ORDER: Open World Object Detection on Road Scenes

Deepak Kumar Singh[1*] **Shyam Nandan Rai**[2*] **K J Joseph**[3] **Rohit Saluja**[1]
Vineeth N Balasubramanian[3] **Chetan Arora**[4] **Anbumani Subramanian**[1] **C.V. Jawahar**[1]
[1]CVIT - IIIT Hyderabad, India, [3]IIT Hyderabad, India, [4]IIT Delhi, India
[1]{deepak.singh, rohit.saluja}@research.iiit.ac.in, [2]shyamnandanraii@gmail.com,
[3]{cs17m18p100001, vineethnb}@iith.ac.in, [4]chetan@cse.iitd.ac.in, [1]{anbumani, jawahar}@iiit.ac.in

Abstract

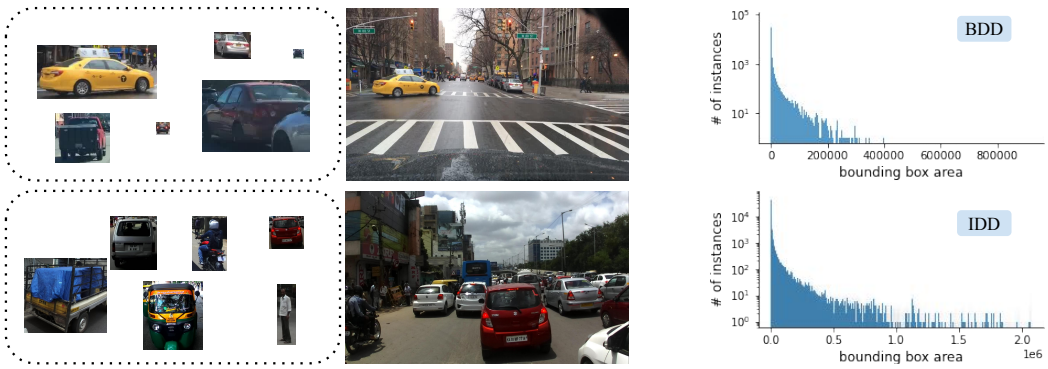
Object detection is a key component in autonomous navigation systems that enables localization and classification of the objects in a road scene. Existing object detection methods are trained and inferred on a fixed number of known classes present in road scenes. However, in real-world or open-world road scenes, while inference, we come across unknown objects that the detection model hasn't seen while training. Hence, we propose Open World Object Detection on Road Scenes (ORDER) to address the aforementioned problem for road scenes. Firstly, we introduce Feature-Mix to improve the unknown object detection capabilities of an object detector. Feature-Mix widens the gap between known and unknown classes in latent feature space that helps improve the *unknown* object detection. Next, we identify that the road scene dataset compared to generic object dataset contains a significant proportion of small objects and has higher intra-class bounding box scale variations, making it challenging to detect the known and unknown objects. We propose a novel loss: Focal regression loss that collectively addresses the problem of small object detection and intra-class bounding box by penalizing more the small bounding boxes and dynamically changing the loss according to object size. Further, the detection of small objects is improved by curriculum learning. Finally, we present an extensive evaluation on two road scene datasets: BDD and IDD. Our experimental evaluations on BDD and IDD shows consistent improvement over the current state-of-the-art method. We believe that this work will lay the foundation for real-world object detection for road scenes.

1 Introduction

Building a robust autonomous navigation system that can reliably maneuver in a real-world scenario is a challenging task. Object detection [20; 2; 14; 19; 27; 5; 11] plays an important role for autonomous navigation systems in identifying and localizing objects in a road scene. Current object detection models are trained on a *closed-set*, where all the *test* classes are *known* at training time [22]. However, in an *open-world* setting the test set has objects from *unknown* categories. Therefore, present object detectors do not generalize well in an open world setting. Recently, Joseph *et al.* [10] introduced Open World Object Detector (ORE) framework that performs open-world object detection. The performance of ORE was evaluated on generic dataset: MS-COCO [12] and PASCAL-VOC [4]. However, we find that the ORE framework shows poor performance when applied to challenging

* Equal Contribution.

Machine Learning for Autonomous Driving Workshop at the 35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.



(a) (b)

Figure 1: **(a)**: We can observe intra-class and inter-class scale variation prominently in some of the categories like car and pedestrian category. This issue is prominent in road scene datasets.**(b)**: Shows the distribution of bounding box area in BDD and IDD, we notice that there are relatively more small bounding boxes than large bounding boxes.

domains, such as road scene datasets. The challenges include: a) *unknown* objects are hard to detect; b) the proportion of small objects (from both *known* and *unknown* set) is significant (fig. 1 [b]), and c) the presence of intra-class scale variation (fig. 1 [a]). The problem of intra-class scale variation is highly pronounced found in road scene datasets. Generic datasets such as MS-COCO and PASCAL-VOC consist of images captured close to the object resulting in smaller variations in scale. Similarly, in aerial object detection dataset [28], the objects are captured at distinctly high altitudes resulting in a consistent intra-class object size.

We propose Open World Object Detection on Road Scenes (ORDER) that addresses the aforementioned problems. We introduce Feature-Mix, inspired by Open-Mix [30] where we combine multiple *unknown* and *known* class instances to improve *unknown* object identification. It is important to note that Open-Mix takes a single instance of known and unknown, hence, it cannot combine multiple *unknown* and *known* class instances that are generally present in road scenes. Feature-Mix overcomes the limitation of Open-Mix by mixing *unknown* and *known* class instances at the feature level, allowing it to mix multiple instances of *known* and *unknown* classes. Next, we propose focal regression loss that handles intra-class variation by including bounding box area and improves small object detection by penalizing more to small bounding boxes than large ones. We further improve small object detection by training the ORDER framework in a curriculum manner. Improving the small object detection and handling the intra-class variation reduces the chances of *known* class detected as *unknown* and improves the *known* object detection. We validate the performance of ORDER and the competitive baselines on the Indian Driving Dataset (IDD) [26] and Berkeley Deep Drive (BDD) [29] datasets. We observe that the ORDER shows state-of-the-art performance on open-world evaluation metrics: Wilderness Impact (WI) and Absolute Open-Set Error (A-OSE). Additionally, an extensive ablation study is also performed on ORDER to show the contribution of proposed components individually.

The key contributions of our work are:

- To the best of our knowledge, ORDER is the first work that addresses open-world object detection for road scene datasets.
- Introduces Feature-Mix, which is integrated in ORDER that significantly improves the feature representation of *unknowns*.
- Identifies and addresses two inherent issues in road scene datasets: intra-class scale variation by proposing focal regression loss and small object detection by curriculum learning.

2 Related Works

Object Detection: The goal of an object detection model is to predict the bounding boxes and the class for an object present in an image. The current set of detectors can be divided into two categories:

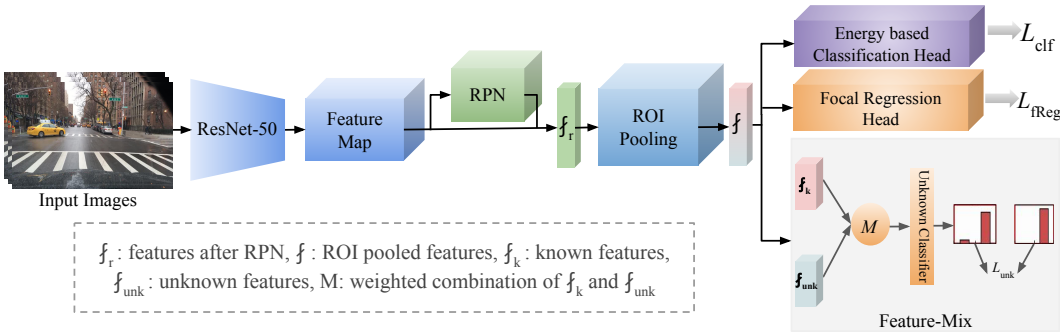


Figure 2: Illustration of ORDER framework. f is RoI pooled features consisting of *known* class features f_k and *unknown* class features f_{unk} that are mixed in Feature-Mix M block. L_{clf}, L_{fReg} and L_{unk} denotes the classification loss, focal regression loss, and feature-mix loss, respectively.

single-stage detectors and two-stage detectors. Two-stage detectors such as Faster R-CNN [20], R-FCN [2] depend on Region Proposal Network (RPN) that generates the region proposals based on an objectness score, which are further refined to get desired object bounding box. Whereas single-stage detectors such as SSD [14], YOLO [19], and SqueezeDet [27] consist of a single network that can predict the object bounding boxes and classes. There are also several works of object detection [13; 27; 8] on road scene datasets. We choose the two-stage detector over the single-stage detector since the former demonstrates better performance.

Open Set Detection: Object detectors trained on fixed set of training and testing classes are not robust in identifying unknown encountered in real-world. Miller *et al.* [16] introduced the open-set object detection for real-world scenarios. They utilized dropout sampling to get the uncertainty present in the object detector and used it to identify *unknown* objects. Next, Miller *et al.* [15] deployed various merging strategies for Monte Carlo (MC) dropout on object detector and evaluated in open-set conditions. Recently, Dhamija *et al.* [3] provided a detailed analysis of object detector performance in the open-set setting and proposed an evaluation metric Wilderness Impact that quantifies the performance of object detection model in real-world.

Open World Detection: Joseph *et al.*[10] introduced the problem of *open-world* object detection and proposed Open World Object Detector (ORE). The key idea of ORE is to identify *unknown* classes and incrementally learn the distinct unknown classes when the labels of those classes are available. ORE uses Faster R-CNN as a base detector and improves its ability to identify *unknown* classes by adding contrastive clustering and an energy-based classifier. However, the ORE is not designed to handle intra-class scale variation explicitly, which is prominently present in road scene datasets. Our proposed framework addresses the issue of scale variation by introducing curriculum-based training along with a novel *Focal Regression Loss* (section 3.4). We also improve the feature representation of *unknown* by introducing *Feature-Mix* (see sections 3.3 and 5.5).

3 Methodology

3.1 Problem Setting

We begin the formulation of Open World Object Detection (OWOD) by considering a set of *known* classes as C_k and a set of *unknown* classes C_{unk}. The *known* classes will have ground-truth bounding boxes, whereas *unknown* classes will be encountered at test time. We train a detection model D on a set of *known* classes and simultaneously ensure that *unknown* class instances are also detected. Next, the detected set of *unknown* class instances could be forwarded to a human annotator to obtain the ground-truth labels. The detection model is incrementally trained on the new ground-truth labels rather than training the entire model from scratch giving an improved detection model $\hat{D}$. We continue the process of the detection model adapting to a new set of classes and detecting *unknown* class objects over a lifetime. In the experimental setting, we define the set of classes as task T.

3.2 ORDER: Open World Object Detection on Road Scenes

ORDER uses Faster-RCNN [7] detector that is molded accordingly to detect *known* and *unknown* objects. It consists of three output heads: Energy Based Classification head, Focal Regression head, and Feature-Mix head. Energy Based Classification head and Focal Regression head are used to learn to differentiate between *known* and *unknown* features by using L_{clf} and L_{unk}. Focal Regression head learns the object bounding boxes. Figure 2 shows the pipeline of the ORDER framework. The ORDER framework is inspired from ORE [10], however, it largely differs in terms of novel components that we introduced to handle the challenges present in road scene datasets. We will discuss the key novelties of ORDER framework in detail.

3.3 Feature-Mix

Unknown class identification is an essential component of an open-world object detector. In autonomous navigation systems, improving *unknown* class identification will increase the chances of avoiding unfavorable situations. We propose a Feature-Mix approach that improves *unknown* class identification by incorporating knowledge of *known* classes. The key intuition behind Feature-Mix is to mix the features of *known* and *unknown* and suppress the activation caused by known features, so that the latent difference between *known* and *unknown* feature is maximized. We begin Feature-Mix formulation by taking Region of Interest (RoI) pooling output features f consisting of *known* class features f_k and *unknown* class features f_{unk}. We mix the known and *unknown* features by:

$$f_{mix} = \lambda f_k + (1 - \lambda) f_{unk}, \tag{1}$$

where, λ is sampled from *beta distribution* parameterized with α and β. Now, the *unknown* classifier C_{unk} utilizes f_{mix} to identify *unknown* objects trained by using a loss L_{unk} given by:

$$L_{unk} = -ylog(softmax(C_{unk}(f_{mix}))), \tag{2}$$

$$y = argmax(log(softmax(C_{unk}(f_{mix})))) \tag{3}$$

y represents the ground-truth label. We use a small held-out validation set, as proposed in ORE [10], consisting of known and unknown data samples to train Feature-Mix.

3.4 Focal Regression Loss

Popular detection methods use Smooth-L1 [20], and Generalized Intersection over Union (GIoU) [21] loss for bounding box regression. However, these losses do not explicitly incorporate the knowledge of intra-class scale variation prominent in road scene datasets. We introduce Focal Regression Loss (L_{fReg}) that addresses the problem of a) detecting small objects by penalizing more for small bounding boxes b) intra-class variation by including bounding box information. We formulate L_{fReg} by adding a regulating component $(1 - IoU)^{\gamma^*}$ to squared IoU loss, where $\gamma^* \in [0, \infty)$ is a focusing parameter. L_{fReg} can be denoted as:

$$L_{fReg} = (1 - IoU)^{\gamma^*} \left\| 1 - IoU \right\|_2^2 \tag{4}$$

$$\gamma^* = \gamma + log(log(\hat{Ar}_{bbox_{gt}})), \hat{Ar}_{bbox_{gt}} = Ar_{Img}/Ar_{bbox_{gt}} \tag{5}$$

$\hat{Ar}_{bbox_{gt}}$ and $Ar_{bbox_{gt}}$ represents the inverse-normalized and unnormalized bounding box area, respectively. Inverse-normalization gives large values for small bounding boxes and small value for large bounding boxes. .$\hat{Ar}_{bbox_{gt}}$ results from dividing image area Ar_{Img} by $Ar_{bbox_{gt}}$. γ^* consists of tunable parameter γ and double logarithmic of inverse-normalized bounding box area. We apply double logarithmic to inverse-normalized bounding box area because a) it prevents overshooting of γ^* when inverse-normalized bounding box area is tiny, b) it smoothens out the variation in the inverse-normalized bounding box area making the training more stable. Note that for small bounding boxes, the value of γ^* would be high, resulting in more penalization as compared to large bounding boxes that are easy to detect.

3.5 Curriculum Training

Detecting smaller objects [9] is a harder task than detecting object instances with larger sizes. In autonomous navigation datasets such as BDD and IDD, the proportion of small objects is significant fig. 1. As per [24], smaller objects are considered harder to learn than larger objects. Hence, we adopt a curriculum learning [1; 6] strategy to gradually train the network from *easy* samples (large objects) to *hard* samples (small objects). We divide the training dataset into three sets: S_{easy}, S_{medium}, and S_{hard}, based on the bounding box area. For an individual task T_i, $i \in \{1, 2, 3\}$, we train the detection model in three steps that can be formulated as:

$$T_i = \begin{cases} S_{easy} & I_1;\ \textit{if}\ Ar_{bbox} < Ar_{easy} \\ S_{easy} + S_{medium} & I_2;\ \textit{if}\ Ar_{bbox} < Ar_{medium} \\ S_{easy} + S_{medium} + S_{hard} & I_3 \end{cases} \tag{6}$$

I_1, I_2, and I_3 are the number of iterations each group of the sets are trained. Ar_{easy} and Ar_{medium} are the area thresholds for selecting large and medium bounding boxes.

4 Experiments and Results

4.1 Datasets Protocol

We adapt the standard evaluation protocol of ORE [10] to demonstrate the efficacy of ORDER. For a given dataset, we divide it into a set of classes. Each class set is denoted by task T_t, t represents the time-stamp of the model having access to only classes of T_t. The dataset can be represented as $\{T = T_1, \cdots T_t, \cdots\}$. At a given time-stamp t, the classes of $\{T_\tau : \tau \leq t\}$ are considered as *knowns* and the classes of $\{T_\tau : \tau > t\}$ as *unknowns*. We follow the protocol discussed above discussed to divide the IDD and BDD datasets into tasks.

The IDD dataset consists of 15 classes. We divide the dataset into three tasks, and each task consists of 5 classes. The BDD dataset consists of 10 classes. We divide the dataset into three tasks; the first task consists of 4 classes, and the rest have 3. For each task, we randomly choose the classes to avoid any bias. The statistics of training and testing instances and the classes for each task are given in table 1. We take a set of $3K$ images from each dataset for validation.

4.2 Evaluation Metrics

We use mean Average Precision (mAP) to evaluate the performance of the model on *known* classes. The IoU threshold for the mAP is taken as 0.5 in accordance with [23; 18; 10]. Now, to quantify the performance of a model for *unknown* identification, we use Wilderness Impact (WI) [3] metric. The WI measures the model's sensitivity to *unknowns* over a range of frequency of frames that may have *unknowns*. The WI is equated as:

$$\text{Wilderness Impact } (WI) = \frac{P_{\mathcal{K}}}{P_{\mathcal{K}\cup\mathcal{U}}} - 1$$

Here, $P_{\mathcal{K}}$ refers to the precision of the model when evaluated on *known* classes, and $P_{\mathcal{K}\cup\mathcal{U}}$ is the precision when evaluated on *known* and *unknown* classes, measured at a recall level(R) of 0.8 in all experiments. Ideally, the WI needs to be close to 0, demonstrating that the precision does not change when *unknowns* are introduced to the test set. Absolute Open-Set Error (A-OSE) [16] is another metric that shows the *unknown* detection performance of a model. It is defined as total number of *unknown* objects getting classified as a *known* object.

4.3 Implementation Details

We use the modified Faster R-CNN with ResNet-50 [7] backbone according to ORE. The shape parameters α and β are chosen to be 1. The contribution of L_{unk} in total loss is 0.001 and 0.1 for IDD and BDD, respectively. The values of hyperparameter γ present in Focal Regression Loss is 0.4 and 0.1 for IDD and BDD, respectively. For the Curriculum training, I_1, I_2, and I_3 values are $36K$ for Ar_{easy} and $72K$ for Ar_{medium} and Ar_{easy} on both IDD and BDD datasets. We train our models on 4 GPUs with a batch size of 8 images.

Table 1: Table shows the division of the datasets into various tasks. For each task the group of classes, training and testing statistics are shown.

	IDD			BDD		
	Task 1	Task 2	Task 3	Task 1	Task 2	Task 3
Categories	person, bicycle, train, truck, traffic sign	traffic light, bus, car, rider, trailer	motorcycle, caravan, animal, autorickshaw, vehicle fallback	pedestrians, motorcycle, traffic sign, bus	bicycle, train, traffic light	car, rider, truck
# training images	23217	25586	25146	61208	40416	69103
# test images	7558	8462	8085	8756	5803	9898
# train instances	103614	155404	122498	345639	195123	734030
# test instances	30025	55204	40831	50270	27938	107825
# unknown instances	96035	40831	0	136763	107825	0

4.4 Results on BDD

We now discuss the results of our experiments on the BDD dataset. As a baseline, we train Faster-RCNN on the first task and finetune it on consecutive tasks as shown in the first-row of table 2 (top). The ORE reduces both WI and A-OSE (lower the better) compared to baseline for the first two tasks of BDD. However, ORE drops in overall mAP by 2 (approx.) compared to baseline for the two tasks (columns 4 and 9 of table 2 [BDD]). On the contrary, ORDER improves mAP by 0.5 and 1.4 for the two tasks, reducing WI by 0.015 and 0.013 compared to the baseline. ORDER also reduces the AOSE by a considerable margin of 9769 and 11385 as compared to the baseline. For Task 3 of BDD, ORDER attains a massive gain in overall mAP of around 6.36 and 5.95, compared to the baseline and ORE (last column of table 2 [BDD]).

4.5 Results on IDD

On the IDD dataset, we observe in table 2 (bottom) that the WI is comparable for the three models. ORDER, however, achieves the best A-OSE for the first two tasks of IDD, reducing it by a margin of 11186 and 10255 compared to baseline and 2796 and 2628 compared to ORE. ORDER's overall mAP is comparable to ORE for Task 1 of IDD and is highest for the remaining tasks (columns 9 and 12 of table 2 [IDD]).

It is also interesting to note that ORDER performs better than ORE for all the columns in table 2 (refer to last two rows of the tables).

5 Discussion and Analysis

5.1 Ablative Study of ORDER

We perform ablative studies to validate the performance of the proposed components of ORDER qualitatively. Table 3 shows the results on Task 1 of IDD. We observe that using all the proposed components shows significant improvement on WI, A-OSE, and mAP over the model trained with only Smooth-L1 loss (row 1 of table 3). It is also essential to infer from the first two rows of table 3 that the proposed focal regression loss shows significant improvement in mAP compared to Smooth-L1.

5.2 Performance Comparison of Focal Regression Loss

We demonstrate the efficacy of our proposed focal regression loss in better identifying *known* objects. We compare the proposed loss with Smooth-L1 [20], GIoU [21], and Least Square IoU [17]. Table 4 (a) shows the mAP on all the losses trained on Task 1 of IDD. We find that *Focal Regression Loss* gives the best performance among all the losses.

Note that all the classes are known for Task 3; hence, the two metrics do not hold.

Figure 3: Qualitative comparison: Row a) images are from the IDD dataset, and b) and c) are from the BDD dataset. The results are inferred from the models trained on Task 2 of the BDD and IDD datasets. In row a), we observe that ORDER is able to detect smaller objects with high confidence. It is interesting to note that the highlighted boxes of a) has `car` instances shows intra-class scale variation. ORDER handles the intra-class scale variation within the `car` instance by detecting it on varying scales. In b) and c), we can see ORDER detects safety-critical classes such as `pedestrian` and `traffic sign` better than ORE. We also notice that ORDER performs better at recognizing overlapping *known* and *unknown* objects and has high confidence in unknown and known predictions. For easy distinction, the red bounding boxes denote *unknown* predictions, whereas the green bounding boxes denote the distinct *known* classes. The blue and pink boxes represents the cropped region. **Best viewed when zoomed**.

Table 2: Quantitative performance of ORDER on road scene datasets. We notice that ORDER shows good performance in identifying unknown classes by giving lower Wilderness Impact and Average Open Set Error and simultaneously performs well in detecting known classes by giving high mean Average Precision. Best results are highlighted in bold.

BDD

Task IDs (→)	Task 1			Task 2					Task 3		
	WI	A-OSE	mAP (↑)	WI	A-OSE	mAP (↑)			mAP (↑)		
	(↓)	(↓)	Current known	(↓)	(↓)	Previously known	Current known	Both	Previously known	Current known	Both
Faster-RCNN [20] + Finetuning	0.04563	12628	46.01	0.02351	14738	42.86	18.31	32.34	28.38	37.96	31.26
ORE [10]	0.03244	6186	44.43	0.01807	5028	37.54	18.65	29.44	27.80	40.70	31.67
ORDER	**0.02994**	**2859**	**46.50**	**0.00983**	**3353**	40.65	24.89	**33.90**	34.35	45.25	**37.62**

IDD

Task IDs (→)	Task 1			Task 2					Task 3		
	WI	A-OSE	mAP (↑)	WI	A-OSE	mAP (↑)			mAP (↑)		
	(↓)	(↓)	Current known	(↓)	(↓)	Previously known	Current known	Both	Previously known	Current known	Both
Faster-RCNN [20] + Finetuning	**0.09559**	21539	**35.79**	0.06279	21134	21.25	27.79	24.52	23.84	23.48	23.72
ORE [10]	0.10702	13149	35.01	**0.05999**	13507	18.17	26.49	22.33	25.76	22.04	24.52
ORDER	0.09984	**10353**	35.20	0.06460	**10879**	20.13	29.88	**25.01**	25.08	24.48	**24.88**

Table 3: Ablation study of proposed components in ORDER on Task 1 of IDD. Best results are highlighted in bold.

Regression Loss	Feature Mix	Curriculum Training	WI	A-OSE	mAP
Smooth-L1 [20]	✗	✗	0.10702	13149	35.01
Focal Regression	✗	✗	0.11021	13084	**36.58**
Focal Regression	✓	✗	0.10996	10563	33.90
Focal Regression	✓	✓	**0.09984**	**10353**	35.20

5.3 Sensitivity Analysis of Feature-Mix:

We show the variation in the performance of ORDER by changing the contribution of the feature-mix in the total loss. Table 4 (b) shows the performance of ORDER on Task 1 of IDD having various loss weights denoting the fraction of feature-mix loss contributed towards total loss. We find that tuning the feature-mix weights to 0.001 gives the best performance on almost all the evaluation metrics.

5.4 Qualitative Results

Qualitative results demonstrating the ORDER's capability to i) handle intra-class scale variations, ii) detect small objects, and iii) discriminate *knowns* from *unknowns* can be seen in fig. 3. We show the sample results of the model trained on task 2 of IDD and BDD datasets. As shown, ORDER performs better than ORE for the three different cases. The key observations are that ORE misses several known objects (especially cars in IDD and pedestrians in BDD) and demonstrates confusion among detected unknown and known objects (especially traffic signs in BDD). On the contrary, ORDER performs considerably better for such cases with high confidence. More qualitative results in appendix A.

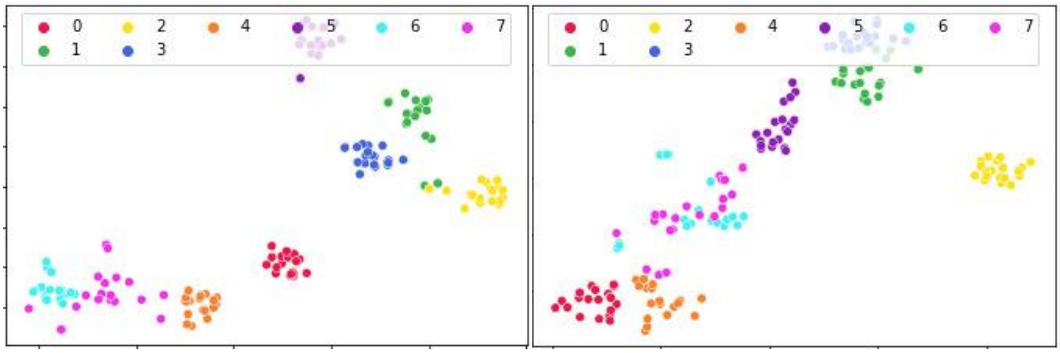
Table 4: (a) Performance of ORDER when trained on various bounding box losses. (b) Sensitivity analysis of Feature-Mix loss contribution. All the experiments are conducted on the Task 1 of IDD. Best results are highlighted in bold.

(a)

Loss	mAP
Smooth-L1 [20]	34.01
GIoU [21]	32.53
Least Square IoU [17]	32.44
Focal Regression (Ours)	**35.20**

(b)

Loss weight	WI	A-OSE	mAP
1	0.10153	10088	35.13
0.1	0.10169	**10069**	35.12
0.01	0.10108	10145	35.14
0.001	**0.09985**	10353	**35.20**



ORDER ORE [10]

Figure 4: We show the t-SNE plots of latent features of ORDER and ORE on Task 2 of BDD. Class label 7 denotes the *unknown* class, and the remaining classes are *known*. We can see that ORDER clearly separates class 7 with 6, whereas in ORE, these classes are intertwined. We can also notice that the separability between the smaller objects such as 0 (*traffic sign*) and 4 (*traffic light*) is better in ORDER.

5.5 Latent Feature Visualization

We show the visualization of latent features of ORDER and ORE. These features are obtained after RoI pooling from the model trained on BDD Task 2 and then visualized using t-SNE [25]. Figure 4 shows the clusters formed by latent features belonging to various classes. 7 represents the *unknown* class and the rest as *known* class labels. We can observe the features cluster of ORDER have better quality compared to ORE and can better differentiate *unknown* class feature label (7) with the known class feature (6).

6 Conclusion

This work introduces the ORDER framework designed to handle Open World Object Detection challenges on road scene datasets. We demonstrate that ORDER outperforms the current state-of-art open world detector. The key contribution consists of *Feature-Mix* that improves the *unknown* object identification. Further, we handle the intra-class scale variation and small object detection by proposed *Focal Regression Loss* and curriculum learning. Currently, ORDER trains on the tasks that belong to a single road scene dataset. In the future work, we plan to extend ORDER to be trainable on tasks that belong to multiple road scene datasets captured in different geographic locations. We hope this work will open doors for further research to make vision models more robust in real-world scenarios, resulting in safer and reliable autonomous navigation systems.

Acknowledgement: This work was partly funded by IHub-Data at IIIT-Hyderabad and DST through the IMPRINT program.

References

[1] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009. 5

[2] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016. 1, 3

[3] Akshay Dhamija, Manuel Gunther, Jonathan Ventura, and Terrance Boult. The overlooked elephant of object detection: Open set. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1021–1030, 2020. 3, 5

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 1

[5] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7036–7045, 2019. 1

[6] Guy Hacohen and Daphna Weinshall. On the power of curriculum learning in training deep networks. In *International Conference on Machine Learning*, pages 2535–2544. PMLR, 2019. 5

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 5

[8] David-Traian Iancu, Alexandru Sorici, and Adina Magda Florea. Object detection in autonomous driving - from large to small datasets. In *2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages 1–6, 2019. 3

[9] Licheng Jiao, Fan Zhang, Fang Liu, Shuyuan Yang, Lingling Li, Zhixi Feng, and Rong Qu. A survey of deep learning-based object detection. *IEEE access*, 7:128837–128868, 2019. 5

[10] KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Vineeth N Balasubramanian. Towards Open World Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5830–5840, 2021. 1, 3, 4, 5, 7, 8, 9, 12

[11] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 1

[12] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 1

[13] Guangrui Liu. *Real-Time Object Detection for Autonomous Driving Based on Deep Learning*. PhD thesis, Texas A&M University-Corpus Christi, 2017. 3

[14] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 1, 3

[15] Dimity Miller, Feras Dayoub, Michael Milford, and Niko Sünderhauf. Evaluating merging strategies for sampling-based uncertainty techniques in object detection. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 2348–2354. IEEE, 2019. 3

[16] Dimity Miller, Lachlan Nicholson, Feras Dayoub, and Niko Sünderhauf. Dropout sampling for robust object detection in open-set conditions. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3243–3249. IEEE, 2018. 3, 5

[17] Xiang Ming, Fangyun Wei, Ting Zhang, Dong Chen, and Fang Wen. Group sampling for scale invariant face detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3446–3456, 2019. 6, 9

[18] Can Peng, Kun Zhao, and Brian C Lovell. Faster ilod: Incremental learning for object detectors based on faster rcnn. *Pattern Recognition Letters*, 140:109–115, 2020. 5

[19] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 1, 3

[20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015. 1, 3, 4, 6, 8, 9

[21] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 658–666, 2019. 4, 6, 9

[22] Walter J. Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E. Boult. Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1757–1772, 2013. 1

[23] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *Proceedings of the IEEE international conference on computer vision*, pages 3400–3409, 2017. 5

[24] Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum self-paced learning for cross-domain object detection. *Computer Vision and Image Understanding*, 204:103166, 2021. 5

[25] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 9

[26] Girish Varma, Anbumani Subramanian, Anoop Namboodiri, Manmohan Chandraker, and CV Jawahar. IDD: A dataset for exploring problems of autonomous navigation in unconstrained environments. In *WACV*, 2019. 2

[27] Bichen Wu, Forrest Iandola, Peter H Jin, and Kurt Keutzer. Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 129–137, 2017. 1, 3

[28] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3974–3983, 2018. 2

[29] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. BDD100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020. 2

[30] Zhun Zhong, Linchao Zhu, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. Openmix: Reviving known knowledge for discovering novel visual categories in an open world. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

A Appendix

Figure 5: The images in row 1,2, and 3 show the result of ORDER on Task 1,2, and 3 respectively. We can notice that the `rider` and `motorcycle` class which are *unknown* in the results of Task 1, are subsequently learnt in Task 2 and Task 3. **Best viewed when zoomed**.

Figure 6: Each row represnts the qualitative results on ORDER and ORE framework. In row b) we can notice that the `bicycle` class is not being detected by ORDER whereas it is being detected as *unknown* by ORE, which is a mis-detection by both the frameworks. Rows a) and c) shows better *known* and *unknown* detections by ORE.