

基于YOLOv8n的甲骨文图像识别技术研究

张振文^{1*}, 黄柏锋^{1*}, 韦易霖¹, 覃军¹, 刘映伶^{2#}

¹桂林信息科技学院机电工程学院, 广西 桂林

²桂林信息科技学院基础教研部, 广西 桂林

收稿日期: 2025年9月9日; 录用日期: 2025年9月29日; 发布日期: 2025年10月10日

摘要

随着深度学习技术的发展, 物体检测算法在图像识别领域取得了显著进展。针对甲骨文图像识别技术的研究, 本文基于YOLOv8n框架构建了一种高效的甲骨文物体检测与分类方法。通过数据预处理、数据增强、模型训练、模型预测等环节, 结合CSPDarknet骨干网络、PAFPN特征金字塔和CIoU损失函数, 提升了检测精度。该验证结果表明目标检测模型的 P 值和召回率 R 分别达到0.855和0.807, 其中 $mAP@0.5$ 的指标达到0.876, $mAP@[0.5:0.95]$ 的值为0.53。分类模型的准确率高达93.6%, 比YOLOv5提高了6.6%。本研究为甲骨文的数字化与智能分析提供了新的技术方案, 探索了提高识别与分类精度及泛化能力的方法。

关键词

YOLOv8n, 甲骨文, 物体检测, 深度学习, 图像识别

Research on Oracle Bone Script Image Recognition Technology Based on YOLOv8n

Zhenwen Zhang^{1*}, Bofeng Huang^{1*}, Yilin Wei¹, Jun Qin¹, Yingling Liu^{2#}

¹School of Mechanical and Electrical Engineering, Guilin Institute of Information Technology, Guilin Guangxi

²Division of Fundamental Education and Research, Guilin Institute of Information Technology, Guilin Guangxi

Received: September 9, 2025; accepted: September 29, 2025; published: October 10, 2025

Abstract

With the development of deep learning technology, object detection algorithms have achieved significant progress in the field of image recognition. Focusing on the recognition of oracle bone script images, this paper proposes an efficient object detection and classification method based on the

*共同第一作者。

#通讯作者。

文章引用: 张振文, 黄柏锋, 韦易霖, 覃军, 刘映伶. 基于 YOLOv8n 的甲骨文图像识别技术研究[J]. 图像与信号处理, 2025, 14(4): 467-485. DOI: 10.12677/jisp.2025.144043

YOLOv8n framework. Through stages such as data preprocessing, data augmentation, model training, and prediction, and by integrating the CSPDarknet backbone, PAFPN feature pyramid, and CIoU loss function, the detection accuracy is effectively improved. Experimental results show that the object detection model achieves a precision (P) of 0.855 and a recall (R) of 0.807, with an mAP@0.5 of 0.876 and an mAP@[0.5:0.95] of 0.53. The classification model attains an accuracy of 93.6%, representing a 6.6% improvement over YOLOv5. This study provides a novel technical solution for the digitization and intelligent analysis of oracle bone inscriptions, and explores methods to enhance recognition accuracy, classification performance, and model generalization capability.

Keywords

YOLOv8n, Oracle Bone Script, Object Detection, Deep Learning, Image Recognition

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

甲骨文是最早成熟的汉字之一, 承载着丰富的历史和文化信息。它不仅是汉字的起源, 也是中华文明的瑰宝, 对研究古代中国的历史、社会、宗教等多个方面具有重要意义。然而, 由于甲骨文复杂的字形结构以及图像噪声问题, 传统的人工解读与识别工作面临巨大挑战[1]。随着计算机视觉与深度学习技术的迅速发展, 甲骨文自动识别技术逐渐成为研究热点, 并为甲骨文的保护与传承提供了新的途径[2]。

甲骨文的自动识别面临多重困难。首先, 甲骨文字符种类繁多且排列不规则, 尤其是破损、磨损等因素的干扰使得图像处理方法的应用变得非常困难[3]。其次, 现有的甲骨文图像数据集相对稀缺, 给深度学习算法的训练带来了巨大挑战。近年来, 基于深度学习的图像处理技术, 尤其是目标检测算法(如 YOLO 系列), 在甲骨文识别方面取得了显著进展[4]。这些方法通过优化图像预处理、网络结构与损失函数, 逐步提高了甲骨文自动识别的精度[5]。

例如, YOLOv5 和 YOLOv7 模型已经被应用于甲骨文的检测与识别, 并在一些实验中达到了较高的检测精度[6]。然而, 传统 YOLO 模型在处理甲骨文时仍然面临低检测精度和泛化能力不足的问题[7]。为此, 近年来研究者提出了改进的 YOLOv8n 模型, 进一步提升了该模型在复杂图像环境下的性能, 通过引入特征金字塔和优化损失函数等技术, 改进了模型的表现[8]。

随着 YOLOv8n 模型的优化, 它已成为甲骨文图像识别的重要工具[9]。通过引入 CSPDarknet 骨干网络、PAFPN 特征金字塔结构以及 CIoU 损失函数, YOLOv8n 能够提高小目标的检测能力, 同时保证高精度, 并改善模型的训练效率与泛化能力[10]。这些技术的结合将甲骨文铭文的数字化保护和智能分析推向了新的阶段, 为文化遗产的保护与传承提供了有力的技术支持[11]。

2. 研究目标

本研究旨在基于 YOLOv8n 框架, 探索一种高效的甲骨文图像目标检测与分类翻译方法。本文的主要目标是通过优化数据预处理、数据增强和甲骨文图像的模型训练, 提升甲骨文图像的识别准确率和泛化能力。具体来说, 本研究的目标是设计一个目标检测系统, 能够有效提高检测精度, 减少干扰噪声的影响, 并具有高分类精度。此外, 结合改进的 YOLOv8n 模型, 本研究还将进一步探讨如何提升模型在复杂背景和低质量图像中的表现, 推动甲骨文数字化与智能分析技术的应用与发展[12]。

3. 构建甲骨文特征目标检测图像与数据集

3.1. 边缘检测与甲骨文轮廓特征提取

由于拍摄角度、光照不均和甲骨文图像的历史性损坏, 以及缺乏对甲骨文字符的形态先验特征和其干扰因素的特殊考虑, 传统的代表性图像分割方法无法有效区分文本目标、点噪声、人造纹理及甲骨文原图中的固有纹理。这导致其误分割率较高, 在处理甲骨文拓片图像时存在一定的局限性[13]。为提高后续分割和识别的准确性, 本研究采用了以下预处理方法:

1) 非局部均值去噪: 这是由于甲骨文图像存在高斯噪声、椒盐噪声等问题, 主要由图像的年代久远、扫描设备差异或保存环境不同等因素引起。非局部均值去噪通过搜索图像中与目标像素块相似的区域, 进行像素值的加权平均, 从而有效去除各种噪声, 恢复甲骨文的真实笔画和纹理, 避免后续分析中的噪声干扰。与传统的均值滤波(可能会模糊边缘)相比, 非局部均值去噪能够保留甲骨文的笔画边缘和细节[14]。

2) CLAHE 对比度增强: CLAHE 将图像划分为多个局部区域, 并独立地对每个区域进行对比度增强, 这可以有效突出甲骨文的笔画细节, 并平衡图像亮度, 使得整体视觉效果更加均匀, 为后续的边缘检测和特征提取提供便利[14]。

3) 自适应阈值分割: 此方法可以将甲骨文图像中的文字符号从背景中分离出来。通过分割得到的二值图像是边缘检测的重要输入, 能够为模型训练提供高质量的特征数据(目标识别与 YOLOv8n)。本文使用局部高斯加权均值来动态生成阈值。对于每个像素(x, y), 取其周围 15×15 邻域内的像素值, 常数偏移为 6。

4) 形态学闭操作优化: 对甲骨文图像轮廓的边缘进行修复、噪声抑制、曲线平滑等处理。这些改进直接提升了轮廓的完整性和准确性, 为后续分析提供了高质量的数据, 特征轮廓提取流程图如图 1。



Figure 1. Main trunk of feature extraction

图 1. 特征轮廓提取流程图

3.2. 数据格式转换

1) 数据集的数量与划分: 数据集的数量和划分按 8:2 的比例分为训练集和验证集。目标检测模型训练集包含 4865 张图片, 验证集包含 1217 张图片。此外, 分类模型构建数据集的训练集包含 32,522 张图片, 分类模型的验证集包含 8094 张图片。

2) 处理 json 文件, 将 json 文件中的 ann 数据转换为 YOLOv8n 可以读取的数据, 用于提取目标文本进行训练。转换公式如下:

$$C_x = \frac{x_1 + x_2}{2w} \quad (1)$$

其中 x_1 为注释框左上角的 x 坐标, x_2 为注释框右下角的 x 坐标, C_x 为归一化中心的 x 坐标, w 为图像宽度

$$C_y = \frac{y_1 + y_2}{2h} \quad (2)$$

y_1 为注释框左上角的 y 坐标, y_2 为注释框右下角的 y 坐标, C_y 为归一化中心的 y 坐标, h 为图像

高度

$$W = \frac{x_2 - x_1}{w} \quad (3)$$

$$H = \frac{y_2 - y_1}{h} \quad (4)$$

W 是归一化宽度, H 是归一化高度

3) 输入图像被调整为 640×640 像素, 以确保每个输入图像的尺寸一致, 从而满足 YOLOv8n 的输入要求。

4) 像素值被归一化到 $[0, 1]$ 范围内, 以便可以输入到 YOLOv8n 模型中。归一化后的目标分布特征图 (图 2) 显示, 甲骨文铭文集中分布在图像的中间, 而右侧显示甲骨文铭文通常较小。

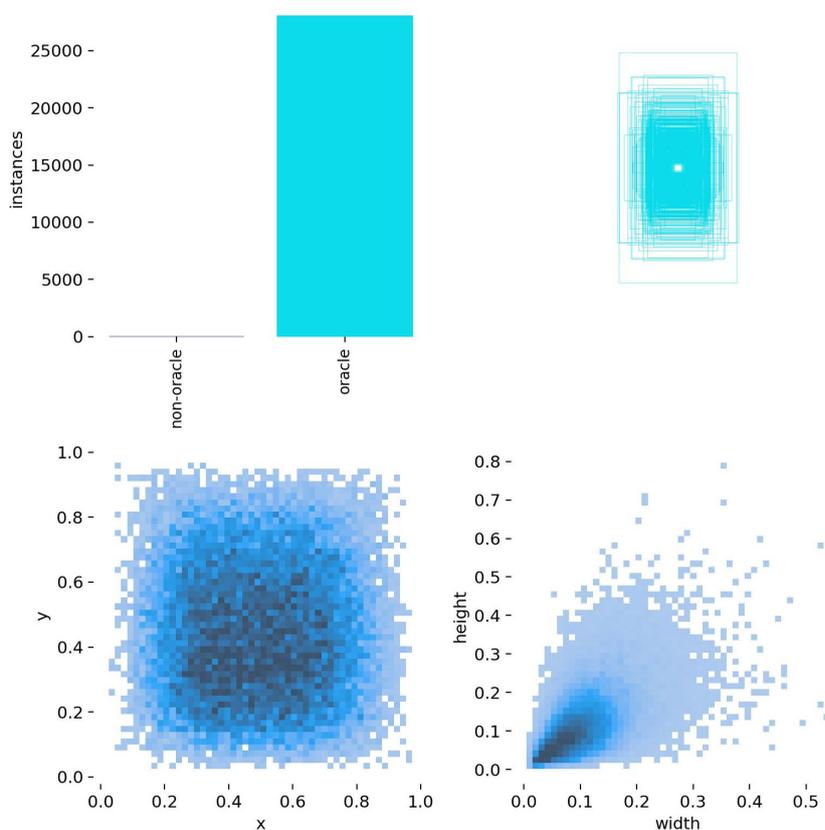


Figure 2. Target distribution characteristics
图 2. 目标分布特征

4. YOLOv8n 网络结构概述

4.1. 特征提取骨干网络

YOLOv8n Backbone 采用了 CSPDarknet 结构, 这是 YOLOv4 和 YOLOv5 中使用的骨干网络的改进版本 (见图 3) 与 YOLOv5 中的 C3 模块相比, C2f 模块具有更强的特征提取能力。具体的改进包括将第一个卷积层的卷积核大小从 6×6 改为 3×3 ; 所有的 C3 模块都被替换为 C2f, 并添加了跳跃层和分割操作, 如下图所示; 块的数量从 C3 的 3-6-9-3 改为 C2f 的 3-6-6-3, 从而实现轻量化处理。

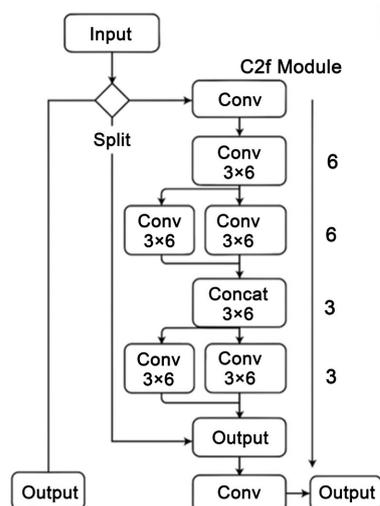


Figure 3. Main trunk of feature extraction
图 3. 特征提取主干

4.2. 特征融合层

YOLOv8n 的 Neck 采用了 PANet 结构, 这是一个双向通道网络, 使用自上而下和自下而上的路径。该 Neck 主要由 SPFF 模块、PAA 模块和 PAN 模块组成。SPFF 模块用于不同尺度特征图的池化操作, 以拼接不同尺度的特征图; PAA 模块用于智能分配锚框, 以优化正负样本的选择; PAN 模块用于特征路径聚合, 通过自上而下和自下而上的路径增强特征检测能力。

mAP (均值平均精度)是对所有类别的 AP (平均精度)进行平均得到的结果, 其中 AP 是精度 - 召回曲线(PR 曲线)下的积分。在目标检测任务中, 通常设置不同的 IoU 阈值(0.5 或 0.5:0.95)来计算 mAP, 分别记录为 mAP@0.5 和 mAP@[0.5:0.95], 它们是衡量模型精度的重要指标。

用于智能分配锚框, 优化正、阴性样本的选择; PAN 模块通过自顶向下和自底向上两种路径对不同层次的特征进行路径聚合, 增强特征检测能力。

4.3. 目标检测与分类检测头

在 YOLOv8n 中有两种类型的检测头, 一种用于目标检测任务, 另一种用于分类任务。检测头预测每个锚框的边界回归值以及目标存在的置信度; 分类头则对每个特征图进行分类, 并输出每个类别的概率分布。

5. 模型训练

鉴于甲骨文图像独特的灰度分布和形态学特征, 例如, 非局部均值去噪能够在去噪过程中保留甲骨文的笔画边缘和细节, CLAHE 对比度增强可以限制局部对比度增强的幅度, 同时增强有用的细节并避免过度的噪声增强。除了模型自身的随机水平翻转、MixUp 增强、Mosaic 数据增强和 CutOut 增强外, 这些方法也得到了应用。随机水平翻转以 50% 的概率对图像进行水平翻转, 提高模型对左右对称物体的识别能力。MixUp 增强按比例混合两张图像并同时混合标签, 增强模型对复杂场景的泛化能力。Mosaic 数据增强模拟了甲骨文拓片中常见的密集字符分布场景, 通过随机拼接四张图像, 增强了小字符的可识别性。CutOut 增强随机阻塞图像的矩形区域, 改善了模型对部分被遮挡目标的适应能力, 以下图 4 为整个模型工作流程。

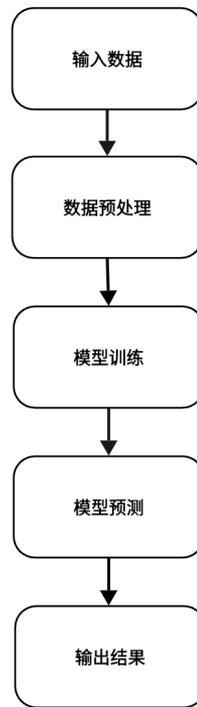


Figure 4. Model workflow chart
图 4. 模型工作流程图

5.1. YOLOv8n 模型训练

在训练 YOLOv8 目标检测模型时, 也会产生很多指标。通过观察以下指标可以了解模型训练过程的准确性。

1) IoU (Intersection over Union)是衡量预测框(A)与真实框(B)重叠程度的核心指标

$$\text{IoU} = \frac{A \cap B}{A \cup B} \quad (5)$$

$A \cap B$ 表示预测框与真实框的相交区域, $A \cup B$ 表示两者的并集区域。

2) P (Precision)精确度是指模型预测的所有样本中实际为正的样本所占的比例。

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (6)$$

其中, TP (True Positives)为真正的正例, 即模型正确预测为正例的样本数量; FP (False Positives)为假正例, 即模型错误预测为正例的样本数量。

3) R (Recall)回召率指的是模型在所有实际为正例的样本中, 正确预测为正例的样本比例。

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (7)$$

TP (True Positives)是一个真正例子, 即模型正确预测为正的样本数量; FN (False Negatives)是一个假阴性例子, 即模型错误预测为阴性的样本数量。

F_1 分数是衡量回召率和精确率之间的平衡, 适用于正负样本不平衡场, 避免单一指标的片面性。

$$F_1 = \frac{R \times P}{R + P} \quad (8)$$

R (Recall)是回召率, P (Precision)是精确度

从图 5 中训练过程中产生的数据可以看出, 经过持续的训练, 损失率在不断地降低, 模型在不断地优化。在训练过程中, 模型的平均准确率 P 不断上升, 最终达到 85.5%, 这意味着模型预测的正例中有很多是真正例, 即假阳性较少。平均召回率 R 也在不断上升, 最终达到 80.7%, 表明该模型可以找到大部分实际的正例, 即漏检较少。最终的 $mAP@0.5$ 和 $mAP@[0.5:0.95]$ 分别达到 87.6%和 53%, 这意味着当 IoU 阈值为 0.5 时, 模型的平均准确率表现良好, 满足了小而密集的甲骨文对象的要求。

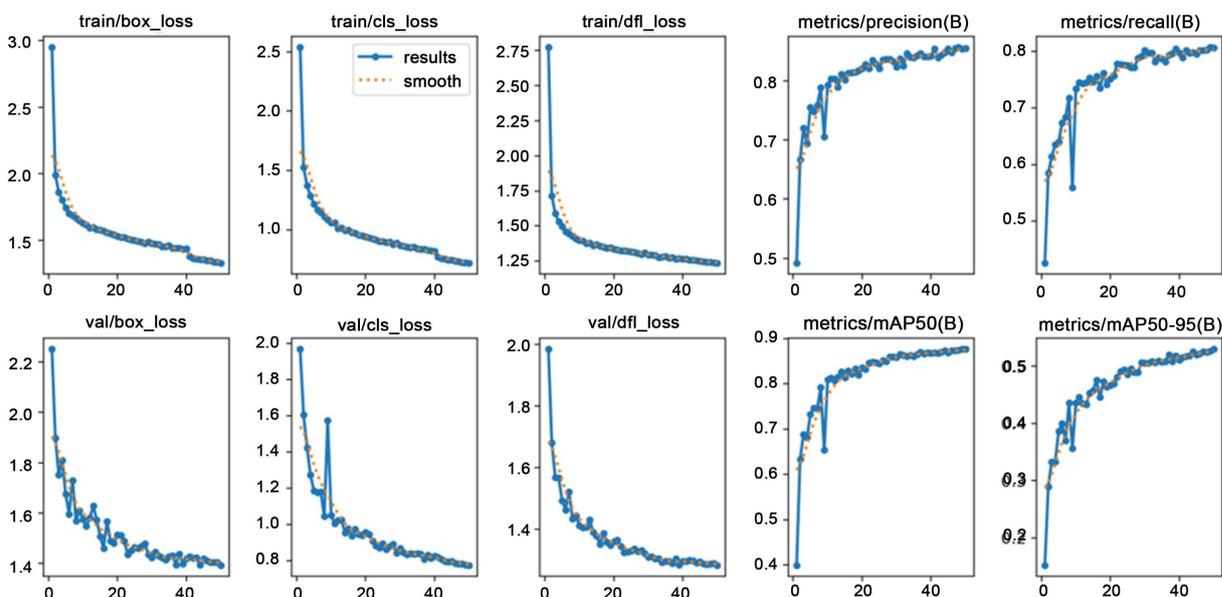


Figure 5. The training variation diagram of the YOLOv8n model

图 5. YOLOv8n 模型训练变化图

5.2. YOLOv8n-cls 模型训练

在训练 YOLOv8-cls 分类模型时, 会产生许多不同于目标检测模型的指标。分类模型训练过程的准确性也可以通过观察以下指标来了解。BCE 损失(二进制交叉熵损失)可以用来评价模型预测概率的准确性。该值越小, 模型预测的概率越接近真实标签。在训练过程中, 通常将对数损失最小化作为优化目标。

$$BCE = -\frac{1}{N} \sum_{i=1}^n (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) \quad (9)$$

其中 N 为样本数, y_i 为值为 0 和 1 的第 i 个二元变量(真标签), p_i 为第 i 个预测为 1 (正类)的概率。

A_1 (Top-1 accuracy)是指模型预测的类别与真实类别一致的概率最高的样本占总样本的比例。

$$A_1 = \frac{TP + TN}{N} \quad (10)$$

N 是样本的总数; TP (True Positives)表示真正的正例, 即模型正确预测为正例的样本数量; TN (True Negatives)表示真正的负例, 即模型正确预测为负例的样本数量。

其中, N 是样本的总数; TP (True Positives)表示真正的正例, 即模型正确预测为正例的样本数量; TN (True Negatives)表示真正的负例, 即模型正确预测为负例的样本数量。

A_5 (Top-5 准确率)是模型预测的前五个类别中, 包含真实类别的样本比例与样本总数的比值。首先, 定义:

$$K_i = \begin{cases} 0 & (k_i \notin S_i) \\ 1 & (k_i \in S_i) \end{cases} \quad (11)$$

S_i 为模型第 i 个样本预测的 5 个类别; k_i 是第 i 个样本的实际类别。

$$A_5 = \frac{1}{N} \sum_{i=1}^N K_i \quad (12)$$

N 是样本总数。当样本的真实类别属于模型预测的五个类别之一时为真, 累积值除以样本数。

图 6 显示了模型的损失率不断下降, 并最终稳定在 0.1339, 表明模型在训练集上具有良好的拟合效果。此外, A_1 (Top1 准确率) 在训练过程中不断提高, 最终达到了 93.6%, 表明模型能够准确预测大多数样本的类别。 A_5 (Top-5 准确率) 也在经过多轮训练后持续增长, 最终达到了 93.6%。模型对少数样本的预测可能由于类别相似性而未能排在第一位, 但它仍然位于模型预测的前五个类别中。因此, 0.1339 的损失率表明模型在训练数据上具有良好的拟合效果。结合验证集上 A_1 达到 93.6% 的准确率, 可以认为模型能够满足甲骨文分类翻译任务的要求。

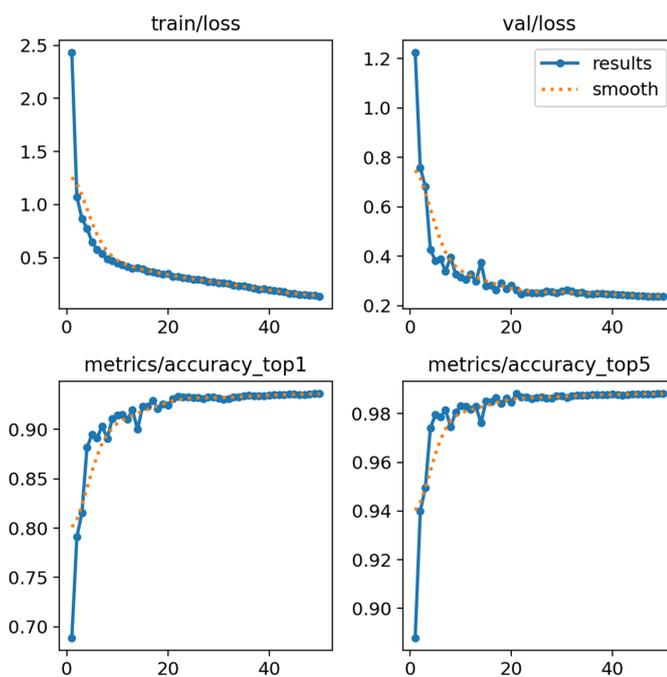


Figure 6. Training variation diagram of the YOLOv8n-cls model
图 6. YOLOv8n-cls 模型训练变化图

6. 实验结果与分析

6.1. 检测与分类结果

本文使用 2024 年第 14 届 MatorCup 数学应用挑战赛提供的数据集, 其中使用 YOLOv8n 模型对图片中的甲骨文进行检测, 使用 YOLOv8n-cls 分类模型对甲骨文进行分类。通过训练得到的预测结果如图 7 所示。

结果表明, YOLOv8n 能够准确地目标检测甲骨文图像中定位甲骨文, 并能准确地对甲骨文进行分类和识别, 通过分类实现翻译效果。这些结果充分验证了 YOLOv8n 在甲骨文目标检测和分类中的可行性。

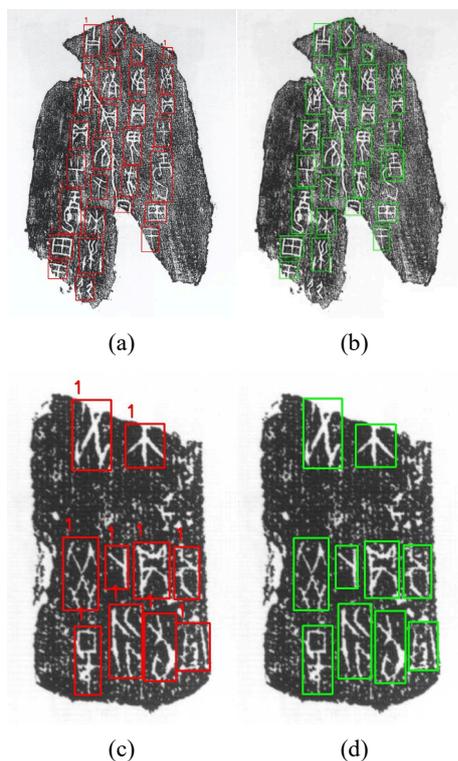


Figure 7. Object detection comparison
图 7. 目标检测对比

红色框表示真实目标的位置, 这些框代表人工标注的甲骨文字符。绿色框表示模型识别的目标区域。这些绿色框展示了模型成功检测到的甲骨文字符区域。通过比较左侧和右侧的框, 我们可以直观地看到目标检测模型的准确度, 表明绿色框与红色框高度重叠, 模型具有较高的检测准确性。

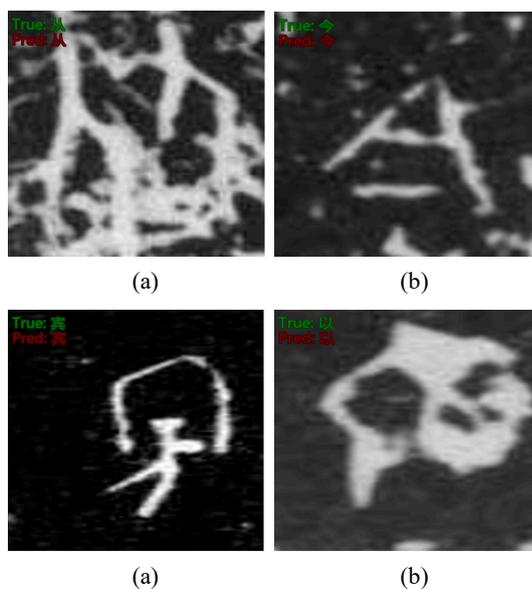


Figure 8. Classification test images
图 8. 分类测试图像

图 8 显示了左上角四个字符的正确结果和预测结果, 所有四个字符的预测均正确。分类模型的准确率为 93.6%, 显示出良好的分类能力。在处理一些形状相似的字符时, 识别的难度, 例如字符“从”和甲骨文字符, 依然能够正确识别, 这表明模型的泛化能力得到进一步提高, 尤其是在处理包含复杂背景或受损字符形状的样本时。

6.2. 性能分析

在目标检测模型的准确度中, $mAP@0.5$ 为 0.876, 表明该模型能够准确识别大多数目标。当该指标的 $mAP@0.5:0.95$ 为 0.53 时, 模型在阈值下仍具有稳定的检测能力。当目标检测模型 P 为 0.855, 召回率 R 为 0.807 时, 大多数被检测到的目标是真正的目标, 大部分实际目标都能够被覆盖。分类模型的 Top-5 准确率从最初的 93.6% 提高至 93.6%, 这表明目标检测模型专注于字符区域的精确提取, 而分类模型则强调对单字图像的精确识别。

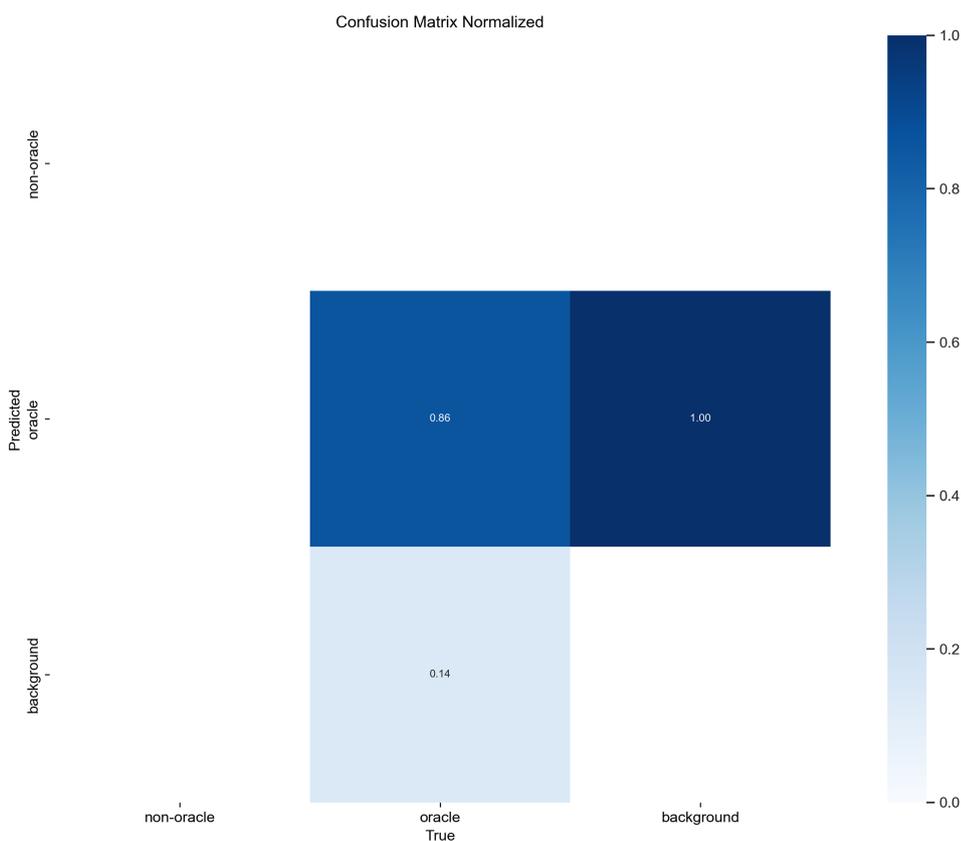


Figure 9. YOLOv8 Standardized confusion Matrix for Object detection

图 9. YOLOv8 目标检测标准化混淆矩阵

如图 9 所示, YOLOv8n 目标检测标准化混淆矩阵显示了模型在最终训练轮次中对每种样本类型的识别率, 特别是在“甲骨文”类别中, 识别率达到了 86%, 有 14% 被错误预测为背景。“背景”类别的识别准确率高达 100。

图 10 显示了 YOLOv8n-cls 分类模型的标准混淆矩阵。图示表明, 在充分训练后, 模型对大多数字符类别的识别能力得到了显著提升, 且主对角线明显突出, 表明分类性能已经稳定, 误差得到了良好的控制, 通过比较这个图, 可以直观地看到最终优化结果。

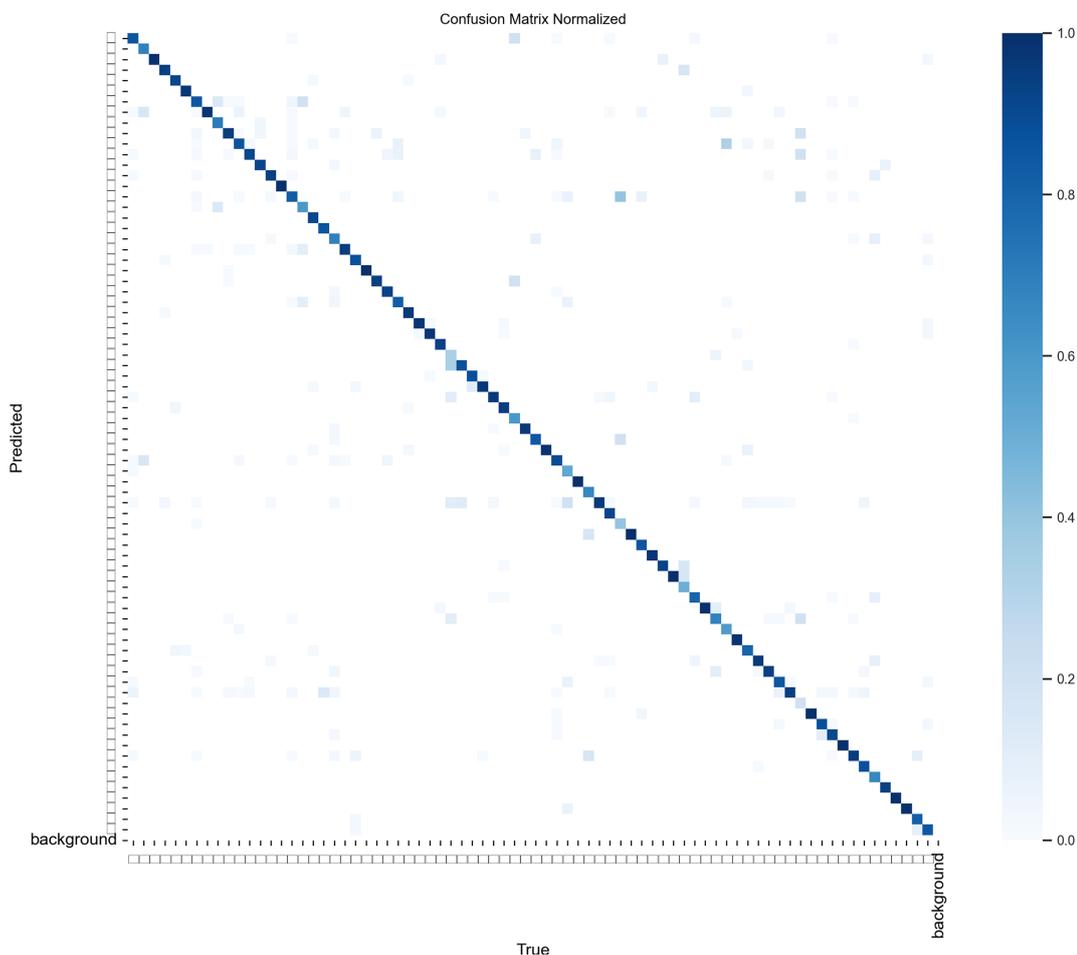


Figure 10. The standardized confusion matrix of the YOLOv8-cls classification model
图 10. YOLOv8-cls 分类模型的标准化混淆矩阵

6.3. 实验环境

所有实验均在 Ubuntu 22.04.3 系统环境下进行。本实验使用的 Pycharm 软件配备了以下环境：pytorch1.13.1, python3.7, CUDA12.2；硬件环境及相关模型参数见表 1。

Table 1. Hardware environment and related model parameters
表 1. 硬件环境及相关型号参数

Items	Description
CPU	12th Gen Intel Core (TM) i5-12500H
MEMORY	512G
STORAGE	16G
GPU	0

6.4. 对比实验

从表 2 可以看出, YOLOv8n 模型在 Precision、Recall 和 F_1 三项指标上均优于其他典型目标检测模

型。相比 U²-Net、SSD、Faster R-CNN 以及改进版 YOLOv3, YOLOv8n 在检测精度方面实现了明显提升, 其中 Precision 达到 0.855, Recall 达到 0.807, F₁ 值为 0.830, 整体性能最优。这表明 YOLOv8n 模型在甲骨文图像检测任务中具有更强的准确性和稳定性, 尤其适用于小目标、复杂背景下的字符识别场景。

为了进一步验证本研究中提出的 YOLOv8n 分类模型在甲骨文图像识别任务中的性能优势, 本文将该模型的分​​类准确率与典型的卷积神经网络(如 MobileViT、AlexNet、GoogleNet、ResNet、ResNeXt、EfficientNet 等)及其优化变种(MobileViTv2、v3、FM-MobileViT) [14] 进行比较。比较结果见表 3。从表 3 中的数据可以看出, 该模型在准确性方面超越了现有的主流结构, 反映了 YOLOv8n 分类模型的轻量化结构和多尺度感知模块。该模型不仅保留了高准确率, 而且在处理甲骨文字符时具有更好的鲁棒性和泛化能力, 尤其是在处理字体模糊和边界损坏的字符时表现尤为突出。

Table 2. Comparison of YOLOv8n experimental results

表 2. YOLOv8n 实验结果对比

Model	Precision	Recall	F ₁
U ² NeT [15]	0.738	0.721	0.729
SSD [3]	0.748	0.758	0.753
Faster R-CNN [3]	0.754	0.778	0.7658
The improved Yolov3 [14]	0.794	0.808	0.8009
Yolov8n	0.855	0.807	0.830

Table 3. Comparison of YOLOv8n-cls experimental results

表 3. YOLOv8n-cls 实验结果对比

Models	Accuracy (%)
YOLOv8n-cls	93.6
MobileViT	90.6
AlexNet	87.9
GoogleNet	89.3
ResNet	89.3
ResNeXt	90.2
EfficientNet	89.8
MobileViTv2	91.6
MobileViTv3	91.8
FM-MobileViT	92.3

7. 结论

基于 YOLOv8n 框架, 提出了一种高效的甲骨文图像目标检测与分类方法。通过数据预处理、数据增强、模型训练与优化, 本文提出的方案显著提高了甲骨文目标检测的准确性和分类能力。结合 CSPDarknet 骨干网络、PAFPN 特征金字塔和 CIoU 损失函数, 模型在复杂背景下的检测效果得到提升。所提出的方法在 MathorCup 2024 甲骨文数据集上取得了优异的结果。目标检测模型的 mAP@0.5 达到 0.876, mAP@[0.5:0.95] 为 0.53, 表明在不同准确度要求下表现良好。同时, 分类模型的准确率达到 93.6%, 比

YOLOv5 提高了 6.6%, 表明本文提出的优化方法有效提高了甲骨图像的分类准确率。通过本研究的探索与实验验证, 基于 YOLOv8n 的甲骨文图像目标检测与分类技术为甲骨文的数字化保护与智能分析提供了可行的技术解决方案, 具有广泛的应用前景。未来的工作可以进一步提高数据集的多样性, 增强模型的鲁棒性, 扩展其在其他古文字识别领域的应用。本研究为甲骨文铭文的智能识别提供了新的技术路径, 并为文化遗产数字化、古文字研究等领域的深入探索奠定了基础。

基金项目

广西壮族自治区大学生创新训练项目《基于神经网络算法的机器学习模型的甲骨文图像识别技术的应用研究》, 桂林信息科技学院韦易霖主持。编号: S202413644044。

参考文献

- [1] Tang, H., Tang, R. and Wang, H. (2024) Oracle Bone Script Intelligent Recognition: Automatic Segmentation and Recognition of Original Rubbing Single Characters. 2024 *5th International Conference on Electronic Communication and Artificial Intelligence (ICECAI)*, Shenzhen, 31 May-2 June 2024, 414-418. <https://doi.org/10.1109/icecai62591.2024.10675079>
- [2] Liu, J., Li, R., Fu, T., Yin, Z. and Yang, Z. (2022) Interpretation Study of Unknown Oracle Bones Based on Self-Training Learning. *Proceedings of the 2022 4th International Conference on Video, Signal and Image Processing*, Shanghai, 25-27 November 2022, 7-11. <https://doi.org/10.1145/3577164.3577166>
- [3] Zhen, Q., Wu, L. and Liu, G. (2024) An Oracle Bone Inscriptions Detection Algorithm Based on Improved YOLOv8. *Algorithms*, **17**, Article No. 174. <https://doi.org/10.3390/a17050174>
- [4] Xu, H., Zhang, Z., Liang, Q. and Lin, Z. (2024) Research on the Intelligent Oracle Bone Script Recognition Model Based on OTSU and Improved YOLOv8. 2024 *IEEE 4th International Conference on Electronic Technology, Communication and Information (ICETCI)*, Changchun, 24-26 May 2024, 972-976. <https://doi.org/10.1109/iceteci61221.2024.10594291>
- [5] Li, D. and Du, B. (2024) Research on Oracle Bone Inscription Segmentation and Recognition Model Based on Deep Learning. 2024 *IEEE 4th International Conference on Electronic Technology, Communication and Information (ICETCI)*, Changchun, 24-26 May 2024, 1309-1314. <https://doi.org/10.1109/iceteci61221.2024.10594164>
- [6] Sun, W., Zhai, G., Gao, Z., Chen, T., Zhu, Y. and Wang, Z. (2020) Dual-View Oracle Bone Script Recognition System via Temporal-Spatial Psychovisual Modulation. 2020 *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, Shenzhen, 6-8 August 2020, 193-198. <https://doi.org/10.1109/mipr49039.2020.00047>
- [7] Huang, S., Wang, H., Liu, Y., Shi, X. and Jin, L. (2019) OBC306: A Large-Scale Oracle Bone Character Recognition Dataset. 2019 *International Conference on Document Analysis and Recognition (ICDAR)*, Sydney, 20-25 September 2019, 681-688. <https://doi.org/10.1109/icdar.2019.00114>
- [8] Zhu, M., Wu, D., Xu, B., Cao, S., Cao, J. and Wang, K. (2024) Oracle Bone Inscriptions Image Recognition and Classification Method Based on Deep Learning. 2024 *IEEE 7th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chongqing, 20-22 September 2024, 1960-1965. <https://doi.org/10.1109/itnec60942.2024.10733031>
- [9] Song, L., Hong, C., Gao, T. and Yu, J. (2024) Lightweight Facial Landmark Detection Network Based on Improved MobileViT. *Signal, Image and Video Processing*, **18**, 3123-3131. <https://doi.org/10.1007/s11760-023-02975-4>
- [10] Jiang, H., Pan, Y., Chen, J., Liu, Z., Zhou, Y., Shu, P., Li, Y., Zhao, H., Mihm, S., Howe, L. and Liu, T. (2024) OracleSage: Towards Unified Visual-Linguistic Understanding of Oracle Bone Scripts through Cross-Modal Knowledge Fusion.
- [11] Xu, H., Zhang, Z., Liang, Q. and Lin, Z. (2024) Research on the Intelligent Oracle Bone Script Recognition Model Based on OTSU and Improved YOLOv8. 2024 *IEEE 4th International Conference on Electronic Technology, Communication and Information (ICETCI)*, Changchun, 24-26 May 2024, 972-976. <https://doi.org/10.1109/iceteci61221.2024.10594291>
- [12] Zheng, Q., Saponara, S., Tian, X., Yu, Z., Elhanashi, A. and Yu, R. (2023) A Real-Time Constellation Image Classification Method of Wireless Communication Signals Based on the Lightweight Network MobileViT. *Cognitive Neurodynamics*, **18**, 659-671. <https://doi.org/10.1007/s11571-023-10015-7>
- [13] Zhao, L., Li, Q., Zhang, L., Luo, Z., Jiang, H., Jiang, C., et al. (2020) A Virtual Teaching Machine System Based on Oracle-Bone-Script Learning. 2020 *IEEE International Conference on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, Exeter, 17-19 December 2020, 1144-1149.

<https://doi.org/10.1109/ispa-bdcloud-socialcom-sustaincom51426.2020.00170>

- [14] Mehta, S. and Rastegari, M. (2021) MobileViT: Light-Weight, General-Purpose, and Mobile-Friendly Vision Transformer.
- [15] Fu, X., Zhou, R., Yang, X. and Li, C. (2024) Detecting Oracle Bone Inscriptions via Pseudo-Category Labels. *Heritage Science*, **12**, Article No. 107. <https://doi.org/10.1186/s40494-024-01221-5>

附录

伪代码:

1、数据预处理伪代码 1:

算法: 甲骨文图像数据 YOLO 格式转换流程

输入: 原始图像文件夹路径(字符串, 记为 P)

输出:

训练集 / 验证集图像(路径: P/processed_data/images/{train/val})

YOLO 格式标签(路径: P/processed_data/labels/{train/val})

处理日志(D:/甲骨文论文/2_Train.log)

Procedure ConvertToYoloFormat(P: string)

var

valid_pairs: 图像-JSON 对列表; train_set, val_set: 分割后的数据集

Begin

// 1. 初始化目录

创建训练 / 验证图像、标签目录及模型保存目录

// 2. 收集有效数据对

For 文件 in P 中所有文件 Do

If 文件是图像文件 Then

json_path := P + "/" + 文件名(不含扩展名) + ".json"

If json_path 存在 Then

valid_pairs += (文件, json_path)

Else

记录警告日志("图片无对应 JSON")

End If

End If

End For

// 3. 划分训练 / 验证集(80%-20%)

(train_set, val_set) := 数据集划分(valid_pairs, 测试集比例 = 0.2)

// 4. 处理数据集

处理数据集(train_set, "train")

处理数据集(val_set, "val")

End ConvertToYoloFormat

子过程(被调用)

Function 处理数据集(file_pairs: 列表, dataset_type: string)

For (img_file, json_path) in file_pairs Do

Begin

// 读取图像与解析标注

img := 读取图像(P + "/" + img_file)

If img = 空 Then

```

记录错误日志("图像读取失败"); Continue
End If
annotations := 解析 JSON (json_path, 图像宽度, 图像高度)
If annotations = 空 Then
记录警告日志("无有效标注"); Continue
End If
// 保存图像与标签
保存图像(img, P + "/processed_data/images/" + dataset_type + "/" + img_file)
保存标签(annotations, P + "/processed_data/labels/" + dataset_type + "/" + 文件名(不含扩展名) + ".txt")
End For
End 处理数据集
关键辅助函数
Function 解析 JSON (json_path: string, w: int, h: int) returns 标注列表
// 读取 JSON 文件, 提取标注信息
// 验证坐标有效性:  $0 \leq x_1 < x_2 \leq w$  且  $0 \leq y_1 < y_2 \leq h$ 
// 转换为 YOLO 格式: 中心点坐标(x,y)、宽高(w,h) 归一化到 [0,1]
返回 转换后的标注列表
End 解析 JSON
数据预处理伪代码 2:
算法: 甲骨文图像特征提取流程
输入: 原始图像文件夹路径(字符串, 记为 P)
输出:
处理后图像集(路径: P/processed_results/processed_images)
特征表(P/processed_results/features/features.csv)
错误日志(P/processed_results/error_log.csv, 若有错误)
Procedure ImageFeatureExtraction(P: string)
var
img_list: 图像文件列表; error_log, features: 数组; i: 整数
Begin
// 1. 初始化目录与文件列表
创建目录 P/processed_results 及子目录;
img_list := 筛选 P 中格式为.jpg/.png 的文件
// 2. 批量处理图像
For i := 0 To len(img_list)-1 Do
Begin
img_path := P + "/" + img_list[i]
// 预处理(去噪 + 增强)
processed_img := Preprocess(img_path)
If processed_img = 空 Then
error_log += (img_list [i], "预处理失败"); Continue

```

```

End If
// 特征提取(轮廓 + 几何特征)
feats := ExtractFeats(processed_img, img_list[i])
If len(feats) = 0 Then
error_log += (img_list [i], "无有效特征"); Continue
End If
// 保存结果
保存图像(processed_img, 输出目录); features += feats
End For
// 3. 导出结果
保存 CSV (features, 特征表路径); 保存 CSV (error_log, 错误日志路径)
End ImageFeatureExtraction
子过程(被调用)
Function Preprocess (img_path: string) returns 图像
// 去噪: 非局部均值(h=12); 增强: CLAHE(clipLimit=4.0)
返回 去噪(增强(读取图像(img_path)))
End Preprocess
Function ExtractFeats (img: 图像; fname: string) returns 特征数组
// 轮廓筛选(面积 > 100); 特征: 边界框(x,y,w,h)、宽高比 w/h、填充率 S/(w×h)
返回 计算特征(筛选轮廓(二值化(img)))
End ExtractFeats

```

3、yolov8n 训练模型伪代码:

```

Begin
// 1. 初始化模型
model := 加载预训练模型("yolov8n.pt")
// 2. 配置训练参数
epochs := 50; imgsz := 640; batch := 8
lr0 := 0.01; device := "cpu"
// 3. 启动训练
model. 训练(
数据路径 = data_path,
轮次 = epochs,
图像尺寸 = imgsz,
批次 = batch,
初始学习率 = lr0,
设备 = device
)
// 4. 训练完成(模型自动保存)
Output "训练结束, 模型保存至默认路径"
End TrainYOLOv8

```

4、YOLOv8n-cls 训练模型:

Begin

// 1. 初始化 YOLOv8 分类模型

model := 加载预训练模型(model_path)

// 2. 配置并启动训练

model. 训练(

数据 = data_path,

轮次 = epochs,

图像尺寸 = imgsiz,

批次大小 = batch,

初始学习率 = lr0

)

// 3. 输出训练结果路径

Output "训练完成! 模型权重保存在: runs/classify/train/weights/best.pt"

End TrainYOLOv8Classifier

5、YOLOv8n 模型预测:

Begin

// 1. 初始化模型与输出目录

model := 加载预训练模型(model_path)

创建目录(txt_output)

创建目录(img_output)

// 2. 批量预测(使用原图尺寸)

results := model(image_folder, save=true, verbose=false)

// 3. 处理每个预测结果

For result in results Do

Begin

// 提取检测信息

boxes := result.boxes

classes := boxes.cls // 类别 ID

xyxy := boxes.xyxy // 边界框坐标 [x1,y1,x2,y2]

conf := boxes.conf // 置信度

// 获取图像尺寸用于归一化

w, h := result.orig_shape[1], result.orig_shape[0]

// 4. 保存 YOLO 格式检测结果

txt_path := txt_output + "/" + 文件名(无扩展名) + ".txt"

打开文件(txt_path) Do

For i := 0 To len(classes)-1 Do

// 计算归一化中心点和宽高

x_center := (xyxy[i][0] + xyxy[i][2]) / (2w)

y_center := (xyxy[i][1] + xyxy[i][3]) / (2h)

```

width := (xyxy[i][2] - xyxy[i][0]) / w
height := (xyxy[i][3] - xyxy[i][1]) / h
写入 (f"{classes [i]} {x_center:.6f} {y_center:.6f} {width:.6f} {height:.6f} {conf [i]:.6f}\n")
End For
End 打开文件
// 5. 保存带检测框的图像
img := 读取图像(result.path)
绘制矩形(img, boxes.xyxy, 颜色 = 绿色, 线宽 = 2)
保存图像(img, img_output + "/" + 文件名)
6、YOLOv8n-cls 预测:
Begin
// 1. 初始化输出目录与模型
创建目录(output_dir)
model := 加载模型(model_path)
valid_exts := ('.jpg', '.jpeg', '.png', '.bmp', '.tif')
// 2. 遍历处理所有图像
For filename in 获取目录文件列表(input_dir) Do
If 文件名(filename) 以 valid_exts 结尾 Then
img_path := input_dir + "/" + filename
// 3. 读取图像(支持中文路径)
img := 读取图像(img_path)
If img = 空 Then
输出警告("无法读取图像:" + filename)
Continue
End If
// 4. 模型预测
results := model(img)
pred_class_idx := 获取最高概率类别索引(results)
pred_class := 获取类别名称(model, pred_class_idx)
pred_conf := 获取置信度(results)
// 5. 获取正确标签(默认 "未知")
true_class := true_labels.get (filename, "未知")
// 6. 保存标注后的图像(支持中文路径)
output_path := output_dir + "/" + filename
保存图像(img, output_path)
输出成功信息("已处理:" + filename + "→ 保存至" + output_path)
End If
End For
End BatchPredictAndAnnotate.

```