

CFA: Constraint-based Finetuning Approach for Generalized Few-Shot Object Detection

Karim Guirguis^{1†} Ahmed Hendawy^{1,2†} George Eskandar² Mohamed Abdelsamad^{1,2} Matthias Kayser¹ Jürgen Beyerer^{3,4}
Robert Bosch GmbH¹ University of Stuttgart² Karlsruhe Institute of Technology³ Fraunhofer IOSB⁴

karim.guirguis@de.bosch.com

Abstract

Few-shot object detection (FSOD) seeks to detect novel categories with limited data by leveraging prior knowledge from abundant base data. Generalized few-shot object detection (G-FSOD) aims to tackle FSOD without forgetting previously seen base classes and, thus, accounts for a more realistic scenario, where both classes are encountered during test time. While current FSOD methods suffer from catastrophic forgetting, G-FSOD addresses this limitation yet exhibits a performance drop on novel tasks compared to the state-of-the-art FSOD. In this work, we propose a constraint-based finetuning approach (CFA) to alleviate catastrophic forgetting, while achieving competitive results on the novel task without increasing the model capacity. CFA adapts a continual learning method, namely Average Gradient Episodic Memory (A-GEM) to G-FSOD. Specifically, more constraints on the gradient search strategy are imposed from which a new gradient update rule is derived, allowing for better knowledge exchange between base and novel classes. To evaluate our method, we conduct extensive experiments on MS-COCO and PASCAL-VOC datasets. Our method outperforms current FSOD and G-FSOD approaches on the novel task with minor degeneration on the base task. Moreover, CFA is orthogonal to FSOD approaches and operates as a plug-and-play module without increasing the model capacity or inference time.

1. Introduction

Object detection is one fundamental building block for visual perception systems. Given an image, the class label and the spatial location are jointly identified for each object. In recent years, the rise of deep convolutional neural networks (CNN) [1] has enabled a wide variety of data-driven object detectors [2–8], delivering competitive performance. However, the training of object detectors requires a significant amount of labeled images, which are time-consuming

† Authors have equally contributed to this work.

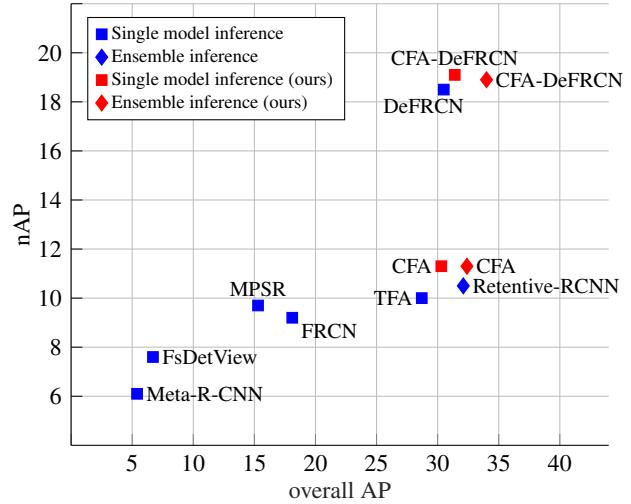


Figure 1. Performance of FSOD methods in G-FSOD task on MS-COCO dataset under 10-shot setting. The proposed finetuning approach, CFA, is applied to various FSOD frameworks, enhancing their performance in the overall and novel AP. Inference with an ensemble of base and novel models result in the highest overall AP but suffer from an increased in computations and inference time. Inference with a single CFA-finetuned model achieves the best trade-off between performance and speed.

and labor-intensive to acquire and to label.

Inspired by the human cognitive ability of learning new concepts with few examples, the few-shot learning (FSL) paradigm has been introduced to tackle the aforementioned issues. This is achieved by leveraging prior knowledge from previous tasks with abundant data to rapidly learn new concepts under a low-data regime. Currently, there are three different ways to integrate this prior knowledge [9]: (1) learning a data augmentation to increase the number of data samples, (2) creating a model that limits the hypothesis space, or (3) developing an algorithm that enables an effective exploration of the hypothesis space.

In 2018, Chen et al. [10] introduced the first few-shot object detection (FSOD) framework. The aim of FSOD is to adapt object detectors to learn novel categories with limited data. Since then, a number of progressively en-

hanced frameworks were introduced, which can be grouped into two main approaches: meta-learning based and transfer learning based methods. Meta-learning based methods [11–17] perform an instance-level exemplar search utilizing a support set of few annotated images. On the other hand, transfer learning based methods [10, 18–20] utilize the previous knowledge from the training on the base classes by finetuning the pre-trained model on the novel classes. Nevertheless, most of the aforementioned approaches have ignored the catastrophic forgetting of the base categories and have focused on improving the detection performance of the novel classes. But in many use cases, it is crucial for perception systems to learn new classes while maintaining their performance on the previously seen ones. For example, in a pick and place task, a robot should be able to learn new objects without forgetting how to handle the old ones.

For this purpose, the task of detecting both base and novel classes, named Generalized Few-Shot Object Detection (G-FSOD), was introduced by [18, 21]. The two-stage finetuning approach [18] (TFA) was among the first to tackle the G-FSOD problem. It jointly finetunes the detector on a balanced set of base and novel classes in a slow learning setting. In this case, only the classification and box regression layers are finetuned while keeping the backbone and the RPN frozen. Although this has resulted in less forgetting, it limits the performance on novel classes. As a remedy, Fan *et al.* [21] has proposed Retentive R-CNN, a transfer-learning-based approach that eliminates the forgetting of the base classes by utilizing an ensemble of base and novel models during inference. Although it succeeds in preventing the performance drop on base classes, the model capacity is increased along with inference time and memory. Furthermore, it restricts the knowledge transfer to the novel task resulting in a lower performance.

Another line of work that shares a similar interest in alleviating catastrophic forgetting is the continual learning (CL) paradigm. The key objective of CL methods is to accumulate knowledge across previous tasks to quickly learn new ones without forgetting. Following [22], the three main CL approaches are categorized as replay-based, regularization-based, and parameter isolation methods. Replay methods [23–35] store or generate samples from previous tasks to be replayed while learning a new task. In contrast, regularization-based methods [36–41] merely introduce a regularization term to the objective function to incorporate knowledge from previous tasks without storing any data. Parameter isolation methods [42–48] assign separate model parameters to each task eliminating forgetting. Nonetheless, CL methods have not yet been exploited in the context of G-FSOD.

Inspired by the CL replay-based methods, this paper proposes a Constrained finetuning Approach (CFA) to allow better transfer of knowledge between base and novel tasks.

CFA extends a replay-based constrained optimization CL method, namely Average Gradient Episodic Memory (A-GEM) [33], to G-FSOD. In A-GEM, catastrophic forgetting is considered to occur when the angle between the loss gradient vectors of previous tasks and the proposed gradient update for the current tasks is obtuse. Hence, A-GEM projects the proposed gradients for the current task orthogonal to the previous task gradients in case of violation. However, starting from the assumption that this may hinder effective knowledge exchange between tasks in G-FSOD, we further constrain the objective function deriving a new gradient update rule. Extensive experiments on MS-COCO and PASCAL-VOC show, that our CFA achieves state-of-the-art performance in the G-FSOD and FSOD settings (as shown in Fig. 1).

2. Related Works

Object Detection. There are two main types of object detectors: two-stage and one-stage detectors. The former [2–4] feature a proposal generation stage. In Faster R-CNN [4], this stage consists of a Region Proposal Network (RPN), which comprises a three-layer CNN that classifies and refines the predicted proposals. Next, the proposals are fed into the classification and localization heads to finally output the detected objects. On the other hand, one-stage detectors [5–8] directly classify and localize the objects.

Few-shot Object Detection. FSOD approaches can be categorized into two groups: transfer learning-based and meta-learning-based methods. Firstly, the transfer-learning-based frameworks [10, 18, 19] learn to transfer knowledge from source categories to novel categories via finetuning. LSTD [10], introduces two regularization modules for background suppression and constrained knowledge transfer for less confusion on base classes. TFA [18] finetunes only the box predictor on a small balanced training set containing both base and novel classes along with a cosine similarity based box classifier. MPSR [19] tackles the high scale variations by generating multi-scale positive samples as object pyramids to refine the prediction at multiple scales. On the other hand, meta-learning-based methods [12–17] learns-to-learn solving a set of unrelated tasks. The aim is to perform an exemplar search in the instance level using only a few annotated images support sets.

Generalized Few-shot Object Detection. G-FSOD [18, 21, 49] is an emerging sub-discipline of FSOD focusing on detecting both the base and novel classes. While TFA [18] attempts to avoid forgetting by finetuning on base and novel classes, ONCE [49] tackles the problem in an incremental setting using a meta-learning approach with a CenterNet [50] detector. The key idea is to meta-learn a class code generator that incrementally learns to synthesize a class code to the novel classes. Retentive R-CNN [21] proposes a transfer-learning-based approach that eliminates

forgetting on base classes. It leverages the base-trained detector head along with the finetuned detector head to detect both classes. Similarly, it integrates both the pre-trained RPN on base classes and another RPN to finetune.

Continual Learning. Recently, continual learning (CL) paradigms have gained more recognition in various computer vision tasks [22]. Firstly, parameter isolation methods [42, 44–48] tackle the task-incremental setting, requiring the task identity during inference. Secondly, regularization-based methods [36–41] consolidate prior knowledge while learning the new task by regularizing the model parameters. Finally, replay methods [23–35] leverages stored samples from previous tasks [23] or generates pseudo-samples using a generative model [24]. The forgetting is mitigated by replaying the samples in a joint training form with the new task samples. Moreover, constrained methods [32–35] alleviate the forgetting of previous tasks by constraining the optimization of the new task loss. Specifically, gradient episodic memory (GEM) [32] and average gradient episodic memory (A-GEM) [33] avoid the joint training between the new and old tasks by guiding the new task’s optimization process. Recently, Riemer *et al.* [34] has proposed an experience replay algorithm that utilizes a meta-learning optimization approach. In [35], a selection scheme for storing necessary samples from previous tasks is proposed.

3. Approach

In this section, the G-FSOD problem is formally introduced. Next, we revisit a continual learning approach, namely A-GEM [33], and show how it can be integrated into the context of G-FSOD. Finally, we present our CFA algorithm which extends A-GEM by further constraining the gradient update on the novel task to foster a mutual knowledge transfer across tasks.

3.1. Problem Formulation

In FSOD and G-FSOD, a dataset is split into a base dataset \mathcal{D}_b containing abundant examples of base classes \mathcal{C}_b , and a novel dataset \mathcal{D}_n comprising a handful examples of novel classes \mathcal{C}_n (i.e., $\mathcal{C}_b \cap \mathcal{C}_n = \emptyset$). Formally, $\mathcal{D}_b = \{ (x, y) \mid y = \{(c_i, b_i)\}, c_i \in \mathcal{C}_b \}$, $\mathcal{D}_n = \{ (x, y) \mid y = \{(c_i, b_i)\}, c_i \in \mathcal{C}_n \}$, where $x \in \mathcal{X}$ is an input image, and $y \in \mathcal{Y}$ is the corresponding annotation. c_i and b_i are the class label and bounding box coordinates of each instance i in the image, respectively. The objective of G-FSOD task is to find a hypothesis $h(\cdot)$ in the hypothesis space that is able to learn the novel classes without forgetting the previously seen base classes.

The few-shot training comprises of two stages: base and novel training. The former trains on \mathcal{C}_b with abundant examples whereas the latter leverages only a handful of examples from \mathcal{C}_n . The novel training scheme varies depending on the task. In FSOD, only a few-shots of \mathcal{C}_n are uti-

lized, while in G-FSOD a balanced set from both categories is used. Although in G-FSOD, the model has the advantage of interacting with the base classes to avoid any performance drop, we argue that the few-shots can not perfectly represent the base data distribution, leading to overfitting on the limited examples. This has been shown in previous works [18, 21, 51], where better performance is achieved on the base task with degradation on the novel task. In this work, we seek to bridge the performance gap between G-FSOD and FSOD. More specifically, we strive to alleviate the catastrophic forgetting without increasing the model capacity or hindering novel classes’ performance.

As previously mentioned in Sec. 2, there are two main approaches of FSOD: finetuning based and meta-learning based. While finetuning approaches can achieve lower performance on \mathcal{C}_n compared to meta-based detectors, they tend to perform better on \mathcal{C}_b . Moreover, they offer more flexibility allowing for easier integration into different detection frameworks. One finetuning method, namely Retentive-R-CNN [21], has eliminated forgetting by reusing a frozen RPN and RoI head from the base task; however, the performance declined on the novel task. To remedy this issue, we present an alternative finetuning approach for G-FSOD.

3.2. Revisiting the Gradient Episodic Memory Algorithm

Inspired by the replay-based continual learning (CL) approaches [32, 33], the proposed CFA alters the search strategy to find optimal parameters that lead to a better generalization across tasks. This approach has a two-fold advantage: it does not need any tailored data augmentations or specific model modifications while being flexible enough to integrate into various detectors independent of their architecture. Specifically, CFA regularizes the gradient update via a subset of the base dataset stored in the episodic memory, analogous to A-GEM [33].

Episodic memory based approaches [32–35] aim to alleviate catastrophic forgetting by maintaining an episodic memory \mathcal{M}_k for each task k . Not only do they prevent the losses of the previous tasks from increasing, but they also allow their decrease resulting in a positive backward transfer. Meaning the performance on previous tasks can be improved while learning new ones. Rather than optimizing for all the samples in the episodic memory as originally proposed [32] instead, A-GEM [33] tries to guarantee that the average episodic memory loss does not increase over a mini-batch of samples from the episodic memory.

Firstly, we show how A-GEM [33] can be exploited in the context of G-FSOD. The employed episodic memory \mathcal{M}_b stores few-shots K of the base classes that are randomly sampled from \mathcal{D}_b , where K is chosen to match the number few-shots in the novel set \mathcal{D}_n . In contrast to the CL

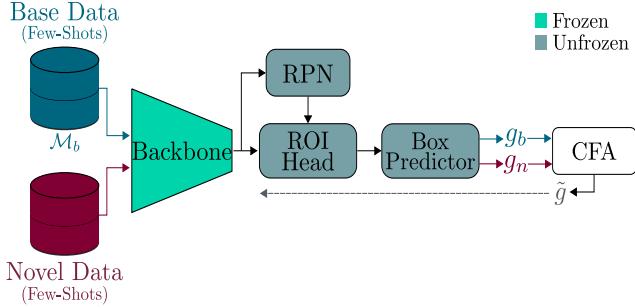


Figure 2. Overview of the fine-tuning stage with CFA on Faster R-CNN [4]. Gradients are computed from each mini-batch. CFA calculates the final gradient update rule which back-propagates on the unfrozen components.

scheme, we train batch-wise where the data is seen more than once and is not handled one by one on-the-fly. Moreover, during novel training, the episodic memory is static meaning that no further samples are added.

The finetuning is then conducted as follows: a mini-batch is sampled from \mathcal{M}_b to compute the base gradient, g_b . Then, another mini-batch is sampled from \mathcal{D}_n and the novel gradient, g_n , is calculated. As defined in [32, 33], a positive knowledge transfer occurs when the angle between g_b and g_n is acute. If this constraint is satisfied, g_n is back-propagated as it is. Otherwise, g_n is projected to a region in the hypothesis space closer to the base task gradients, dictated by g_b , before back-propagating on the model.

We formulate the objective function for the G-FSOD task as follows:

$$\begin{aligned} & \text{minimize}_{\theta} \quad \mathcal{L}(h_{\theta}(x), y) \\ & \text{subject to} \quad \mathcal{L}(h_{\theta}, \mathcal{M}_b) \leq \mathcal{L}(h_{\theta}^b, \mathcal{M}_b), \end{aligned} \quad (1)$$

where $(x, y) \in \mathcal{D}_n$. h_{θ}^b is the pretrained model on \mathcal{D}_b . The loss function on \mathcal{M}_b is given by:

$$\mathcal{L}(h_{\theta}, \mathcal{M}_b) = \frac{1}{|\mathcal{M}_b|} \sum_{(x_i, y_i) \in \mathcal{M}_b} \mathcal{L}(h_{\theta}(x_i), y_i). \quad (2)$$

\mathcal{L} is the overall detection loss function and is defined in [4] as:

$$\mathcal{L} = \mathcal{L}_{rpn} + \mathcal{L}_{cls} + \mathcal{L}_{reg}, \quad (3)$$

where \mathcal{L}_{rpn} is the RPN loss function. \mathcal{L}_{cls} and \mathcal{L}_{reg} are the classification and bounding box regression loss.

Analogous to [32, 33], the optimization problem in Eq. (1) can be reduced to a quadratic programming (QP) problem in the context of A-GEM as follows:

$$\begin{aligned} & \text{minimize}_{\tilde{g}_n} \quad \frac{1}{2} \|g_n - \tilde{g}_n\|_2^2 \\ & \text{subject to} \quad \tilde{g}_n^\top g_b \geq 0, \end{aligned} \quad (4)$$

where \tilde{g}_n is the projected gradient update for the novel task. Formally, the closed-form A-GEM gradient update rule is realized as:

$$\tilde{g}_n = g_n - \frac{g_n^\top g_b}{g_b^\top g_b} \cdot g_b, \quad (5)$$

which implies that only in case of violation of the A-GEM constraint, the novel gradient is projected orthogonal to the base gradient.

3.3. Constraint-based Finetuning Approach for G-FSOD

Compared to finetuning approaches, A-GEM applied in the context G-FSOD, provides a better regularization to the learning process preventing early overfitting. However, we argue that depending merely on the above mentioned constraint might hinder knowledge transfer across tasks, due to the following reasons: (1) the base gradient is only back-propagated in case of violation and thus has little influence during finetuning while learning the novel classes, (2) projecting the novel gradient orthogonally is too restrictive for diverse novel feature learning.

Motivated by the aforementioned observations, we propose to minimize the angle between g_b and g_n instead of always projecting g_n orthogonally in case of violation. The proposed CFA algorithm arises from a joint optimization problem on both tasks, where a further constraint is inflicted to account for the performance on base categories. In CFA, both the novel and base gradients are back-propagated. The scheme is illustrated in Fig. 2. In case of violation, g_n is projected orthogonally to \tilde{g}_n with respect to g_b , while g_b is, in turn, projected orthogonally to \tilde{g}_b with respect to g_n . Formally, our proposed constrained optimization problem is formulated as follows:

$$\begin{aligned} & \text{minimize}_{\tilde{g}_b, \tilde{g}_n} \quad \frac{1}{2} \|g_n - \tilde{g}_n\|_2^2 + \frac{1}{2} \|g_b - \tilde{g}_b\|_2^2 \\ & \text{subject to} \quad \tilde{g}_n^\top g_b \geq 0, \\ & \quad \tilde{g}_b^\top g_n \geq 0, \end{aligned} \quad (6)$$

where \tilde{g}_b and \tilde{g}_n denotes the projected gradient update for the base task and novel task, respectively. Solving the above constrained optimization problem using the method of Lagrange multipliers, the gradient update rules are derived as follows:

$$\tilde{g}_n = g_n - \left(\frac{g_n^\top g_b}{g_b^\top g_b} \right) \cdot g_b, \quad (7)$$

$$\tilde{g}_b = g_b - \left(\frac{g_b^\top g_n}{g_n^\top g_n} \right) \cdot g_n. \quad (8)$$

The formal proof of the aforementioned update rules for CFA is provided in the supplementary material section.

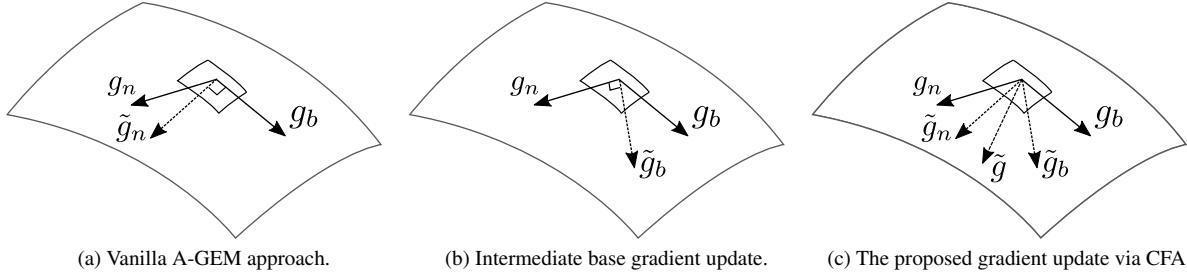


Figure 3. Visualization of the gradient update for vanilla A-GEM and the proposed CFA. In Fig. 3a, we present the novel task gradient update using the A-GEM where the novel task gradient is projected orthogonally to base task gradient. Fig. 3b shows the the solution for the proposed constraint where the base task gradient is projected to a right angle with the novel task gradient. Finally, the final gradient update for the CFA algorithm is shown in Fig. 3c.

Algorithm 1 CFA

```

1: procedure TRAIN( $f_\theta$ ,  $\mathcal{D}_b$ ,  $\mathcal{D}_n$ )
2:    $\mathcal{M}_b \sim \mathcal{D}_b$ 
3:   for  $n_{epoch} = 1, \dots, N_{epoch}$  do
4:     for  $(x_n, y_n)$  in  $\mathcal{D}_n$  do
5:        $(x_b, y_b) \sim \mathcal{M}_b$ 
6:        $g_b \leftarrow \nabla_\theta \mathcal{L}(f_\theta(x_b), y_b)$ 
7:        $g_n \leftarrow \nabla_\theta \mathcal{L}(f_\theta(x_n), y_n)$ 
8:       if  $g_n^\top g_b \geq 0$  then
9:          $\tilde{g} \leftarrow \frac{g_n + g_b}{2}$ 
10:      else
11:         $\tilde{g} \leftarrow \frac{1}{2} \left( 1 - \frac{g_n^\top g_b}{g_b^\top g_b} \right) \cdot g_b + \frac{1}{2} \left( 1 - \frac{g_b^\top g_n}{g_n^\top g_n} \right) \cdot g_n$ 
12:      end if
13:       $\theta \leftarrow \theta - \eta \tilde{g}$ 
14:    end for
15:  end for
16:  return  $f_\theta$ 
17: end procedure

```

Instead of performing two gradient updates, a single update rule can be realized by averaging \tilde{g}_b and \tilde{g}_n :

$$\tilde{g} = \frac{1}{2} \left(1 - \frac{g_n^\top g_b}{g_b^\top g_b} \right) \cdot g_b + \frac{1}{2} \left(1 - \frac{g_b^\top g_n}{g_n^\top g_n} \right) \cdot g_n. \quad (9)$$

Eq. (9) can be interpreted as an adaptive re-weighting of the two gradients g_b and g_n that attempts to balance their contribution during the finetuning stage. The advantage of the CFA algorithm, as shown in Algorithm 1, is two-fold. Firstly, the base gradients always contribute to the finetuning stage. Secondly, the algorithm searches for the optimal direction of back-propagation by re-weighting the contributions of each gradient while keeping the angle between the last gradient update \tilde{g} and the base gradient g_b less than 90°. A visualization of the differences between A-GEM and CFA gradient update rules is shown in Fig. 3.

4. Experiments

To evaluate our approach, we follow the well-established G-FSOD benchmarks [18, 21, 51], where experiments on

PASCAL VOC [52] and MS-COCO [53] datasets are conducted. For a fair comparison, we use the same class and data splits as in previous works [12, 18, 19].

4.1. Experimental setting

PASCAL VOC. The dataset features three different sets, where each comprises 20 categories. Moreover, the classes are split randomly into 15 and 5, base and novel classes, respectively. The data is sampled from both the VOC 2007 and VOC 2012 train/val set for base and novel training. For testing, the VOC 2007 test set is utilized. The results are reported for $K = 1, 2, 3, 5, 10, 30$ shots.

MS-COCO. The dataset consists of 80 classes where we use 60 base categories disjoint with VOC, while the remaining 20 are used as novel classes. During testing, we use the $5k$ images from the validation set, while the rest is utilized for training. We report the results for $K = 5, 10, 30$ shots.

Evaluation setting. We consider the G-FSOD evaluation protocol, where a few-shots of the base classes are utilized during novel training. The main aim of G-FSOD is to assess the overall performance of the base and novel categories, which is the more practical case for computer vision tasks. For evaluation metrics, we report AP, bAP, nAP denoting the overall, base, and novel mean average precision, respectively. For PASCAL VOC, only the AP50, bAP50, and nAP50 are utilized.

Implementation details. We adopt Faster R-CNN [4] as a primary detection framework using a ResNet-101 [54] as a backbone and a Feature Pyramid Network (FPN) [55]. The learning rate is set to 0.02 for base training and 0.001 for novel training. We optimize the model with a stochastic gradient descent (SGD) optimizer using a momentum of 0.9 and a weight decay of 0.0001. The batch size is set to 16 across all experiments using 4 Nvidia GeForce 1080Ti GPUs.

Analogous to TFA [18], we evaluate our method with a fully-connected base classifier (CFA w/fc) and a cosine similarity-based box classifier (CFA w/cos). Moreover, we apply CFA to the state-of-the-art finetuning-based approach

Methods / Shots	w/E	5 shot			10 shot			30 shot		
		AP	bAP	nAP	AP	bAP	nAP	AP	bAP	nAP
FRCN-ft-full [18]	✗	18.0	22.0	6.0	18.1	21.0	9.2	18.6	20.6	12.5
TFA w/ fc [18]	✗	27.5	33.9	8.4	27.9	33.9	10.0	29.7	35.1	13.4
TFA w/ cos [18]	✗	28.1	34.7	8.3	28.7	35.0	10.0	30.3	35.8	13.7
MPSR [19]	✗	-	-	-	15.3	17.1	9.7	17.1	18.1	14.1
DeFRCN [51]	✗	28.7	33.1	15.3	30.6	34.6	18.6	31.6	34.7	22.5
ONCE [49]	✗	13.7	17.9	1.0	13.7	17.9	1.2	-	-	-
Meta R-CNN* [14]	✗	3.6	3.5	3.8	5.4	5.2	6.1	7.8	7.1	9.9
FSRW [12]	✗	-	-	-	-	-	5.6	-	-	9.1
FsDetView* [16]	✗	5.9	5.7	6.6	6.7	6.4	7.6	10.0	9.3	12.0
CFA w/ fc	✗	30.1	37.1	9.0	30.8	37.6	10.5	31.9	37.7	14.7
CFA w/ cos	✗	29.7	36.3	9.8	30.3	36.6	11.3	31.7	37.0	15.6
CFA-DeFRCN	✗	30.1	35.0	15.6	31.4	35.5	19.1	32.0	35.0	23.0
Retentive R-CNN [21]	✓	31.5	39.2	8.3	32.1	39.2	10.5	32.9	39.3	13.8
CFA w/ fc	✓	31.8	39.5	8.8	32.2	39.5	10.4	33.2	39.5	14.3
CFA w/ cos	✓	32.0	39.5	9.6	32.4	39.4	11.3	33.4	39.5	15.1
CFA-DeFRCN	✓	33.0	38.9	15.6	34.0	39.0	18.9	34.9	39.0	22.6

Table 1. G-FSOD experimental results for 5,10,30-shot settings on MS-COCO. We report AP, bAP, nAP for all, base, and novel classes, respectively. w/E denotes whether the ensemble-learning based evaluation protocol of Retentive-RCNN [21] was used. Colored results represent the **best** and **second-best**. '*' represents results reported in [21] and [51]. '-' denotes unreported results in the original work.

Methods / Shots	w/E	All Set 1					All Set 2					All Set 3				
		1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
FRCN-ft-full [18]	✗	55.4	57.1	56.8	60.1	60.9	50.1	53.7	53.6	55.9	55.5	58.5	59.1	58.7	61.8	60.8
TFA w/ fc [18]	✗	69.3	66.9	70.3	73.4	73.2	64.7	66.3	67.7	68.3	68.7	67.8	68.9	70.8	72.3	72.2
TFA w/ cos [18]	✗	69.7	68.2	70.5	73.4	72.8	65.5	65.0	67.7	68.0	68.6	67.9	68.6	71.0	72.5	72.4
MPSR [19]	✗	56.8	60.4	62.8	66.1	69.0	53.1	57.6	62.8	64.2	66.3	55.2	59.8	62.7	66.9	67.7
DeFRCN [51]	✗	73.1	73.2	73.7	75.1	74.4	68.6	69.8	71.0	72.5	71.5	72.5	73.5	72.7	74.1	73.9
Meta R-CNN* [14]	✗	17.5	30.5	36.2	49.3	55.6	19.4	33.2	34.8	44.4	53.9	20.3	31.0	41.2	48.0	55.1
FSRW [12]	✗	53.5	50.2	55.3	56.0	59.5	55.1	54.2	55.2	57.5	58.9	54.2	53.5	54.7	58.6	57.6
FsDetView* [16]	✗	36.4	40.3	40.1	50.0	55.3	36.3	43.7	41.6	45.8	54.1	37.0	39.5	40.7	50.7	54.8
CFA w/ fc	✗	69.5	68.2	69.8	73.5	74.3	66.0	66.9	69.2	70.1	71.1	67.7	69.0	70.9	72.6	73.5
CFA w/ cos	✗	69.1	69.8	71.9	73.6	73.9	64.8	66.5	68.3	69.5	70.5	67.7	69.7	71.9	73.0	73.5
CFA-DeFRCN	✗	73.8	74.6	74.5	76.0	74.4	69.3	71.4	72.0	73.3	72.0	72.9	73.9	73.0	74.1	74.6
Retentive R-CNN [21]	✓	71.3	72.3	72.1	74.0	74.6	66.8	68.4	70.2	70.7	71.5	69.0	70.9	72.3	73.9	74.1
CFA w/ fc	✓	70.3	69.5	71.0	74.4	74.9	67.0	68.0	70.2	70.8	71.5	69.1	70.1	71.6	73.3	74.7
CFA w/ cos	✓	71.4	71.8	73.3	74.9	75.0	66.8	68.4	70.4	71.1	71.9	69.7	71.2	72.6	74.0	74.7
CFA-DeFRCN	✓	75.0	76.0	76.8	77.3	77.3	70.4	72.7	73.7	74.7	74.2	74.7	75.5	75.0	76.2	76.6

Table 2. G-FSOD experimental results for 1,2,3,5,10-shot settings on the three all sets of Pascal VOC (AP50). w/E denotes whether the ensemble-learning based evaluation protocol of Retentive-RCNN [21] was used. Colored results represent the **best** and **second-best**. '*' represents results reported in [21] and [51].

DeFRCN [51], referred to as CFA-DeFRCN, and we follow the original hyperparameters of the paper. Unlike CFA w/fc and CFA w/cos, CFA-DeFRCN does not employ an FPN, analogous to the baseline DeFRCN.

Retentive R-CNN [21] has introduced a model-growth based evaluation protocol where the base RPN (RPN_b) and base detector (DET_b) are leveraged during inference along with the finetuned novel RPN (RPN_n) and the novel detector (DET_n). Specifically, proposals are generated from RPN_b and RPN_n based on maximum objectness score, concurrently. Next, the proposals are fed to DET_b and DET_n

where detections from DET_b (with added 0.1 bonus) are encouraged for C_b via a non-maximum suppression. For fair comparison, we further evaluate our methods (CFA w/fc, CFA w/cos, and CFA-DeFRCN) using the aforementioned evaluation protocol, similar to Retentive-RCNN [21]. Both evaluation protocols are illustrated in the supplementary materials.

4.2. Comparison Experiments

We perform quantitative comparisons against both transfer-learning [18, 19, 21, 51] and meta-learning [12, 14,

Methods / Shots	w/E	Novel Set 1					Novel Set 2					Novel Set 3				
		1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
FRCN-ft-full [18]	✗	15.2	20.3	29.0	25.5	28.7	13.4	20.6	28.6	32.4	38.8	19.6	20.8	28.7	42.2	42.1
TFA w/ fc [18]	✗	36.8	29.1	43.6	55.7	57.0	18.2	29.0	33.4	35.5	39.0	27.7	33.6	42.5	48.7	50.2
TFA w/ cos [18]	✗	39.8	36.1	44.7	55.7	56.0	23.5	26.9	34.1	35.1	39.1	30.8	34.8	42.8	49.5	49.8
MPSR [19]	✗	42.8	43.6	48.4	55.3	61.2	29.8	28.1	41.6	43.2	47.0	35.9	40.0	43.7	48.9	51.3
DeFRCN [51]	✗	57.0	58.6	64.3	67.8	67.0	35.8	42.7	51.0	54.4	52.9	52.5	56.6	55.8	60.7	62.5
Meta R-CNN* [14]	✗	16.8	20.1	20.3	38.2	43.7	7.7	12.0	14.9	21.9	31.1	9.2	13.9	26.2	29.2	36.2
FSRW [12]	✗	14.8	15.5	26.7	33.9	47.2	15.7	15.3	22.7	30.1	39.2	19.2	21.7	25.7	40.6	41.3
MetaDet [13]	✗	18.9	20.6	30.2	36.8	49.6	21.8	23.1	27.8	31.7	43.0	20.6	23.9	29.4	43.9	44.1
FsDetView* [16]	✗	25.4	20.4	37.4	36.1	42.3	22.9	21.7	22.6	25.6	29.2	32.4	19.0	29.8	33.2	39.8
CFA w/ fc	✗	40.0	35.5	40.9	54.1	56.9	22.2	27.1	35.2	38.5	40.9	29.7	35.1	39.5	47.2	51.3
CFA w/ cos	✗	41.2	43.6	49.5	56.5	57.3	21.3	27.4	35.3	39.1	42.1	31.7	39.1	44.6	49.9	52.6
CFA-DeFRCN	✗	58.2	63.3	65.8	68.9	67.1	37.1	45.5	51.3	55.2	53.8	54.7	57.8	56.9	60.0	63.3
Retentive R-CNN [21]	✓	42.4	45.8	45.9	53.7	56.1	21.7	27.8	35.2	37.0	40.3	30.2	37.6	43.0	49.7	50.1
CFA w/ fc	✓	39.0	34.9	41.4	54.8	57.0	21.8	26.1	35.3	37.1	40.1	29.9	34.3	40.1	47.0	52.6
CFA w/ cos	✓	42.4	43.9	50.3	56.6	57.3	21.0	27.5	35.3	38.6	41.4	32.3	38.0	44.5	49.8	52.7
CFA-DeFRCN	✓	59.0	63.5	66.4	68.4	68.3	37.0	45.8	50.0	54.2	52.5	54.8	58.5	56.5	61.3	63.5

Table 3. G-FSOD experimental results for 1,2,3,5,10-shot settings on the three novel sets of Pascal VOC (nAP50). Colored results represent the **best** and **second-best**. '*' represents results reported in [21] and [51]. We achieve state-of-the-art novel performance (nAP50) across the three different splits under different few-shot settings.

[16, 49] based methods under G-FSOD setting. We also compare to Retentive-RCNN using their evaluation protocol [21]. To show the effectiveness of our method, we report our results for CFA w/fc, CFA w/cos, and CFA-DeFRCN.

Results on MS-COCO. We also evaluate our approach on MS-COCO in Tab. 1 across $K = 5, 10, 30$ -shot settings. The standard MS-COCO mean average precision (mAP) metric is used for the base (bAP) and novel (nAP) categories. In the G-FSOD setting, the results show that, independent of the architecture, CFA can help achieve less forgetting on the base categories compared to TFA, while improving on the novel tasks. Tab. 1 shows that CFA-DeFRCN consistently achieves the best overall and novel performance across the three few-shot configurations, compared to both FSOD and G-FSOD methods.

Furthermore, we report the results of our method using the Retentive-RCNN evaluation protocol. In comparison to Retentive-RCNN [21], the CFA-finetuned models achieve better performance on both base and novel classes. CFA w/cos performs slightly better than CFA w/fc. This finding is in line with previous observations in TFA [18] and Retentive-RCNN [21], where cosine similarity classifiers allow better generalization due to their robustness against variations of feature norms between base and novel classes.

Although the ensemble-based evaluation protocol leads to less forgetting on the base classes, it suffers from increased inference time and model capacity (additional experiments are presented in the supplementary material). As shown in Tab. 1, single CFA-finetuned methods can achieve very comparable performance to Retentive-RCNN [21] on base classes and superior results on novel classes. In this ensemble setting, CFA-DeFRCN does not only employ base and novel RPN and detectors but also base and novel backbones. This is due to the fact that the gradient decoupling

Method	Backbone	RPN	RoI Head	AP	bAP	nAP
TFA		✓	✓	27.9	33.9	10.0
				29.9	37.2	7.9
		✓	✓	28.9	35.4	9.6
		✓	✓	28.9	35.1	10.2
CFA		✓	✓	24.1	29.0	9.2
				29.6	36.0	10.4
		✓	✓	30.3	37.4	9.3
		✓	✓	30.8	37.8	9.6
		✓	✓	30.8	37.6	10.5
		✓	✓	23.9	28.6	10.1

Table 4. Effect of unfreezing different components while finetuning with CFA in comparison to TFA [18]. ✓ denotes unfreezing a component. We adopt the FC classifier based detector model [4]. The results are reported for MS-COCO under 10-shots.

layer allows finetuning the backbone with gradients backpropagated from the RoI head.

Results on PASCAL VOC. The overall and novel performance of the G-FSOD models on PASCAL VOC dataset are shown in Tab. 2 and Tab. 3, respectively. The effectiveness of CFA is highlighted across different splits, where single CFA-finetuned models are able to generalize better than the ensemble of base and novel models used in Retentive R-CNN [21]. Additional results on MS-COCO and PASCAL VOC can be found in the supplementary material.

4.3. Ablation Study

To further assess the capability of our proposed CFA algorithm, ablation experiments are conducted on the MS-COCO dataset with a $K = 10$ -shot scenario.

Impact of unfreezing different components. We study the influence of various model components in the G-FSOD setting. The results are presented in Tab. 4. Firstly, we no-

Method	Base-Shots	AP	bAP	nAP
TFA	1-Shots	22.2	26.3	9.8
	2-Shots	24.8	29.8	9.9
	3-Shots	26.1	31.5	10.1
	5-Shots	27.0	32.6	10.2
	10-Shots	27.9	33.9	10.0
CFA	1-Shots	28.8	34.9	10.5
	2-Shots	30.0	36.5	10.5
	3-Shots	30.3	37.0	10.3
	5-Shots	30.5	37.2	10.4
	10-Shots	30.8	37.6	10.5

Table 5. Impact of variable number of base shots on the catastrophic forgetting of base classes. We compare CFA against TFA [18]. The experiments are conducted on MS-COCO dataset given 10-shots of the novel categories. We report the results on the FC classifier based detector model [4].

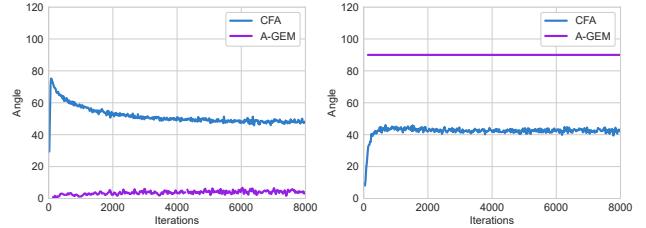
Model	AP	bAP	nAP
A-GEM w/ fc	30.1	36.8	10.1
CFA w/ fc	30.8	37.6	10.5
A-GEM w/ cos	28.2	34.5	9.3
CFA w/ cos	30.3	36.6	11.3
A-GEM-DeFRCN	30.3	35.6	14.4
CFA-DeFRCN	31.4	35.5	19.1

Table 6. Comparison between fine-tuning different models with CFA and A-GEM [33]. The results are reported for MS-COCO under the 10-shot setting.

tice that unfreezing either the RPN or the RoI Head alone leads to suboptimal results. Although the bAP tends to slightly increase compared to the frozen model, there exists a slight drop in the nAP, because the unfrozen component overfits the few novel shots. Secondly, we get the best results in both TFA and CFA by unfreezing the RPN and RoI head. Unfreezing the backbone however leads to deteriorated results. Thirdly, CFA is able to better guide the gradients in the increased search space when unfreezing the RPN and RoI head.

Influence of the number of base shots. In Tab. 5, we show the effect of using unbalanced datasets by finetuning with $K = 1, 2, 3, 5, 10$ base shots and 10 novel shots. Albeit both methods nearly attain their novel task performance, TFA indicates higher sensitivity towards a lower number of base shots. In case of using one base shot, TFA exhibits a significant drop in bAP ($\sim 22.4\%$). On the other hand, CFA is shown to be more robust to fewer base shots, experiencing a ($\sim 7.2\%$) reduction, and thus achieving less forgetting with fewer base shots. Using only 3-shots, CFA reaches a similar performance to the 10-shot scenario with only 0.6 points reduction in bAP. Based on the previous observations, CFA is more memory efficient where it can leverage fewer base shots to assist in achieving less forgetting.

Finetuning with A-GEM. Since CFA extends A-GEM, we compare finetuning with A-GEM against CFA across



(a) Angle between \tilde{g} and g_n (b) Angle between \tilde{g} and g_b

Figure 4. In 10-shot setting for MS-COCO dataset, the angle between the projected gradient \tilde{g} and both g_n (a) and g_b (b) are depicted for A-GEM [33] and CFA.

the three models utilized throughout this work. As shown in Tab. 6, regardless of the model, CFA exhibits less forgetting and helps improve the novel task performance compared to A-GEM. Moreover, not only does CFA consistently attain better overall performance, but it also improves the nAP. This implies that the proposed constraint in CFA results in more forward knowledge transfer because, in each update step, both base and novel gradients adaptively contribute to the gradient update rule, which lead to minimize the expected risk on both tasks.

Impact of CFA and A-GEM on the projected gradient.

In Fig. 4 (a), it is shown that as the network converges the projected gradient is quite close to the novel gradients direction ($\sim 4^\circ$) in case of A-GEM, whereas CFA provides a gradient update at a higher angle ($\sim 43^\circ$). This shows that A-GEM is biased towards the novel tasks compared to the base tasks. In Fig. 4 (b), CFA provides a projected gradient much closer to the base task loss gradients ($\sim 45^\circ$) compared to the consistent orthogonal projection by the A-GEM. This has shown to allow for better transfer of knowledge when learning the novel tasks as well as achieving less forgetting.

5. Conclusion

We propose a new finetuning approach, CFA, for G-FSOD. It readapts a gradient episodic memory approach by replaying a few-shot of the base objects from a static memory buffer. A new gradient update rule is derived, which averages the base and novel gradients when the angle between them is acute. It also adaptively reweights them in case the novel gradients point towards a direction that could lead to forgetting. CFA strives to encourage the knowledge transfer between base and novel classes while being versatile enough to be integrated with FSOD frameworks without incurring overhead in model capacity or inference time. Experiments on MS-COCO and PASCAL-VOC demonstrate the effectiveness of the proposed method in G-FSOD on the overall AP. Moreover, CFA outperforms the state-of-the-art in FSOD on the novel AP. Ablation studies show that CFA is robust enough with a few base shots. In future work, we plan to extend our approach to a continual learning setting.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521:436–44, 2015. 1

[2] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. 1, 2

[3] Ross Girshick. Fast R-CNN. In *IEEE International Conference on Computer Vision*, pages 1440–1448, 2015. 1, 2

[4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015. 1, 2, 4, 5, 7, 8

[5] 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, 2016. 1, 2

[6] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016. 1, 2

[7] Joseph Redmon and Ali Farhadi. YOLO9000: better, faster, stronger. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7263–7271, 2017. 1, 2

[8] Mingxing Tan, Ruoming Pang, and Quoc Le. EfficientDet: Scalable and efficient object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 10778–10787, 2020. 1, 2

[9] Yaqing Wang and Quanming Yao. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys*, 53(3):63:1–63:34, 2020. 1

[10] Hao Chen, Yali Wang, Guoyou Wang, and Yu Qiao. LSTD: A low-shot transfer detector for object detection. In *Conference on Artificial Intelligence*, pages 2836–2843, 2018. 1, 2

[11] Leonid Karlinsky, Joseph Shtok, Sivan Harary, Eli Schwartz, Amit Aides, Rogerio Feris, Raja Giryes, and Alex M. Bronstein. RepMet: Representative-based metric learning for classification and few-shot object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5192–5201, 2019. 2

[12] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *IEEE International Conference on Computer Vision*, pages 8419–8428, 2018. 2, 5, 6, 7

[13] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Meta-learning to detect rare objects. In *IEEE International Conference on Computer Vision*, pages 9924–9933, 2019. 2, 7

[14] Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaduan Liang, and Liang Lin. Meta R-CNN: Towards general solver for instance-level low-shot learning. In *IEEE International Conference on Computer Vision*, pages 9577–9586, 2019. 2, 6, 7

[15] Qi Fan, Wei Zhuo, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4012–4021, 2020. 2

[16] Yang Xiao and Renaud Marlet. Few-shot object detection and viewpoint estimation for objects in the wild. In *European Conference on Computer Vision*, pages 192–210, 2020. 2, 6, 7

[17] Bohao Li, Boyu Yang, Chang Liu, Feng Liu, Rongrong Ji, and Qixiang Ye. Beyond Max-Margin: Class margin equilibrium for few-shot object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7359–7368, 2021. 2

[18] Xin Wang, Thomas E. Huang, Trevor Darrell, Joseph E. Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. In *International Conference on Machine Learning*, pages 9919–9928, 2020. 2, 3, 5, 6, 7, 8

[19] Jiaxi Wu, Songtao Liu, Di Huang, and Yunhong Wang. Multi-scale positive sample refinement for few-shot object detection. In *European Conference on Computer Vision*, pages 456–472, 2020. 2, 5, 6, 7

[20] Bo Sun, Banghuai Li, Shengcai Cai, Ye Yuan, and Chi Zhang. FSCE: few-shot object detection via contrastive proposal encoding. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7352–7362, 2021. 2

[21] Zhibo Fan, Yuchen Ma, Zeming Li, and Jian Sun. Generalized few-shot object detection without forgetting. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4527–4536, 2021. 2, 3, 5, 6, 7

[22] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory G. Slabaugh, and Tinne Tuytelaars. Continual learning: A comparative study on how to defy forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2, 3

[23] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5533–5542, 2017. 2, 3

[24] Nitin Kamra, Umang Gupta, and Yan Liu. Deep generative dual memory network for continual learning. *CoRR*, abs/1710.10368, 2017. 2, 3

[25] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Greg Wayne. Experience replay for continual learning. In *Advances in Neural Information Processing Systems*, pages 348–358, 2019. 2, 3

[26] David Isele and Akansel Cosgun. Selective experience replay for lifelong learning. In *Conference on Artificial Intelligence*, pages 3302–3309, 2018. 2, 3

[27] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet Kumar Dokania, Philip

H. S. Torr, and Marc’Aurelio Ranzato. Continual learning with tiny episodic memories. *CoRR*, abs/1902.10486, 2019. [2](#), [3](#)

[28] Matthias De Lange and Tinne Tuytelaars. Continual prototype evolution: Learning online from non-stationary data streams. In *IEEE International Conference on Computer Vision*, 2021. [2](#), [3](#)

[29] Craig Atkinson, Brendan McCane, Lech Szymanski, and Anthony V. Robins. Pseudo-recursal: Solving the catastrophic forgetting problem in deep neural networks. *CoRR*, abs/1802.03875, 2018. [2](#), [3](#)

[30] Frantzeska Lavda, Jason Ramapuram, Magda Gregorova, and Alexandros Kalousis. Continual classification learning using generative models. In *Advances in Neural Information Processing Systems Workshops*, 2018. [2](#), [3](#)

[31] Jason Ramapuram, Magda Gregorova, and Alexandros Kalousis. Lifelong generative modeling. *Neurocomputing*, 404:381–400, 2020. [2](#), [3](#)

[32] David Lopez-Paz and Marc’Aurelio Ranzato. Gradient episodic memory for continuum learning. In *Advances in Neural Information Processing Systems*, pages 6470–6479, 2017. [2](#), [3](#), [4](#)

[33] Arslan Chaudhry, Marc’Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *International Conference on Learning Representations*, 2019. [2](#), [3](#), [4](#), [8](#)

[34] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *International Conference on Learning Representations*, 2019. [2](#), [3](#)

[35] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. In *Advances in Neural Information Processing Systems*, 2019. [2](#), [3](#)

[36] James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114, 2016. [2](#), [3](#)

[37] Zhizhong Li and Derek Hoiem. Learning without forgetting. In *European Conference on Computer Vision*, pages 614–629, 2016. [2](#), [3](#)

[38] Friedemann Zenke, Ben Poole, and Surya Ganguli. Improved multitask learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995, 2017. [2](#), [3](#)

[39] Heechul Jung, Jeongwoo Ju, Minju Jung, and Junmo Kim. Less-forgetting learning in deep neural networks. *CoRR*, abs/1607.00122, 2016. [2](#), [3](#)

[40] Amal Rannen Triki, Rahaf Aljundi, Matthew B. Blaschko, and Tinne Tuytelaars. Encoder based lifelong learning. In *IEEE International Conference on Computer Vision*, pages 1329–1337, 2017. [2](#), [3](#)

[41] Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasçi, Larry P. Heck, Heming Zhang, and C.-C. Jay Kuo. Class-incremental learning via deep model consolidation. In *IEEE Winter Conference on Applications of Computer Vision*, pages 1120–1129, 2020. [2](#), [3](#)

[42] Arun Mallya and Svetlana Lazebnik. PackNet: Adding multiple tasks to a single network by iterative pruning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7765–7773, 2018. [2](#), [3](#)

[43] Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *CoRR*, abs/1606.04671, 2016. [2](#)

[44] Arun Mallya and Svetlana Lazebnik. Piggyback: Adding multiple tasks to a single, fixed network by learning to mask. In *European Conference on Computer Vision*, pages 72–88, 2018. [2](#), [3](#)

[45] Joan Serrà, Dídac Surís, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *International Conference on Machine Learning*, pages 4555–4564, 2018. [2](#), [3](#)

[46] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. Expert Gate: Lifelong learning with a network of experts. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7120–7129, 2017. [2](#), [3](#)

[47] Ju Xu and Zhanxing Zhu. Reinforced continual learning. In *Advances in Neural Information Processing Systems*, pages 907–916, 2018. [2](#), [3](#)

[48] Amir Rosenfeld and John K. Tsotsos. Incremental learning through deep adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(3):651–663, 2020. [2](#), [3](#)

[49] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy M. Hospedales, and Tao Xiang. Incremental few-shot object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13843–13852, 2020. [2](#), [6](#)

[50] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. CenterNet: Keypoint triplets for object detection. In *IEEE International Conference on Computer Vision*, pages 6568–6577, 2019. [2](#)

[51] Limeng Qiao, Yuxuan Zhao, Zhiyuan Li, Xi Qiu, Jianan Wu, and Chi Zhang. DeFRCN: Decoupled faster R-CNN for few-shot object detection. In *IEEE International Conference on Computer Vision*, 2021. [3](#), [5](#), [6](#), [7](#)

[52] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. [5](#)

[53] 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 *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 5

[54] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[55] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017. 5