Pedestrian Detection using DINO Object Detection Model

Author: Arindam Chakrabory

Date: September 24, 2024

1. Introduction

This project aims to detect pedestrians using the DINO object detection model. The task was part of the Computer Vision and Machine Learning Internship at the Vision and Graphics Lab, Department of CSE, IIT Delhi. The dataset consists of 200 images annotated in COCO format and was collected on the IIT Delhi campus. The task involves fine-tuning a pre-trained DINO model and evaluating its performance on a custom pedestrian detection dataset.

DINO, which stands for "DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection," is a state-of-the-art object detection model designed to improve upon the vanilla DETR model with better precision and efficiency in detecting objects.

2. Dataset Preparation and Visualization

The dataset consists of 200 images, with bounding box annotations for pedestrians. The following steps were taken to prepare the dataset:

Split the dataset:

- Training set: 160 images.
- Validation set: 40 images.

The splitting process was automated using the script train_val_split.py, ensuring that the dataset was correctly separated for model training and evaluation

Format of the Dataset Directory for Training the model:

```
custom_dataset/
    ├── train_images/
    ├── val_images/
    └── annotations/
        ├── instances_train_imagesd.json
        └── instances_val_images.json
```

2.1 Bounding Box Visualizations

The bounding box visualizations were generated using the bb_visualize.py script. Here, red bounding boxes around detected pedestrians confirmed accurate detections in most cases.

Figure 1:Visualization of bounding box

Figure 2:Visualization of bounding box

3. Model Architecture and Setup

3.1 DINO Model Overview

DINO uses a transformer-based architecture similar to DETR but with improved de-noising techniques for better accuracy and faster convergence. For this task, we use the pre-trained DINO-4scale model with a ResNet-50 (R50) backbone, which has been proven effective in detecting objects like pedestrians in crowded environments.

3.2 Repository and Environment Setup

Mentioned in README.md

4. Experimentation and Results

4.1 Initial Evaluation of Pre-trained Model

The DINO-4scale model was evaluated on the validation set using pre-trained weights. The bounding box Average Precision (AP) values were computed using COCO metrics.

Metric	Value
Bounding Box AP (IoU=0.50:0.95, area=all)	52.8
Bounding Box AP (IoU=0.50, area=all)	88.2
Bounding Box AP (IoU=0.75, area=all)	59.3
Average Recall (IoU=0.50:0.95, maxDets=100)	61.8
Average Recall (IoU=0.50:0.95, area=medium)	70.3
Average Recall (IoU=0.50:0.95, area=large)	71.2

The initial results showed strong performance for detecting pedestrians in both medium and large objects, with slightly lower performance for smaller objects

Figure 3:Pre-Trained model Predictions (for only person class)

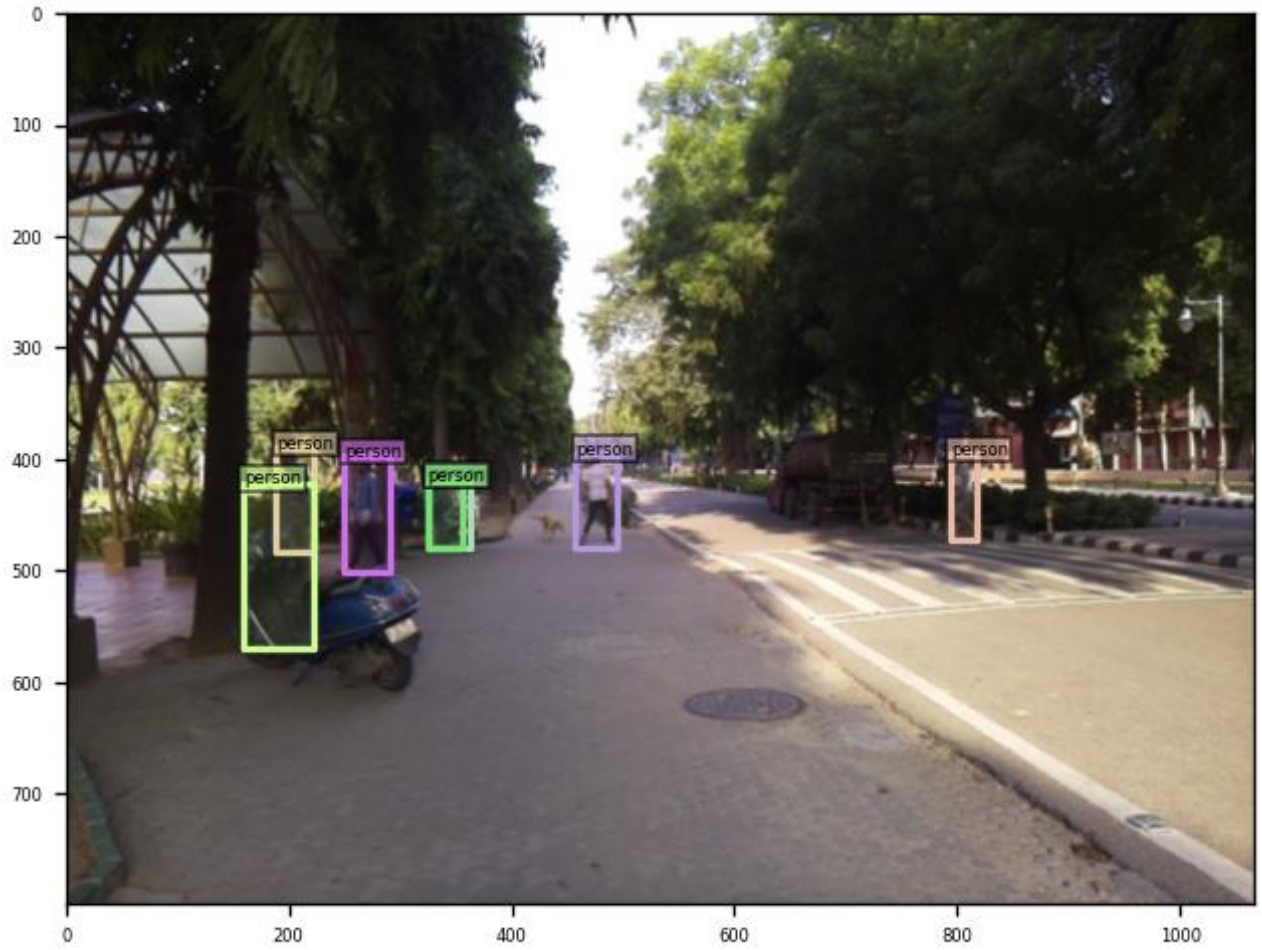

Figure 4:Pre-Trained model Predictions (for only person class)

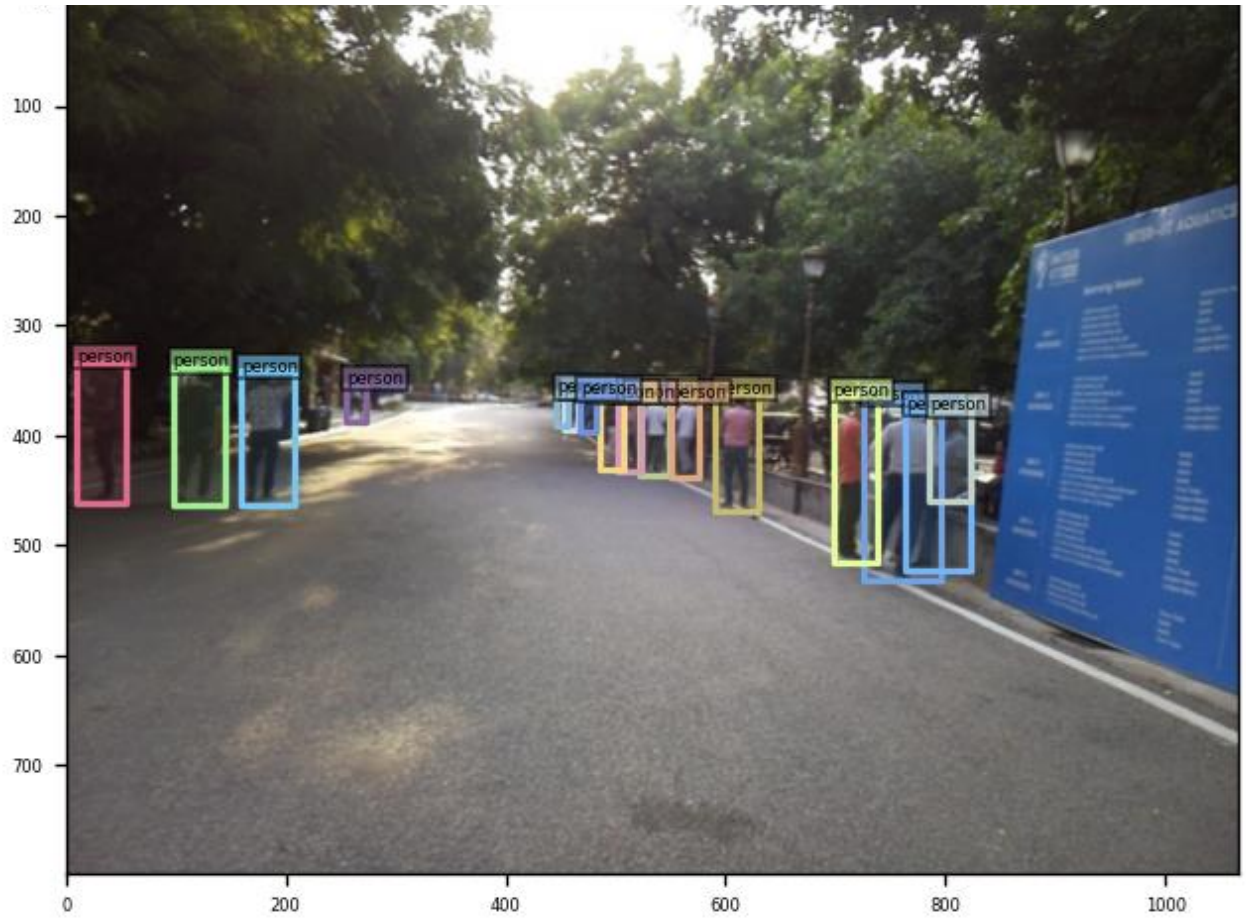

Figure 6: Pre-Trained model Predictions (for only person class)

4.2 Fine-tuning the Model

To improve the model's performance on the custom dataset, the pre-trained DINO model was fine-tuned on the training set (160 images). The fine-tuning process was carried out by:

4.3 Fine-tuned Model Evaluation

After fine-tuning, the performance of the model improved:

Metric	Value
Bounding Box AP (IoU=0.50:0.95, area=all)	58.3
Bounding Box AP (IoU=0.50, area=all)	91.1
Bounding Box AP (IoU=0.75, area=all)	67.6
Average Recall (IoU=0.50:0.95, maxDets=100)	70.4
Average Recall (IoU=0.50:0.95, area=medium)	78.1
Average Recall (IoU=0.50:0.95, area=large)	80.6

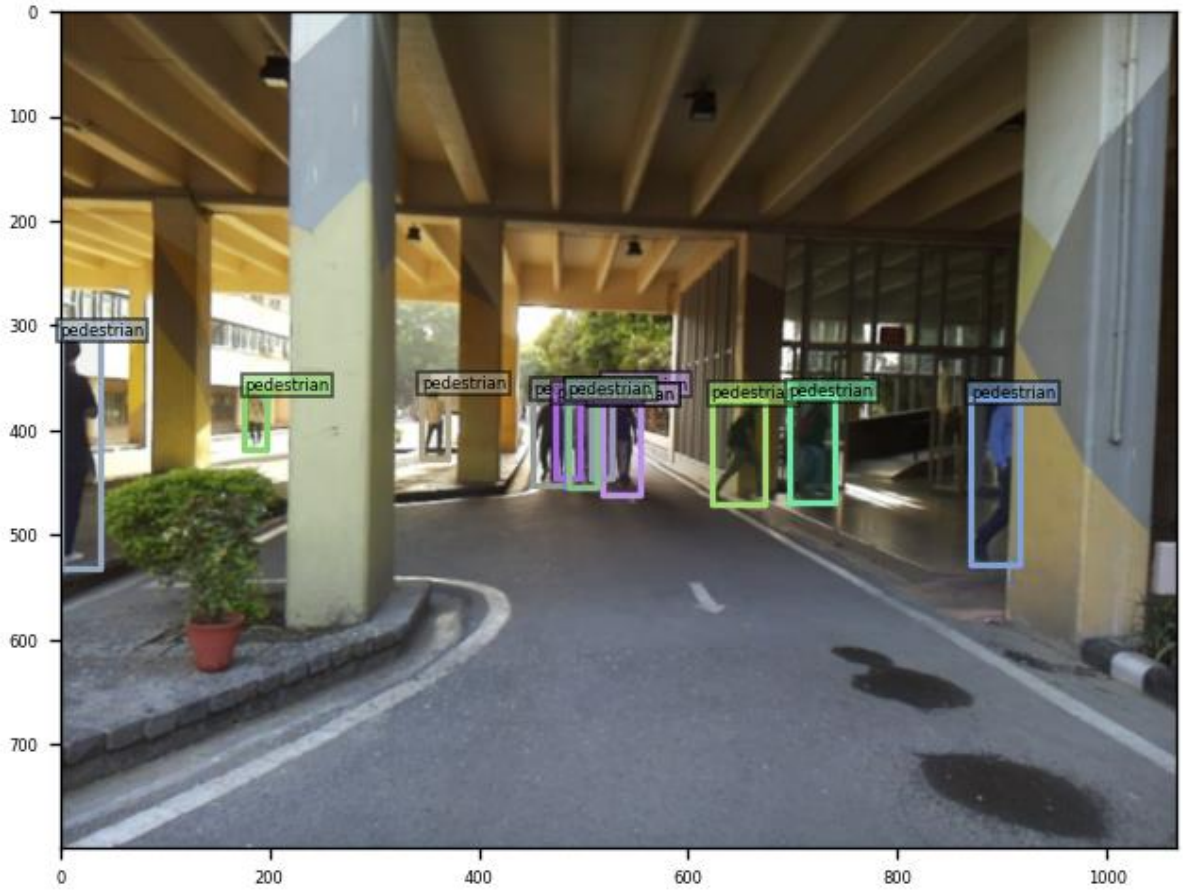

Figure 7: Finetuned Model's Prediction

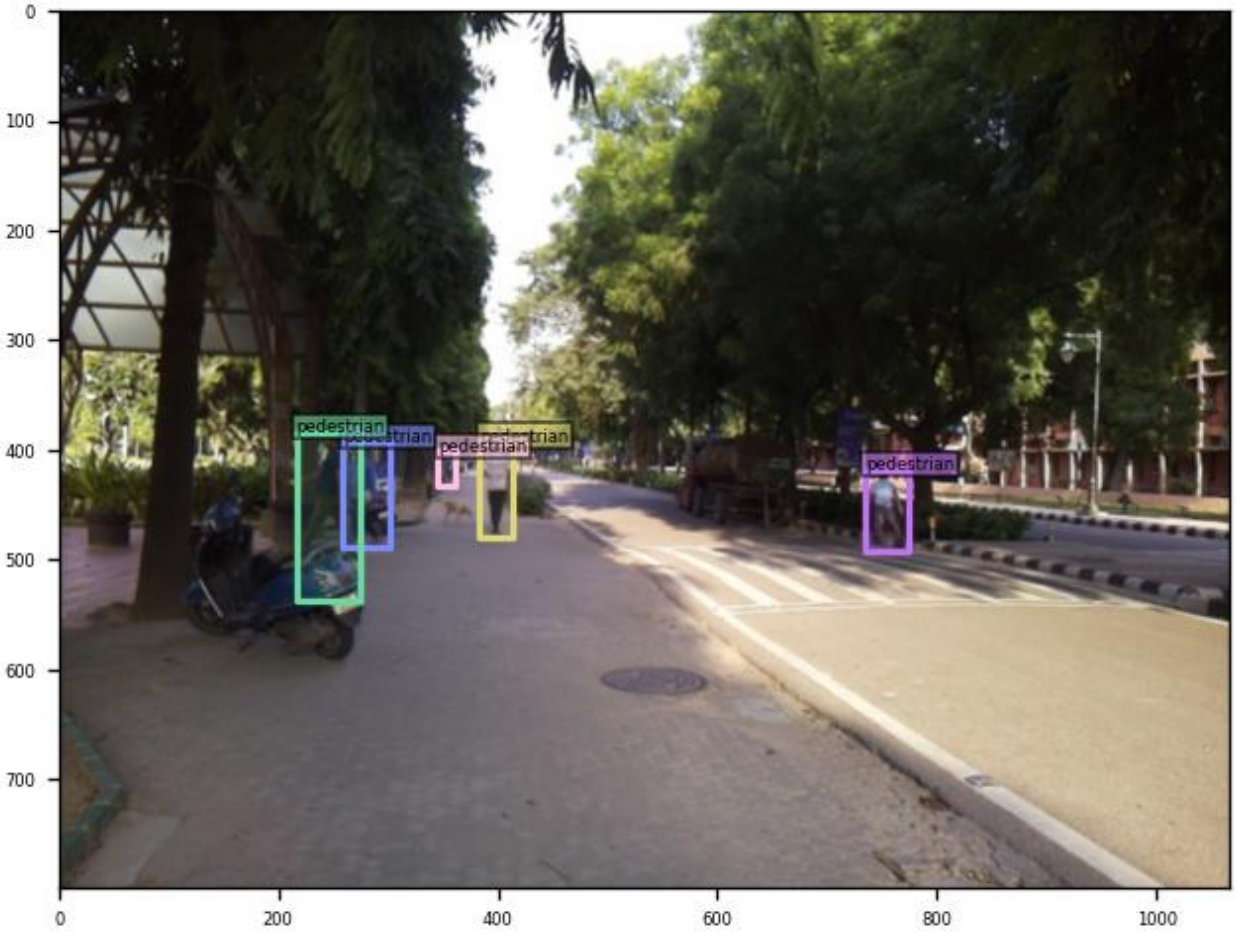

Figure 8: Finetuned Model's Prediction

Figure 9: Finetuned Model's Prediction

4.4 Loss Graphs

Loss values during training showed a clear decrease, indicating that the model was learning effectively:

Class Error:	0.00
Total Loss:	3.9371
Loss (bbox):	0.0565
Loss (giou):	0.3790
Loss (ce):	0.0692

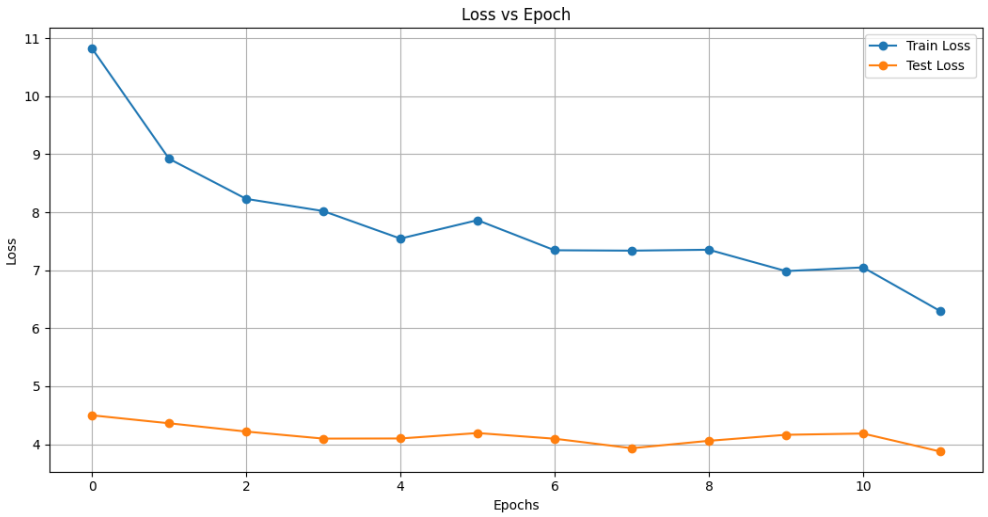

Figure 10: Loss vs Epoch

The final model exhibited improvements across most metrics, especially in high Intersection over Union (IoU) scenarios, demonstrating better precision in detecting small pedestrian objects.

5. Error Analysis

5.1 Failure Cases

Despite the improvements, some failure cases were identified:

Occlusions: Pedestrians partially blocked by objects were sometimes missed.

Complex Backgrounds: Some instances with busy or cluttered backgrounds led to missed detections or false positives.

Small Objects: Performance for very small objects was lower than for medium and large objects.

These issues may be improved with additional data augmentation, longer fine-tuning, or using a more robust backbone for small object detection.

6. Conclusion and Future Work

The DINO object detection model, fine-tuned on the custom pedestrian dataset, achieved promising results. Fine-tuning led to significant improvements in average precision, recall, and loss values. However, there were still challenges with detecting small pedestrians and handling complex occlusions.

Future Work:

Data Augmentation: Applying additional augmentation techniques like random cropping or color jittering might help improve model robustness.

Larger Dataset: Increasing the size of the training dataset would likely enhance the model's generalization ability.

Advanced Architectures: Exploring larger backbones (e.g., ResNet-101) or other transformer-based detection models could yield further improvements.

7. References

DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection