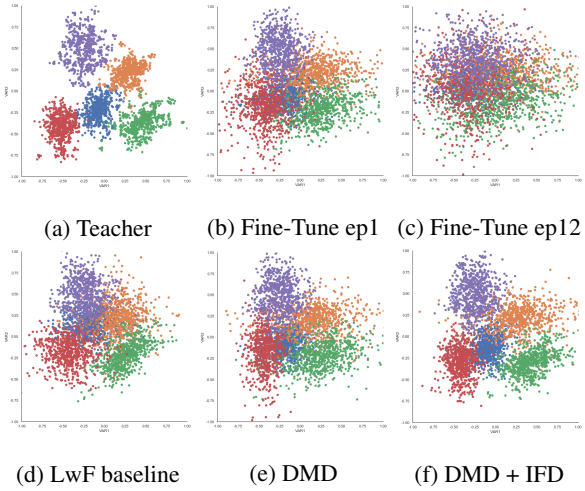
Alleviating Catastrophic Forgetting of Incremental Object Detection via Within-Class and Between-Class Knowledge Distillation

Mengxue Kang[1], Jinpeng Zhang[1]*, Jinming Zhang[2], Xiashuang Wang[3],
Yang Chen[4], Zhe Ma[1]*, Xuhui Huang[1]
[1]Intelligent Science & Technology Academy of CASIC, Beijing 100043, China
[2] Xinjiang University, Xinjiang, 830046, China
[3] The Second Academy of China Aerospace Science and Industry Corporation, Beijing 100854, China
[4] Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
kangmengxue@hotmail.com, zhjphust@126.com, mazhe_thu@163.com

Abstract

Incremental object detection (IOD) task requires a model to learn continually from newly added data. However, directly fine-tuning a well-trained detection model on a new task will sharply decrease the performance on old tasks, which is known as catastrophic forgetting. Knowledge distillation, including feature distillation and response distillation, has been proven to be an effective way to alleviate catastrophic forgetting. However, previous works on feature distillation heavily rely on low-level feature information, while under-exploring the importance of high-level semantic information. In this paper, we discuss the cause of catastrophic forgetting in IOD task as destruction of semantic feature space. We propose a method that dynamically distills both semantic and feature information with consideration of both between-class discriminativeness and within-class consistency on Transformer-based detector. Between-class discriminativeness is preserved by distilling class-level semantic distance and feature distance among various categories, while within-class consistency is preserved by distilling instance-level semantic information and feature information within each category. Extensive experiments are conducted on both Pascal VOC and MS COCO benchmarks. Our method outperforms all the previous CNN-based SOTA methods under various experimental scenarios, with a remarkable mAP improvement from 36.90% to 39.80% under one-step IOD task.



(a) Teacher (b) Fine-Tune ep1 (c) Fine-Tune ep12

(d) LwF baseline (e) DMD (f) DMD + IFD

Figure 1. Visualization of semantic feature space of old categories. (a) represents the semantic feature space of old categories before adding new categories. (b)-(f) represent the semantic feature space of old categories after adding new categories. (b) is at epoch 1 using fine-tuning method. (c) is at epoch 12 using fine-tuning method. **The top three figures illustrate the cause of catastrophic forgetting as destruction of within-class consistency and between-class discrimnativeness.** (d) using LwF baseline method. (e) using our DMD method. (f) using DMD method and IFD method at the same time. **The bottom three figures prove that our method can alleviate catastrophic forgetting via maintaining the within-class consistency and between-class discrimnativeness from teacher to student.**

1. Introduction

In real-world scenarios, learning often occurs incrementally from streaming data. However, traditional object detection models lack this capability. They usually make implicit assumptions about a fixed or stationary data distribution [9]. Directly fine-tuning a model based on newly added data may result in a sharp decrease of its performance on the old data, which is well-known as catastrophic forget-

*Corresponding author

ting. Catastrophic forgetting is the key problem for incremental learning/continual learning task [50]. Depending on whether the task identity is explicitly given or must be inferred, incremental learning (IL) is divided into three types: task/domain/class IL [29]. In this paper, we focus on the most complicated scenario: class incremental learning.

Incremental image classification task has been studied thoroughly [20, 32, 34, 37, 43], but only a few researches focus on incremental object detection (IOD) task [29, 9]. Unlike incremental classification task where there is only one category of objects within each image, IOD task may contain various categories of objects within each image. When adding new categories, only annotations for the new objects are provided and all the old objects are categorized as backgrounds. The "background" labels interfere with the memory of the previously-learnt old labels, thus resulting in catastrophic forgetting of old categories during the incremental process. Here we do a visualization of semantic feature space before and after adding new categories to illustrate the idea above. As shown in Fig.1(a)-(c), adding new categories results in a much messier and severely distorted semantic feature space of the old categories. We thus innovatively propose a method of maintaining the semantic feature space in order to alleviate catastrophic forgetting of the old categories in IOD task.

Previous researches mostly utilize knowledge distillation [14] to reduce catastrophic forgetting under IOD task [4]. There are three kinds of knowledge distillation, including feature-based distillation, response-based distillation, and relation-based distillation. Most works [31, 30, 45] design feature distillation by manually selecting specific layers to mimic the low-level features of old categories. For example, fine-grained feature distillation method [41] and multi-view correlation distillation method [45] selectively utilized intermediate layers to preserve the pattern of old classes. However, this kind of methods heavily relies on low-level feature selection, while under-exploring the importance of high-level semantic information.

In this paper, we focus on how to take advantage of the high-level semantic feature space to improve the knowledge distillation methods. Generally, the representation of instance includes class-specific semantic knowledge, consisting of both within-class knowledge and between-class knowledge. Within-class knowledge represents the consistency of feature expressions in a certain category, while between-class knowledge represents the distinction of feature expressions among various categories. Previous work [15] shows the potential of using within-class and between-class knowledge in incremental classification task. During incremental object detection, all old classes are categorized as the same background, thus affecting their original distinct feature distributions, and destroying both within-class consistency and between-class discriminativeness. However, previous works in feature distillation have not explicitly discussed the cause of catastrophic forgetting via these two components. **Therefore, maintaining both semantic differences among various categories and semantic consistency within each category should be fully considered, in order to mitigate the issue of catastrophic forgetting in incremental object detection task**.

To tackle the problem of within-class consistency, **IFD (Interactive Feature Distillation) method** is proposed to force information of the same category to remain close-by via mimicking information within the same category from teacher to student. The information includes instance-wise interaction between high-level semantics and low-level features. Moreover, to tackle the problem of between-class discrimnativeness, **DMD (Distance Matrix Distillation) method** is proposed to keep the class-wise between-class semantic difference and feature difference of student the same as that of teacher. Between-class semantic difference and feature difference are here represented as between-class semantic distance and feature distance between each two classes. Here we distill between-class semantic distance pattern and feature distance pattern from teacher to student, so as to keep between-class dissimilarity.

The main contributions of this work can be summarized below. **(i)** To the best of our knowledge, we are the first to discuss catastrophic forgetting in IOD task as the destruction of within-class consistency and between-class discrimnativeness. **(ii)** We propose a novel instance-wise feature distillation method based on the interaction between high-level semantics and low-level features to keep the within-class consistency. **(iii)** We propose a novel class-wise distance distillation method based on distance matrix of high-level semantics and low-level features to keep the between-class discriminativeness.

2. Related Work

2.1. Incremental Object Detection

The goal of incremental object detection (IOD) is to learn a sequence of tasks and have the ability to localize and identify all the involved classes during the test phase. It is less explored and more complicated than incremental classification [29, 32]. In recent years, parameter isolation [22], sample replay [28], and knowledge distillation [31] were used in incremental object detection task. Besides, [17] proposed a meta-learning scheme that shares optimal information across incremental tasks. [27] proposed a weight consolidation scheme by applying EWC [19] to Faster RCNN. [35] and [6] focused on incremental few-shot scenarios. [40] presented a new online incremental object detection dataset, and [44] used prototypical task correlation guided gating mechanism to solve it. [21] designed an incremental object detection system with RetinaNet detec-

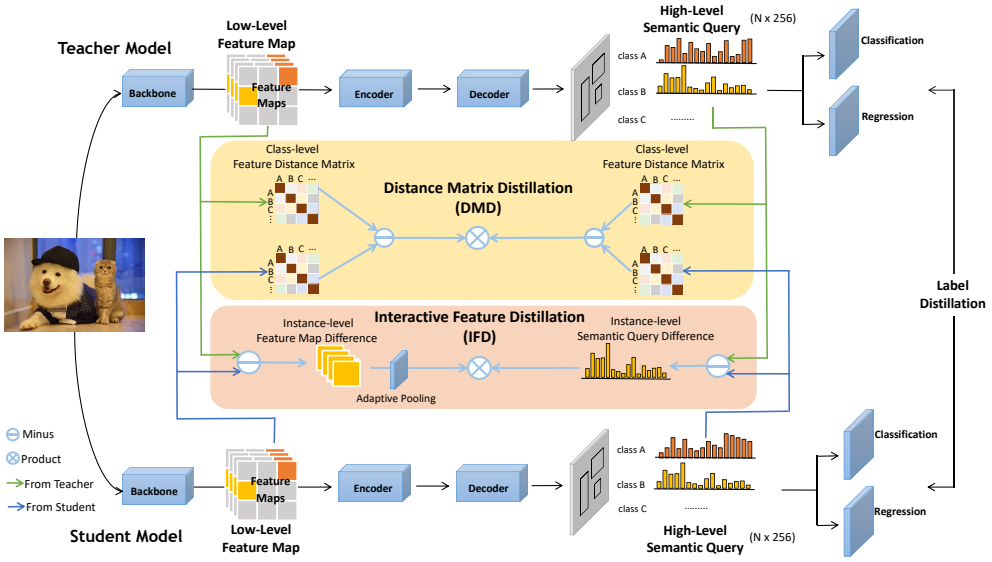


Figure 2. Overall framework of within-class and between-class knowledge distillation for incremental object detection. Between-class DMD method focuses on class-wise interaction between high-level semantics and low-level features among various categories, and within-class IFD method focuses on instance-wise interaction between high-level semantics and low-level features within each single category.

tor on edge devices. [16] introduced the concept of Open World Object Detection (OWOD) task, which combines incremental learning with open set learning. [47] followed along this path to further detect unknown objects based on feature space. [11] extended to use DETR for OWOD task. However, these methods do not devote sufficient attention to the collaborative utilization of within-class knowledge and between-class knowledge.

2.2. Knowledge Distillation for Incremental Object Detection

Knowledge distillation is an effective way to transfer information of old knowledge from teacher to student, thus alleviating catastrophic forgetting. It is widely used in incremental classification tasks [32, 50], and is now frequently involved in incremental object detection tasks. LwF [24] is recognized as a solid baseline for this problem, which firstly applied knowledge distillation under incremental object detection task. SID [31] discussed the appropriate distillation locations by considering outputs, intermediate layers, and the relationship among different instances. MVCD [45] proposed the preservation of channel-wise, point-wise, and instance-wise correlations between feature maps of teacher and student. ILOD [38], Faster ILOD [30], RILOD [21] and RD-IOD [46] distilled different types of knowledge to mimic teacher's behavior on object classification, bounding box regression, and feature extraction. The state-of-the-art method, ERD [9], was proposed as a fully response-based distillation method focusing on classification predictions and bounding boxes. All these works emphasize the importance of knowledge selection and discuss what should be transferred from teacher to student.

However, the feature distillation methods employed in incremental object detection heavily rely on the selection of low-level features, often neglecting the significance of high-level semantic information. When compared with low-level features, high-level features positioned near the classifier encompass considerably more abstract semantic information and offer a more robust representation. Semantic features provide conclusive information about the image, and thus, should be harnessed to more effectively guide the selection of crucial information. Consequently, our emphasis is on combining both low-level and high-level information, along with incorporating between-class and within-class information into a cohesive framework aimed at enhancing the incremental object detection task.

3. Methods

3.1. Overall Framework

The overall framework of our method is shown in Fig.2. The teacher model and the student model are Transformer-based detectors (e.g Deformable DETR [52], AdaMixer [10]). The "Distance Matrix Distillation (DMD)" module is used to preserve the between-class discrimnativeness, while the "Interactive Feature Distillation (IFD)" module is used to preserve the within-class consistency.

The overall loss function of the student is defined as,

$$L_{total} = L_{new} + L_{label} + \alpha L_{DMD} + \beta L_{IFD} \quad (1)$$

where L_{new} is classification loss and localization loss to train student detector for new classes, following the original Deformable DETR [52]. L_{label} is the response distillation loss to train student detector for old classes, following the baseline method LwF [24]. L_{DMD} is between-class distance matrix distillation loss, and L_{IFD} is within-class interactive feature distillation loss. The last three losses help transfer old knowledge from teacher to student. α and β are the hyper-parameters to balance the loss.

Low-level feature maps and high-level semantic queries are used in the calculation of L_{DMD} and L_{IFD}. As illustrated in Fig.2, low-level feature maps refer to the pyramid features generated by Feature Pyramid Network (FPN), containing more concrete details. High-level semantic queries refer to the queries generated by Transformer decoder, containing more abstract semantics.

3.2. Distance Matrix Distillation (DMD)

In order to inherit the between-class discriminativeness, the student has to learn both semantic difference and feature difference from the teacher. They are further represented as high-level semantic distance and low-level feature distance among different classes, which can be calculated using high-level semantic queries and low-level feature maps, respectively.

Concretely, we first calculate the average semantic query Q_{class_i} and average feature map F_{class_i} for each class i as:

$$Q_{class_i} = \frac{1}{N_i}\sum_{n=1}^{N_i} Q_{i,n} \tag{2}$$

$$F_{class_i} = \frac{1}{N_i}\sum_{n=1}^{N_i} AdaptivePooling(F_{i,n}) \tag{3}$$

where N_i refers to the total number of semantic queries for class i. $Q_{i,n}$ refers to the n^{th} semantic query for class i. $F_{i,n}$ refers to the n^{th} feature map for class i with any height and width. $AdaptivePooling$ helps resize the feature map to a standard size of 32×32. We then calculate the Euclidean distance of average semantic queries by Eq.4 and the Euclidean distance of average feature maps by Eq.5 for every two classes. The results are called semantic distance matrix (denoted as $SMat$) and feature distance matrix (denoted as $FMat$), respectively:

$$SMat = \|Q_{class_i} - Q_{class_j}\|_2, \quad 1 \le i \le C, 1 \le j \le C \tag{4}$$

$$FMat = \|F_{class_i} - F_{class_j}\|_2, \quad 1 \le i \le C, 1 \le j \le C \tag{5}$$

where C refers to the number of old classes. i and j refer to two different classes. Finally, we distill the semantic distance matrix $SMat$ and feature distance matrix $FMat$ of all the old classes from teacher to student by Eq.6.

$$L_{DMD} = \frac{1}{C}\|(SMat^S - SMat^T)(FMat^S - FMat^T)\|_2 \tag{6}$$

where $SMat^S$ and $SMat^T$ are the semantic distance matrix calculated from the student and the teacher. $FMat^S$ and $FMat^T$ are the feature distance matrix calculated from the student and the teacher.

3.3. Interactive Feature Distillation(IFD)

In order to keep the within-class consistency, the student has to ensure information similarity with the teacher for each class. Here the information is represented as high-level semantic information and low-level feature information, which are further represented by high-level semantic queries and low-level feature maps. Based on this understanding, IFD method is proposed to realize within-class information transfer.

Concretely, IFD method includes three steps: For each instance, we first calculate the difference of high-level semantic queries between teacher and student by Eq.7. Then, we calculate the corresponding difference of low-level feature maps by Eq.8. After that, we minimize the interaction between high-level semantic differences and low-level feature differences by Eq.9.

$$Q_{diff_i} = \frac{1}{N_i}\sum_{n=1}^{N_i}(Q_{i,n}^S - Q_{i,n}^T) \tag{7}$$

$$F_{diff_i} = \frac{1}{N_i}\sum_{n=1}^{N_i} AdaptivePooling(F_{i,n}^S - F_{i,n}^T) \tag{8}$$

$$L_{IFD} = \sum_{i=1}^{C} Q_{diff_i} \times F_{diff_i} \tag{9}$$

where N_i refers to the total number of instances predicted by teacher for class i on an image. $Q_{i,n}^S$ and $Q_{i,n}^T$ refer to the semantic query of student and teacher for the n^{th} instance under class i. Q_{diff_i} refers to the semantic difference between student and teacher for class i. Similarly, $F_{i,n}^S$ and $F_{i,n}^T$ refer to the feature map of student and teacher for the n^{th} instance under class i. F_{diff_i} represents the feature difference between student and teacher for class i. $AdaptivePooling$ helps resize the feature map to a standard size of 32×32. C refers to the number of old classes.

Both DMD and IFD methods depend on high-level semantic queries and low-level feature maps, but they tend towards two different aspects of knowledge distillation, between-class knowledge distillation and within-class knowledge distillation. Moreover, DMD adopts class-wise operation and IFD adopts instance-wise operation. Their collaboration achieves better dynamic knowledge distillation capabilities.

4. Experiments and Discussions

We conduct experiments under various scenarios on MS COCO 2017[2] and Pascal VOC [8] to demonstrate the effectiveness of our method. We also perform ablation studies to prove each component.

Datasets and Evaluation Metric. MS COCO and Pascal VOC are two object detection datasets with 80 and 20 categories, respectively. Each dataset is split into old classes and new classes following alphabetic order. The models are trained with 12 epochs (1x schedule) for each incremental step. The evaluation metrics include: (1) standard COCO protocols: $\mathbf{mAP}$, mAP_{50}, mAP_{75}, mAP_S, mAP_M and mAP_L; standard VOC protocols: mAP_{50}. (2) **AbsGap** and **RelGap** [27, 29] represent the absolute gap and relative gap, respectively, between final mAP of incremental learning and mAP of full-training (a.k.a. upper-bound). These gaps are only meaningful when both methods are compared with the same base framework and training protocol [13]. (3) **Omega ($\mathbf{\Omega}$)** [13], defined in Eq.10, represents the cumulative capability of multi-step incremental learning step by step.

$$\Omega = \frac{1}{T}\sum_{t=1}^{T}\frac{mAP_{final,t}}{mAP_{upper,t}} \tag{10}$$

where T is the number of total tasks and t refers to the t^{th} task in incremental learning. $mAP_{final,t}$ and $mAP_{upper,t}$ refer to the task-average mAP and upper-bound mAP on all testing data containing learned categories after task t. A higher value of the Ω metric corresponds to a better capability of reducing cumulative knowledge forgetting Ω metrics here can be easily extended under different IoU and area thresholds to achieve a more complete performance evaluation, including Ω_{50}, Ω_{75}, Ω_S, Ω_M and Ω_L.

Experiment Setup. In order to thoroughly study our methods, we implement our models under different experimental scenarios (denoted as A+B, where A is the normal training at the first step, and +B is the incremental training afterward). For example, 2-step COCO scenario of 40+20+20 means 20 new classes are added to the previously learned classes at each step. For simplicity, it can be denoted as COCO(40+20+20). Experimental scenarios include: **(i) One-step**: 40+40, 50+30, 60+20, 70+10, last 40 + first 40 for MS COCO; 10+10, 15+5, 19+1 for Pascal VOC. **(ii) Multi-step**: 40+20+20, 40+10+10+10+10 for MS COCO; 15+1+1+1+1+1, 5+5+5+5 for Pascal VOC.

Implementation details. We build our model based on Deformable DETR detector. Teacher and student detectors defined in our experiments are standard Deformable DETR architecture. All experiments are performed on 4 NVIDIA Tesla V100 GPU with a batch size of 8 images per GPU. The image size is 640×640. We use AdamW as the optimizer for all the incremental steps. The learning rate is set as 2e-4, and divided by 10 at 8th and 11th epoch. The weight decay is 0.0001.

5. Overall Performance

One-step. We report the incremental performance of COCO under 40+40, 50+30, 60+20 and 70+10 scenario in Table 1. Under 40 classes + 40 classes scenario, our method has a much larger final mAP of 39.10 and a smaller gap of 1.10 toward the upper bound, compared with LwF [24], RILOD [21], SID [31] and ERD [9]. Since the state-of-the-art methods have different upper bounds, base frameworks, and training protocols, we also use Ω metrics to make them comparable. All the Ω metrics of our method are larger than those of the current SOTA ERD method by [9], demonstrating the effectiveness of our model under different IoU and area thresholds. Similarly, for all the other incremental conditions (50+30, 60+20, and 70+10), our method also keeps the best performance over other typical incremental object detection approaches. We provide the result under last 40 classes + first 40 classes scenario as well. The performance is improved with a larger Ω_{all} of 0.995, indicating our method can alleviate catastrophic forgetting with no influence of the category orders.

We also report incremental performance of VOC under scenarios of 10+10, 15+5, and 19+1 in Appendix Table 10. It shows that our method outperforms all other methods on VOC dataset. Fig.1(c)(f) illustrates the changes of semantic feature space of old categories. By adding our distillation method (DMD+IFD), within-class knowledge becomes more consistent while between-class knowledge becomes more distinct, proving the effectiveness of our method.

Multi-step. We report incremental learning results under multi-step settings, so as to reveal its ability of long-term incremental learning. Table 2 shows the results under 3-step VOC scenario. Table 4 shows the results under 2-step COCO scenario. Appendix Table 9 and Appendix Table 11 show the results under 4-step COCO scenario and 5-step VOC scenario, respectively. Under all these scenarios, our incremental results all realize a smaller gap to the corresponding full-training results and show a larger Ω value, which demonstrates its excellent capability of alleviating catastrophic forgetting even over multiple steps. Meanwhile, our method does not increase network parame-

Table 1. Incremental results (%) on COCO benchmark under different scenarios. AbsGap and RelGap represents the absolute gap and the relative gap toward upper bound.

Scenarios	Method	AbsGap	RelGap↓	$\Omega_{all}\uparrow$	$\Omega_{50}\uparrow$	$\Omega_{75}\uparrow$	$\Omega_{S}\uparrow$	$\Omega_{M}\uparrow$	$\Omega_{L}\uparrow$	Final mAP	Upper Bound
40 classes + 40 classes	LwF[24]	23.00	57.21%	0.714	0.718	0.713	0.341	0.417	0.466	17.20	40.20
	RILOD[21]	10.30	25.62%	0.872	0.886	0.867	0.681	0.748	0.776	29.90	40.20
	SID[31]	6.20	15.42%	0.923	0.941	0.916	0.793	0.871	0.860	34.00	40.20
	ERD[9]	3.30	8.21%	0.959	0.967	0.954	0.918	0.916	0.910	36.90	40.20
	Ours	**0.50**	**1.24%**	**0.994**	**0.992**	**0.998**	**0.980**	**0.986**	**0.993**	**39.80**	40.30
50 classes + 30 classes	LwF[24]	35.20	87.56%	0.562	0.581	0.553	0.216	0.152	0.109	5.00	40.20
	RILOD[21]	11.70	29.10%	0.854	0.870	0.846	0.664	0.717	0.728	28.50	40.20
	SID[31]	6.40	15.92%	0.920	0.937	0.914	0.759	0.864	0.864	33.80	40.20
	ERD[9]	3.60	8.96%	0.955	0.963	0.946	0.836	0.916	0.920	36.60	40.20
	Ours	**1.50**	**3.72%**	**0.981**	**0.983**	**0.982**	**0.975**	**0.963**	**0.962**	**38.80**	40.30
60 classes + 20 classes	LwF[24]	34.40	85.57%	0.572	0.593	0.561	0.172	0.193	0.148	5.80	40.20
	RILOD[21]	14.80	36.82%	0.816	0.833	0.807	0.599	0.658	0.646	25.40	40.20
	SID[31]	7.50	18.66%	0.907	0.927	0.897	0.741	0.853	0.833	32.70	40.20
	ERD[9]	4.40	10.95%	0.945	0.954	0.940	0.888	0.893	0.891	35.80	40.20
	Ours	**2.00**	**4.96%**	**0.975**	**0.978**	**0.976**	**0.946**	**0.949**	**0.943**	**38.30**	40.30
70 classes + 10 classes	LwF[24]	33.10	82.34%	0.588	0.606	0.580	0.207	0.215	0.192	7.10	40.20
	RILOD[21]	15.70	39.05%	0.805	0.825	0.795	0.612	0.621	0.642	24.50	40.20
	SID[31]	7.40	18.41%	0.908	0.920	0.901	0.737	0.837	0.852	32.80	40.20
	ERD[9]	5.30	13.18%	0.934	0.945	0.929	0.806	0.880	0.872	34.90	40.20
	Ours	**2.70**	**6.70%**	**0.967**	**0.972**	**0.968**	**0.961**	**0.947**	**0.916**	**37.60**	40.30
Last 40 classes + First 40 classes	LwF[24]	19.70	49.00%	0.755	0.756	0.753	0.780	0.755	0.742	20.50	40.20
	RILOD[21]	6.10	15.17%	0.924	0.938	0.922	0.912	0.931	0.920	34.10	40.20
	SID[31]	6.70	16.67%	0.917	0.937	0.916	0.909	0.927	0.912	33.50	40.20
	ERD[9]	2.70	6.72%	0.966	0.973	0.963	0.959	0.966	0.962	37.50	40.20
	Ours	**0.40**	**0.99%**	**0.995**	**0.994**	**0.997**	**0.993**	**0.999**	**0.990**	**39.90**	40.30

Table 2. Incremental results (%) on VOC benchmark under 5+5+5+5 three-step setting, when five classes are added sequentially.

Method	mAP				Final mAP	AbsGaP↓	RelGaP↓	Ω ↑	Upper Bound
	A (1-5)	+B(6-10)	+B(11-15)	+B(16-20)					
CF	1.25	2.34	3.12	36.32	11.31	43.94	61.37%	0.6362	70.64
RILOD [31]	22.11	34.70	37.24	29.80	30.97	39.63	56.13%	0.6876	70.60
SID [31]	27.26	40.10	43.02	34.44	36.21	35.40	49.43%	0.7360	71.60
ILOD [38]	29.55	43.47	46.65	37.34	39.25	30.55	43.76%	0.7548	69.80
CIFRCN [12]	34.60	44.10	55.60	59.60	48.48	22.04	31.25%	0.7972	70.51
ERD [9]	41.25	57.38	63.57	53.12	53.83	16.77	23.55%	0.9021	70.60
Ours	**46.14**	**60.50**	**69.53**	50.54	**58.72**	**12.07**	**17.05%**	**0.9218**	70.79

ters and extra FLOPs for detection inference.

Qualitative Analysis. Fig.3 shows the detection results of Teacher, LwF and "DMD+IFD" on an image from COCO validation set, in which the two old categories (handbag v.s. backpack) have similar appearance and semantics. Fig.3(a) shows the best confidence differentiation obtained by Teacher, while Fig.3(b) and Fig.3(c) respectively shows the worst and better confidence differentiation obtained by LwF and "DMD+IFD". This reflects that our method effectively preserve the teacher knowledge about between-class discriminativeness and within-class similarity, thus relieving catastrophic forgetting.

(a) Teacher (b) LwF (c) DMD+IFD

Figure 3. Incremental object detection results for COCO(70+10).

Table 3. Ablation study (%) using the COCO benchmark under first 40 classes + last 40 classes.

Method		AbsGap	RelGap↓	$\Omega_{all}\uparrow$	$\Omega_{50}\uparrow$	$\Omega_{75}\uparrow$	$\Omega_{S}\uparrow$	$\Omega_{M}\uparrow$	$\Omega_{L}\uparrow$	Final mAP	Upper Bound
Baseline		1.41	3.50%	0.982	0.982	0.984	0.951	0.972	0.969	38.89	40.30
DMD	Manhattan Distance	1.16	2.87%	0.986	0.985	0.985	0.956	0.972	0.971	39.14	40.30
	Cosine Distance	0.87	2.16%	0.989	0.988	0.991	0.961	0.982	0.983	39.43	40.30
	Euclidean Distance	0.72	1.80%	0.991	0.990	0.992	0.966	0.982	0.985	39.58	40.30
IFD	Feats	1.37	3.40%	0.983	0.985	0.984	0.926	0.970	0.978	38.93	40.30
	ForeFeats	0.94	2.33%	0.988	0.988	0.991	0.949	0.980	0.991	39.36	40.30
	SemanticForeFeats	0.62	1.55%	0.992	0.992	0.995	0.971	0.984	0.991	39.68	40.30
SemanticForeFeats + Euclidean Distance		**0.50**	**1.24%**	**0.994**	**0.992**	**0.998**	**0.980**	**0.986**	**0.993**	**39.80**	40.30

Table 4. Incremental results (%) on COCO under the 40+20+20 two-step setting. A(a-b) is the one-step normal training for classes a-b and +B(c-d) is the incremental training for classes c-d.

	A(1-40)				
	+B(40-60)				
	mAP	AbsGap	RelGap↓	$\Omega_{all}\uparrow$	Upper Bound
CF	10.70	29.10	73.38%	0.634	39.80
RILOD[21]	27.80	12.00	30.85%	0.849	39.80
SID[31]	34.00	5.80	15.42%	0.927	39.80
ERD[9]	36.70	3.10	7.79%	0.961	39.80
Ours	39.30	**0.60**	**1.53%**	**0.992**	39.90
	+B(60-80)				
	mAP	AbsGap	RelGap↓	$\Omega_{all}\uparrow$	Upper Bound
CF	9.40	30.80	76.62%	0.501	40.20
RILOD[21]	15.80	24.40	60.70%	0.697	40.20
SID[31]	23.80	16.40	40.80%	0.815	40.20
ERD[9]	32.40	7.80	19.40%	0.909	40.20
Ours	36.60	**3.70**	**10.11%**	**0.964**	40.30

6. Ablation Study

To test each component, we implement our methods under COCO first 40 + last 40 scenario in Table 3. "Baseline" denotes LwF[24] method, which distills predicted labels from teacher to student, with no feature distillation or relation distillation. Our model is built based on this baseline model (with distillation on predicted labels) plus two modules (DMD and IFD). For IFD module, "Feats" denotes distilling the entire feature map from teacher to student. After that, since foreground objects have meaningful semantic information, we select all the foreground objects out of background for further discussion. "ForeFeats" represents distillation on low-level feature map for foreground objects only. "SemanticForeFeats" denotes distillation on the interaction between low-level feature map and high-level semantic information for foreground objects only. For DMD module, we respectively use "Manhattan Distance", "Cosine Distance" (cosine similarity) and "Euclidean Distance" as the measurement of between-class distance.

(a) Original (b) w/o Distill (c) w/ Distill

Figure 4. Effect of within-class IFD method. (a) is the orginal picture. (b) is the activation map without IFD method. (c) is the activation map with IFD method.

Table 3 demonstrates that our model reaches its best performance when combining the two modules, with a final mAP of 39.80% and a 0.50% gap to the upper bound. All Ω values are higher than the baseline model and than all the other state-of-the-art models, supporting the effectiveness of our method.

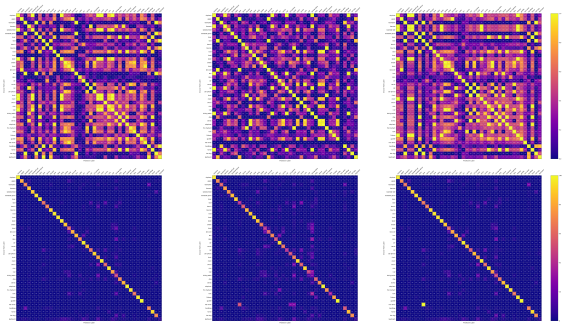
6.1. Interaction Feature Distillation (IFD) module

Table 3 shows that adding IFD increases the final mAP from 38.89% to 39.68% and reduces the gap toward full training from 1.41% to 0.62%. All Ω values are dramatically improved. Moreover, we illustrate the effectiveness of this module in Fig.1 and Fig.4. Fig.1(e)(f) shows that IFD helps concentrating information from the same category so as to keep within-in class consistency. Fig.4 shows adding high-level semantic information to low-level feature map results in more precise attention to important regions. Both the quantitative and qualitative results demonstrate the usefulness of within-class feature distillation module.

Comparison with traditional feature distillation. We also discuss different feature distillation methods in Table 3. Adding a full feature map distillation has almost the same performance as the baseline. But selecting the meaningful foreground feature map improves the final mAP from 38.89% to 39.36%. After that, adding semantic information to foreground feature map shows a better performance of

39.68% final mAP and 0.62% as the gap to the upper bound. Our IFD method is thus demonstrated to be an effective way compared with traditional feature distillation methods.

6.2. Distance Matrix Distillation (DMD) module



(a) Teacher (b) w/o Distill (c) w/ Distill

Figure 5. Distance matrix (upper) and confusion matrix (bottom) of first 40 classes. (a) represents the teacher model. (b) represents student without between-class DMD method. (c) represents student with DMD method.

Table 3 indicates that adding DMD is also meaningful with an improvement of final mAP from 38.89% to 39.58% and a gap descending from 1.41% to 0.72%. All Ω values increase significantly, showing better performance with the help of this module. Further analyses are shown in Fig.1 and Fig.5. Fig.1(d)(e) shows that DMD helps discriminate information from different categories, in order to keep between-class discrimnativeness. Fig.5 (upper) shows the distance pattern among all the old classes. Within each cell of the distance matrix, the number represents $1 - distance$. A larger number means that the two classes are closer to each other. Notably, the distance pattern generated by student with DMD (Fig.5c upper) tends to be more consistent with teacher than that generated by student without this module (Fig.5b upper). In addition, confusion matrix of student with DMD (Fig.5c bottom) shows less misclassification among old classes, and thus has a clearer and more distinguishable relationship. These results indicate the success of using DMD to learn between-class distinctiveness.

Comparing three types of distance functions. In Table 3, we compared three types of distance presented as Euclidean distance, Manhattan distance, and cosine similarity between each two categories. It is obvious that Manhattan distance only has a negligible improvement, while cosine similarity and Euclidean distance are better ways to describe the between-class distance. Specifically, Euclidean distance-based distance distillation improves the final mAP from 38.89% to 39.58%, indicating it is the most effective distance distillation method of the three.

6.3. More Ablation Studies

We provide more ablation studies under COCO (70+10) scenario in Table 5. **(1)** We address the contribution of high-level semantic information and low-level feature information separately. *FeatOnly* calculates L_{DMD}(Eq.6) and L_{IFD}(Eq.9) with only the feature component, while *SemOnly* calculates L_{DMD} and L_{IFD} with only the semantic component. The results show that distilling semantics alone has better performance than distilling features alone. *Feat+Sem* and *Feat×Sem* refer to adding and multiplying the two components in Eq.6 & Eq.9. Multiplying results in better performance than adding. **(2)** We ablate L_{label}(Eq.1) to check how much the method is indeed relying on LwF regularization. *LwF* refers to label distillation only. *Ours* refers to IFD+DMD without label distillation. *LwF+Ours* is their combination. The results show that our method works well alone and improves LwF significantly. **(3)** We discuss the setting of two hyper-parameters α and β in Eq.1. The results show that performances are similar and insensitive for α and β from 1 to 1.5. **(4)** We discuss the sensitivity of our method to the number of queries. The results show that the performance increases as the number of queries increases from 100 to 300.

Table 5. Ablation results (%) on COCO under 70 + 10 scenario.

	Method		mAP	AbsGap↓	RelGap↓	Ω_{all} ↑
1	Effect of semantic information	Feat Only	36.80	3.50	8.68%	0.957
		Sem Only	37.10	3.20	7.94%	0.960
		Feat+Sem	37.50	2.80	6.95%	0.965
		Feat×Sem	37.60	2.70	6.70%	0.967
2	Effect of LwF	LwF	36.40	3.90	9.68%	0.952
		Ours	36.60	3.70	9.18%	0.954
		LwF + Ours	37.60	2.70	6.70%	0.967
3	Hyper params (α, β) in Eq.1	(1, 1)	37.60	2.70	6.70%	0.967
		(1, 1.5)	37.60	2.70	6.70%	0.967
		(1.5, 1)	37.50	2.80	6.95%	0.965
		(1, 2)	37.20	3.10	7.69%	0.962
		(2, 1)	37.10	3.20	7.94%	0.960
4	Num of queries	100	36.20	2.90	7.42%	0.963
		300	37.60	2.70	6.70%	0.967
5	Generality (Teacher+Student)	DETR+DETR	37.60	2.70	6.70%	0.967
		Ada+DETR	38.20	2.10	5.21%	0.974
		Ada+Ada	40.20	1.90	4.51%	0.977

6.4. Generality Studies

To demonstrate the generality of our method on Transformer-based detectors, we perform extended experiments on the latest Adamixer [10] detector, which has query-based structure with bipartite matching process. First, we set two standard AdaMixer detectors as the teacher model and the student model to build incremental Adamixer, with the same incremental learning settings as Section 3. The results of incremental Adamixer under MS-COCO 40+40, 50+30, 60+20, and 70+10 sce-

narios are shown in Appendix Table 6. Our incremental Adamixer exceeds the state-of-the-art ERD [9] method. In addition, we set Adamixer as the teacher model and set Deformable DETR as the student model to validate the generality between different transformer architectures. Table 5(5) shows the experimental results, which indicate that high-level queries carry robust semantic information, and better teacher and better student result in better performance. These experiments fully demonstrate the good generality of our method.

7. Discussion

Transformer vs CNN. Previous researches highlight the influence of model architecture in incremental classification task [32]. Existing IOD methods are all built upon CNN-based detectors, like Fast RCNN[24], Faster RCNN [30], YOLO-series [48], RetinaNet[26], FCOS[39], GFLv1[9].

Here we are the first to use a Transformer-based detector (Deformable DETR [52]) under full-dataset incremental object detection task. Compared with CNN-based detector, Transformer-based detector provides several advantages: **(1) Representation advantages**: Transformer models use the self-attention mechanism to ensure long-range dependencies, providing more effective feature extraction capability and more robust semantic representation. Our method benefits from this characteristic to realize more accurate between-class distance computation and within-class feature guidance. **(2) Architecture advantages**: Compared with the extremely large amount of candidates generated by CNN-based detector (Faster-RCNN: about 6300 candidate boxes per image), Transformer-based detector (Deformable DETR: about 300 candidate queries per image) can generate a relatively small set of queries as its candidates, leading to simplified statistical calculations for both the between-class distance matrix and within-class guidance. This efficiency greatly facilitates the implementation of our method. Leveraging these two advantages, our Transformer-based model outperforms all previous CNN-based models, showing remarkable enhancements. (seen in Table 4, Table 10).

Our methods can also be used in CNN-based detectors with intentional and sophisticated adaptation. To effectively perform DMD+IFD methods, the NMS can be used to filter candidate boxes and generate top-K high-quality predictions. The filtered features closest to the classifier can be used as high-level semantics, and FPN features can be used as low-level features. Obviously, the NMS influences the quality and quantity of high-level semantics. Therefore, it needs to be carefully designed according to the dataset to strike a balance between performance and computation, which can result in certain limitations.

Compared with metric learning. Metric learning aims to decrease the distance between similar objects and increase the distance between dissimilar objects [18]. Our method in incremental object detection task shares the same idea of reducing between-class discriminativeness and enlarging within-class consistency. However, newly added instances destroy the pattern of existing feature space, and thus leading to the forgetting of old knowledge. Since old knowledge is not provided during incremental training, we are not able to use metric learning to reduce the between-class distance and enlarge the within-class distance. We can only transfer the feature space pattern of old knowledge to the student model, so as to keep the within-class distance and between-class distance of old knowledge.

Comparison with OWOD methods. The recent work OW-DETR [11] also uses Transformer architecture. However, OW-DETR is designed for open-world object detection (OWOD) task, which deals with the coexistence of known and unknown objects at the same time, setting it apart from conventional incremental object detection workflows. ORE [16], Topology [47] and OW-DETR [11] are all designed for OWOD tasks, with sample replay as part of their methods. Here we compare our method with all these methods in Appendix Table 7. The results show that our method has the smallest AbsGap and RelGap and the largest Ω value, which indicates the excellent performance of our knowledge distillation method over other OWOD methods.

8. Conclusion

In this paper, we innovatively uncover the cause of catastrophic forgetting in incremental object detection task as the interference of "background" labels and destruction of semantic feature space. We elaborately design a method to alleviate the destruction of the semantic feature space and thereby mitigate the issue of catastrophic forgetting. We employed the distance matrix distillation (DMD) method to preserve the between-class discriminativeness and the interactive feature distillation (IFD) method to maintain the within-class consistency. Our method merges low-level and high-level information, while incorporating both between-class and within-class information into a unified framework. Extensive experiments on COCO and VOC datasets validate our effectiveness and generalization. The results demonstrate that utilizing within-class and between-class knowledge distillation helps exceed the state-of-the-art (SOTA) performance. Moreover, our method is the first to implement knowledge distillation in Transformer structure for full-dataset incremental object detection, which shows the remarkable potential of the Transformer structure in incremental object detection. More details can be found in Appendix.

References

[1] Fabio Cermelli, Antonino Geraci, Dario Fontanel, and Barbara Caputo. Modeling missing annotations for incremen-

tal learning in object detection. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3699–3709, 2022.

[2] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *ArXiv*, abs/1504.00325, 2015.

[3] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *ArXiv*, abs/1504.00325, 2015.

[4] Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4794–4802, 2019.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

[6] Na Dong, Yongqiang Zhang, Ming Ding, and Gim Hee Lee. Incremental-detr: Incremental few-shot object detection via self-supervised learning. *ArXiv*, abs/2205.04042, 2022.

[7] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6568–6577, 2019.

[8] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2010.

[9] Tao Feng, Mang Wang, and Hangjie Yuan. Overcoming catastrophic forgetting in incremental object detection via elastic response distillation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9417–9426, 2022.

[10] Ziteng Gao, Limin Wang, Bing Han, and Sheng Guo. Adamixer: A fast-converging query-based object detector. *ArXiv*, abs/2203.16507, 2022.

[11] Akshita Gupta, Sanath Narayan, K. J. Joseph, Salman Hameed Khan, Fahad Shahbaz Khan, and Mubarak Shah. Ow-detr: Open-world detection transformer. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9225–9234, 2021.

[12] Yu Hao, Yanwei Fu, Yu-Gang Jiang, and Qi Tian. An end-to-end architecture for class-incremental object detection with knowledge distillation. *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2019.

[13] Tyler L. Hayes, Ronald Kemker, Nathan D. Cahill, and Christopher Kanan. New metrics and experimental paradigms for continual learning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2112–21123, 2018.

[14] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531, 2015.

[15] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 831–839, 2019.

[16] K. J. Joseph, Salman Hameed Khan, Fahad Shahbaz Khan, and Vineeth N. Balasubramanian. Towards open world object detection. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5826–5836, 2021.

[17] K. J. Joseph, Jathushan Rajasegaran, Salman Hameed Khan, Fahad Shahbaz Khan, Vineeth N. Balasubramanian, and Ling Shao. Incremental object detection via meta-learning. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2021.

[18] Mahmut Kaya and Hasan S. Bilge. Deep metric learning: A survey. *Symmetry*, 11:1066, 2019.

[19] 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*, 114:3521 – 3526, 2016.

[20] 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*, 114:3521 – 3526, 2017.

[21] Dawei Li, Serafettin Tasci, Shalini Ghosh, Jingwen Zhu, Junting Zhang, and Larry Heck. Rilod: near real-time incremental learning for object detection at the edge. *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 2019.

[22] Wei Li, Q. Wu, Linfeng Xu, and Chao Shang. Incremental learning of single-stage detectors with mining memory neurons. *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, pages 1981–1985, 2018.

[23] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *ArXiv*, abs/2006.04388, 2020.

[24] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:2935–2947, 2018.

[25] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:318–327, 2017.

[26] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:318–327, 2020.

[27] Liyang Liu, Zhanghui Kuang, Yimin Chen, Jing-Hao Xue, Wenming Yang, and Wayne Zhang. Incdet: In defense of elastic weight consolidation for incremental object detection. *IEEE Transactions on Neural Networks and Learning Systems*, 32:2306–2319, 2020.

[28] Xialei Liu, Hao Yang, Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Multi-task incremental learning for object detection. *arXiv: Computer Vision and Pattern Recognition*, 2020.

[29] Angelo G Menezes, Gustavo de Moura, Cézanne Alves, and André CPLF de Carvalho. Continual object detection: A review of definitions, strategies, and challenges. *arXiv preprint arXiv:2205.15445*, 2022.

[30] Can Peng, Kun Zhao, and Brian C. Lovell. Faster ilod: Incremental learning for object detectors based on faster rcnn. *ArXiv*, abs/2003.03901, 2020.

[31] Can Peng, Kun Zhao, Sam Maksoud, Meng Li, and Brian C. Lovell. Sid: Incremental learning for anchor-free object detection via selective and inter-related distillation. *ArXiv*, abs/2012.15439, 2021.

[32] Haoxuan Qu, Hossein Rahmani, Li Xu, Bryan M. Williams, and Jun Liu. Recent advances of continual learning in computer vision: An overview. *ArXiv*, abs/2109.11369, 2021.

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021.

[34] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, G. Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5533–5542, 2017.

[35] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.

[36] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015.

[37] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *NIPS*, 2017.

[38] Konstantin Shmelkov, Cordelia Schmid, and Alahari Karteek. Incremental learning of object detectors without catastrophic forgetting. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3420–3429, 2017.

[39] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9626–9635, 2019.

[40] Jianren Wang, Xin Wang, Yue Shang-Guan, and Abhinav Kumar Gupta. Wanderlust: Online continual object detection in the real world. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10809–10818, 2021.

[41] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng. Distilling object detectors with fine-grained feature imitation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4928–4937, 2019.

[42] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8:415 – 424, 2021.

[43] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Raymond Fu. Large scale incremental learning. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 374–382, 2019.

[44] Binbin Yang, Xinchi Deng, Han Shi, Changlin Li, Gengwei Zhang, Hang Xu, Shen Zhao, Liang Lin, and Xiaodan Liang. Continual object detection via prototypical task correlation guided gating mechanism. *ArXiv*, abs/2205.03055, 2022.

[45] Dongbao Yang, Y. Zhou, and Weiping Wang. Multi-view correlation distillation for incremental object detection. *ArXiv*, abs/2107.01787, 2022.

[46] Dongbao Yang, Yu Zhou, Dayan Wu, Can Ma, Fei Yang, and Weiping Wang. Rd-iod: Two-level residual-distillation-based triple-network for incremental object detection. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 18:1 – 23, 2020.

[47] Shuo Yang, Pei Sun, Yi Jiang, Xiaobo Xia, Ruiheng Zhang, Zehuan Yuan, Changhu Wang, Ping Luo, and Min Xu. Objects in semantic topology. *ArXiv*, abs/2110.02687, 2021.

[48] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4643–4652, 2022.

[49] Chaohui Yu, Qiang feng Zhou, Jingliang Li, Jia-Chao Yuan, Zhibin Wang, and Fan Wang. Foundation model drives weakly incremental learning for semantic segmentation. *ArXiv*, abs/2302.14250, 2023.

[50] Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. Continuous learning of context-dependent processing in neural networks. *Nat. Mach. Intell.*, 1:364–372, 2019.

[51] Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasci, Larry Heck, Heming Zhang, and C.-C. Jay Kuo. Class-incremental learning via deep model consolidation. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1120–1129, 2019.

[52] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *ArXiv*, abs/2010.04159, 2021.