Analysis of Few Shot Object Detection with Fine-tuning

Chetan Reddy Jean-Bernard Uwineza Sayak Nag

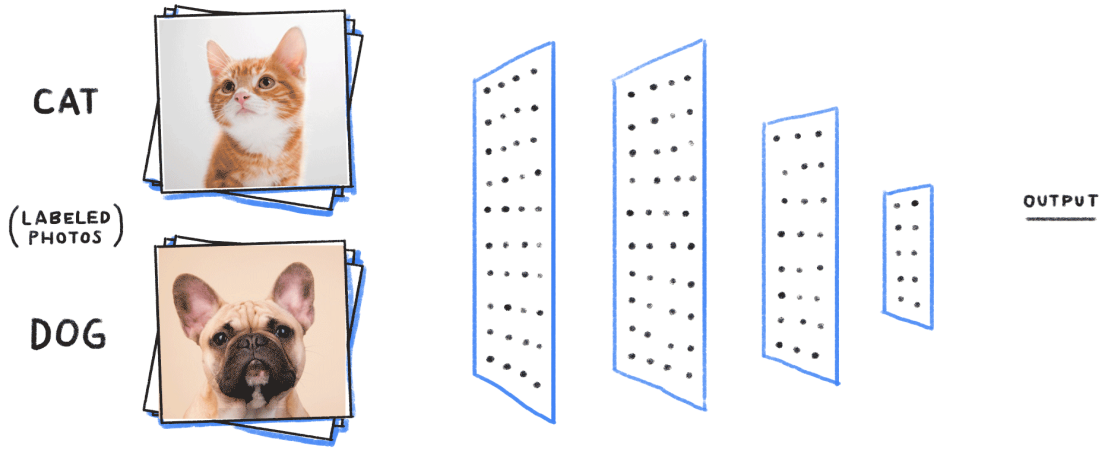
Om Shankar Ohdar Vikarn Bhakri

EE260: Intro to Deep Learning

UC RIVERSIDE | UNIVERSITY OF CALIFORNIA
Electrical and Computer Engineering

What is Image Classification?

- **Object Classification** involves recognizing specific pre-specified objects or object classes.



Images are classified into different classes based on object they contain.

It could be **binary**, or **multi-class**.

What is Object Localization?

- **Object Localization** aims to locate the main (or most visible) object in an image.

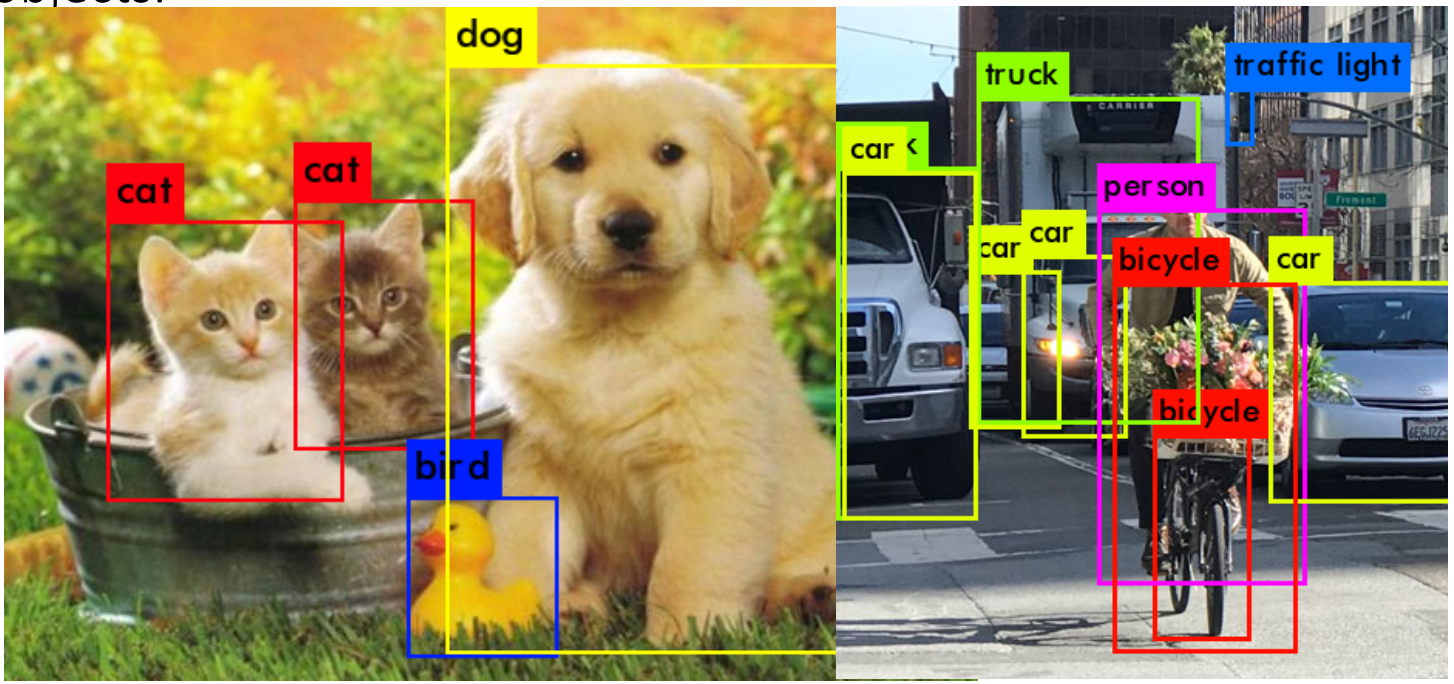
CAT CAT

Different from image classification as it require both classification of an object and its localization.

A **Bounding box** is usually put around the recognized object.

Object Detection

- **Object Detection** combines classification and localization for all objects.



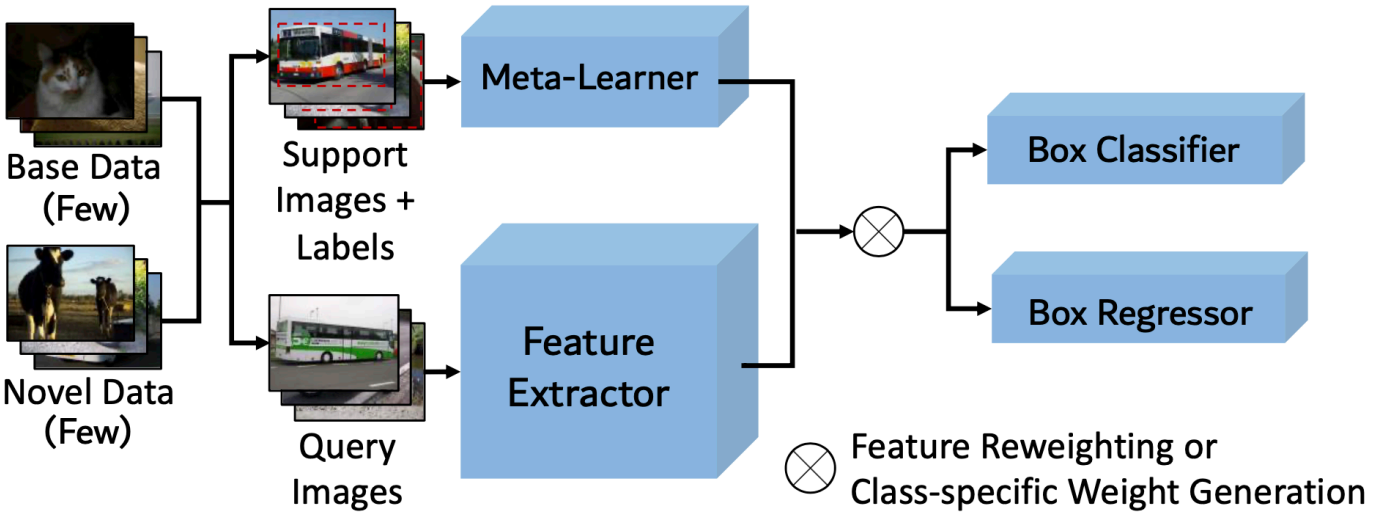
Each recognized object is given a **location (bounding box)** and **a class label.**

Few Shot Detection

- Training a detector requires many training examples per class.
- To introduce new classes, there should be many examples to **re-train** or **fine-tune** the model.
- **Few Shot Detection** is an active area that attempts to use as few training examples for **novel classes** of objects.
- There are two prominent approaches:

  Meta-learning & Few-shot finetuning

Meta-Learning

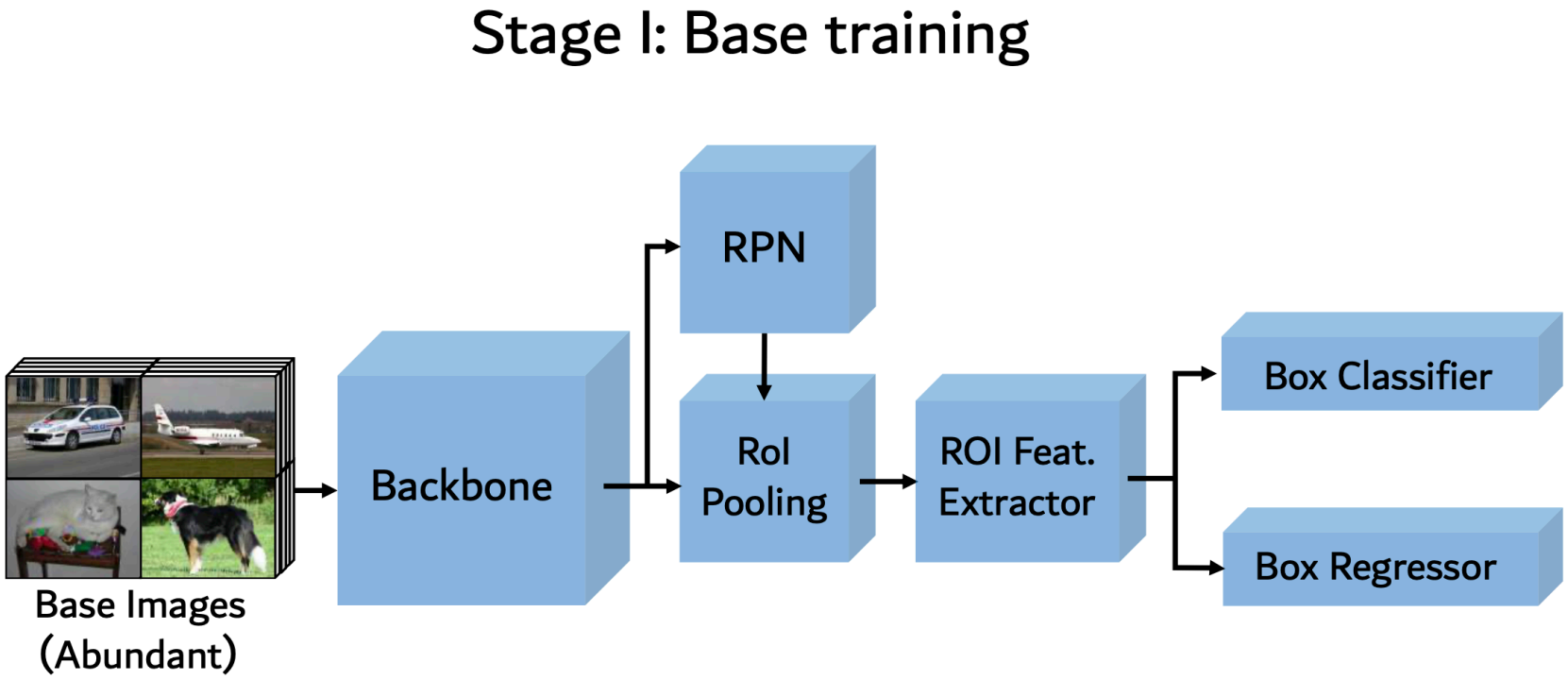


A meta-learner obtains class-level meta features.

The model generalizes to novel classes through **feature re-weighting** (Kang et al. 2019) or **class specific weight generation** (Yan et. al. 2019).

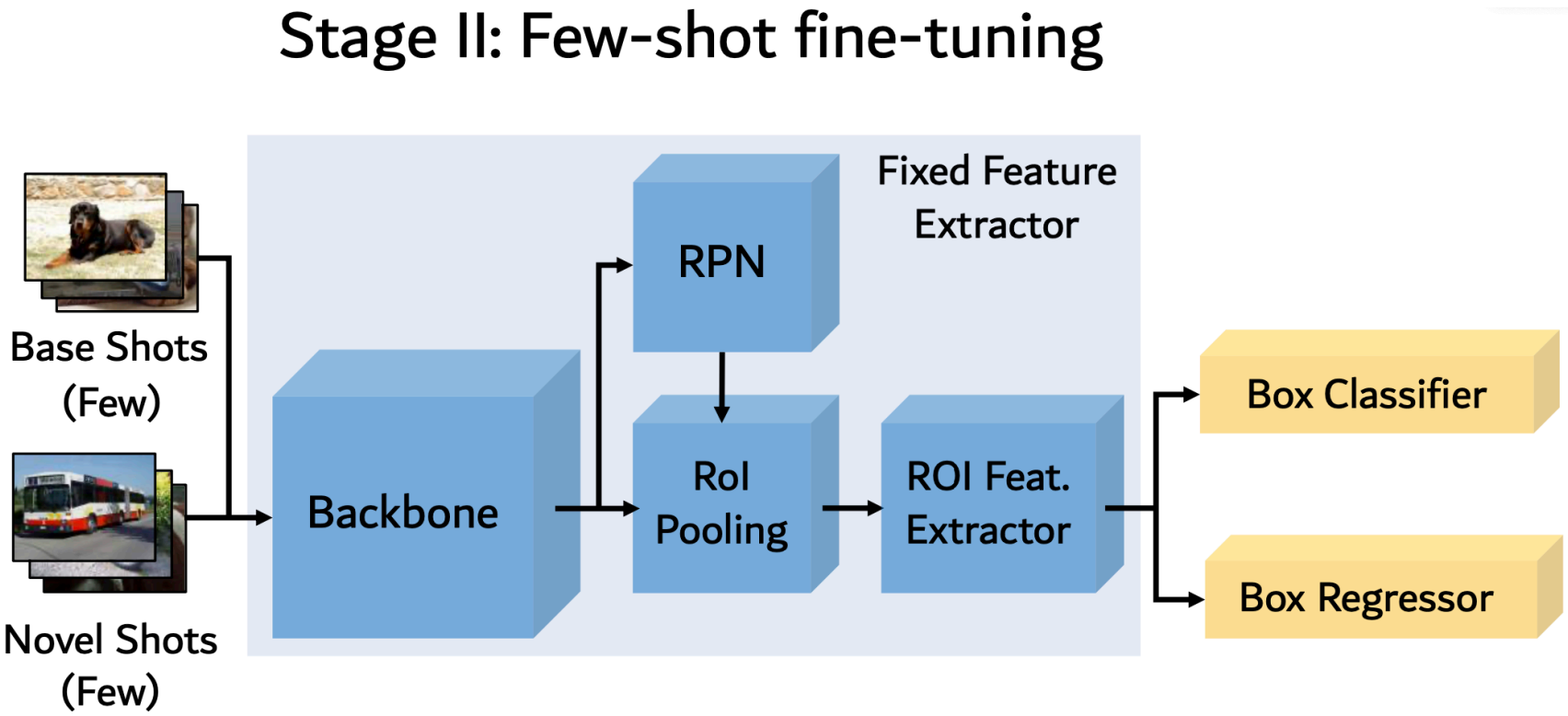
Generally, training is done in two stages.

Few-shot Finetuning



- Backbone: **ResNet**
- ROI Feat. Extractor: **Faster-RCNN** (Ren et al., 2015)

Few-shot Finetuning



Proposed by Wang et. al. 2020, the entire network is not tuned to novel category samples but **only the last layers of the detector**.

Training: Process

- **Base model training:** The feature extractor and the box predictors are trained over the base class datapoints.

Backpropagation is done over

$$L = L_{rpn} + L_{cls} + L_{loc}$$

where L_{rpn} is applied to the output of the RPN, L_{cls} is a cross-entropy loss for the box classifier, and L_{loc} is a smoothed L_1 loss for the box regressor.

- **Few Shot Fine tuning:** The weights of the box classifier and regressor are first randomly initialized and then *K* shots of the novel classes are used to fine tuned these weights.

Training : Euclidean Distance based Similarity

- Wang et. al. 2020 used a **cosine** distance-based similarity in the second fine-tuning stage for their classifier

- We use **Euclidean** distance over Softmax for the similarity score. It is given as,

$$s_{i,j} = \frac{e^{-\left\|F(x)_i - w_j\right\|_2^2}}{\sum_j \left(e^{-\left\|F(x)_i - w_j\right\|_2^2}\right)},$$

where $s_{i,j}$ is the similarity score between the ith object proposal of the input x and the weight vector of class j, $w_j \in R^d$ and $[w_i, w_2, \ldots, w_c] = \mathrm{W} \in R^{d \times c}$, and $F(x)_i$ is the extracted features for the ith proposal object.

Data

- Model was evaluated on the COCO, PASCAL-VOC and KITTI dataset.

- The COCO dataset (2014) is a common object dataset with 80 classes. It has 80000 training images.
- The PASCAL-VOC dataset (2007) is a visual object dataset with 20 classes. It has 9963 total images. 5000 were used for training.
- The KITTI dataset is a real-world dataset of traffic images such as pedestrians, cars and buses. We take 50 images and separate them into two classes of pedestrian and cars for our few shot experiments.

Experimental Setup

- The model was trained using **SGD** with **momentum=0.9** along with weight decay of 0.0001, and a batch size of 16.
- We used a learning rate of **0.02** for **base training** and **0.001** for **novel class** training.

- For base training, the model was trained for **20 epochs** in total with early stopping for COCO and Pascal-VOC and then fined-tuned for the novel categories for **10 epochs**.

- For KITTI the PASCAL-VOC base feature detector was fixed and only the **last layers we fine-tuned for 20 epochs**.

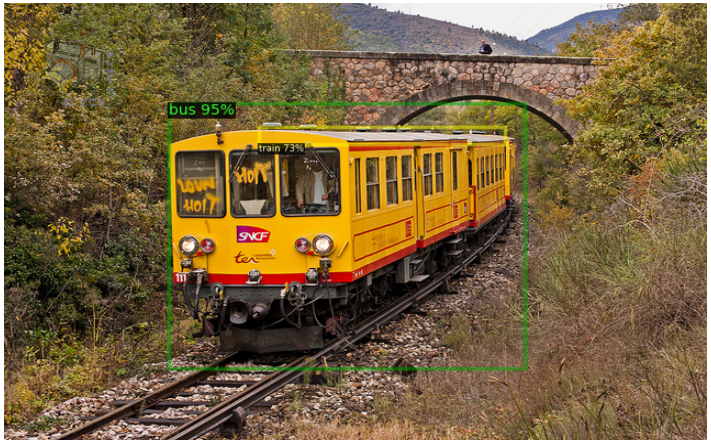
Results

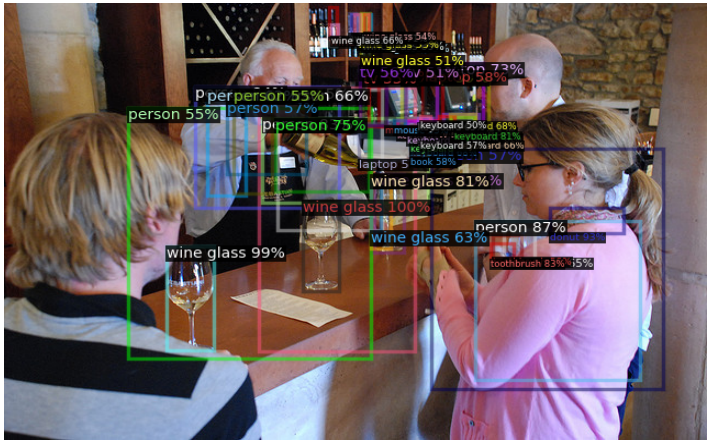
mAP50(threshold=0.5) results on test points

	1-Shot (Ours)	1-Shot(C.S.)	5-Shot (Ours)	5-Shot(C.S.)	10-Shot (Ours)	10-Shot(C.S.)
COCO	36.3	39.8	43.1	41.2	48.1	42.2
PASCAL-VOC	37.1	39.8	56.3	55.7	49.6	56
KITTI	22.5	N/A	31.7	N/A	29.8	N/A

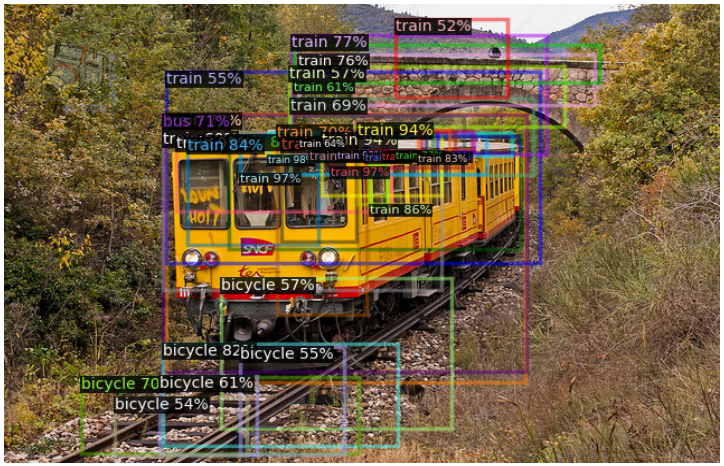
C.S.: Cosine Similarity

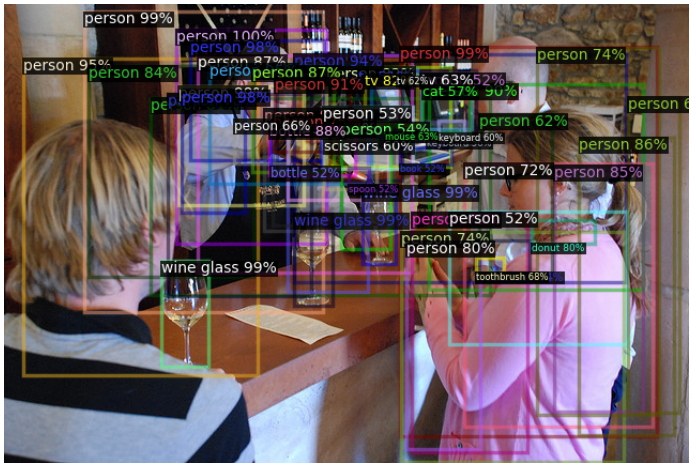
Results : COCO- 1 Shot





Results : COCO- 5 Shot

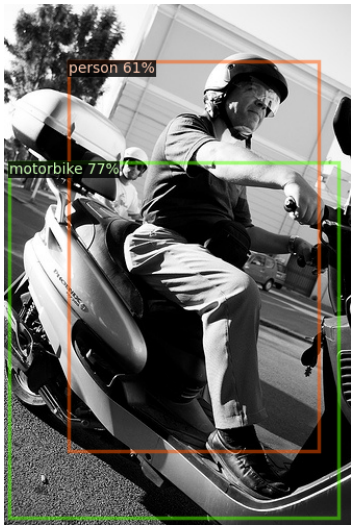


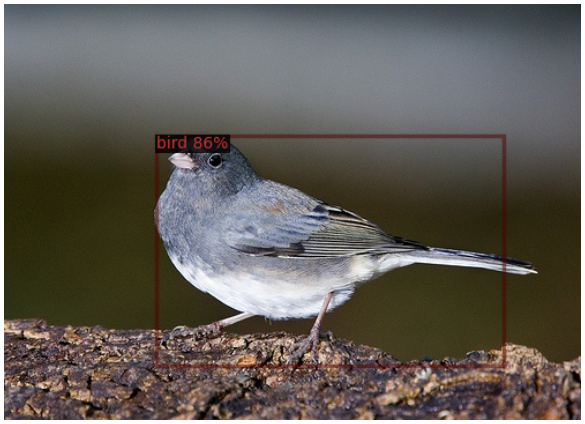


Results : COCO- 10 Shot

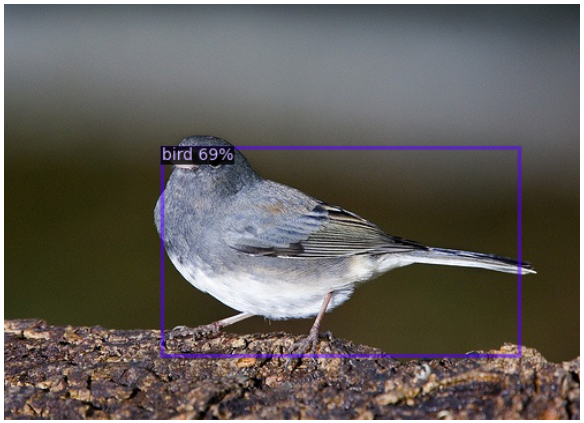
Results: PASCAL- 1 Shot

Results: PASCAL- 5 Shot

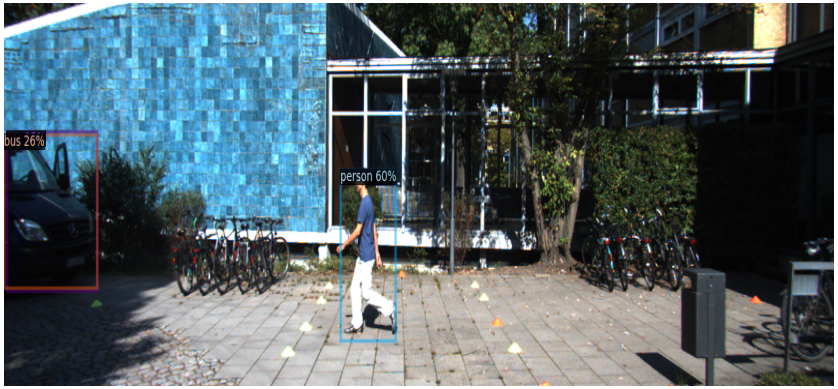


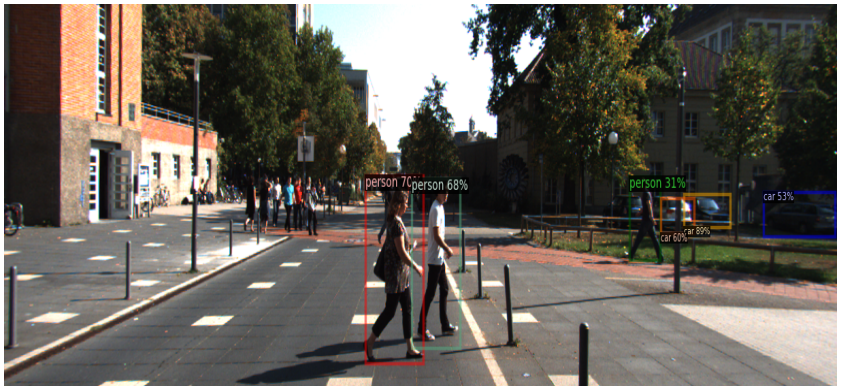


Results: PASCAL- 10 Shot

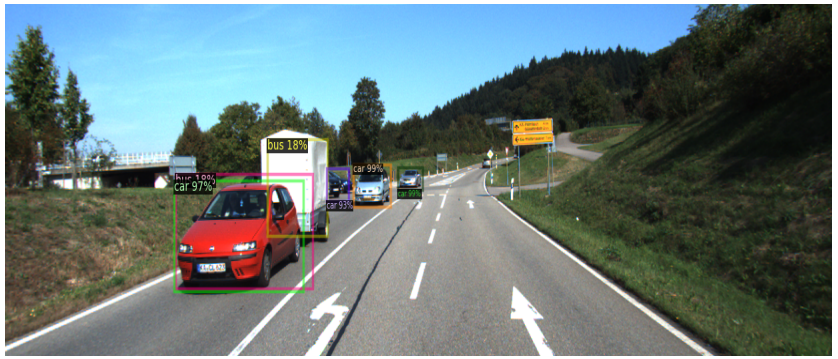


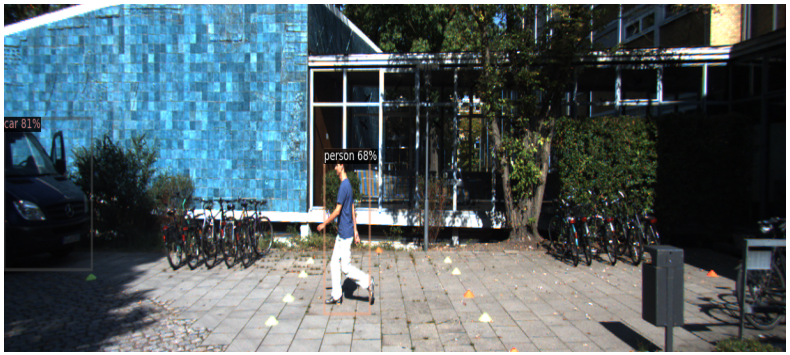
Results: KITTI –1 Shot

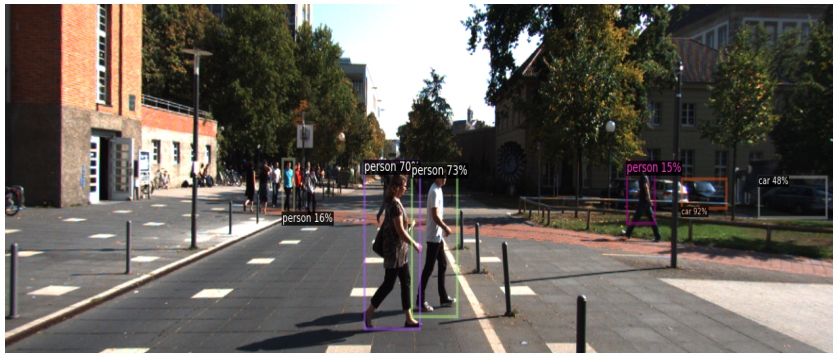




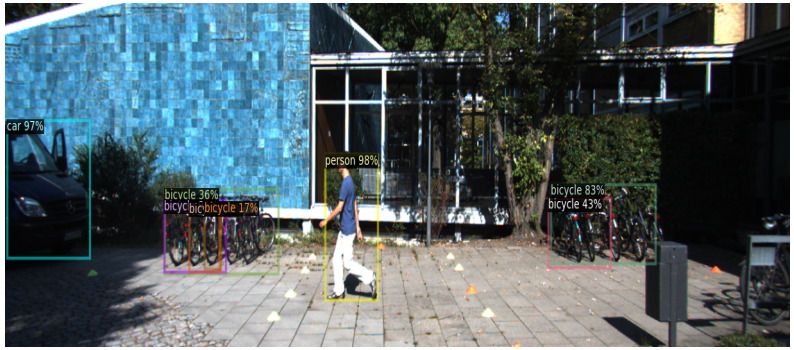
Results: KITTI –5 Shot

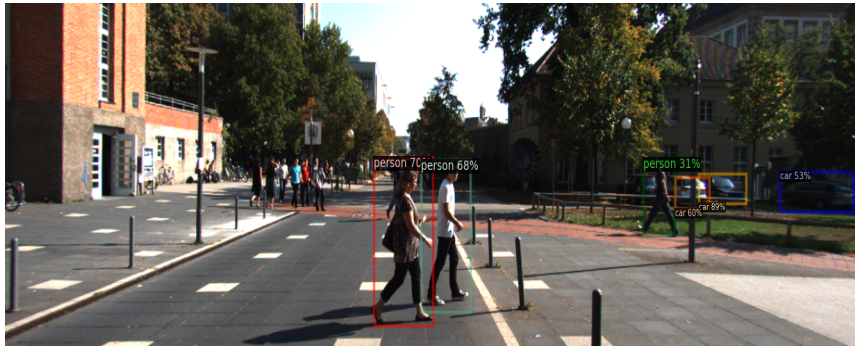






Results: KITTI –10 Shot





Conclusion

- We use **Euclidean distance** based Softmax **similarity score** in our few-shot experiments and showed that it gives comparable results to cosine similarity.
- We also show how few shot detection can be used to detect objects in traffic images by evaluating on the KITTI dataset.
- **One shot** is usually not enough, but **10 shots** are plenty.
- Further, the capability of these models for **3D object detection** can also be explored.

Thank you!

Questions?

(Not you, Miraj & Abhishek.)