

Object Detection Using YOLOv8 on KITTI 2D Dataset

Project Overview

The objective of this proof of concept (POC) project is to create an object detection model using the KITTI 2D dataset. The model aims to accurately detect various objects in normal weather conditions using the YOLOv8 architecture, leveraging the Ultralytics framework for model development. Additionally, the model is deployed on the NVIDIA Jetson Orin Developer Kit for real-time inference.

Detailed Description

Background

Object detection is a crucial task in computer vision with applications in autonomous driving, surveillance, and robotics. The KITTI 2D dataset, which provides labeled images of urban scenes, is a benchmark dataset widely used for evaluating object detection models. YOLO (You Only Look Once) is a state-of-the-art object detection framework known for its speed and accuracy. YOLOv8, the latest version, improves upon previous versions by incorporating advanced techniques and optimizations. Ultralytics, the team behind YOLOv8, has developed an easy-to-use framework that simplifies the process of implementing and training object detection models.

Objectives

The primary goal of this project is to develop and validate an object detection model capable of accurately identifying and localizing various objects such as cars, pedestrians, and cyclists in urban environments. The specific objectives include:

1. **Data Preparation:** Preprocess the KITTI 2D dataset to be compatible with the YOLOv8 framework, including annotation conversion and data augmentation.
2. **Model Training:** Train the YOLOv8 model using the Ultralytics framework on the prepared dataset, fine-tuning hyperparameters to optimize performance.
3. **Model Evaluation:** Evaluate the model's performance using standard metrics such as precision, recall, and mean Average Precision (mAP).
4. **Model Deployment:** Convert the trained model to ONNX format and deploy it on the NVIDIA Jetson Orin Developer Kit for real-time inference.
5. **Result Analysis:** Analyze the results to identify strengths and weaknesses, and propose improvements.

Dataset Details:

- **Training Set:** 7481 images
- **Test Set:** 7518 images
- **Classes:** Car, Van, Truck, Pedestrian, Person_sitting, Cyclist, Tram, Misc and Don't Care

key	value	Description
Type	1	String describing the type of object: [Car, Van, Truck, Pedestrian, Person_sitting, Cyclist, Tram, Misc or Don't Care]
Truncated	1	Float from 0 (non-truncated) to 1 (truncated), where truncated refers to the object leaving image boundaries
Occluded	1	Integer (0,1,2,3) indicating occlusion state: 0 = fully visible 1 = partly occluded 2 = largely occluded 3 = unknown
Alpha	1	Observation angle of object ranging from $[-\pi, \pi]$
BBox-coordinates	4	2d Bounding box coordinates

- Converted the dataset annotations to the YOLO format.
- Applied data augmentation techniques to enhance model generalization.

Model Training:

- Set up the YOLOv8 environment using the Ultralytics framework.
- Configure the YOLOv8 model with appropriate hyperparameters.
- Trained the model on the training set, used the validation set for hyperparameter tuning.
- Implemented early stopping and learning rate scheduling to optimize training.

Model Evaluation:

- Used the test set to evaluate the trained model.
- Calculated performance metrics: precision, recall, F1-score, and mAP.
- Visualized detection results to qualitatively assess model performance.

Model Deployment:

- **Convert to ONNX:** Use the Ultralytics framework to converted the trained YOLOv8 model to ONNX format.
- **Deploy on NVIDIA Jetson Orin:** Load the ONNX model onto the NVIDIA Jetson Orin Developer Kit and set up the inference pipeline for real-time object detection.

Result Analysis and Reporting:

- Analyzed the performance metrics to identify areas of improvement.
- Compared the results with baseline models and previous works.
- Document the findings and propose future work for enhancing the model's performance.

Expected Outcomes

- A trained YOLOv8 model capable of accurately detecting and localizing objects in images from the KITTI 2D dataset under normal weather conditions.
- A comprehensive report detailing the data preparation process, model training, evaluation metrics, and analysis of the results.
- A deployed model on the NVIDIA Jetson Orin Developer Kit demonstrating real-time object detection capabilities.
- Recommendations for further improvements and potential applications of the developed model in real-world scenarios.

Tools and Technologies

- **Programming Languages:** Python
- **Deep Learning Frameworks:** Ultralytics YOLOv8
- **Libraries:** OpenCV, Albumentations, Matplotlib, ONNX
- **Hardware:** GPU-enabled system for training, NVIDIA Jetson Orin Developer Kit for deployment
- **Dataset:** KITTI 2D Object Detection Dataset

Conclusion

This POC demonstrates the feasibility of using YOLOv8 for object detection on the KITTI 2D dataset. Transfer learning proved to be an effective approach for training the model in a reasonable timeframe while achieving good accuracy.

Reference

I'd like to give a shout out to the Ultralytics team for their exceptional work on the YOLOv8 model and the intuitive framework they have developed for object detection. The advancements in YOLOv8 have significantly improved both speed and accuracy, making it an invaluable tool for our projects. The ease of use and comprehensive documentation provided by Ultralytics have greatly simplified the process of implementing state-of-the-art object detection models. Their dedication to creating high-quality, accessible tools is truly commendable. Thank you, Ultralytics, for your outstanding contributions to the field of computer vision!