Solar Panel Object Detection Project Documentation

Project Overview

This project focuses on object detection of solar panels using deep learning models, specifically leveraging the YOLOv8 model. It involves dataset preparation, model training, evaluation, and inference.

Directory Structure

- **images/**: Contains image data used for training, validation, and testing.
    - `image_chips_native-*`: Subfolder storing preprocessed image chips.
- **labels/**: Holds annotation files in YOLO format, mapping bounding boxes to the respective images.
- **mlruns/**: Stores logs and metadata related to MLflow experiment tracking.
- **train/**: Directory containing training images and their corresponding labels.
- **val/**: Directory containing validation images and their corresponding labels.
- **test/**: Directory containing test images and their corresponding labels for evaluating model performance.
- **solar_dataset.yaml**: Configuration file specifying dataset structure, classes, and paths for YOLO model training.
- **solar-p-detection.ipynb**: Jupyter Notebook for training and evaluating the solar panel detection model.
- **solar-panel.ipynb**: Jupyter Notebook for additional analysis, including visualization and performance metrics.
- **yolov8n.pt**: Pretrained YOLOv8 model weights used for fine-tuning on the solar panel detection dataset.

Workflow

1. **Dataset Preparation**
    - Organize images and annotation files.
    - Convert dataset into YOLO format if necessary.
    - Update `solar_dataset.yaml` with dataset paths and class details.
2. **Model Training**

    - Load YOLOv8 model (`yolov8n.pt`).
    - Train using `solar-p-detection.ipynb`.
    - Log training performance using MLflow.
3. **Evaluation**
    - Compute metrics (IoU, mAP) using Pascal VOC and COCO methods.
    - Analyze results in `solar-panel.ipynb`.
4. **Inference & Deployment**
    - Run detection on test images.
    - Deploy the model for real-time solar panel detection.

Dependencies

- Python 3.x
- Ultralytics YOLOv8
- OpenCV
- Pandas
- Matplotlib
- MLflow
- Shapely (for IoU calculation)
- supervision (for evaluation metrics)

Future Scope

- Improve model accuracy with data augmentation.
- Optimize inference for real-time applications.
- Integrate the detection pipeline with IoT-based solar panel monitoring systems.

Author: Ashutosh Ranjan **Date:** 28 February 2025