Few-Shot Batch Incremental Road Object Detection via Detector Fusion

Anuj Tambwekar [*]   Kshitij Agrawal   Anay Majee   Anbumani Subramanian
PES University   Intel Corporation

anujstam@gmail.com   firstname.lastname@intel.com

Abstract

Incremental few-shot learning has emerged as a new and challenging area in deep learning, whose objective is to train deep learning models using very few samples of new class data, and none of the old class data. In this work we tackle the problem of batch incremental few-shot road object detection using data from the India Driving Dataset (IDD). Our approach, DualFusion, combines object detectors in a manner that allows us to learn to detect rare objects with very limited data, all without severely degrading the performance of the detector on the abundant classes. In the IDD OpenSet incremental few-shot detection task, we achieve a mAP_{50} score of 40.0 on the base classes and an overall mAP_{50} score of 38.8, both of which are the highest to date. In the COCO batch incremental few-shot detection task, we achieve a novel AP score of 9.9, surpassing the state-of-the-art novel class performance on the same by over 6.6 times.

1. Introduction

Object detection is a challenging task that aims to detect the presence and location of desired objects in an image. It is a crucial part of smart transportation, where it is used by both autonomous vehicles, as well as advanced driver assistant systems (ADAS) [25].

Current state of the art object detection networks such as Faster R-CNN [19], YOLO [18] and RetinaNet [12] perform well on canonical object detection datasets such as PASCAL VOC [3] and COCO [13]. These object detection networks are reliant on large-scale annotated datasets for training. Acquiring such large volumes of data is both expensive or requires tremendous amount of human effort. The traditional object detection datasets, are well-balanced and contain a sufficiently large number of examples per category. However, in a real world, classes of interest may be rare and hard to come by, thus resulting in a data scarcity due to the nature of the objects in question. This problem is

one of the key challenges in detecting rare objects found on Indian roads [15], where there is a significant class imbalance between the frequently occurring classes, and the rarer classes.

Few shot learning has been proposed to overcome these challenges and learn with limited amount of data. While significant progress has been made in this area in both the classification [26, 22, 23] and detection [10, 9, 27, 4, 30] tasks, most techniques assume that the data from the abundant classes is always available. This requirement cannot always be met, as there may be scenarios where we require to increase the capability of the model to detect new classes without access to old data. For example if the models are deployed in autonomous vehicles and the vehicle needs to detect new objects on the road. This challenge of learning to detect new classes in addition to existing classes, while only being exposed to the new classes, is termed incremental object detection. As described in [17], learning one class at a time is termed class incremental object detection, while learning to detect multiple new classes in one attempt is termed batch incremental object detection. Few-shot object detection techniques also suffer from class confusion [29], a phenomenon where objects are located correctly, but classified into the wrong class. Base-novel confusion refers to this incorrect classification, where an object belonging to a base class is incorrectly labeled as an object of a novel class, or vice-versa.

Most existing works treat few-shot detection and incremental detection as two separate tasks, when in reality it is important to address both simultaneously. One of the key requirements of existing incremental learning techniques [20, 2, 7] is a reliance on having a large amount of data corresponding to the new classes, making them ineffective in a few-shot setting. Additionally, incremental learning techniques are prone to *catastrophic forgetting* [5, 16], a phenomenon where the incremental detector fails to detect the previously learnt classes and thus forgets the previously learnt features. In a safety-critical environment such as an ADAS or an autonomous vehicle, it is crucial to ensure that none of the the prior learning is forgotten, making existing techniques unsuitable for these tasks.

---

[*]Work done as an intern at Intel Corporation

In this work, we propose *DualFusion*, a method to address few-shot batch incremental learning to detect road objects using only 10 annotated instances of each novel class, and without any base class data. The proposed architecture combines class-specific object detectors using a proposal segregation technique. This improves the classification ability of the combined detector to detect novel and base classes from overlapping regions of interest, while still retaining knowledge from the individual classifiers as no retraining is involved. We specifically design this technique for contextual datasets like road object detection where rare novel classes are found alongside bases classes.

The rest of this paper is divided as follows - Section 2 provides an overview of the background and related work. Section 3 describes the architecture of DualFusion in detail, along with its training strategy. Section 4 goes over the experimental setup used to validate DualFusion on road object detection with the India Driving Dataset (IDD)[24] as well as more generalized cross-domain setting on COCO, where it achieves SOTA performance on novel classes. Finally in Section 5 we conclude and discuss the advantages, drawbacks and scope for extension.

2. Background and Related Work

2.1. Object Detection

Standard object detection techniques require large amount of data to train and aim to localize and classify objects present in an image. They are broadly categorized into two groups - single stage detectors and two-stage detectors. Single Stage detectors such as SSD [14], Retinanet [12] and YOLO [18] extract regions of interest and classify objects in a single step. Meanwhile, two stage detectors such as Faster R-CNN [19] first identify regions of interest via a Region Proposal Network (RPN), and then determine the bounds of the object and its class based on the contents of the region of interest. These techniques cannot be applied directly when dealing with heavy class imbalance and when only a limited amount of data is available as shown by prior work such as [9, 27].

2.2. Few-Shot Object Detection

Few-shot learning [28] is a technique that learns a data distribution using very limited training data. Formally, if N new classes are learnt in the few-shot setting, and there are K training examples per class, it is termed N-way K-shot learning.

Recent successful few-shot object detection techniques are episodic approaches that operate by reweighting the features between classes. such as Feature-Reweight [9]. Meta-RCNN [30] extends the previous approach to a Faster R-CNN based network. It is a meta-learning technique that uses an episodic training regime to attempt to train the net-

work to learn how to learn better. In a similar fashion Kim et al. [10] uses a meta-learner that employs knowledge transfer to create class-vector prototypes that are used to provide weights to the features. MetaDet [29], uses meta-learning to create a model that learns to transform weights learnt during the few-shot training stage to large-scale training weights.

Two-stage finetuning [27], provided a different viewpoint, and instead suggested that the lower layers of the detectors produced class agnostic features, while the higher layers produced class specific features. The technique, known as FsDet simply called for fine-tuning the final layer using cosine similarity in order to learn the new classes. However, FsDet is reliant on a well balanced training dataset which contains both the base and the novel classes in the fine-tuning stage. This reliance on base class labels during the fine tuning stage makes it unsuitable for an incremental learning setting.

One of the best performing few-shot object detection techniques is FewX [4], which uses attention maps as a way to search for the presence of the novel classes in an image. In theory, this allows the network to detect novel classes without any kind of fine-tuning, if trained on a sufficiently varied dataset, but for context specific situations, fine-tuning greatly improves performance. While this architecture claims the highest performance on the COCO [13] dataset, it suffers from the glaring limitation of only being able to detect the novel classes. FewX uses attention as a mechanism to find candidate proposals and then classify objects in an image. Hence, if no base classes are present in the support set, FewX will never be able to detect them. This makes it an impractical solution when any of the classes have sufficient data and need not be treated as few-shot, as few-shot performance is always less than standard training performance. It also poses a challenge when access to the base class data is withheld.

2.3. Incremental Object Detection

Incremental object detection learns to detect new classes in addition to the existing classes. In this setting, the detectors are trained to retain the base class performance and while only being exposed to the new classes. Incremental learning can be divided into two categories based on the number of classes being added in every stage - single class incremental learning where new categories are introduced one at a time, and batch incremental learning, where multiple new categories are learnt together.

Most existing incremental object detection techniques [21, 2, 7] operate on knowledge distillation [8]. [21] uses a balanced loss combing distillation as well as cross-entropy to teach a new network to detect new classes, as well as to attempt to transfer the knowledge from the old detector. [2] uses intermediate proposals, along with two terms they call hint loss and confidence loss, to transfer feature maps
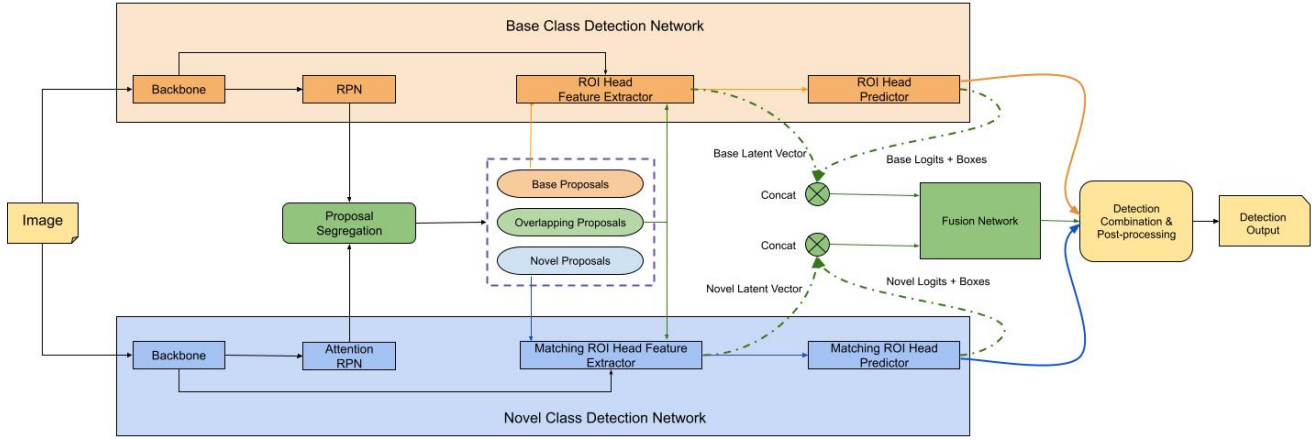
Figure 1: Architecture of DualFusion. Components in orange denote parts of the base detector, those in blue denote parts of the novel detector, and those in green denote the Fusion Network. Coloured arrows denote the flow of the various proposal categories - orange for base categories, blue for novel, and green for overlapping.

from the original model. [7] expands the foreground domain of the region proposal network, and uses a nearest-prototype classifier as well as distillation loss to expand the detector's ability to add in new classes. The chief drawback with all these techniques is that they assume an abundance of data corresponding to the new classes. The current state-of-the-art technique, Open-ended Centre Net (ONCE) [17], is the only existing few-shot incremental object detection network, and works by re-weighting image features based on a class code generator. The reported quantitative results for the novel classes are, however, very low.

3. Our Approach : DualFusion

In this work we target few-shot batch incremental object detection. In this setting, the objective is to learn to detect multiple novel classes at once, while retaining the ability to detect the base classes, using only novel class data in the incremental addition stage.

One of the biggest challenges with any incremental learning scenario is *catastrophic forgetting* [5]. While techniques such as knowledge distillation [8] have been proposed to combat this limitation, they still result in a heavy loss of performance on the base classes [1, 21]. We propose a simple yet powerful solution to circumvent this problem - instead of re-training the existing network, or training a new network using a distillation loss, we retain the original network, and combine it with another network specifically trained to discriminate for novel classes. Our approach can thus be extended to detect multiple novel classes incrementally without the need for base class data.

Formally, we can define the task as follows - Given a set of base classes B, a detector capable of detecting the base

classes D_B, and novel classes N, our goal is to create a new detector D', capable of detecting all classes from B and N, while only having access to 10 annotated instances of each class in N, and no annotated instances of classes in B.

3.1. Architecture

The DualFusion architecture comprises of 3 components

- A Faster R-CNN based base class detector to detect the base classes, that is trained only on base class data, with no exposure to novel classes.

- A few shot novel class detector based on FewX modified, so as to combine the outputs of each attention head into an intermediary layer before the classifier and box regressor.

- The Fusion Network. This network is designed to classify objects that are flagged as possible candidates by both the base class and the novel class detector. It is akin to an ROI head predictor that learns to combine the features obtained by the base and novel detector to classify objects.

Figure 1 provides a pictorial representation of the architecture. The black arrows indicate the flow of data before and after the proposal segregation step. The coloured arrows indicate the flow of data corresponding to the various proposal categories. Orange arrows denote the flow of data corresponding to the base class proposals, Blue arrows denote the flow of data corresponding to the flow of novel class proposals, and green arrows denote the flow of data corresponding to the overlapping proposals.

3.1.1 Architectural Setup and Flow

The final DualFusion model comprises of the three models mentioned earlier. Details about the training process are described subsequently in Section 3.2. In short, the faster RCNN base class detector is trained only on base classes, the novel class FewX detector is only trained on few-shot samples of the novel classes, and the Fusion network is trained using features extracted from the mined dataset whose collection is described in the Training section.

An input image is passed through the backbone and RPN of both the novel class and base class detectors. The proposals are then segregated into 3 categories - base proposal regions, novel proposal regions and overlapping proposal regions. We describe why this is needed and how it is exactly done in Section 3.1.2. The base proposals are passed through the base class detector as usual to obtain a set of features, which are then converted into a detection output. In a similar fashion, the novel class detector acts on novel proposals and identifies the rare novel classes.

The overlapping proposals are passed to **both** detectors' feature extraction networks, and the penultimate outputs and the predicted boxes and logits are extracted. These latent feature and prediction combinations are first passed through a single fully-connected layer, following which they are concatenated and passed to the fusion network, which learns to distinguish between the base and novel detections, thereby reducing base-novel confusion. Each input is passed to a corresponding two layer deep fully connected network that uses the SeLU [11] activation function - this allows the features to be transformed into the same feature space. The transformed features are then combined and passed through two layers with ReLU activation, for the class score and bounding boxes predictions.

As result, we have a collection of three detection outputs at the end of a forward pass of an image. These detection outputs are combined, so as to remove overlapping predictions of the same region, and to reduce intra-novel confusion, by comparing the IoU between the three detection sets and selecting the detections with larger confidence scores whenever the IoU between detections from different detectors exceeds 0.5. This step is similar to the standard Non-Maximal Suppression (NMS) post-processing step of object detectors, however it considers the output of different detectors in the same region, as opposed to detections of the same class.

3.1.2 Proposal Segregation

The obvious question to ask is - "Why can't we just use two detectors?". A simple strategy of using two different detectors does not work as novel objects with similar features may be incorrectly classified as base class objects and vice-versa. Additionally, regions with features present in both the base and novel classes may be detected by both detectors, and thus the same region may be marked as two different objects (see Figure 4 for an example of the same). We term this phenomenon as base-novel confusion. Our design of fusing the intermediate outputs of each of the networks allows for a degree of communication between detectors, and thereby reduces this confusion. We utilize FewX as a novel class detector as it is, at the time of writing, the best performing few-shot detection model on COCO. Additionally, it is a two-stage approach, which allows us to implement proposal segregation.

Proposal segregation allows us to retain base and novel class performance by using the appropriate detector in regions without confusion. We first generate a collection of candidate proposals from the base detector P_B and candidate proposals from the novel detector P_N. Following this, we compute the element-wise Intersection over Area (IoA) between P_B and P_N. The proposals from P_B whose IoA with every proposal from P_N is below a certain threshold are considered valid base proposals. Likewise all proposals from P_N whose IoA with every proposal from P_B is below the threshold are considered valid novel proposals. The rest of the proposals are considered as overlapping proposals. We use IoA instead of Intersection over Union (IoU), as it provides a better estimate of proposal overlap between base and novel classes, and used a threshold value of 0.5.

3.2. Training

DualFusion is trained over a series of 4 steps

- **Step 1. Base Model Training**: To setup the base trained models, we train a Faster R-CNN based detector to detect IDD base classes. Following this, we train a FewX based model on the IDD base classes, using just 10 images from each class - this is done in order to allow for domain adaptation. The Faster R-CNN detector is frozen and from this point forward, we assume we have no access to the base training data.

- **Step 2. Novel Model Fine-Tuning** : We fine-tune the FewX based few-shot object detector specifically to detect novel classes. The only data we have access to at this point are driving scene images where novel classes are present. We assume 10 images per novel class, giving us a total of $10 \times N$ training samples, where N is the number of novel categories. Furthermore, we only use the ground truth annotations of the novel classes present in these training samples.

- **Step 3. Fusion Network Dataset Creation** : Due to the nature of images in IDD, the novel classes occur in driving scenes which contain the presence of base classes. As such, while the training samples only contain ground truth annotations of novel classes, there

exist un-annotated instances of some of the base class objects in the training data. By using the predictions from base class detector, we generate pseudo-ground truth labels for the base classes and use them for training. Any objects detected with a bounding box IoU of over 50% with a novel class are removed. It is not required, nor expected of the data to contain instances of every base object class as part of this scenario, as the goal is to simply teach the detector how to distinguish between the features of the base and novel detectors.

- **Step 4. Fusion Network Training** : Combining the pseudo labels of base class from Step 3, we generate the final training set. We train the fusion network that combines features from both the base and novel detectors to provide predictions in regions of high base-novel confusion.

(a) Tractor

(b) Street Cart

(c) Water Tanker

(d) Excavator

Figure 2: Examples of novel classes in IDD-OS

4. Experimental Setup and Results

To validate the efficacy of DualFusion as a batch incremental few-shot learning paradigm, we run a set of tests comparing it to existing techniques for both the India Driving Dataset (IDD) and COCO.

4.1. IDD

For our experiments we use the IDD-OpenSet (IDD-OS) split as described by [15] over conventional driving datasets such as Berkeley Deepdrive [31] and KITTI [6], due to the fact that it possesses the additional challenge of large amount of class imbalance, along with a well-defined splits for consideration of base classes and novel classes. Examples of the objects belonging to the novel categories can be seen in Figure 2.

Given an object detector D trained on the IDD Base Set classes - *Autorickshaw, Bicycle, Bus, Car, Person, Motorcycle, Rider, Traffic Light, Traffic Sign,* and *Truck*, train a detector D', that can detect the rare novel classes from IDD-OS - *Tractor, Street Cart, Water Tanker* and *Excavator*, in a batch incremental setting. The detector D' is trained to detect novel classes with only 10 annotations of each, while simultaneously having access to the original detector D.

We compare the performance of DualFusion to various techniques. We test two Faster R-CNN finetuning strategies. The first is standard Faster R-CNN Finetuning, where we simply add 4 additional outputs to the detector and train it on the novel classes. The second is Faster R-CNN Last-Layer Finetuning, where we finetune only the last layer. For both, we train for 2000 iterations at a learning rate of 0.002 and a learning rate decay step of 1000 iterations. We also test FsDet without the rebalancing finetuning stage as discussed by the authors in [27]. In this step, the merged network is finetuned on a balanced dataset containing annotations from all the classes. As this requires base class data, we skip this step, and instead use the model obtained after combining the model trained on the base data, and the model obtained by training on novel data. Lastly, we compare with FewX. All five networks are initially trained on IDD-Base, following which, they are trained on only the 10 novel examples from IDD-OS. As per [15], we report the mAP_{50} score for each category. Feature Reweight [9] was not used due to out-of-memory issues while training. Our Faster R-CNN base detector roofline gave us a base class mAP_{50} detection score of 50.1. We found that fine-tuning FewX was required in order to allow it to adapt to the driving domain. The FewX network and the DualFusion novel arm were trained for 2500 iterations, at a learning rate of 0.002 with steps at 1000 and 1500 iterations and a batch size of 2.

Results : Table 1 shows a comparison between the mAP_{50} scores of DualFusion and the prior mentioned Few-shot detection techniques. From the table we can observe that DualFusion is able to learn the novel classes extremely well, while significantly retaining base class knowledge, resulting in the highest base mAP_{50} and the the highest overall mAP_{50} performance. Additionally, DualFusion also performs better for certain novel classes (street cart and excavator) compared to FewX, due to its ability to reduce base-novel confusion. Our technique is able to obtain 80% of the base class performance of a conventional Faster R-CNN, achieving a mAP_{50} of 40.0, all while extending the ability of the detector to detect the novel classes. A few example detections predicted by DualFusion on IDD-OS can be seen in Figure 3 where they are contrasted with detections predicted by Faster R-CNN finetuning method. As seen in the figure, while finetuning is able to learn to pick out novel classes, it depicts severe base class forgetting. Du-

Method	Novel Classes (mAP_{50})				$mAP_{50,novel}$	$mAP_{50,base}$	$mAP_{50,all}$
	Tractor	Street Cart	Water Tanker	Excavator			
Faster R-CNN Finetuning	16.8	8.5	0.1	21.6	11.8	0.0	3.4
Faster R-CNN Finetuning (only last layer)	12.9	9.1	0.1	13.2	8.8	0.0	2.5
FsDet (no rebalancing finetuning)	0.0	0.0	0.0	0.0	0.0	9.42	6.73
FewX	**48.9**	**50.1**	31.5	22.0	**38.1**	-	10.9
DualFusion (ours)	30.5	28.6	**32.0**	**42.1**	33.3	**40.0**	**38.8**

Table 1: mAP_{50} Scores on the IDD Open Set 10-shot Batch Incremental Task. Cells filled with '-' indicate that the technique cannot be used for the task, and hence a score is not applicable. While FewX performs well on novel classes, it cannot detect the base classes. On the other hand, DualFusion achieves the highest base class and overall class performance on the task, and is clearly able to retain base class knowledge while learning to detect the novel classes.

(a) Tractor   (b) Street Cart   (c) Water Tanker   (d) Excavator

Figure 3: Examples of novel (green bounding boxes) and base (blue bounding boxes) class detections on IDD-OS via Faster R-CNN Finetuning (top row) and DualFusion (bottom row). Base class forgetting and erroneous novel class detections are clearly seen in Finetuning, but both are avoided by DualFusion.

alFusion is able to retain base classes as well as provide better detections for the novel classes. While FewX does outperform DualFusion in two novel classes, it is unable to detect any base classes due to the absence of base class supports in the incremental setting, showcasing DualFusion's superiority.

Figure 4 shows how the FusionNet helps reduce confusion between the base and novel classes, by handling overlapping proposal regions between the base and novel detectors. As seen in the figure, the autorickshaw class is incorrectly detected, as part of the excavator. By using the fusion approach, this region is evaluated as being part of the excavator, which gets rid of the incorrect prediction.

4.2. COCO

DualFusion is specifically designed for contextual datasets like IDD, as it leverages heavily on the presence of the novel classes in settings with abundant bases classes. To benchmark our DualFusion approach in a more generalized setting, we compare our network's performance to ONCE [17] on the COCO incremental learning task.

We use the incremental batch of 20 classes setup as de-

scribed by [17]. Our training methodology is identical to the training described in Section 3, with the only change being the use of the COCO dataset as opposed to IDD. We treat the 20 classes in common with the PASCAL VOC [3] dataset as novel (*aeroplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dining table, dog, horse, motorbike, person, potted plant, sheep, sofa, train* and *tv/monitor*), and the remainder as base classes. We consider the 10-shot setting once again, and the training examples used are the same as [4]. Due to computational constraints, we only trained our base Faster R-CNN on a single GPU, using the default training settings. As such, the base class Faster R-CNN detector was only able to reach a base class AP score of 19. We use the default FewX training settings to train our novel class detector. Our fusion model was trained for 10 epochs at a learning rate of 0.001. We used an IoA overlap threshold of 0.8 during the fusion training stage. The regular FewX model achieved an AP score of 13.1 on the novel classes, but as stated earlier, it fails for the base cases. As such we do not include it in Table 2.

Results: As seen in Table 2 the results clearly show that DualFusion has superior novel class performance compared

(a) Without FusionNet - Autorickshaw is incorrectly detected   (b) With FusionNet - No incorrect Autorickshaw detection

Figure 4: Example of an erroneous prediction removed via fusion. The FusionNet is able to prevent an object from being classified as both a base class and a novel class object by handling proposals where such confusion may occur.

Method	Base Classes AP	Novel Classes AP	All Classes AP
Fine-Tuning	2.8	0.6	2.3
Feature Reweight	3.7	1.5	3.1
ONCE	**17.9**	1.2	**13.7**
DualFusion (ours)	10.8	**9.9**	10.6

Table 2: Performance on COCO 20-Class Batch Incremental Few-shot Setting. DualFusion results in the highest novel class performance compared to the other techniques, achieving 6.6 × the performance of the closest competitor.

to ONCE, while still retaining a good degree of base class performance. The results for Finetuning, Feature Reweight and ONCE are from [17]. Finetuning refers to vanilla fine-tuning of a CentreNet [32] model on novel class data, and Feature Reweight is a CentreNet implementation of [9]. Since DualFusion is designed specifically for contextual settings like driving, it is encouraging to see good performance, even in a generalized object detection challenge like COCO, achieving almost 8 times the novel class performance. Granted, our initial base class performance was only 19 AP and as such, we retain over 56% of the base class performance in this setting, and retain roughly 76% of FewX's novel class performance. Interestingly, we also observed that 71 of the 80 classes were detected during the Fusion Network training stage, indicating our method of generating pseudo labels for base classes is quite efficient even in generalized settings. Moreover, it enables us to train the detector with no knowledge of base classes in a truly incremental setting. These results show that our work is applicable to more than just road object detection, and is applicable to wider settings.

5. Conclusion

In summary, through this work we present an architecture for batch incremental object detection for road object detection, that employs the simple, yet effective strategy of fusing specialized object detectors. DualFusion allows for incrementally adding the ability to detect new classes, using less data, with the only requirement being access to data from all the novel classes. This approach allows for a high degree of base class performance retention and prevention of catastrophic forgetting, while providing extremely high novel class performance. We solidify our claims by obtaining the highest base mAP_50 and overall mAP_50 scores (40.0 and 38.8 respectively), on the IDD-OS incremental task. While this technique is designed for road object detection, it is also suitable for other object detection problems where the classes of interest do not occur in isolation. This can be seen by its performance on the COCO dataset where it achieves the highest recorded novel class detection score of 9.9 AP in the 20 class few-shot batch incremental setting.

The chief limitation of DualFusion is its reliance to permanent access to novel class data. While it requires access to the base class data only once, all the novel few-shot data must be retained. Additionally, the evaluation time grows substantially with the addition of more novel classes. Techniques to overcoming these limitations could pave the way for lifelong few-shot object detection, which in turn would greatly increase the robustness and reliability of autonomous vehicles and similar safety critical systems.

References

[1] Francisco M. Castro, Manuel J. Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Computer Vision – ECCV 2018*. Springer International Publishing, 2018.

[2] Li Chen, Chunyan Yu, and Lvcai Chen. A new knowledge distillation for incremental object detection. In *2019 International Joint Conference on Neural Networks (IJCNN)*, 2019.

[3] Mark Everingham, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The pascal visual

object classes (voc) challenge. *Int. J. Comput. Vision*, 88(2):303–338, June 2010.

[4] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *CVPR*, 2020.

[5] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4), 1999.

[6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[7] Yu Hao, Yanwei Fu, Yu-Gang Jiang, and Qi Tian. An end-to-end architecture for class-incremental object detection with knowledge distillation. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, 2019.

[8] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[9] Bingyi Kang, Z. Liu, Xin Wang, F. Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[10] Geonuk Kim, Hong-Gyu Jung, and Seong-Whan Lee. Few-shot object detection via knowledge transfer. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3564–3569, 2020.

[11] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[12] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[13] Tsung-Yi Lin, M. Maire, Serge J. Belongie, James Hays, P. Perona, D. Ramanan, Piotr Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[14] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In *Computer Vision – ECCV 2016*, pages 21–37. Springer International Publishing, 2016.

[15] Anay Majee, Kshitij Agrawal, and Anbumani Subramanian. Few-shot learning for road object detection. In *AAAI Workshop on Meta-Learning*, 2021.

[16] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

[17] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy M. Hospedales, and Tao Xiang. Incremental few-shot object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[18] Joseph Redmon, S. Divvala, Ross B. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[19] Shaoqing Ren, Kaiming He, Ross B. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 2015.

[20] K. Shmelkov, C. Schmid, and Alahari Karteek. Incremental learning of object detectors without catastrophic forgetting. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[21] K. Shmelkov, C. Schmid, and Alahari Karteek. Incremental learning of object detectors without catastrophic forgetting. *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[22] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[23] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[24] G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker, and C. Jawahar. Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Los Alamitos, CA, USA, Jan 2019. IEEE Computer Society.

[25] Rodrigo Verschae and Javier Ruiz-del Solar. Object detection: Current and future directions. *Frontiers in Robotics and AI*, 2:29, 2015.

[26] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[27] Xin Wang, Thomas E. Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. July 2020.

[28] Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3), June 2020.

[29] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Meta-learning to detect rare objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[30] Xiongwei Wu, Doyen Sahoo, and Steven Hoi. Meta-rcnn: Meta learning for few-shot object detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, page 1679–1687, New York, NY, USA, 2020. Association for Computing Machinery.

[31] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[32] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points, 2019.