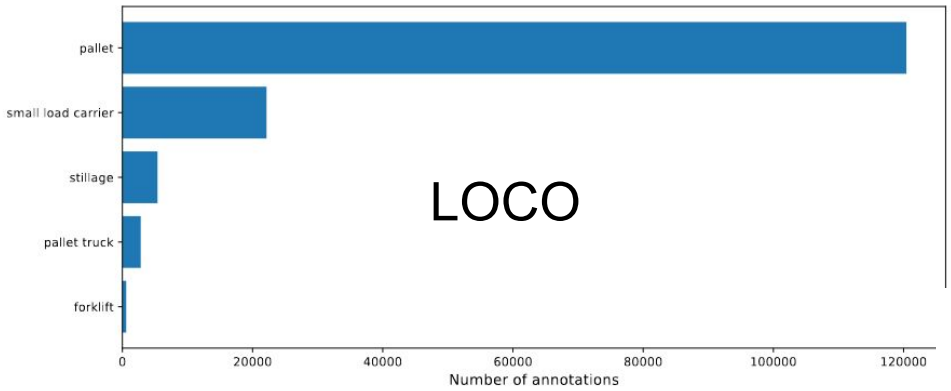
Object Detection with LOCO

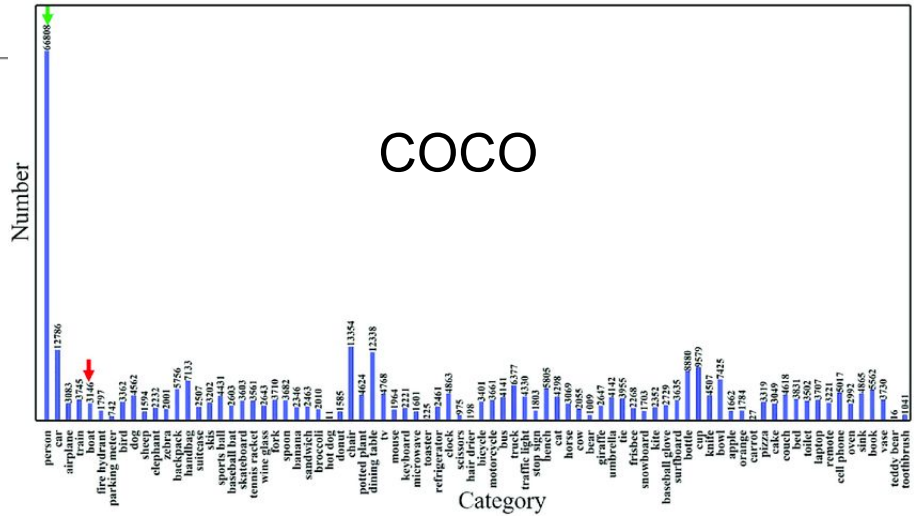
in PyTorch

Timeline

- **13.06:** Task received
- **14.06:** Extension obtained until 22.06
- **15.06:** Registered Paperspace account, bought Pro version; started reading into the task; first attempts with PyTorch; mainly research, initial problems with LOCO dataset
- **15.06 - 19.06:** Traveled to Maastricht (little free time); attempted solutions and problem solving
- **20.06:** Dataset problems largely resolved, first functioning learning pipeline, still errors in the code
- **21.06:** Debugging; code refinement; first results and result analysis
- **21.06:** Further fine-tuning; visualizing results; final finish

LOCO: Logistics Objects in Context Dataset





LOCO: Logistics Objects in Context Dataset

Deep Learning Model: Faster R-CNN

- far from current SOTA (according to https://paperswithcode.com/sota/object-detection-on-coco)
  - Faster R-CNN on COCO = 43.9 mAP
  - current SOTA ≈ 60 mAP (InterImage=65.4mAP; YOLOv7-E6E=56.8mAP)
- But:
  - works “out of the box” in PyTorch & is already pre-trained
  - Faster R-CNN was also used in the LOCO paper
  - goals:
    - build a learning pipeline in PyTorch for LOCO
    - try to rebuild the results of the paper

TABLE II

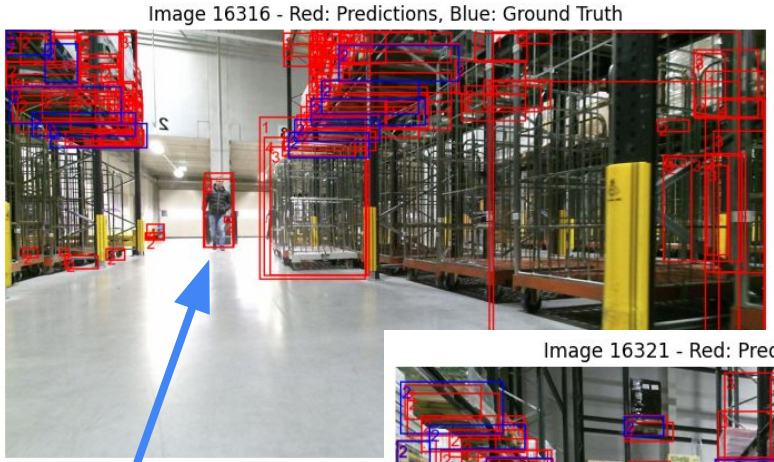
PRELIMINARY ANALYSIS. TABLE SHOWS EVALUATION RESULTS FOR *YOLOv4-608*, *YOLOv4-tiny* AND *Faster R-CNN* TRAINED ON *LOCO*.

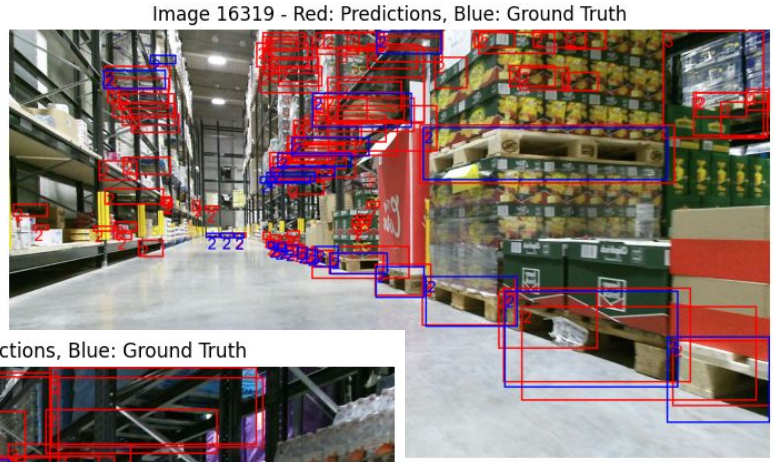
Model	*YOLOv4-608*		*YOLOv4-tiny*		*Faster R-CNN*	
Dataset	*LOCO*	*COCO*	*LOCO*	*COCO*	*LOCO*	*COCO*
mAP@0.50	41.0%	65.7%	22.1%	40.2%	20.2%	60.0%
Small load carrier	27.7%	N/A	18.1%	N/A	28.3%	N/A
Pallet	65.0%	N/A	36.2%	N/A	19.8%	N/A
Stillage	53.1%	N/A	31.3%	N/A	37.6%	N/A
Forklift	31.3%	N/A	11.6%	N/A	2.9%	N/A
Pallet truck	28.1%	N/A	13.3%	N/A	12.5%	N/A

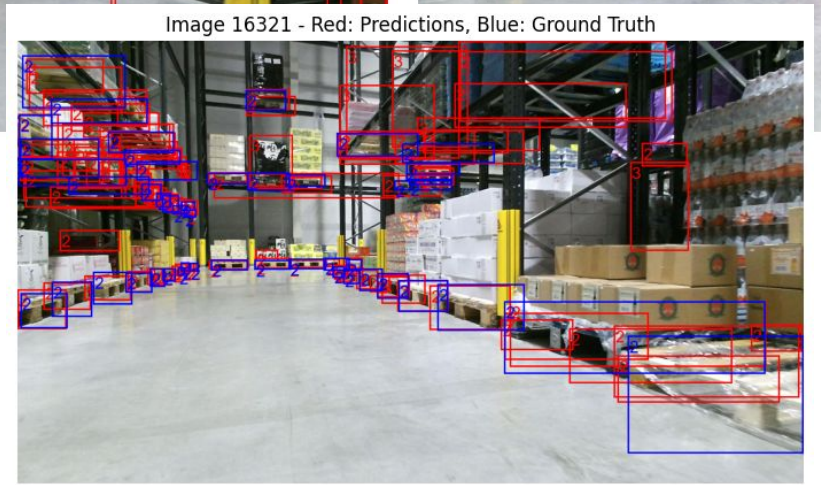
Faster R-CNN - Results on Test

- Class counts in predictions: 1: 1910; 2: 101439; 3: 3145; 4: 2397
- Class counts in ground truths: 1: 119; 2: 39022; 3: 29; 4: 559
- mAP: 9.4%
- mAP / Class:
  - Class Small Load Carrier: 2.19%
  - Class Pallet: 18.55%
  - Class Stillage: N/A
  - Class Forklift: 0.048%
  - Class Pallet Truck: 17.09%
- **mAP@.50: 24.3%**
- mAP@.75: 5.2%

Faster R-CNN - Results on Test







Outlook

- since Faster R-CNN not SOTA anymore, use other models
- YOLOv7 (56.8mAP @ COCO) seems to have an already existing and promising training pipeline
  - https://github.com/WongKinYiu/yolov7
- Refine the Dataset
  - add other classes that might be important in a Logistics environment as well (e.g. Persons, Cars, Trucks)
  - more pictures of forklifts, pallet trucks and stillages necessary (likely not in our scope?)