernando Tello Gamarra. Mobile robot navigation using an object recognition software with rgbd images and the yolo algorithm. *Applied Artificial Intelligence*, 33(14):1290–1305, 2019.

[3] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European Conference on Computer Vision*, pages 659–675. Springer, 2022.

[4] J Redmon. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[5] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[6] Ali Farhadi and Joseph Redmon. Yolov3: An incremental improvement. In *Computer vision and pattern recognition*, volume 1804, pages 1–6. Springer Berlin/Heidelberg, Germany, 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] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, Yonghye Kwon, Kalen Michael, Jiacong Fang, Zeng Yifu, Colin Wong, Diego Montes, et al. ultralytics/yolov5: v7. 0-yolov5 sota realtime instance segmentation. *Zenodo*, 2022.

[9] Chuyi Li, Lulu Li, Yifei Geng, Hongliang Jiang, Meng Cheng, Bo Zhang, Zaidan Ke, Xiaoming Xu, and Xiangxiang Chu. Yolov6 v3. 0: A full-scale reloading. *arXiv preprint arXiv:2301.05586*, 2023.

[10] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475, 2023.

[11] Rejin Varghese and M Sambath. Yolov8: A novel object detection algorithm with enhanced performance and robustness. In *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, pages 1–6. IEEE, 2024.

[12] CY Wang, IH Yeh, and HYM Liao. Yolov9: Learning what you want to learn using programmable gradient information. arxiv 2024. *arXiv preprint arXiv:2402.13616*.

[13] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[14] X Zhu, W Su, L Lu, B Li, X Wang, and J Dai. Deformable detr: Deformable transformers for end-to-end object detection. arxiv 2020. *arXiv preprint arXiv:2010.04159*, 2010.

[15] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.