Supplementary Material for "Rethinking Few-shot Class-incremental Learning: Learning from Yourself"

Yu-Ming Tang, Yi-Xing Peng, Jingke Meng, and Wei-Shi Zheng

A1 Detailed Information of The Corner Cases

In Sec 3. of the main paper, we analyze the necessity of the proposed *generalized accuracy* (*gAcc*) metric. In Fig. 2 of the main paper, we analyze some possible corner cases of the FSCIL setting on the proposed *gAcc* metric. Detailed information on these cases is shown below. An extension of Fig. 2 of the main paper is shown in Fig. A1.

Lazy. To analyze the situation of an algorithm that totally ignores the novel tasks and only retrains the first-task (base) performance, we design this corner case called 'Lazy' to show its performance under existing metrics and our proposed *gAcc*. The detailed accuracies of each task are shown in Table A1. We set the base class performance as 85% after training on the first task, *i.e.* $A_1^1 = 85\%$. We set the model completely **ignoring** the novel task and base task performance remains **unchanged** during incremental tasks, *i.e.* $\forall j \geq i > 1, A_i^j = 0, A_i^1 = A_1^1$. In Fig. A1 (c), the accuracy (*aAcc*) of 'Lazy' is very close to those of recent SOTAs. This is not satisfying since the primary goal of continual learning is to learn new knowledge. In comparison, the proposed *gAcc* correctly distinguishes this situation and presents a more balanced evaluation, which gives a low AUC to 'Lazy' and shows the margin of this case and other methods (see Fig. A1 (b)). In conclusion, the *aAcc* alone fails to reflect the performance of FSCIL methods properly and our *gAcc* can provide a more balanced perspective.

Greedy and Greedy-NF. We also show two other extreme examples. The 'Greedy' situation is that an algorithm only focuses on the novel classes and does not care about the previous forgetting problem, *i.e.* for all novel tasks, $A_i^j = 100\%$, when $i = j$ else 0. The A_1^1 is set to 85%. See Table A2 for the detailed performance. It is clear that, under the metric of *gAcc* and *aAcc*, the 'greedy' performs far worse than other methods/cases. Furthermore, we also show the non-forget version of the 'Greedy' as 'Greedy-NF'. For 'Greedy-NF', we set A_i^j to 100% for $\forall j > 1, j < i$. See Table A3 for the detailed performance at each task. From Fig. A1, we can see that, after removing the forgetting problem of the 'Greedy', the performance under *gAcc* is boosted significantly. It should be clarified that even though the 'Greedy-NF' performs well on the novel classes, their lame performance on the base classes affects the final *gAcc*. These results show that the *gAcc* will not over-emphasize the novel class performance and provide more balanced perspectives of base-novel performance.

Vit baseline. We further show our vit baseline in the Fig. A1. This baseline is conducted by employing a pure vit trained on the first task to extract features for

Table A1: Detailed accuracies setting of the corner case **Lazy**. In this case, the model only focuses on the base task and learns nothing about the novel tasks. f_i: model after trained on task $\mathcal{T}_i$. A^i: accuracy on task $\mathcal{T}_i$ of the corresponding model f.

(%)	A^1	A^2	A^3	A^4	A^5	A^6	A^7	A^8	A^9	$aAcc$	$gAcc$
f_1	85	-	-	-	-	-	-	-	-	85.00	81.46
f_2	85	0	-	-	-	-	-	-	-	78.46	66.29
f_3	85	0	0	-	-	-	-	-	-	72.86	57.15
f_4	85	0	0	0	-	-	-	-	-	68.00	50.61
f_5	85	0	0	0	0	-	-	-	-	63.75	45.58
f_6	85	0	0	0	0	0	-	-	-	60.00	41.55
f_7	85	0	0	0	0	0	0	-	-	56.67	38.22
f_8	85	0	0	0	0	0	0	0	-	53.68	35.42
f_9	85	0	0	0	0	0	0	0	0	51.00	33.01
Avg										65.49	49.92

Table A2: Detailed accuracies setting of the corner case **Greedy** In this case, the model only focuses on the novel task and does not care the forgetting of the old task. f_i: model after trained on task $\mathcal{T}_i$. A^i: accuracy on task $\mathcal{T}_i$ of the corresponding model f.

(%)	A^1	A^2	A^3	A^4	A^5	A^6	A^7	A^8	A^9	$aAcc$	$gAcc$
f_1	85	-	-	-	-	-	-	-	-	85.00	81.46
f_2	0	100	-	-	-	-	-	-	-	7.69	22.01
f_3	0	0	100	-	-	-	-	-	-	7.14	16.38
f_4	0	0	0	100	-	-	-	-	-	6.67	13.49
f_5	0	0	0	0	100	-	-	-	-	6.25	11.59
f_6	0	0	0	0	0	100	-	-	-	5.88	10.23
f_7	0	0	0	0	0	0	100	-	-	5.55	9.17
f_8	0	0	0	0	0	0	0	100	-	5.26	8.33
f_9	0	0	0	0	0	0	0	0	100	5.00	7.64
Avg										14.94	20.03

novel classes. This experiment is done on *mini*ImageNet dataset. Under the $gAcc$ metric, it is shown that even though the vit baseline gets a high performance on $gAcc(1)$, it does not perform well on the novel classes thus the AUC of the baseline is worse than SOTA methods.

A2 Detailed Network Structures

A2.1 Backbone

In this work, we conduct experiments on the vanilla Vision Transformer (ViT) [4]. To compare with the existing FSCIL methods fairly, we modified the ViT configuration to match the number of parameters. Specifically, we adjust the feature dimension and number of heads of the ViT backbone. See Table A4 for the detailed configuration.

A2.2 The MLPs used in this work

For the feature rectification (FR) module proposed in the main paper, we use MLPs to construct the FR module. We use the MLP structure in the Vision

Table A3: Detailed accuracies setting of the corner case **Greedy-NF**. On top of the 'Greedy', we keep the novel tasks' performance (**N**on-**F**orgetting). f_i: model after trained on task $\mathcal{T}_i$. A^i: accuracy on task $\mathcal{T}_i$ of the corresponding model f.

(%)	A^1	A^2	A^3	A^4	A^5	A^6	A^7	A^8	A^9	$aAcc$	$gAcc$
f_1	85	-	-	-	-	-	-	-	-	85.00	81.46
f_2	0	100	-	-	-	-	-	-	-	7.69	22.01
f_3	0	100	100	-	-	-	-	-	-	14.29	32.76
f_4	0	100	100	100	-	-	-	-	-	20.00	40.46
f_5	0	100	100	100	100	-	-	-	-	25.00	46.37
f_6	0	100	100	100	100	100	-	-	-	29.41	51.12
f_7	0	100	100	100	100	100	100	-	-	33.33	55.03
f_8	0	100	100	100	100	100	100	100	-	36.84	58.33
f_9	0	100	100	100	100	100	100	100	100	40.00	61.16
avg										32.40	49.86

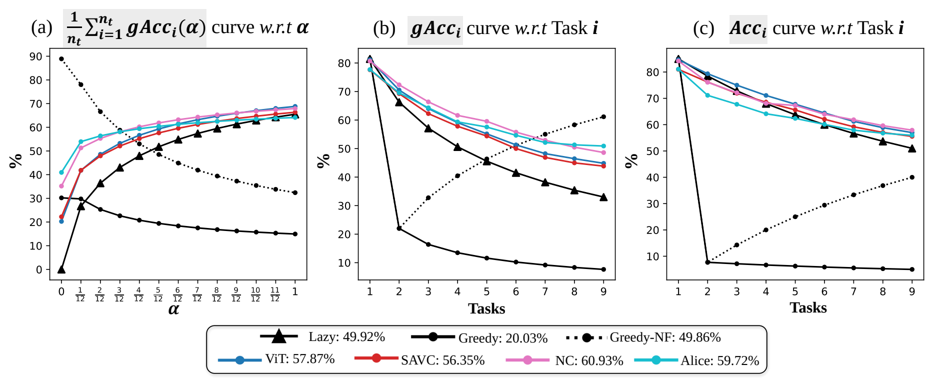

Fig. A1: Detailed illustration of our proposed $gAcc$. This figure is an extension of the Figure 2 in the main paper. Performance of methods is obtained on the *mini*ImageNet. In the legend, we show the average AUC across tasks (Eq. (6) in the main paper) of each method. **(a)**: The $gAcc$ curve (averaged across all n_t tasks) v.s. param α. **(b)**: The $gAcc$ AUC of each task $\mathcal{T}_i$ (see Eq. (5)) in the main paper) at each task. **(c)**: The $aAcc_i$ (see Eq. (1) in the main paper) at each task. Best viewed in color.

Transformer, *i.e.* for an input x, the MLP process it like this: $FC_1 \rightarrow$ GELU $\rightarrow$ $LN \rightarrow FC_2$, where FC_1 and FC_2 are two fully connected layers with the same dimension for both input and output. GELU is the activated function and LN is the LayerNorm normalization. FR modules are lightweight and only consist of 0.6M parameters for all three datasets.

A3 Dataset statistics

In this work, we use three datasets to evaluate our proposed method.
CIFAR-100 [8]. There are 50,000 images for training and 10,000 images for testing provided in the CIFAR-100 dataset. There exist 100 classes in total. For FSCIL, we follow pioneer works to select the first 60 classes as the base task for

Table A4: Details of the adjusted backbone. We slightly adjust the vanilla ViT to match the parameters of the ResNet-18.

Model	Embed Dim	#Heads	#Layers	#Params
Vit-Tiny [16]	192	3	12	5M
Vit-Small [16]	384	6	12	22M
Vit-Base [4, 16]	768	12	12	86M
ResNet-18	512	-	-	11M
Ours	256	8	12	10M

training. The rest 40 classes are split into 8 tasks and each task employs a 5-way 5-shot learning. The resolution of CIFAR-100 dataset is 32×32.

***mini*ImageNet [12].** With 100 classes, the *mini*ImageNet dataset has a total of 50,000 images for training and 10,000 images for testing. The resolution of *mini*ImageNet dataset is 84×84. For FSCIL, we follow prior works to select the first 60 classes for the base training and the rest 40 classes will be separated into 8 tasks.

CUB200 [17]. In this work, we use 'CUB-200' to refer to the CUB-200-2011 dataset, which contains 11,788 images of 200 different types of birds. The official training set contains 5994 images for training and there are 5794 samples for testing. The resolution of CUB-200 dataset is 224×224. Following previous works, we use the first 100 classes to form the base task, and the rest 100 classes are used to form 10 novel classes. Each novel task follows a 10-way 5-shot setting for FSCIL.

A4 Visualization

Corresponding to Figure 3 in the main paper, we further show the t-SNE [10] visualization of the rectified features as well as the original raw features from the Vit backbone. The visualization results are shown in Fig. A2. As discussed in the main paper, we see that the features from deep layers are more aggregated for the base classes (light-colored scatters in the figure), but their discriminating ability for novel classes is poor; on the contrary, although the shallow features are relatively more dispersed, causing the poor discriminating ability for base classes, the dispersion makes it more discriminative for novel classes. The FR module is proposed to take advantage of both features, *i.e.* being more discriminative on novel classes while keeping the base-class performance. The rectified features are shown in the lower part. As shown in the figure, the rectified features are clearly more distinguishable for different novel classes (red circle in the figure). At the same time, the proposed training strategy effectively constrains base features and the rectified features on base class are clearly more aggregated than that of the raw intermediate features (blue circle in the figure). A similar conclusion can be made across different layers shown in the figure.

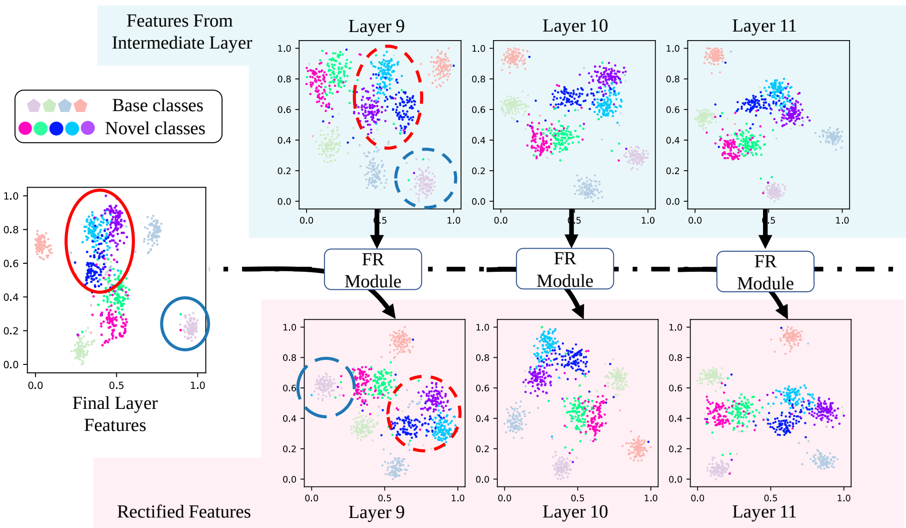

Fig. A2: Visualization results of the intermediate features, final features and rectified features on the test set of *mini*ImageNet using t-SNE [10]. This figure is an extension of the Figure 3 in the main paper. It is shown in this figure that the final features of novel classes are less discriminative compared with the intermediate features (red circles). However, Since the shallow features are more scattered, the performance of base classes will be worse (blue circles). We also show the results of FR modules (*i.e.* rectified features). It is clear that the rectified features absorb the advantages of both and perform better. Best viewed in color.

A5 More Experimental Results

In this section, we provide more experimental results to show the effectiveness of our method. Firstly, we show the CIFAR-100 experiment results in Sec. A5.1. Secondly, a detailed study of novel class performance is shown in Sec. A5.2. Later, more ablation studies including losses ablation on CUB-200, and sensitivity analysis on hyper-parameters are shown in Sec. A5.3.

A5.1 More Results on CIFAR-100

Experimental results conducted on the CIFAR-100 dataset are shown in Table A5. Although our method does not outperform NC-FSCIL in terms of average acc ($aAcc$), our method achieves higher generalized accuracy ($gAcc$) than NC-FSCIL. When we look into the detailed accuracy in each task, we can see that as the tasks proceed, our method achieves higher accuracies than NC-FSCIL (task $\mathcal{T}_9$). This is mainly because our method gets better novel class performance than NC-FSCIL. The detailed evaluation on $gAcc$ of the CIFAR-100 dataset is shown in Fig. A3. We can observe similar phenomena discussed in the main paper. For example, a recent SOTA SubNetwork [7] does not necessarily perform better

Table A5: Comparison with FSCIL methods on CIFAR-100. The best and second best results are marked **bolded** and <u>underlined</u>. Methods with†are reported with the original results in their paper. Other results are reimplemented by us.

Methods	*aAcc* in each task (%)									*aAcc*	*gAcc*
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**		
TOPIC [15]†	64.1	55.9	47.1	45.2	40.1	36.4	34.0	31.6	29.4	42.62	-
Self-promoted	64.1	65.9	61.4	57.5	53.7	50.8	48.6	45.7	43.3	54.52	-
BiDistFSCIL [21]†	79.5	75.4	71.8	68.0	65.0	62.0	60.2	57.7	55.9	66.14	-
MetaFSCIL [3]†	74.5	70.1	66.8	62.8	59.5	56.5	54.4	52.6	50.0	60.79	-
DSN [18]†	73.0	68.8	64.8	62.6	59.4	57.0	54.0	51.6	50.0	60.14	-
CEC [20]	73.2	69.1	65.4	61.2	58.0	55.5	53.2	51.3	49.2	59.56	50.85
CLOM [25]	74.2	69.8	66.1	62.2	59.0	56.1	54.0	51.9	49.8	60.35	50.36
C-FSCIL [5]	76.8	72.0	67.2	63.1	59.7	56.7	54.4	51.9	49.7	61.28	50.62
DF-Replay [9]	72.9	68.1	65.1	61.6	58.5	54.8	52.8	51.0	49.1	59.31	51.53
LIMIT [23]	76.1	71.7	67.3	63.2	60.1	57.3	55.2	53.0	50.8	61.64	52.16
SAVC [13]	78.4	72.7	68.4	64.0	61.2	58.3	56.0	54.1	51.7	62.76	53.29
FACT [22]	78.5	72.7	68.8	64.5	61.3	58.6	56.8	54.4	52.3	63.12	54.56
SubNetwork [7]	80.0	75.7	71.6	67.8	64.7	61.4	59.3	57.1	55.1	64.85	56.63
S3C [6]	78.0	73.9	70.2	66.1	63.3	60.1	58.3	56.7	54.0	64.51	58.04
ALICE [11]	80.3	71.9	67.0	63.2	60.7	58.3	57.3	55.5	53.7	63.10	59.72
NC-FSCIL [19]	82.9	77.5	73.7	69.0	65.6	62.0	59.7	57.8	<u>55.4</u>	**67.06**	<u>59.73</u>
Ours	82.9	76.3	72.9	67.8	65.2	62.0	60.7	58.8	**56.6**	<u>67.02</u>	**60.32**

than S3C [6] when considering the novel class performance. In fact, it is shown in the figure that although SubNetwork achieves better *aAcc* ($\alpha = 1$) than S3C, the S3C instead achieves a far larger AUC than SubNetwork. Similar observation can be found when comparing method C-FSCIL [5] with DF-replay [9]. It is clear that the *gAcc* curve and the corresponding AUC can offer a more fine-grained and more detailed perspective when evaluating FSCIL methods.

Table A6: Novel class performance in incremental tasks on *mini*ImageNet.

Methods	Novel class performance in incremental tasks (%)									*aAcc*	*gAcc*
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Avg		
CEC [20]	21.4	21.3	19.9	20.0	18.0	17.0	17.5	17.7	19.1	57.91	48.64
Self-Prompted [24]	6.6	6.3	5.0	4.1	6.4	6.8	6.7	7.0	6.11	52.62	41.53
LIMIT [23]	31.0	23.8	22.3	21.7	19.4	19.0	19.4	20.0	22.01	58.15	49.47
FACT [22]	18.8	16.6	15.0	14.1	12.4	11.4	11.7	12.2	14.03	60.79	49.56
CLOM [25]	6.4	10.5	11.2	11.5	10.6	9.7	10.7	11.6	10.28	57.23	46.11
C-FSCIL [5]	36.2	34.6	35.3	35.8	33.7	31.5	32.2	32.6	33.99	55.39	50.46
S3C [6]	25.0	26.0	27.9	25.7	23.7	22.5	23.7	23.5	24.75	62.74	53.70
SAVC [13]	31.6	27.1	26.6	26.3	22.8	21.4	21.6	22.4	24.98	66.66	56.35
BiDistFSCIL [21]	32.2	32.4	29.3	31.2	26.6	25.8	26.2	25.8	28.69	61.32	53.49
Regularizer [1]	47.2	44.8	37.1	34.0	39.8	22.0	25.4	26.2	34.56	62.82	55.62
SubNetwork [7]	35.4	33.1	30.1	29.0	27.0	26.2	26.4	27.8	29.38	64.31	55.88
NC-FSCIL [19]	52.0	45.4	42.9	40.2	37.4	34.4	32.7	31.5	39.56	67.90	60.93
Vit-Baseline	22.4	24.0	24.3	23.6	21.8	21.3	22.5	22.6	22.81	68.74	57.87
+YourSelf (Ours)	51.4	49.0	44.1	43.5	40.3	38.5	38.1	39.0	**42.99**	**68.80**	**62.20**

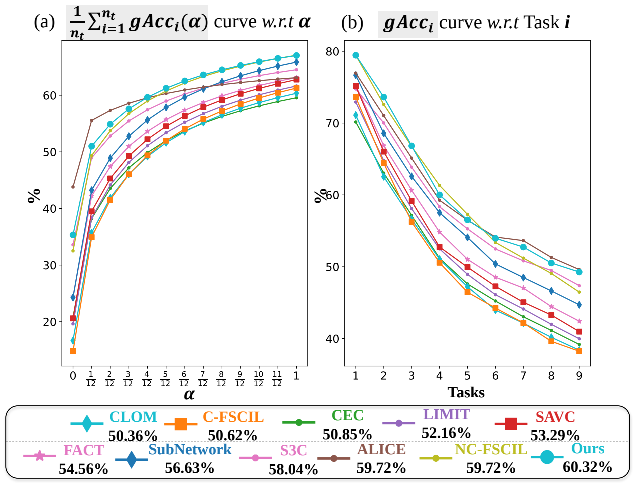

Fig. A3: FSCIL performances on the CIFAR-100 dataset are shown in *generalized accuracy*. In the legend, we show the average AUC across tasks(Eq. (6) in the main paper) of each method. **(a)**: The $gAcc$ curve (averaged across all n_t tasks) v.s. param α. **(b)**: The $gAcc$ AUC of each task $\mathcal{T}_i$ (see Eq. (5)) in the main paper) at each task. Best viewed in color.

A5.2 Novel classes performance

As a supplement to Sec. 5.2 in the main paper, we also provide results of only novel classes ($gAcc(0)$) learning only across incremental learning tasks. Results on *mini*ImageNet and CIFAR-100 are shown in Tab. A6 and Tab. A7 respectively. Similar to the conclusion stated in Sec. 5.2, we can see from the tables that the $aAcc$ metric can not effectively reflect the change of novel class performance. For example, in Tab. A6, models CLOM and C-FSCIL share similar $aAcc$ performance while their novel class performance is not even close. Similar conclusions can also be summarized by comparing FACT and S3C in Tab. A7. This validates the necessity of our proposed $gAcc$, which can effectively show the performance gap on novel tasks, complementary to the existing $aAcc$ metric. Experimental results in the two tables also show the effectiveness of our proposed method. It is clear that our proposed method effectively boosts the novel class performance compared to the baseline in both datasets. Furthermore, our proposed Yourself method outperforms other methods on novel tasks on average.

A5.3 More Ablation Studies

Ablation studies on CUB-200. The ablation study regarding the effectiveness of each loss on CUB-200 is shown in Table A8. We can see that the introduction of the FR module as well as the loss $\mathcal{L}_{cos}$ and loss $\mathcal{L}_{NovCE}$ can bring 1.26% improvement on $gAcc$. Additionally, the $gAcc$ performance can be boosted by

Table A7: Novel class performance in incremental tasks on CIFAR-100.

Methods	Novel class performance in incremental tasks (%)										
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Avg	$aAcc$	$gAcc$
CEC [20]	30.2	27.3	22.0	21.4	22.1	21.9	21.5	20.9	23.41	59.56	50.85
CLOM [25]	22.6	22.0	18.6	17.5	16.7	17.7	17.5	17.4	18.75	60.35	50.36
C-FSCIL [5]	23.4	16.3	14.0	13.2	16.3	16.9	15.8	17.2	16.64	61.27	50.62
DF-Replay [9]	43.0	33.6	27.9	27.1	22.4	22.4	23.4	22.3	27.76	59.31	51.53
LIMIT [23]	27.2	24.4	21.3	21.0	21.0	21.2	20.5	20.1	22.09	61.64	52.16
SAVC [13]	30.0	25.6	20.0	21.7	22.3	22.3	22.3	21.2	23.18	52.76	53.29
FACT [22]	35.2	31.0	26.6	25.2	25.7	26.8	25.1	24.2	27.48	63.12	54.56
S3C [6]	49.2	40.8	36.2	35.0	35.2	35.2	35.5	35.2	37.79	64.50	58.04
NC-FSCIL [19]	46.0	41.4	39.1	36.4	34.0	33.6	32.1	30.0	36.58	**67.06**	59.72
Vit-Baseline	23.6	16.5	15.1	13.5	14.0	14.3	14.5	15.0	15.81	66.35	54.21
+YourSelf (Ours)	58.8	44.7	37.3	34.8	35.8	32.3	34.5	35.8	**39.25**	67.02	**60.32**

0.4% and 1.47% when introducing the loss $\mathcal{L}_{CR}$ and loss $\mathcal{L}_{IR}$ respectively. This indicates the effectiveness of each loss we proposed.

Table A8: Ablation studies on CUB-200 dataset.

FR	$\mathcal{L}_{Cos}$ $+\mathcal{L}_{NovCE}$	$\mathcal{L}_{CR}$	$\mathcal{L}_{IR}$	$aAcc$ in each task (%)											$aAcc$	$gAcc$
				1	2	3	4	5	6	7	8	9	10	11		
				84.0	79.6	76.4	72.5	70.1	67.4	66.3	65.2	63.5	63.2	52.3	70.16	62.37
✓	✓			83.4	79.9	76.8	73.2	70.8	68.0	66.9	66.3	64.3	64.2	63.3	70.57	63.63
✓	✓	✓		83.4	78.7	75.7	72.1	70.2	67.3	65.7	65.3	63.4	63.6	62.7	69.93	64.03
✓	✓	✓	✓	83.4	77.0	75.3	72.2	69.0	66.8	66.0	65.6	64.1	64.5	63.6	69.85	65.50

The impact of trade-off parameters. Recall that the overall loss function described in the main paper is as follows.

$$\mathcal{L} = \beta_{cos}(\mathcal{L}_{cos} + \mathcal{L}_{NovCe}) + \beta_{CR}\mathcal{L}_{CR} + \beta_{IR}\mathcal{L}_{IR} \tag{A1}$$

In order to study the effect of different parameter values on the final performance, we conduct experiments on three hyper-parameters β_{cos}, β_{CR} and β_{IR}. Experiments are conducted on CIFAR-100 and *mini*Imagenet. Results are shown in Fig. A4.

- **The effect of β_{cos}.** We can see in the Fig. A4 (a) that the $\mathcal{L}_{cos}$ and $\mathcal{L}_{NovCe}$ can effectively improve the $gAcc$ without altering the $aAcc$ significantly. When we adjust the weight β_{cos} ranging from 0 to 1, we can see the $aAcc$ remains almost unchanged (except for the outlier of $\beta_{cos} = 1$ on *mini*ImageNet) but the $gAcc$ first rise and then fall. By default, we set the $\beta_{cos} = 0.1$ in all other experiments in this work.
- **The effect of β_{CR}.** We conduct experiments by setting the hyper-parameter β_{CR} of the loss $\mathcal{L}_{CR}$ ranging in $[0, 1]$. From Fig. A4 (b), it can be concluded that the introduction of the loss $\mathcal{L}_{CR}$ can bring up to 4% performance gain on $gAcc$. Furthermore, we can see that for both $aAcc$ and $gAcc$, the performance rises first and goes down when the β_{CR} gets larger. We set $\beta_{CR} = 0.1$ as

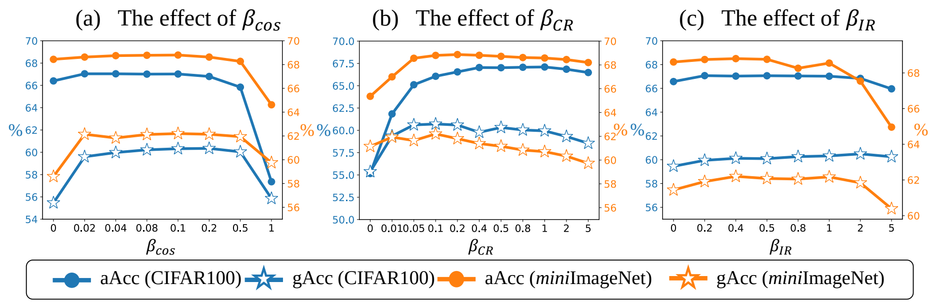

Fig. A4: Ablation studies on the impact of three hyper parameters β_{cos}, β_{CR} and β_{IR}. We use orange to indicate the results on *mini*ImageNet and Blue to indicate the results from CIFAR-100. In each sub-figure, the y-axis for the miniImageNet dataset is on the right in orange, while the y-axis for the CIFAR-100 dataset is on the left in blue. We report both $aAcc$ (●) and $gAcc$ (☆) results in the figure. Best viewed in color.

our default setting on *mini*ImageNet and $\beta_{CR} = 0.5$ on CIFAR-100, which is applied to all experiments in this work.

- **The effect of β_{IR}.** Similarly, we show the experiment about the effect of different β_{IR}. When changing values of β_{IR} form 0 to 1, the $gAcc$ is rising up and then falling down while the $aAcc$ on *mini*ImageNet is falling down. It shows that with proper weights, the $\mathcal{L}_{IR}$ can effectively boost the $gAcc$ without doing much harm to the $aAcc$. We set $\beta_{IR} = 1$ as the default setting for CIFAR-100 and $\beta_{IR} = 0.5$ for *mini*ImageNet.

A6 Discussion about our $gAcc$ and other alternative metrics in FSCIL

Harmonic Accuracy [11]. It is worth mentioning that in previous work Alice [11] proposed a metric called *harmonic accuracy* based on the harmonic mean to tackle the problem of the base-novel balance problem. The *harmonic accuracy (hAcc)* at task $\mathcal{T}_i$ can be denoted as:

$$hAcc_i = \frac{2A_i^1 A_i^{novel}}{A_i^1 + A_i^{novel}}, \tag{A2}$$

where A_i^1 is the base class accuracy and the $A_i^{novel} = \frac{1}{i-1}\sum_{j=2}^{i} A_i^j$ is the averaged performance of the novel tasks. The $hAcc$ is derived from the harmonic mean which was originally designed for calculating the average of proportions (*e.g.* velocity). Direct use of the harmonic mean to calculate accuracy loses its clear physical meaning. Further, the $hAcc$ has the following limitation when evaluating the FSCIL methods.

- **Symmetry Issue.** The harmonic mean resembles the arithmetic mean, where in the FSCIL context, the arithmetic mean represents the 'task-wise accuracy' (as detailed in Sec. 3.1 of the main paper). Both arithmetic mean and harmonic mean treat the performance from base tasks and novel tasks equally. As shown in Eq. (A2), the roles of A_i^1 and A_i^{novel} are symmetry. Unlike our *gAcc*, this will ignore the difficulty of the base task and over-emphasize the performance of the novel classes. A better solution to balancing this 'base-novel' dilemma is to analyze the performance from more fine-grain perspectives like our proposed *gAcc*.

– **Heterogenicity.** When comparing different methods using a metric, it's routine to utilize the absolute difference between their values to signify the performance gap. However, when employing *hAcc* for the partial derivative of A_i^{novel}, the equation

$$\frac{\partial(hAcc)}{\partial A_i^{novel}} = \frac{2(A_i^1)^2}{(A_i^1 + A_i^{novel})^2}, \tag{A3}$$

indicates a non-linear change rate in *hAcc*. This non-linearity is prominent when $(A_i^1 + A_i^{novel}) < 1$ (common in FSCIL), causing the derivative to exceed one. Essentially, with a small $(A_i^1 + A_i^{novel})$ and a constant A_i^1 while A_i^{novel} decreases, the reduction in *hAcc* amplifies. Consequently, when evaluating methods with low A_i^{novel}, the absolute difference in *hAcc* becomes inaccurate due to this amplification. Termed as 'heterogeneity', this phenomenon hampers the visualization of differences between methods. In contrast, our proposed *gAcc* exhibits linear changes, ensuring that the absolute difference accurately reflects performance gaps among methods.

– **Instability.** It is clear from Eq. (A2) that if A_i^{novel} or A_i^1 equals zero, the overall accuracy will be zero (or not defined when we consider the original definition of the harmonic mean). This makes the analysis of corner cases (like we do in the main paper and Sec. A1) more difficult.

Forgetting metrics. In addition to metrics like PD and KR stated in Sec. 3.3 of the main paper, we further discuss two more forgetting metrics Performance Dropping (PD) and Relative Performance Dropping [14](RPD) here. Both PD and RPD are metrics to measure the forgetting of the incremental learning methods. For a particular task $\mathcal{T}_i$, PD evaluates the performance by PD $= A_{n_t}^i - A_i^i$, which calculates the performance drops between the first seen task $\mathcal{T}_i$ and the last task $\mathcal{T}_{n_t}$. Another similar metric forgetting [2] also reflects the performance drops but it focuses on strictly the *maximum* drop: $F = \max_j A_j^i - \min_j A_j^i$. RPD [14] is the extension of PD that calculates the ratio of the PD by the accuracy of the first task, *i.e.* RPD$= \frac{\text{PD}}{A_i^i}$

These metrics show the forgetting of FSCIL methods serve a different unique role in the evaluation and our proposed *gAcc* is complementary to them. Similar to the conclusion in Sec. 3.3, all these forgetting metrics will fail to correctly reflect the poor performance of cases like 'Lazy' while ours can effectively capture changes on the novel class performance.

References

1. Akyürek, A.F., Akyürek, E., Wijaya, D.T., Andreas, J.: Subspace regularizers for few-shot class incremental learning. In: International Conference on Learning Representations (2022)
2. Chaudhry, A., Dokania, P.K., Ajanthan, T., Torr, P.H.: Riemannian walk for incremental learning: Understanding forgetting and intransigence. In: Proceedings of the European conference on computer vision (ECCV). pp. 532–547 (2018)
3. Chi, Z., Gu, L., Liu, H., Wang, Y., Yu, Y., Tang, J.: Metafscil: A meta-learning approach for few-shot class incremental learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2022)
4. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021)
5. Hersche, M., Karunaratne, G., Cherubini, G., Benini, L., Sebastian, A., Rahimi, A.: Constrained few-shot class-incremental learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
6. Kalla, J., Biswas, S.: S3c: Self-supervised stochastic classifiers for few-shot class-incremental learning. In: European Conference on Computer Vision (2022)
7. Kang, H., Yoon, J., Madjid, S.R.H., Hwang, S.J., Yoo, C.D.: On the soft-subnetwork for few-shot class incremental learning. In: International Conference on Learning Representations (2023)
8. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Technical report, University of Toronto (2009)
9. Liu, H., Gu, L., Chi, Z., Wang, Y., Yu, Y., Chen, J., Tang, J.: Few-shot class-incremental learning via entropy-regularized data-free replay. In: European Conference on Computer Vision (2022)
10. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research (2008)
11. Peng, C., Zhao, K., Wang, T., Li, M., Lovell, B.C.: Few-shot class-incremental learning from an open-set perspective. In: European Conference on Computer Vision (2022)
12. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision (2015)
13. Song, Z., Zhao, Y., Shi, Y., Peng, P., Yuan, L., Tian, Y.: Learning with fantasy: Semantic-aware virtual contrastive constraint for few-shot class-incremental learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
14. Tan, Z., Ding, K., Guo, R., Liu, H.: Graph few-shot class-incremental learning. In: Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining (2022)
15. Tao, X., Hong, X., Chang, X., Dong, S., Wei, X., Gong, Y.: Few-shot class-incremental learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020)
16. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning (2021)

17. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)
18. Yang, B., Lin, M., Zhang, Y., Liu, B., Liang, X., Ji, R., Ye, Q.: Dynamic support network for few-shot class incremental learning. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)
19. Yang, Y., Yuan, H., Li, X., Lin, Z., Torr, P., Tao, D.: Neural collapse inspired feature-classifier alignment for few-shot class incremental learning. In: International Conference on Learning Representations (2023)
20. Zhang, C., Song, N., Lin, G., Zheng, Y., Pan, P., Xu, Y.: Few-shot incremental learning with continually evolved classifiers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2021)
21. Zhao, L., Lu, J., Xu, Y., Cheng, Z., Guo, D., Niu, Y., Fang, X.: Few-shot class-incremental learning via class-aware bilateral distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
22. Zhou, D.W., Wang, F.Y., Ye, H.J., Ma, L., Pu, S., Zhan, D.C.: Forward compatible few-shot class-incremental learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2022)
23. Zhou, D.W., Ye, H.J., Ma, L., Xie, D., Pu, S., Zhan, D.C.: Few-shot class-incremental learning by sampling multi-phase tasks. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)
24. Zhu, K., Cao, Y., Zhai, W., Cheng, J., Zha, Z.J.: Self-promoted prototype refinement for few-shot class-incremental learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2021)
25. Zou, Y., Zhang, S., Li, Y., Li, R.: Margin-based few-shot class-incremental learning with class-level overfitting mitigation. In: Advances in neural information processing systems (2022)