YOLO-Based Object Detection for Tooth Analysis

Executive Summary

This project aimed to enhance dental diagnostics through the application of deep learning, specifically using the YOLOv8 object detection model. We developed a system that detects cavities and plaque in intraoral dental images. Our team created a full pipeline from data preprocessing to model training, testing, and packaging for deployment. This final deliverable includes a polished and documented version of our work ready for reproduction or future extension by clinical researchers or developers.

Project Goals

- Develop a machine learning pipeline capable of detecting dental anomalies from intraoral images.
- Train and evaluate a YOLOv8 model to identify cavities and plaque.
- Create a lightweight, reproducible system deployable by dental professionals or researchers.
- Deliver clear documentation, user instructions, and a demo for end users.

Project Methodology

- **Data Preprocessing**: Stratified splitting of labeled data, normalization, and augmentation using `ImageDataGenerator`.
- **Visual Validation**: Verified class distributions using bar plots.
- **Model Architecture**: Trained a CNN model for binary classification and YOLOv8 for object detection tasks.
- **Model Training**: Used validation loss monitoring, early stopping, and regularization techniques to prevent overfitting.
- **Evaluation**: Tested model on unseen intraoral images, and analyzed performance using accuracy, loss curves, and sample predictions.
- **Packaging**: Project is fully documented and uploaded to GitHub with installation and usage instructions.

Results / Findings

- Successfully implemented a YOLOv8 object detection model to identify two target classes: cavities and plaque.
- Achieved stable validation accuracy (~85%) after 10 epochs of training.
- Identified early overfitting, which was mitigated with L2 regularization and dropout layers.
- Deployed model to test on unseen samples, with real-time inference capabilities.
- Final model and documentation uploaded to GitHub for public access.

Key Outcomes:

- Labeled dataset prepared and visually validated.
- YOLOv8 trained for dental object detection.
- Binary CNN model also explored for baseline classification.
- Project packaged for reproducibility and deployment.
- Presentation and demo materials completed.

Install Instructions

Requirements

- Python 3.10+
- pip
- `ultralytics` (for YOLOv8)
- TensorFlow (for CNN version)
- OpenCV
- Matplotlib
- Jupyter Notebook or VSCode

Installation Instructions

1. Clone the repository:

```
git clone https://github.com/your-username/YOLO-Tooth-Detection.git
cd YOLO-Tooth-Detection
```

2. Create a virtual environment (optional but recommended): python -m venv yolovenv source yolovenv/bin/activate # On Windows: yolovenv\Scripts\activate
3. Install dependencies: pip install -r requirements.txt
4. Download YOLOv8 weights (optional): yolo download model=yolov8n.pt

Getting Started

To Preprocess Images:

1. Place labeled images into the data/raw/ folder.
2. Run: python src/preprocess_images.py

To Train YOLO

1. Ensure your dataset is in YOLO format (images/train, images/val, and labels/).
2. Run: yolo task=detect mode=train model=yolov8n.pt data=dental.yaml epochs=20 imgsz=640

To Make Predictions

1. Place test images into a folder (data/test/).
2. Run: yolo task=detect mode=predict model=runs/detect/train/weights/best.pt source=data/test/