Name - Rakshith Mahishi **Role - Junior AI Engineer**

1. Labelling Process:
As a part of the labelling process, I used the extracted frames and fed it to DINO model(not finetuned) to label the dataset with the common items found. Then once that was done, I uploaded the labelled dataset from DINO to roboflow to label the images manually with the required labels.

I kept the labelling process simple and to the point since it was manual labelling I had to spend some time to carefully label. One hard part about this was labelling the bottles in the customer's hand which was small and hard for any model to detect so going through almost 775 images to label them was a challenge for me.
Class Distribution: Initial dataset showed severe class imbalance where the person had 1,639, refrigerator had 1,067, backpack 153, and bottle 145.
In order to balance the above i had 3 options:

a. Data augmentation to train on more examples to detect the bottle in majority of the cases
b. Peanlize the weights for bottle more than usual to make the learning better
c. Fine-tune the model post training to detect bottles.

I chose option A, considering the time constraint and resource constraints.

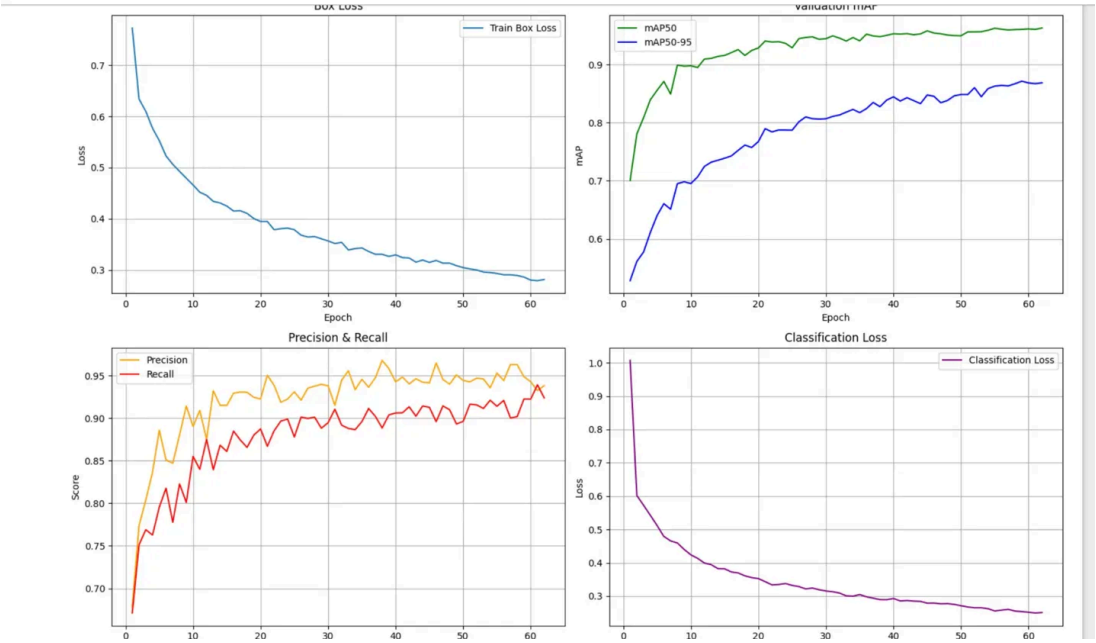
2. Training Setup:
Used the model YOLOv8s available in the ultralystics repo provided.
Compute: Google Colab T4 GPU (12GB VRAM), training time was about 50 minutes per experiment.

Data Augmentation Strategy:
Applied 8 augmentation techniques (flip, brightness, contrast, HSV shifts, blur, noise) specifically to the bottle class.
Increased bottle instances from 145 → 2,688
Final training dataset: 13,752 instances across all classes.

Hyperparameters:
Epochs: 100 (early stopping patience: 15 in case of no improvement)
Batch size: 16
Image size: 640×640
Optimizer: SGD with learning rate = 0.01

3. Results:
Split Dataset: Standard train/val/test split - resulted in poor validation of about overall 67.5%.

Combined Dataset: Merged all splits for maximum training data - achieved 87.6% overall mAP50-95.



4. Challenges and Limitations:
   a. Bottle Detection Challenge
   b. Small object size in top-down camera view (bottles in hand ~20-40 pixels)
   c. Partial occlusion by hands/body
   d. Limited validation examples (only 6 instances)
   e. Training mAP score can be higher, but it cannot be validated properly
   f. Class Imbalance
   g. Camera angle makes the small objects (bottles, backpacks) challenging to detect compared to larger objects (person, refrigerator).

5. Conclusion

Overall, I got to work on YOLO from scratch instead of just using it to inference i actually learnt a few things about how the model itself and how we can also leverage world models for object detection in the future. The results were conclusive and everything has been pushed to github and the drive link will be provided in the README.