Bone Fracture Detection with RT-DETR

Overview

This repository implements a real-time bone fracture detection system using RT-DETR (PekingU/rtdetr_r50vd) built on PyTorch Lightning. The pipeline covers data preprocessing, model training, evaluation, conversion, and inference. Key deliverables include documented milestones, source code, and evaluation results.

Table of Contents

Milestones

1. **Dataset Preparation**: Organized COCO-format dataset; implemented custom `CocoDetection` and visualization of annotations.
2. **Data Loader & Collation**: Built data pipelines with `DataCollatorForObjectDetection` and custom `collate_fn`.
3. **Lightning Integration**: Wrapped RT-DETR in a PyTorch Lightning module (`RTDetrLightning`) with training, validation, and test steps.
4. **Model Training**: Configured optimizer, learning-rate schedules, checkpointing, and early stopping; executed end-to-end fine-tuning.
5. **Evaluation**: Developed COCO-mAP evaluation script using `pycocotools` and `torchmetrics`; visualized GT vs. predictions.
6. **Model Conversion & Inference**: Exported to ONNX/TorchScript; implemented real-time inference with `RTDetrImageProcessor`.

Requirements

- ipykernel
- supervision==0.3.0
- transformers
- pytorch-lightning
- timm
- cython
- pycocotools
- scipy
- python-dotenv

Data Preprocessing

1. Download and organize images & annotations in COCO format under `data/{train,valid,test}`.
2. Implemented `CocoDetection` (src/dataset.py) to load images, annotations, and apply the `RTDetrImageProcessor` for bounding boxes and labels.
3. Visualize samples using `supervision` to ensure correctness.

Training

1. Configure hyperparameters in `train.py` (learning rates, weight decay, batch size, epochs).
2. Initialize `RTDetrLightning`, callbacks (ModelCheckpoint, EarlyStopping), and `Trainer`.
3. Monitor training & validation losses and COCO mAP.

Evaluation

- Outputs include class-wise AP, mAP@[.50:.95], AR, and precision/recall curves.

Results

Metric	Value
mAP (IoU=0.50:0.95)	0.32
AP@0.50	0.55
AP@0.75	0.28
AP (small)	0.18
AP (medium)	0.35
AP (large)	0.48
AR (maxDets=100)	0.62