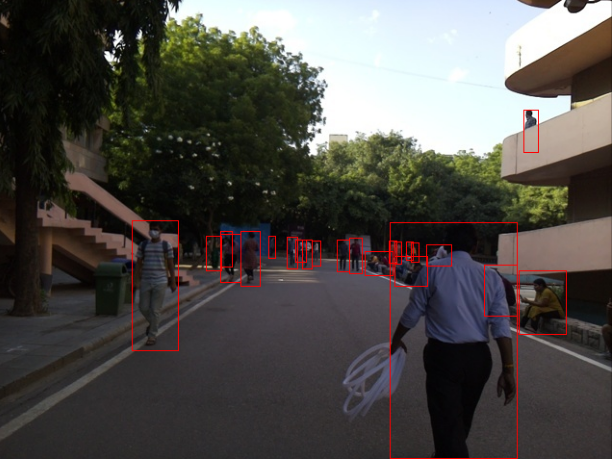
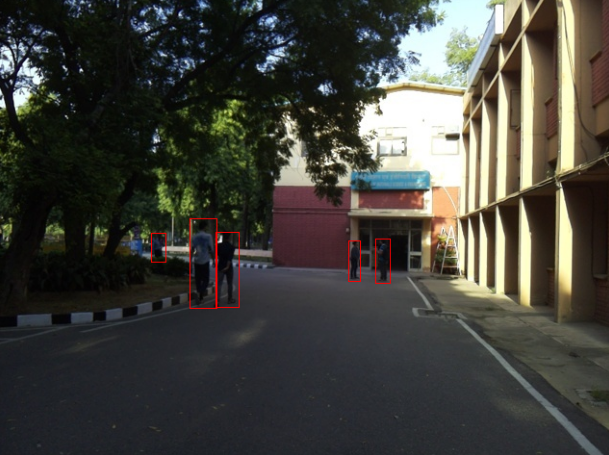
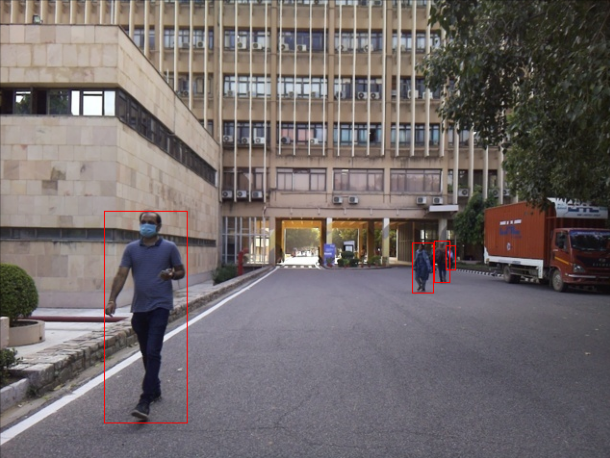
**CV\_Intern\_Visionlab\_IITD\_Assignment\_Sep\_2024**

This report outlines the experiments and results of the assignment given by IITD. It involved the evaluation, analysis and fine tuning of the DINO object detection model on a pedestrian dataset containing 200 images.

**Ground Truth Visualization**









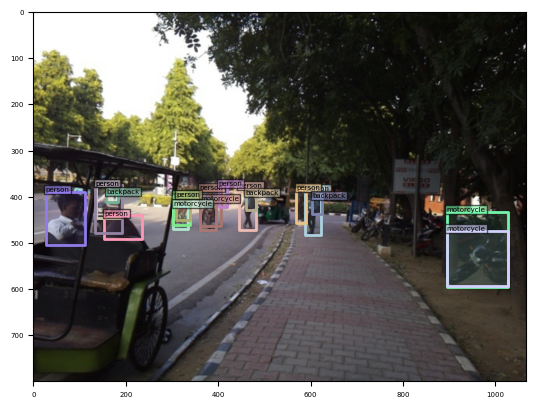
**Inference and Evaluation on pre-trained weights**

Bounding box AP values with pre-trained model:

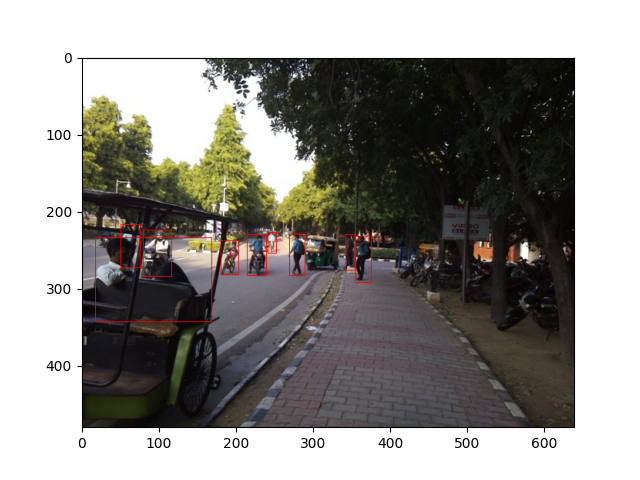


**Analysis**

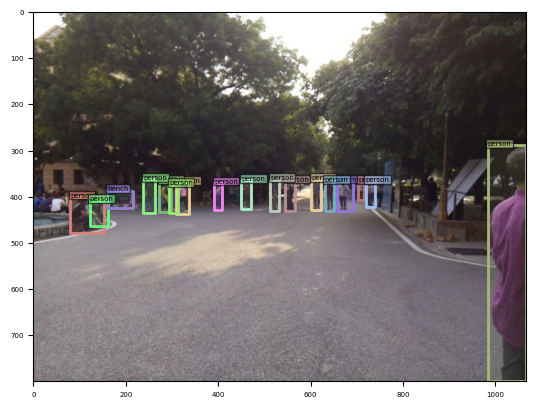
Inference with pre-trained model (checkpoint0033\_4scale.pth):



Ground truth:



* Results are fairly consistent with the ground truth. Larger objects are detected with more average precision while smaller objects with lesser AP.
* False negatives for people who are sitting and are farther away (see middle left)



**Errors encountered**

* DINO\_train.sh and DINO\_eval.sh needed specific versions of numpy and yapf to work. Solved it by “pip install numpy==1.23.5 yapf==0.40.1”
* While setup, test.py gave CUDA out of memory error for dimensions higher than 71 in the following code:

for channels in [30, 32, 64, 71, 1025, 2048, 3096]:

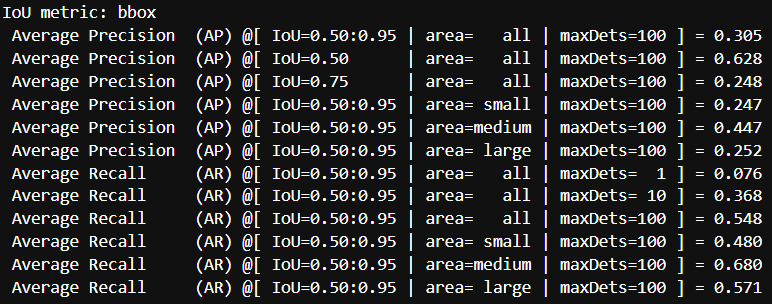
check\_gradient\_numerical(channels, True, True, True)

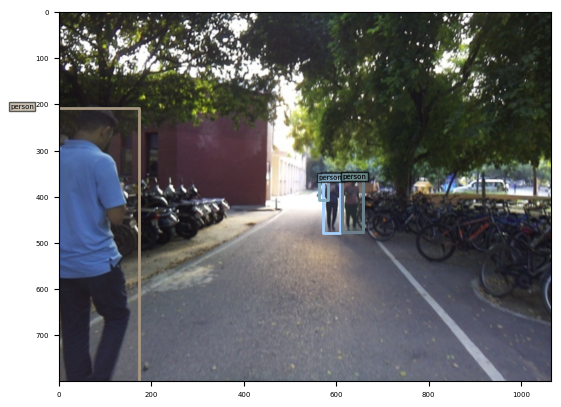
Since it was only a testing code, I removed dimensions >= 1025

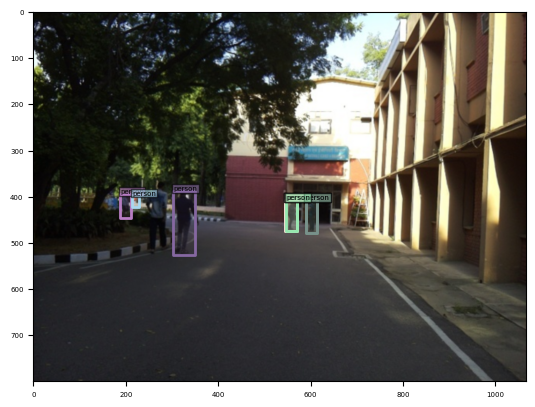
* Few config errors that were resolved by correcting the params (num\_classes, dn\_labelbook\_size, etc)
* When trying to add the –save\_results flag in DINO\_train.sh and DINO\_eval.sh, it threw a RuntimeError: Sizes of tensors must match except in dimension 1. This error was caused in engine.py within the “if arg.save\_results:” block. Couldn’t resolve it in time.

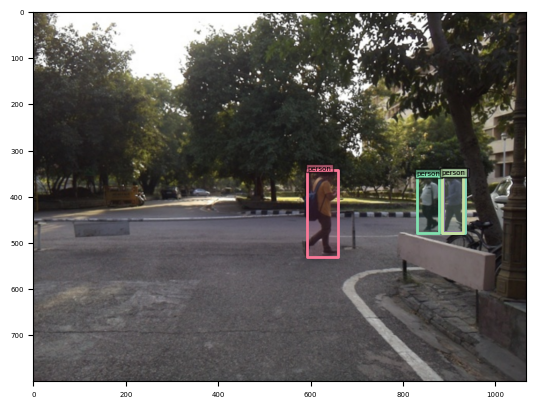
**Inference and Evaluation on fine-tuned weights**

* lr=0.00025
* epochs=30





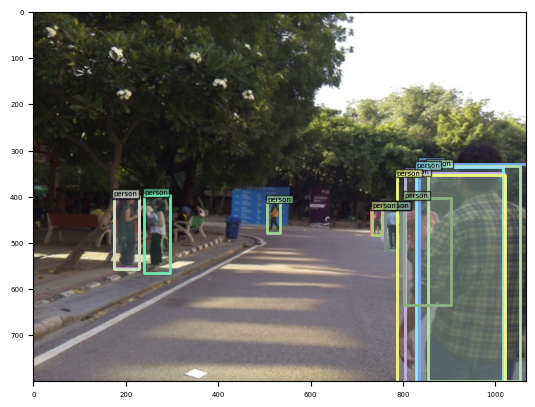




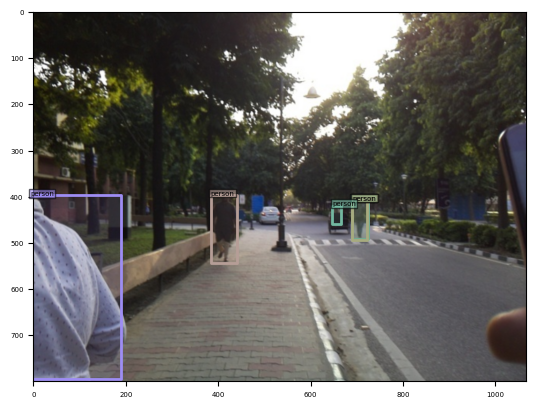
**Observations:**

* Lots of false positives (overlapping bounding boxes) for small and large objects. Possibly, due to IOU threshold value.

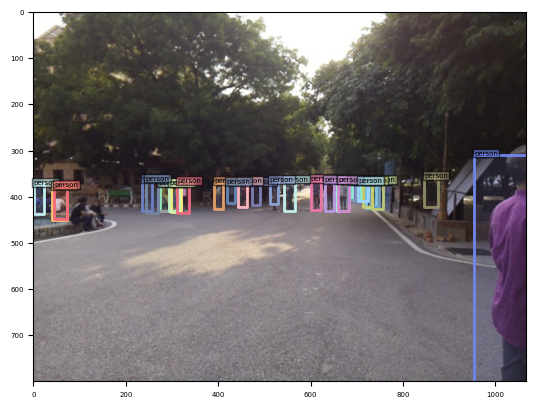




* Low sample size but occlusion handled well in this case:



* False negative for sitting people



Due to time constraints (practical exam on 23rd Sept), I wasn’t able to experiment more. If I could, I would’ve experimented by doing the following:

* Lower learning rate, around 0.0001 (I used 0.00025)
* If I had access to more GPU memory, I would have increased batch size (I used batch size of 2)
* Since the dataset is smaller, I would have run it for more epochs (I ran it for 30 epochs)
* In DETR based models, num\_queries generally refers to the number of object queries used in the model. It correlates to the number of objects the model can detect in a single image. Default value is 900. I would’ve tried with a smaller value (200-500)
* Since it’s a small dataset which I’m running for more epochs, I would slightly increase weight decay to prevent overfitting. Weight decay adds a penalty term to the loss function that is proportional to the sum of the squared weights in the model.