Table 2: Roofline estimates on object detection models evaluated on IDD dataset. Faster-RCNN gives the best performance among all models.

| Method | Backbone and head | $mAP$ | $mAP_{50}$ | $mAP_{75}$ |
|---|---|---|---|---|
| YOLO -V3[4](**?**) | Darknet-53 | 11.7 | 26.7 | 8.9 |
| Poly-YOLO[4] (**?**) | SE-Darknet-53 | 15.2 | 30.4 | 13.7 |
| Mask-RCNN[4] (**?**) | ResNet-50 | 17.5 | 30.0 | 17.7 |
| Retina-Net (**?**) | ResNet-50 + FPN | 22.1 | 35.7 | 23.0 |
| Faster-RCNN (**?**) | ResNet-101 + FPN | **27.7** | **45.4** | **28.2** |

Table 3: Few-shot object detection performance ($mAP_{50}$) on IDD-10 split 1 using 10-shot samples.

| Method | $mAP$ | $mAP_{base}$ | $mAP_{novel}$ |
|---|---|---|---|
| Meta-RCNN | 17.3 | 24.0 | 4.2 |
| Add-Info | 26.0 | 34.0 | 7.5 |
| Meta-Reweight | 21.8 | 26.4 | 10.9 |
| **TFA (FsDet)** | **28.5** | **31.2** | **22.1** |

Table 4: Few-shot object detection performance ($mAP_{50}$) on IDD-OS split using 10-shot samples.

| Method | $mAP$ | $mAP_{base}$ | $mAP_{novel}$ |
|---|---|---|---|
| Meta-RCNN | 20.1 | 27.9 | 4.2 |
| Add-Info | 32.0 | 33.0 | 36.0 |
| **TFA (FsDet)** | **43.0** | **47.4** | **37.0** |

adopt episodic training strategy to adapt to few-shot data in both the base training and fine-tuning stages. We adopt this technique as well, using a two-stage training pipeline with an additional loss term $L_{meta}$ during both training stages. An important benefit of this technique is that it can be applied to both proposal-free and proposal-based object detection architectures as described in section 2.2.

## 4 Experiments

In this Section, we evaluate various few-shot techniques on IDD based on categories of data described in 3.3 and demonstrate the performance of the few-shot object detection networks. We also evaluate several object detection baselines that help us to establish roofline estimates for the few-shot models.

### 4.1 Object Detection Baselines

We establish a roofline (upper bound) performance by training object detection algorithms considering both proposal-free and proposal-based methods like YOLO (**?**), RetinaNet (**?**), Faster-RCNN (**?**) on IDD dataset. The experiments generate an estimate of the performance of object detection methods given abundant data samples. The baseline evaluation is conducted on the 10 class dataset (IDD-10cls) described in section 3.3.

Following state-of-the-art object detection approaches we report the mean Average Precision (mAP) in Table 2 for all methods trained on IDD (backbones are pre-trained on Imagenet). We see that Faster-RCNN based networks outperform the compared object detection networks and we consider it as the roofline for evaluating the few-shot detection task.

### 4.2 Few-Shot Detection on IDD Dataset

We discuss four state-of-art techniques namely, Meta-Reweight (**?**), Add-Info (**?**), Meta-RCNN (**?**) and FsDet (**?**) which are evaluated on representative data-splits from IDD dataset. Following previous works we report mean Average Precision at 50% overlap of intersection over union ($mAP_{50}$) as the performance metric to evaluate the few-shot object detection architectures. All our experiments are conducted in a 10-shot setting.

**Implementation:** For meta-learning architectures (Meta-RCNN and Add-Info), we train a ResNet-101 based model for 9 epochs in the base training stage at 0.001 learning rate, followed by 9 epochs for fine-tuning. We train Meta-Reweight with a DarkNet-19 based model for 30 epochs at a 0.0001 learning rate for base training, followed by 40 epochs at 0.000001 learning-rate for finetuning.

The metric learner, FsDet, uses a ResNet-101 with Feature Pyramid Network based architecture and is trained for 40k iterations at 0.005 learning rate for the base training stage, and then followed by 15k iterations at 0.001 learning rate for fine-tuning.

In the base training stage, all methods use a batch size of 2. All backbones are initialized with ImageNet pretrained weights and standard data augmentation like horizontal flip and random crop are applied.

**Evaluation in Same Domain:** First, experiments are performed on IDD-10 splits containing 7 base classes and 3 novel classes per split. Table 3 reports the performance of the networks in terms of $mAP_{50}$ for all, base ($mAP_{base}$) and novel ($mAP_{novel}$) classes in the validation set of IDD. The results from split 2 (not described due to space limitations) are very similar to split 1.

**Evaluation in Open Set:** We conduct experiments on the IDD-OS split which contains 10 base classes (same used for roofline estimates) and 4 novel classes (only available in IDD). Table 4 lists the performance of all networks on IDD-OS split in terms of $mAP_{50}$ for similar class definitions mentioned above. We were not able to conduct experiments with Meta-Reweight due to memory limitations during training. Figure 5 illustrates the classwise performance of FsDet (metric-learning) and meta-learning based architectures (Meta-RCNN and Add-Info) on IDD-OS split.

[4]Results are from (**?**).

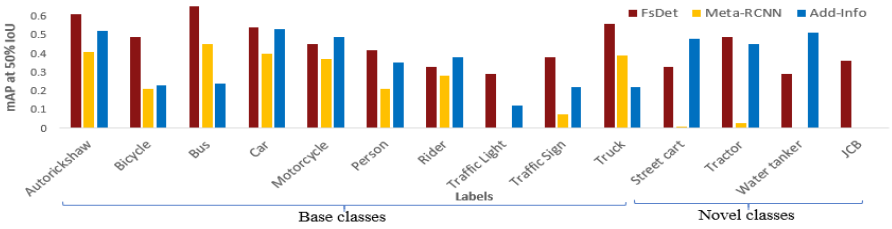


Figure 5: Classwise performance of metric-learning and meta-learning based techniques in detecting rare objects in IDD dataset. The last 4 classes represent the rare categories. FsDet, a metric learning based approach, performs better than meta-learning approaches in both base and novel classes.

### 4.3 Discussion

Our experiments uncovered several findings on the nature of road objects and the behaviour of few-shot object detection networks in the context of driving. Results from Tables 3 and 4 demonstrate that cosine similarity based TFA architectures (FsDet) outperforms meta-learning based architectures (Meta-RCNN, Meta-Reweight and Add-Info) on novel-class performance by 11.2 $mAP$ points on IDD-10 (split 1) and 1.0 $mAP$ point on IDD-OS split. We attribute lower inter-class distance between new object categories as the probable reason for the lower performance of meta-learning over similarity-based methods. This aspect of IDD makes it a unique dataset well suited for evaluation of few-shot object detection in a real-world, driving scenarios.

Comparing the base class performance in Table 4 against the roofline performance metrics in Table 2, we demonstrate a lower degradation in base-class performance when adopting TFA architecture (FsDet) over its meta-learning counterparts, after the introduction of novel classes. Meta-learning techniques like Meta-RCNN and Add-Info suffer a significant reduction in base-class performance except when additional features were provided to the final prediction head of the object detector.

Figure 6 shows class-level confusion among all classes in IDD-OS split trained on 10-shot data samples using TFA architecture. In particular, the confusion between $truck$ vs. $car$, $bicycle$ vs. $motorcycle$ and $water\text{-}tanker$ vs. $car$ classes are high, with the maximum being 40%. This observation can possibly be explained by the fact that road objects in context, share a large number of low-level features with other object classes, thus posing a challenge for few-shot algorithms to differentiate. This observation here is in line with that by the authors of MetaDet (**?**) that confusion between classes is the primary challenge in few-shot learning scenarios. This is further echoed by the authors of Meta-Reweight (**?**) that there exists a high confusion of 50% among classes in PASCAL VOC dataset.

## 5 Conclusion

We analyzed the performance of state-of-the-art methods for few-shot object detection, using a real-world dataset (IDD) which inherently contains class-imbalanced data from driving scenarios. Our evaluation of methods was for two tasks: same-domain and open-set representations. To evaluate these settings, we expanded a publicly available dataset

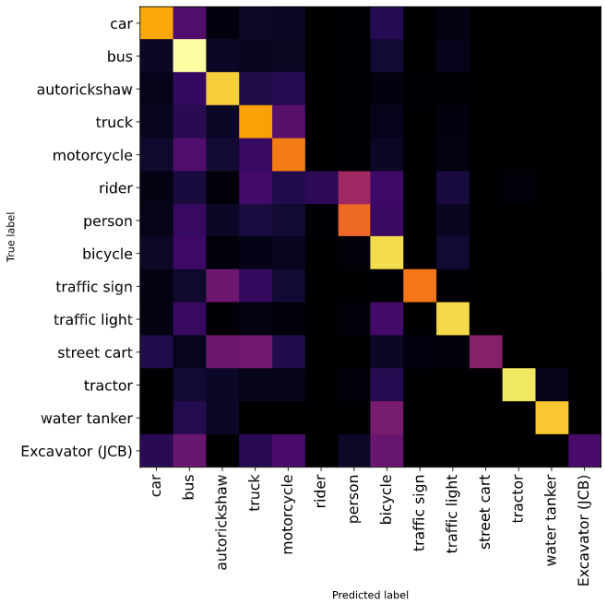


Figure 6: Confusion matrix plotted for class prediction results from IDD validation dataset showing confusion between classes when trained on IDD-OS 10-shot split on FsDet network.

with additional class labels in the open-set representation. By creating an extension of IDD, we hope to pave a way for many future works in few-shot learning with real-world datasets. Based on our experiments, we conclude that cosine similarity based TFA network (FsDet) outperforms meta-learning based networks in both the tasks by 11.2 and 1.0 $mAP$ points in novel class performance respectively. We conclude that meta-learning networks while achieving great strides, tend to under perform even simpler baselines from metric-learning based methods. We also observe that class-confusions remains an open challenge in any few-shot learning paradigm and can be the focus of further improvements.

Modi aperiam quam deleniti unde, a eveniet iusto laborum eligendi minima ut et qui soluta, ducimus laboriosam iusto, deserunt quis ratione, impedit veniam blanditiis aperiam architecto est officia sed qui?Repellendus assumenda accusantium ullam quis culpa qui, expedita impedit mollitia tempore accusamus est?