New Objects on the Road? No Problem, We'll Learn Them Too

New Objects on the Road? No Problem, We'll Learn Them Too

Deepak Kumar Singh*,[1] Shyam Nandan Rai*,[1,4] K J Joseph[2] Rohit Saluja[1]
Vineeth N Balasubramanian[2] Chetan Arora[3] Anbumani Subramanian[1] C.V. Jawahar[1]

***Abstract*— Object detection plays an essential role in providing localization, path planning, and decision making capabilities in autonomous navigation systems. However, existing object detection models are trained and tested on a fixed number of known classes. This setting makes the object detection model difficult to generalize well in real-world road scenarios while encountering an unknown object. We address this problem by introducing our framework that handles the issue of unknown object detection and updates the model when unknown object labels are available. Next, our solution includes three major components that address the inherent problems present in the road scene datasets. The novel components are a) Feature-Mix that improves the unknown object detection by widening the gap between known and unknown classes in latent feature space, b) Focal regression loss handling the problem of improving small object detection and intra-class scale variation, and c) Curriculum learning further enhances the detection of small objects. We use Indian Driving Dataset (IDD) and Berkeley Deep Drive (BDD) dataset for evaluation. Our solution provides state-of-the-art performance on open-world evaluation metrics. We hope this work will create new directions for open-world object detection for road scenes, making it more reliable and robust autonomous systems.**

I. INTRODUCTION

Autonomous navigation systems such as self-driving cars have become increasingly popular over recent years. However, the generalization of autonomous navigation systems across various geographic locations are challenging. Suppose a self-driving car is trained on German streets where it learns to localize the objects such as pedestrians, trucks which helps the car manoeuvre safely is deployed in India. On Indian streets, the car encounters novel classes such as autorickshaw (tuk-tuk) and finds it challenging to navigate accurately, as indicated in fig. 1. We address this problem by improving an object detector's novel object localization capabilities, which plays a vital part in autonomous navigation systems.

Existing object detection models [22], [3], [16], [21], [27], [7], [13] are trained and tested on a fixed number of classes known as *closed-set setting*. Miller *et al.* [18] introduced *open-set* object detection that made the object detection model capable of detecting unknown objects present in the test set without training on them. However, *open-world* [6] is a more natural problem setting. Joseph *et al.*[12] introduced open-world for object detection. In this setting, along with detecting an unknown object, the model also updates itself if unknown object labels are available. We address the problem of open-world object detection for road scenes by presenting

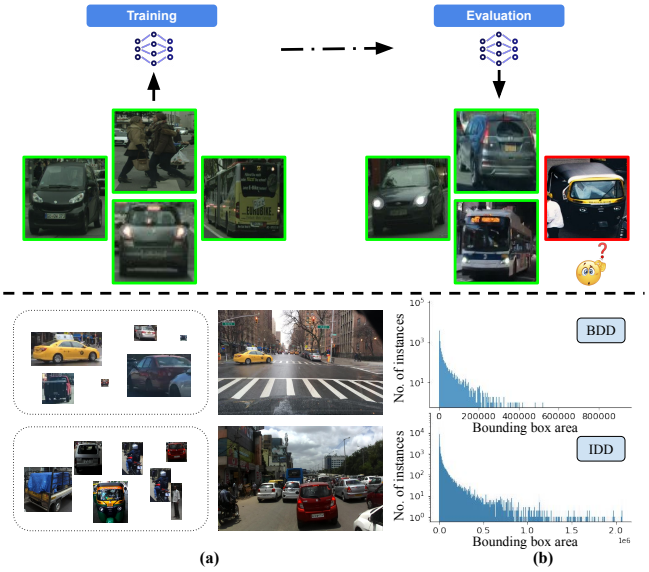

Fig. 1: **Top**: Displays the problem statement when an object detector is trained on a fixed number of classes (indicated in the green box) and struggles to correctly detect novel/unknown class (indicated in the red box). **Bottom**: Shows the challenges in road scene datasets. (a): We can observe intra-class 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.

our framework that encapsulates the capabilities to localize unknown classes on road scenes and update the framework when the labels of unknown classes are available. Another important aspect of our work is we identify the inherent challenges present in road scene datasets such as Berkeley Deep Drive (BDD), which includes a) *unknown* objects that are hard to detect compared to the generic dataset such as MS-COCO [14] and PASCAL-VOC [5]; b) the proportion of small objects (from both *known* and *unknown* sets) is significant (fig. 1 [b]), and c) the presence of pronounced intra-class scale variation (fig. 1 [a]). Generic datasets such as MS-COCO and PASCAL-VOC consist of images captured close to the object resulting in smaller variations in scale. Likewise, in the aerial object dataset [28], the objects are captured at high altitudes resulting in a small intra-class object size variation.

We introduce three major components into our solution

*Equal Contribution.
[1]CVIT - IIIT Hyderabad, India, [2]IIT Hyderabad, India, [3]IIT Delhi, India , [4]Politecnico di Torino, Italy. Email: deepak.singh.@research.iiit.ac.in

to address the inherent challenges present in the road scene dataset. Firstly, we present Feature-Mix that improves *unknown* object identification. It works by combining multiple *unknown* and *known* class features and then maximizing their differences in the latent space. Feature-Mix is inspired by Open-Mix [30]. However, Open-Mix lacks the ability to combine multiple *unknown* and *known* class instances commonly present in road scenes. Next, we propose Focal Regression Loss that jointly addresses the problem of significant intra-class variation and small object detection. It dynamically changes the loss according to object size by giving heavier weightage to small bounding boxes than large bounding ones, making the model more focused on small object detection.

Lastly, we train our framework in a curriculum style by initially training on easy samples (large bounding boxes) then progressively training on hard samples (small bounding boxes). Curriculum learning improves the small object detection and lowers the chances of a *known* class detected as *unknown*. The backbone of our framework is motivated by Joseph *et al.* [12] Open World Object Detector (ORE) framework. However, it is important to note that ORE shows poor performance when applied to road scenes as the framework struggles to handle the challenges present in road scenes. Our framework is validated on Indian Driving Dataset (IDD) [26] and Berkeley Deep Drive (BDD) [29] datasets. We use Wilderness Impact (WI) and Absolute Open-Set Error (A-OSE) as open-world evaluation metrics to measure the performance of our method and the baselines. We also perform qualitative and quantitative ablation studies to show our method's efficacy on road scenes. Our work contribution can be summarised as follows:

- To the best of our knowledge, our framework is the first effort to perform open-world object detection for road scenes.
- We introduce Feature-Mix that remarkably improves the identification of *unknowns*.
- Focal Regression Loss and Curriculum learning are introduced in our framework to address intra-class scale variation and improve small object detection present in road scene datasets.

II. Related Works

Object detection methods have made significant progress with the introduction of deep-learning-based object detectors [22], [21]. Existing object detection models can be classified into two classes of detectors: a) Two-stage detectors, which rely on Region Proposal Network to get the final object bounding box. Faster R-CNN [22], R-FCN [3], FPN [13], and Cascade R-CNN [2] are the recognized object detectors in this category. b) Single-stage detector that relies on a single network to predict object bounding box. SSD [16], YOLO [21], and SqueezeDet [27] are examples of notable single-stage object detectors. Object detection models are used in the autonomous navigation system to localize and classify the object class. Some of the recent works by [15], [27], [10] focus object detection on road scene dataset applicable for autonomous navigation systems.

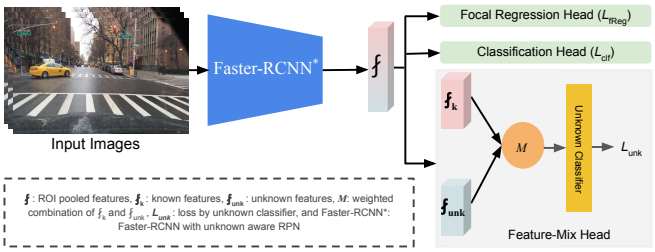

Fig. 2: Illustration of our framework. f is RoI pooled features consisting of *known* class features f_k and *unknown* class features f_{unk} mixed in Feature-Mix M block. L_{clf}, L_{fReg} and L_{unk} denotes the classification loss, focal regression loss, and feature-mix loss, respectively.

Miller *et al.* [18] introduced open-set object detection to improve the generalization capabilities of an object detector in real-world scenarios. They used the dropout sampling method to identify *unknown* objects present in the test set. Further, Miller *et al.* [17] used different merging methods for Monte Carlo dropout to improve object detection in an open-set setting. A thorough insight is recently given by Dhamija *et al.* [4] regarding the performance of object detectors in the open-set setting. Additionally, they propose Wilderness Impact(WI), an evaluation metric that quantifies the performance of the object detection model in the real world.

However, a more practical problem setting would be addressing the problem of open-world object detection, which is presented by Joseph *et al.* [12]. In this problem setting, the model detects the unknown object and updates the model incrementally when the labels of unknown classes are available. Joseph *et al.* [12] introduced Open World Object Detector (ORE) that performs the task of open-world object detection. ORE uses Faster R-CNN as the base detector because it is a two-stage detector with better accuracy than a single-stage object detector. It improves the unknown classes identification by adding contrastive clustering and an energy-based classifier. Although ORE gives good performance on the generic dataset, it is not designed to address the challenges such as intra-class scale variation present in the road scene dataset, which our framework handles.

III. Methodology

A. Basic Problem Setting

In Open World Object Detection, we assume the *known* classes as C_k and *unknown* classes C_{unk}. The ground-truth bounding boxes of *known* classes are available and are used during training and evaluation, whereas the *unknown* class bounding boxes will only be available during evaluation. The detection model D is trained on *known* classes, and simultaneously the *unknown* class instances are also learned in an unsupervised manner. In the subsequent steps, the labels for future classes are provided. Next, D is incrementally trained on newly available ground-truth labels to get an updated detection model $\hat{D}$. This process can be performed

for multiple sets of classes. In the experimental setting, we define the set of classes as task T.

In our basic framework adopted from Joseph *et al.*[12], we use unknown aware FasterRCNN [9] in fig. 2 as our object detector, and contrastive clustering and energy-based unknown identification to learn unknown classes. Next, we introduce three key novel components to the basic framework that helps it adapts to the road scenes. a) Feature-Mix that improved the *unknown* object class detection, b) Focal Regression Loss that minimizes the problem of intra-class variation and small object detection and c) curriculum learning improves small object detection. Now, we will discuss each key component in detail.

B. Feature-Mix

An ideal unknown class identifier should differentiate accurately between *known* and *unknown* classes. So, to improve the differentiability between *known* and *unknown* classes, we introduced Feature-Mix into our framework. Feature-Mix combines *known* and *unknown* features and then suppresses the activation caused by *known* features, making the latent difference between *known* and *unknown* features wider. In Feature-Mix, we take the Region of Interest (RoI) pooling output features f provided by Faster-RCNN in fig. 2. f consists of individual *known* class features f_k and *unknown* class features f_{unk}. Now, we randomly choose a *known* class feature f_{k_i} and a *unknown* class feature f_{unk_j} and mix them according to the following equation:

$$f_{mix_{ij}} = \lambda f_{k_i} + (1-\lambda) f_{unk_j}, \tag{1}$$

where, λ is sampled from *beta distribution* parameterized with α and β, i and j represents the indices of *known* and *unknown* class features. Next, we train *unknown* classifier C_{unk} having $f_{mix_{ij}}$ as an input shown in fig. 2. The loss function L_{unk} is given by:

$$L_{unk} = -y \log \text{softmax}(C_{unk}(f_{mix_{ij}})), \tag{2}$$

$$y = \arg\max \log \text{softmax}(C_{unk}(f_{mix_{ij}})) \tag{3}$$

y represents the ground-truth label. We use a small held-out validation similarly utilized in ORE [12], consisting of known and unknown data samples to train Feature-Mix.

C. Focal Regression Loss

The next inherent challenge present in the road scene dataset is intra-class scale variation. Existing object detection approaches use losses such as Smooth-L1 [22], and Generalized Intersection over Union (GIoU) [23], which are not explicitly modeled to capture the intra-class scale variation of bounding boxes. We address this issue by introducing Focal Regression Loss (L_{fReg}) given by:

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

$$\gamma^* = \gamma + \log\log \hat{Ar}_{bbox_{gt}} \tag{5}$$

$$\hat{Ar}_{bbox_{gt}} = \frac{Ar_{Img}}{Ar_{bbox_{gt}}} \tag{6}$$

L_{fReg} consists of two parts: a) squared IoU loss denoted by $\|1 - IoU\|_2^2$, and b) regulating component as $(1 - IoU)^{\gamma^*}$. The regulating component can vary the squared IoU loss according to the focusing parameter $\gamma^* \in [0, \infty)$ whose value dynamically varies according to the bounding box size. γ^* is higher for small object bounding boxes and thus gives more penalty compared to large object bounding boxes whose γ^* is smaller. In this manner, we can focus on small bounding boxes, improve *known* class object detection, and minimize the possibility of being confused as an *unknown* class object or being left undetected.

The factor γ^* consists of a tunable scalar parameter γ, and a double logarithmic of inverse-normalized bounding box area. We apply double logarithmic to the inverse-normalized bounding box area because it prevents overshooting of γ^* when the bounding box area is tiny. Also, it smoothens out the significant variation in the bounding box area. Hence, this helps minimize intra-class scale variation, making the loss function more stable during training. $\hat{Ar}_{bbox_{gt}}$ and $Ar_{bbox_{gt}}$ represents the inverse-normalized and unnormalized bounding box area, respectively. $\hat{Ar}_{bbox_{gt}}$ results from dividing image area Ar_{Img} by $Ar_{bbox_{gt}}$. $\hat{Ar}_{bbox_{gt}}$ controls the value of γ^* by giving large values for small bounding boxes and small values for large bounding boxes.

D. Curriculum Training

Curriculum Learning [1], [8] is a training method that progressively trains a model from *easy* to *hard* samples. Road scene datasets such as BDD and IDD consist of all scales of objects due to intra-class scale variation, which makes curriculum learning a natural fit to train our model. Hence, we gradually train the network from *easy* samples (large objects) to *hard* samples (small objects). Another important aspect of using curriculum learning is that the road scene dataset consists of significant proportions of small objects, see fig. 1; detecting smaller objects [11] is a harder task than detecting object instances with larger sizes. Therefore, the training detection model is more stable when trained in a curriculum manner, and the model can better detect objects at different scales, especially the smaller objects that are hard to detect.

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:

$$T_i = \begin{cases} S_{easy} & I_1;\ \textit{if } Ar_{bbox} < Ar_{easy} \\ S_{easy} + S_{medium} & I_2;\ \textit{if } Ar_{bbox} < Ar_{medium} \\ S_{easy} + S_{medium} + S_{hard} & I_3 \end{cases} \tag{7}$$

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

IV. Experiments and Results

A. Datasets Protocol

We adapt the standard evaluation protocol of ORE [12] to demonstrate the efficacy of our approach. 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 above protocol 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 the supplementary video[1]. We take a set of $3K$ images from each dataset for validation.

B. 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 [24], [20], [12]. Now, to quantify the performance of a model for *unknown* identification, we use Wilderness Impact (WI) [4] metric. The WI measures the model's sensitivity to *unknowns* over a range of frequencies 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) [18] is another metric that shows the *unknown* detection performance of a model. It is defined as the total number of *unknown* objects getting classified as *known* object.

C. Implementation Details

We use the modified Faster R-CNN with ResNet-50 [9] 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.

[1]Supplementary Video Link: shorturl.at/gqR02

D. 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 I (top). The ORE reduces both WI and A-OSE (lower the better) compared to baseline for the first two tasks[2] of BDD. However, ORE drops in overall mAP by 2 (approx.) compared to baseline for the two tasks (columns 4 and 9 of table I [BDD]). Our method improves mAP by 0.5 and 1.4 for the two tasks and reduces WI by 0.015 and 0.013 compared to the baseline. Our approach also reduces the AOSE by a considerable margin of 9769 and 11385 as compared to the baseline. For task 3 of BDD, our method attains a massive gain in overall mAP of around 6.36 and 5.95, compared to the baseline and ORE (last column of table I [BDD]).

E. Results on IDD

On the IDD dataset, we observe in table I (bottom) that the WI is comparable for the three models. However, our method achieves the best A-OSE for the first two tasks of IDD, reducing it by a margin of 11186 and 10255 compared to the baseline and 2796 and 2628 compared to ORE. Our framework'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 I [IDD]).

It is also interesting to note that the performance of our method is better than ORE for all the columns in table I (refer to the last two rows of the tables).

V. Discussion and Analysis

A. Ablative Study

We perform ablative studies to validate the performance of the proposed components in our framework. Table II 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 II). It is also essential to infer from the first two rows of table II that the proposed focal regression loss shows significant improvement in mAP compared to Smooth-L1.

B. 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 [22], GIoU [23], and Least Square IoU [19]. Table III 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.

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

TABLE I: Quantitative performance on road scene datasets. We notice that our method 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 [22] + Finetuning	0.04563	12628	46.01	0.02351	14738	42.86	18.31	32.34	28.38	37.96	31.26
ORE [12]	0.03244	6186	44.43	0.01807	5028	37.54	18.65	29.44	27.80	40.70	31.67
Ours	**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 [22] + Finetuning	**0.09559**	21539	**35.79**	0.06279	21134	21.25	27.79	24.52	23.84	23.48	23.72
ORE [12]	0.10702	13149	35.01	**0.05999**	13507	18.17	26.49	22.33	25.76	22.04	24.52
Ours	0.09984	**10353**	35.20	0.06460	**10879**	20.13	29.88	**25.01**	25.08	24.48	**24.88**

TABLE II: Ablation study of proposed components in our framework on Task 1 of IDD. Best results are highlighted in bold. FM and CL are abbreviated for Feature-Mix and Curriculum Learning, respectively.

Regression Loss	FM	CL	WI	A-OSE	mAP
Smooth-L1 [22]	✗	✗	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

TABLE III: Performance of our method when trained on various bounding box losses. All the experiments are conducted on Task 1 of IDD. Best results are highlighted in bold.

Loss	mAP
Smooth-L1 [22]	34.01
GIoU [23]	32.53
Least Square IoU [19]	32.44
Ours	**35.20**

TABLE IV: Sensitivity analysis of Feature-Mix loss contribution. All the experiments are conducted on Task 1 of IDD. Best results are highlighted in bold.

β	WI	A-OSE	mAP
1	0.102	10088	35.1
0.1	0.101	**10069**	35.1
0.01	0.10108	10145	35.14
0.001	**0.09984**	10353	**35.20**

C. *Sensitivity Analysis of Feature-Mix:*

We show the variation in performance of our method by changing the contribution of the feature-mix in the total loss. Table IV shows the performance of our method 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.

D. *Qualitative Results*

Qualitative results demonstrate our method'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, our method performs better than ORE for all the three examples. 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,

Fig. 3: Qualitative comparison: Column 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 column a), we observe that our method can detect smaller objects with high confidence. Interestingly, the highlighted boxes of a) have `car` instances that show intra-class scale variation. Our approach handles the intra-class scale variation within the `car` instance by detecting it on varying scales. In column b) and c), we can see that our method detects safety-critical classes such as `pedestrian` and `traffic sign` better than ORE. We also notice that our method better recognizes 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 ones denote the *known* classes. The blue and pink boxes represent the cropped region. **Best viewed when zoomed**.

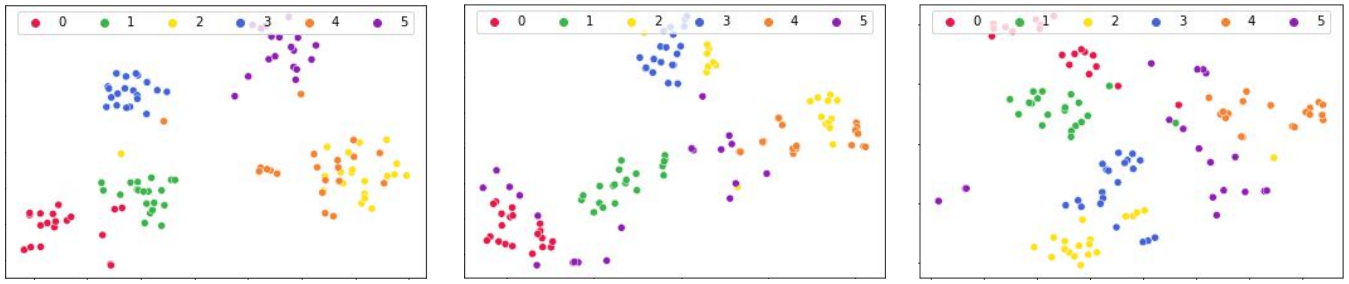

Fig. 4: We show the t-SNE plots of latent features of our method, ORE [12], and the baseline Faster-RCNN+FT(Fine-Tuned) on Task 1 of IDD. Class label 5 denotes the *unknown* class, and the remaining classes are *known*. Our method has tighter clusters that are well separated. Due to the introduction of Feature-Mix, we can notice distinguishable *unknown* cluster which is well separated from *known* classes.

our approach performs considerably better for such cases with high confidence. More qualitative results are in the supplementary video[3].

[3]Supplementary Video Link:`shorturl.at/gqR02`

E. Latent Feature Visualization

We show the visualization of latent features of our method, ORE [12], and the baseline Faster-RCNN+FT(Fine-Tuned). These features are obtained after RoI pooling from the model trained on IDD Task 1. Figure 4 shows the t-SNE [25] clusters

formed by latent features belonging to various classes. The feature of category 5 represents the *unknown* class, and the rest are *known* class labels. Our method produces tighter and well separated clusters. The introduced Feature-Mix component has effectively improved the *unknown* feature representation, which is prominent across each method.

VI. Conclusion

In this work, we introduce a novel approach that detects unknown objects on road scenes and performs open-world object detection. Our method gives a state-of-the-art performance on open-world object detection on various evaluation matrices. Another key contribution of our work was it addressed the inherent challenges such as intra-class scale variation and small object detection present in the road scene dataset by introducing Feature-Mix, Focal Regression Loss and curriculum learning. Currently, our method trains on the tasks that belong to a single road scene dataset. In future work, we plan to extend our approach to be trainable on tasks that consist of 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 more reliable autonomous navigation systems.

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.

[2] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[3] 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.

[4] 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.

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

[6] Dario Fontanel, Fabio Cermelli, Massimiliano Mancini, Samuel Rota Bulo, Elisa Ricci, and Barbara Caputo. Boosting deep open world recognition by clustering. *IEEE Robotics and Automation Letters*, 5(4):5985–5992, 2020.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

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

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

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

[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.

[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.

[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.

[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.

[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.