Appendix

9. More details about our methods

9.1. Distance Matrix Distillation (DMD)

Distance Matrix Distillation (DMD) is proposed to maintain the class-level between-class discriminativeness. Since teacher has already learned a high-quality discrete semantic space, student can learn to be distinct by directly learning from the discrete semantic space of teacher. Moreover, the high-level semantic difference is accumulated by low-level feature difference throughout forward-propagation and back-propagation of neural networks. Thus, in order to keep the difference, we focus on the interaction between both the useful low-level feature difference and the useful high-level semantic difference to better ensure its discriminativeness.

We provide a detailed description of DMD method as illustrated in Algorithm 1. Specifically, for each class i, we first calculate the average high-level semantic feature $\overline{Q}^T_{class_i}$ for teacher and $\overline{Q}^S_{class_i}$ for student. We also calculate the average low-level detail feature $\overline{F}^T_{class_i}$ for teacher and $\overline{F}^S_{class_i}$ for student. Then, we calculate four distance matrices $FMat^T$, $FMat^S$, $SMat^T$ and $SMat^S$ between each two categories i and j. We finally calculate DMD loss L_{DMD} as an interaction between the semantic difference ($SMat^S - SMat^T$) and feature difference ($FMat^S - FMat^T$).

9.2. Interactive Feature Distillation (IFD)

We provide a detailed description of IFD method as illustrated in Algorithm 2. First, for each instance n, we calculate (a) its high-level semantic difference Q_{diffn} between student and teacher, and (b) low-level detail difference F_{diffn} between student and teacher. Based on these two differences, we calculate the interactive difference IF_{diffn} for each instance. Then, average interactive difference $\overline{IF}_{diffi}$ is calculated as the average of all interactive difference IF_{diffn} within category i. The total interactive loss L_{IFD} is calculated as the sum of all average interactive difference $\overline{IF}_{diffi}$.

Algorithm 1 Distance Matrix Distillation (DMD)

Input: Image I, teacher detector θ^T, student detector θ^S, all old categories $Labels_{old}$.
Output: Class-level between-class loss function L_{DMD}.

Inference I with teacher detector θ^T yields potential instances. Each instance n includes low-level detail feature F^T_n, high-level semantic feature Q^T_n, and classification predictions C^T_n.
Inference I with student detector θ^S yields potential instances. Each instance n includes low-level detail feature F^S_n, high-level semantic feature Q^S_n, and classification predictions C^S_n.

// Calculating high-level semantic feature and low-level detail feature for each category.
for i in $Labels_{old}$ **do**
  for all C^T_n = i **do**
    // for teacher model
    Compute $APF^T_n = AdaptivePooling(F^T_n)$
    // for student model
    Compute $APF^S_n = AdaptivePooling(F^S_n)$
  end for
  Compute $\overline{F}^T_{class_i} = mean(APF^T_n)$
  Compute $\overline{Q}^T_{class_i} = mean(Q^T_n)$
  Compute $\overline{F}^S_{class_i} = mean(APF^S_n)$
  Compute $\overline{Q}^S_{class_i} = mean(Q^S_n)$
end for

// Calculating distance matrix among each two categories i and j
Create $FMat^T$, $FMat^S$, $SMat^T$, $SMat^S$
for i in $Labels_{old}$ **do**
  for j in $Labels_{old}$ **do**
    // for teacher model
    Compute $FMat^T_{i,j} = (\overline{F}^T_{class_i} - \overline{F}^T_{class_j})^2$
    Compute $SMat^T_{i,j} = (\overline{Q}^T_{class_i} - \overline{Q}^T_{class_j})^2$
    // for student model
    Compute $FMat^S_{i,j} = (\overline{F}^S_{class_i} - \overline{F}^S_{class_j})^2$
    Compute $SMat^S_{i,j} = (\overline{Q}^S_{class_i} - \overline{Q}^S_{class_j})^2$
  end for
end for

// Calculating DMD loss
Compute $L_{DMD} = (SMat^S - SMat^T) \times (FMat^S - FMat^T)$

return L_{DMD}

10. More results on IOD task

10.1. One-step results

As discussed in paper, Appendix Table 10 indicates 2-step performance of VOC under 10+10, 15+5, 19+1 scenarios. We compare our method with all other methods, including

- other knowledge distillation methods

  SID[31], ILOD[38], Faster ILOD[30], RILOD[21],

Algorithm 2 Interactive Feature Distillation (IFD)

Input: Image I, teacher detector θ^T, student detector θ^S, all old categories $Labels_{old}$.
Output: Instance-level within-class loss function L_{IFD}.

$\overline{IF}_{diff\,i} = 0$
$L_{IFD} = 0$
Inference I with teacher detector θ^T yields potential instances. Each instance n includes low-level detail feature F_n^T, high-level semantic feature Q_n^T, and classification predictions C_n^T.
Inference I with student detector θ^S yields potential instances. Each instance n includes low-level detail feature F_n^S, high-level semantic feature Q_n^S, and classification predictions C_n^S.

// Calculating interactive feature $IF_{diff\,n}$ for each instance, and averaging as $\overline{IF}_{diff\,i}$ for each category.
for i in $Labels_{old}$ **do**
  for all C_n^T = i **do**
    Compute $APF_n^S = AdaptivePooling(F_n^S)$
    Compute $APF_n^T = AdaptivePooling(F_n^T)$
    Compute $F_{diff\,n} = |F_n^S - F_n^T|$
    Compute $Q_{diff\,n} = |Q_n^S - Q_n^T|$
    Compute $IF_{diff\,n} = F_{diff\,n} \times Q_{diff\,n}$
  end for
  Compute $\overline{IF}_{diff\,i} = mean(IF_{diff\,n})$
end for

// Calculating total interactive loss
for i in $Labels_{old}$ **do**
  Compute $L_{IFD} = L_{IFD} + \overline{IF}_{diff\,i}$
end for

return L_{IFD}

MMA[1], Meta-ILOD[17], DMC[51], Topology[47], MVCD[45].

- replay methods
  Meta-ILOD[17], ORE[16], OW-DETR[11].
- methods including external data
  RILOD[21], DMC[51].
- methods including pseudo-labels
  ORE[16], Topology[47], OW-DETR[11], IncDet[27].
- EWC-related methods
  IncDet[27].
- meta-learning methods
  Meta-ILOD[17].

All these methods are listed in Appendix Table 8 for much more comparison and insights. The incremental leaning results in Appendix Table 10 demonstrate that our method is highly competitive.

10.2. Multi-step results

As mentioned in paper, we provide 4-step COCO results in Appendix Table 9, 5-step VOC in Appendix Table 11 and 3-step COCO+VOC results in Appendix Table 7.

Specifically, Appendix Table 9 shows the incremental leaning results on COCO dataset under 40+10+10+10+10 setting. Our method has better AbsGap, RelGap and Ω values than other knowledge distillation methods at each incremental step. It demonstrates the ability of alleviating long-term catastrophic forgetting.

Moreover, Appendix Table11 shows the VOC incremental learning results step-by-step under 15+1+1+1+1+1 setting. The performance of our method exceeds SID [31], the most typical knowledge distillation method. It also helps prove the effectiveness of our method on long-term catastrophic forgetting.

In addition, Appendix Table 7 provides 20+20+20+20 three-step scenario that using a combination of VOC and COCO datasets. We use the exact same dataset provided in [16, 47, 11]. The dataset includes all VOC classes as the first 20-class normal training, and the remaining 60 classes from COCO are grouped into three incremental steps with semantic drifts. The results show that our method has better performances even than ORE [16], Topology [47] and OW-DETR [11], the three methods are designed for open-world object detection task and include extra replay samples.

10.3. Stability and Plasticity

For incremental learning, it is very important to balance the stability on old knowledge and the plasticity on new knowledge. Here we also employs evaluation metrics of RSD, RPD and RSPD [29] to discuss stability and plasticity of incremental detectors. Specifically, RSD and RPD [29], defined in Eq.11 and Eq.12, refer to the rate of stability deficits on old categories and the rate of plasticity deficits on new categories. RPSD [29], defined in Eq.13, represents the balancing performance between stability and plasticity.

$$RSD = \frac{1}{N_{old}} \sum_{t=1}^{N_{old}} \frac{mAP_{upper,t} - mAP_{final,t}}{mAP_{upper,t}} \tag{11}$$

$$RPD = \frac{1}{N_{new}} \sum_{t=N_{old}+1}^{N_{old}+N_{new}} \frac{mAP_{upper,t} - mAP_{final,t}}{mAP_{upper,t}} \tag{12}$$

$$RPSD = RSD + RPD \tag{13}$$

where t refers to t^{th} task in incremental learning. $mAP_{final,t}$ and $mAP_{upper,t}$ refer to the task-average mAP

Table 6. Incremental results (%) with AdaMixer architecture on COCO benchmark under different scenarios. AbsGap and RelGap represent the absolute gap and the relative gap toward upper bound. Ω value represents the incremental capability. We here use ResNet50 as its backbone, and use PVT2[42] in Table 5(5) as its backbone.

Scenarios	Method	AbsGap	RelGap↓	$\Omega_{all}\uparrow$	$\Omega_{50}\uparrow$	$\Omega_{75}\uparrow$	$\Omega_{S}\uparrow$	$\Omega_{M}\uparrow$	$\Omega_{L}\uparrow$
40 classes + 40 classes	LwF[24]	23.00	57.21%	0.714	0.718	0.713	0.670	0.709	0.733
	RILOD[21]	10.30	25.62%	0.872	0.886	0.867	0.841	0.874	0.888
	SID[31]	6.20	15.42%	0.923	0.941	0.916	0.897	0.935	0.930
	ERD[9]	3.30	8.21%	0.959	0.967	0.954	0.959	0.958	0.955
	Ours	**2.58**	**6.78%**	**0.966**	**0.971**	**0.968**	**0.961**	**0.966**	**0.969**
50 classes + 30 classes	LwF[24]	35.20	87.56%	0.562	0.581	0.553	0.608	0.576	0.555
	RILOD[21]	11.70	29.10%	0.854	0.870	0.846	0.832	0.858	0.864
	SID[31]	6.40	15.92%	0.920	0.937	0.914	0.879	0.932	0.932
	ERD[9]	3.60	8.96%	0.955	0.963	0.946	0.918	0.958	0.960
	Ours	**2.16**	**5.68%**	**0.972**	**0.973**	**0.973**	**0.976**	**0.978**	**0.971**
60 classes + 20 classes	LwF[24]	34.40	85.57%	0.572	0.593	0.561	0.586	0.596	0.574
	RILOD[21]	14.80	36.82%	0.816	0.833	0.807	0.800	0.829	0.823
	SID[31]	7.50	18.66%	0.907	0.927	0.897	0.871	0.926	0.917
	ERD[9]	4.40	10.95%	0.945	0.954	0.940	0.944	0.947	0.945
	Ours	**3.38**	**8.89%**	**0.956**	**0.964**	**0.952**	**0.955**	**0.959**	**0.953**
70 classes + 10 classes	LwF[24]	33.10	82.34%	0.588	0.606	0.580	0.603	0.608	0.596
	RILOD[21]	15.70	39.05%	0.805	0.825	0.795	0.806	0.811	0.821
	SID[31]	7.40	18.41%	0.908	0.920	0.901	0.869	0.918	0.926
	ERD[9]	5.30	13.18%	0.934	0.945	0.929	0.903	0.940	0.936
	Ours	**4.27**	**11.24%**	**0.944**	**0.955**	**0.935**	**0.914**	**0.946**	**0.939**

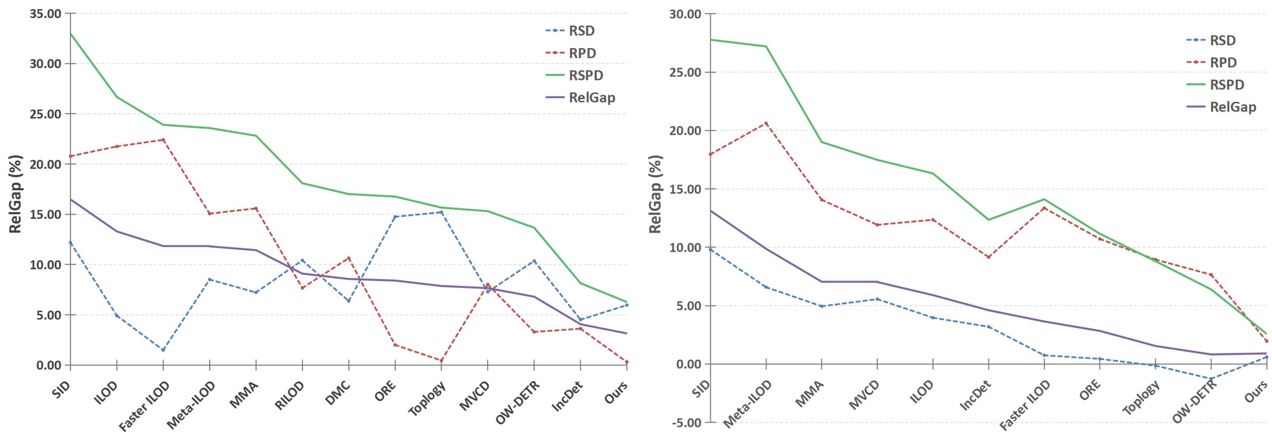
and upper-bound mAP on all testing data containing learned categories after task t, respectively. N_{old} and N_{new} refer to the total number of old categories and new categories. The smaller the RPSD is, the better the total performance of stability and plasticity would be.

As shown in Appendix Table 10, our method has the smallest AbsGap of 2.23 and RelGap of 3.15%, largest Ω of 0.9842 under 10+10 scenarios, indicating its excellent incremental learning ability. Our method also has smallest RSPD of 6.27, RSD of 5.97 and RPD of 0.30, indicating its best trade-off ability between stability and plasticity to achieve an optimal comprehensive performance. Similar results are shown under 15+5 scenarios and 19+1 scenarios, which further support the effectiveness of our method.

In addition, Appendix Table 10 shows taht the RPSD reflects a strong correlation with the final incremental learning performance. To better illustrate their relations, we plot the curves between RSD, RPD, RPSD and RelGap in Fig.6. Obviously, these remarkable results once again reveal that the balance between stability of old knowledge and plasticity of new knowledge is crucial to incremental learning. Compared with most of these methods, our method performs better on both RSD and RPD, therefore leading to better comprehensive performance on RPSD.

11. Further Discussion

In order to preserve the within-class consistency, semantic information for the same category should be forced to remain close-by. [47, 49] calculate a static semantic centroid for each category with pretrain models, like Bert [5] and CLIP [33], and make every instance within that category close to the static semantic centroid. Since these pre-trained models have rich class-level semantic information, it can provide a discrete and static semantic centroid for each category and thus support a discrete semantic feature space. However, it utilizes prior knowledge of the pre-trained model, which is not usually provided. Moreover, the semantic space should not be fixed due to its access to different datasets. Therefore, we discard the static semantic centroid method and propose to use dynamic semantic centroids generated by teacher. Since teacher has already learned a high-quality semantic space, student can learn the



(a) Stability and Plasticity on VOC(10+10)

(b) Stability and Plasticity on VOC(15+5)

Figure 6. Incremental learning performance (RelGap), the stability deficits rate (RSD) of old knowledge, the plasticity deficits rate (RPD) of new knowledge and their total deficits rate (RPSD). (a) results on VOC(10+10). (b) results on VOC(15+5). In both (a) and (b), the incremental learning performance (RelGap) reflects a positive correlation with RSPD. Some previous methods[38, 30, 17, 16, 47, 11] have contrary performance of stability and plasticity on VOC(10+10). They sacrifice one in exchange for the promotion of the other. All methods have consistent performance on VOC(15+5), in which sacrificing the latter has the least impact on final performance. Obviously, our methods achieve state-of-art or very competitive performance by making better trade-off between old and new knowledge.

Table 7. Incremental learning results under VOC+COCO 20+20+20+20 three-step scenario. VOC+COCO refers to the dataset setting of OWOD task proposed by ORE[16]. It combines Pascal VOC[8] and MS COCO[3] together. The first subset consists of 20 categories from Pascal VOC, and other subsets consist of remained 60 categories from MS COCO, which are split by semantic drifts. The colored mAP values of classes(1-20), classes(1-40) and classes(1-60) denote the performance of previously learned classes.

Method	mAP				AbsGap↓	RelGap↓	Ω_{all} ↑	Final mAP	Upper Bound
	A(1-20)	+B(20-40)	+B(40-60)	+B(60-80)					
ORE [16]	56.34				0.00	0.00%	1.0000	56.34	56.34
	52.37	25.58			11.13	22.21%	0.8889	38.98	50.11
	37.77		12.41		15.91	35.18%	0.8087	29.32	45.23
	30.01			13.44	16.23	37.84%	0.7619	26.66	42.89
Toplogy [47]	56.34				0.00	0.00%	1.0000	56.34	56.34
	53.39	26.49			10.17	20.30%	0.8985	39.94	50.11
	38.04		12.81		15.60	34.49%	0.8174	29.63	45.23
	30.11			13.31	16.98	39.59%	0.7641	25.91	42.89
OW-DETR [11]	56.34				0.00	0.00%	1.0000	56.34	56.34
	53.55	33.45			7.19	14.35%	0.9283	42.92	50.11
	38.25		15.82		14.46	31.97%	0.8456	30.77	45.23
	31.38			17.14	15.07	35.14%	0.7964	27.82	42.89
Ours	55.32				0.00	0.00%	1.0000	55.32	55.32
	49.16	38.67			3.90	8.15%	0.9592	43.91	48.33
	35.13		32.63		9.20	21.14%	0.9023	34.30	44.03
	31.04			24.91	**11.66**	**28.32%**	**0.8560**	29.51	42.73

semantic centroids and the entire semantic space by directly learning from its perfect teacher.

Table 8. Main methods and base detectors for class-incremental object detection task in recent years.

Method	Method Type	Base Detector
LwF [24]	Pseudo-Labels	Faster-RCNN[36]
SID[31]	Knowledge Distillation	CenterNet[7], FCOS[39]
ILOD [38]	Knowledge Distillation	Faster-RCNN[36]
RILOD [21]	Knowledge Distillation External Data	RetinaNet[25]
Faster ILOD [30]	Knowledge Distillation	Faster-RCNN[36]
Meta-ILOD [17]	Knowledge Distillation Meta-Learning Exemplar Replay	Faster-RCNN[36]
RD-IOD [46]	Knowledge Distillation	Faster-RCNN[36]
CIFRCN [12]	Knowledge Distillation	Faster-RCNN[36]
MVCD [45]	Knowledge Distillation	Faster-RCNN[36]
MMA [1]	Knowledge Distillation	Faster-RCNN[36]
DMC [51]	Knowledge Distillation External Data	RetinaNet[25]
ORE [16]	Pseudo-Labels Exemplar Replay	Faster-RCNN[36]
Topology [47]	Knowledge Distillation Exemplar Replay	Faster-RCNN[36]
OW-DETR [11]	Pseudo-Labels Exemplar Replay	DETR
IncDet [27]	Pseudo-Labels EWC	Faster-RCNN[36]
ERD [9]	Knowledge Distillation	GFL v1[23]
Ours	Knowledge Distillation	Deformable DETR[52] AdaMixer[10]

Table 9. Incremental results (%) on COCO benchmark under the 40+10+10+10+10 four-step setting.

	A(1-40)				
	+B(40-50)				
	mAP	AbsGap	RelGap↓	Ω_{all} ↑	Upper Bound
CF	5.80	32.20	84.74%	0.576	38.00
RILOD[21]	25.40	12.60	33.16%	0.834	38.00
SID[31]	34.60	3.40	8.95%	0.955	38.00
ERD[9]	36.40	1.60	4.21%	0.979	38.00
Ours	39.10	**1.10**	**2.81%**	**0.986**	40.20
	+B(50-60)				
	mAP	AbsGap	RelGap↓	Ω_{all} ↑	Upper Bound
CF	5.70	34.10	85.68%	0.432	39.80
RILOD[21]	11.20	28.60	71.86%	0.650	39.80
SID[31]	24.10	15.70	39.45%	0.839	39.80
ERD[9]	30.80	9.00	22.61%	0.911	39.80
Ours	35.40	**4.50**	**12.71%**	**0.953**	39.90
	+B(60-70)				
	mAP	AbsGap	RelGap↓	Ω_{all} ↑	Upper Bound
CF	6.30	29.40	82.35%	0.368	35.70
RILOD[21]	10.50	25.20	70.59%	0.561	35.70
SID[31]	14.60	21.10	59.10%	0.731	35.70
ERD[9]	26.20	9.50	26.61%	0.866	35.70
Ours	32.00	**7.80**	**24.38%**	**0.916**	39.80
	+B(70-80)				
	mAP	AbsGap	RelGap↓	Ω_{all} ↑	Upper Bound
CF	3.30	36.90	91.79%	0.311	40.20
RILOD[21]	8.40	31.80	79.10%	0.491	40.20
SID[31]	12.60	27.60	68.66%	0.648	40.20
ERD[9]	20.70	19.50	48.51%	0.796	40.20
Ours	30.30	**10.00**	**33.00%**	**0.883**	40.30

Table 10. Incremental results (%) on VOC benchmark under different scenarios.

Scenarios	Method	AbsGap	RelGap	Ω	RSPD	RSD	RPD	Final mAP			Upper Bound		
								Total	Old	New	Total	Old	New
10 classes + 10 classes	SID [31]	22.30	33.99%	0.8300	67.95	34.59	33.35	43.30	44.36	42.24	65.60	67.82	63.38
	ILOD [38]	9.38	13.30%	0.9335	26.67	4.91	21.75	61.14	67.34	54.93	70.51	70.82	70.20
	Faster ILOD [30]	8.35	11.84%	0.9408	23.90	1.50	22.41	62.16	69.76	54.47	70.51	70.82	70.20
	Meta-ILOD [17]	8.88	11.81%	0.9409	23.58	8.51	15.07	66.31	68.36	64.26	75.19	74.72	75.66
	MMA [1]	8.60	11.44%	0.9428	22.82	7.23	15.59	66.60	69.3	63.9	75.20	74.70	75.70
	RILOD [21]	6.80	9.10%	0.9545	18.09	10.42	7.67	67.90	67.48	68.36	74.70	75.33	74.04
	DMC [51]	6.40	8.57%	0.9572	17.01	6.37	10.64	68.30	70.53	66.16	74.70	75.33	74.04
	ORE [16]	5.93	8.41%	0.9579	16.76	14.76	2.01	64.58	60.37	68.79	70.51	70.82	70.20
	Topology [47]	5.55	7.87%	0.9606	15.67	15.21	0.46	64.96	60.03	69.88	70.51	70.80	70.20
	MVCD [45]	5.48	7.66%	0.9617	15.31	7.29	8.02	66.09	66.15	66.02	71.57	71.35	71.78
	OW-DETR [11]	4.80	6.81%	0.9660	13.67	10.36	3.30	65.71	63.48	67.88	70.51	70.82	70.20
	IncDet [27]	3.00	4.07%	0.9797	8.14	4.52	3.62	70.80	69.70	71.80	73.80	73.00	74.50
	Ours	**2.23**	**3.15%**	**0.9842**	**6.27**	**5.97**	**0.30**	68.56	67.01	70.11	70.79	71.26	70.32
15 classes + 5 classes	SID [31]	13.70	20.88%	0.8956	41.35	24.49	16.86	51.90	52.26	51.54	65.60	69.21	61.99
	Meta-ILOD [17]	7.42	9.87%	0.9507	27.20	6.58	20.62	67.77	71.73	55.90	75.19	76.78	70.42
	MMA [1]	5.30	7.05%	0.9648	19.01	4.95	14.06	69.90	73.00	60.50	75.20	76.80	70.40
	MVCD [45]	5.02	7.02%	0.9649	17.48	5.56	11.92	66.54	69.41	57.92	71.57	73.50	65.76
	ILOD [38]	4.16	5.90%	0.9705	16.32	3.97	12.36	66.35	69.25	57.60	70.51	72.11	65.72
	IncDet [27]	3.40	4.61%	0.9770	12.35	3.20	9.16	70.40	72.70	63.50	73.80	75.10	69.90
	Faster ILOD [30]	2.57	3.64%	0.9818	14.11	0.75	13.36	67.94	71.57	56.94	70.51	72.11	65.72
	ORE [16]	2.00	2.84%	0.9858	11.16	0.44	10.71	68.51	71.79	58.68	70.51	72.11	65.72
	OW-DETR [11]	1.09	1.55%	0.9923	8.81	-0.14	8.95	69.42	72.21	59.84	70.51	72.11	65.72
	Topology [47]	0.58	0.82%	0.9959	6.39	-1.26	7.65	69.93	73.01	60.69	70.51	72.10	65.72
	Ours	**0.64**	**0.91%**	**0.9955**	**2.57**	**0.60**	**1.96**	70.15	71.55	65.88	70.79	71.99	67.20
19 classes + 1 classes	SID [31]	20.10	30.64%	0.8468	60.82	33.82	27.00	45.50	46.33	44.67	65.60	70.01	61.19
	RILOD [21]	9.70	12.99%	0.9351	59.60	10.93	48.67	65.00	66.33	40.40	74.70	74.47	78.70
	Meta-ILOD [17]	4.97	6.60%	0.9670	27.56	5.82	21.74	70.23	70.89	57.60	75.19	75.27	73.60
	DMC [51]	3.90	5.21%	0.9739	17.12	4.80	12.33	70.81	70.90	69.00	74.70	74.47	78.70
	MMA [1]	4.50	5.98%	0.9701	19.44	5.58	13.86	70.70	71.10	63.40	75.20	75.30	73.60
	ILOD [38]	2.79	3.96%	0.9802	11.37	3.97	7.40	67.72	67.72	65.10	70.51	70.52	70.30
	Faster ILOD [30]	1.95	2.77%	0.9862	15.37	2.28	13.09	68.56	68.91	61.10	70.51	70.52	70.30
	MVCD [45]	1.86	2.59%	0.9870	14.28	2.11	12.17	69.71	70.19	60.60	71.57	71.70	69.00
	ORE [16]	1.62	2.30%	0.9885	16.17	1.66	14.51	68.89	69.35	60.10	70.51	70.52	70.30
	Topology [47]	0.69	0.98%	0.9951	11.81	0.43	11.38	69.82	70.22	62.30	70.51	70.52	70.30
	OW-DETR [11]	0.30	0.43%	0.9979	12.28	0.48	11.81	70.21	70.18	62.00	70.51	70.52	70.30
	Ours	**0.17**	**0.24%**	**0.9988**	**6.23**	**-1.65**	**7.88**	70.62	71.86	66.88	70.79	70.69	72.60

Table 11. Incremental results (%) on VOC benchmark under 15+1+1+1+1+1 five-step setting.

Scenarios	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP	SID[31]
A(1-20)	80.40	73.70	77.10	57.00	58.10	74.40	75.70	85.40	54.40	76.40	54.10	83.50	83.80	73.70	72.10	51.00	71.10	58.50	82.80	72.60	70.79	71.60
A(1-15)	81.90	75.90	76.40	58.70	60.90	74.20	76.50	86.40	57.60	78.50	57.50	83.70	84.10	75.00	75.30						73.51	73.71
+B(16, plant)	83.10	82.60	79.90	61.10	60.10	77.50	80.40	89.00	61.40	81.40	68.10	88.10	86.70	79.80	80.10	43.60					**75.18**	68.00
+B(17, sheep)	82.60	74.30	69.20	56.90	60.10	75.50	76.10	76.30	58.50	67.70	63.80	80.80	79.90	74.90	78.30	37.90	47.50				**68.25**	63.00
+B(18, sofa)	79.60	69.70	59.10	51.70	53.40	66.30	74.90	70.10	57.30	53.10	63.00	73.40	67.40	72.60	76.50	26.90	34.70	32.60			**60.13**	57.30
+B(19, train)	68.20	67.70	53.30	41.30	44.80	53.70	69.50	62.10	52.80	37.20	57.20	66.80	67.10	64.80	69.70	21.20	25.00	29.40	46.00		**52.52**	53.20
+B(20, tv)	68.90	63.60	55.60	39.00	46.00	53.90	67.00	67.00	54.30	29.20	56.30	61.10	61.00	57.90	69.10	23.50	23.60	30.00	41.10	59.30	**51.37**	48.90