Supplementary Material: Progressive Prototype Evolving for Dual-Forgetting Mitigation in Non-Exemplar Online Continual Learning

Anonymous Authors

1 DETAILS ABOUT EVALUATION METRIC

Following [1, 4, 5], we choose *Final Accuracy* and *Average Forgetting* to evaluate our methods. For a continual learning task that has T stages. *Final Accuracy* is the accuracy of all the classes after training the model of T-th stage. *Average Forgetting* represents the average performance degradation of different stages. Specifically, the forgetting of j-th continual learning stage after training on t-th stage can be computed as :

$$f_j^t = \max_{l \in \{1,\ldots,t-1\}} a_{l,j} - a_{t,j}, \tag{1}$$

where $a_{t,j}$ represents the accuracy of stage j, after training the model from stage 1 to t. Then the average forgetting at stage t can be computed as:

$$F_t = \frac{1}{t-1} \sum_{j=1}^{t-1} f_j^t. \tag{2}$$

2 MORE EXPERIMENTAL RESULTS

2.1 Average Forgetting

In Table 1, we present the Average Forgetting results of various state-of-the-art Online Continual Learning methods with different memory sizes. Notably, our method employs a memory size of zero for all experiments. It is evident that as the memory size increases, the average forgetting rate of the existing OCL method decreases. This trend can be attributed to the preservation of exemplars from previous samples, explicitly aiding knowledge retention through data replay. Remarkably, even without retaining any exemplars, our method achieves the lowest average forgetting results. This remarkable performance can be attributed to our approach, which leverages information stored in progressive prototypes to mitigate intra-stage forgetting. Additionally, Prototype Similarity Preserving and Prototype-Guided Gradient Constraint components are also introduced to address the inter-stage forgetting problem.

2.2 Intra-stage Forgetting Mitigation

In this section, we will demonstrate the anti-intra-stage forgetting ability of our proposed $\mathcal{L}_{pce}$ method. Apart the results in Section 4.3 in the main paper, to further illustrate the effectiveness of our proposed progressive prototypes in resisting intra-stage forgetting, we present the accuracy of newly encountered classes at each stage on CIFAR-10, CIFAR-100, and MiniImageNet in Figure 1. The results reveal that the method incorporating the proposed $\mathcal{L}_{pce}$ achieves higher accuracy for newly encountered classes across different datasets and different stages. This improvement can be attributed to the progressive prototypes, which accumulate knowledge from previously seen samples during a stage. The inclusion of $\mathcal{L}_{pce}$ takes this accumulated knowledge into consideration, effectively mitigating intra-stage forgetting.

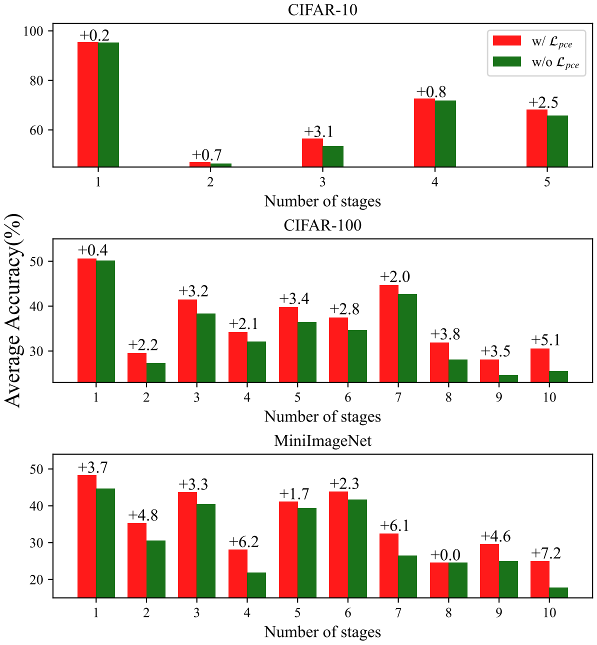

Figure 1: Accuracy of newly encountered classes of each stage on various datasets.

3 INFLUENCE OF HYPERPARAMETERS

In this section, we examine the impact of various hyperparameters on the CIFAR-100 dataset. The results for different α and β, which are used to calculate ϵ_t in Eq.7 in the main paper, are presented in Table 2 and Table 3. The parameter α serves as the intercept of the linear function and a higher α means that a higher threshold ϵ_t is used to construct the projection matrix. β defines the scope of the linear function, representing how changes in the similarity between prototypes, s_t, influence ϵ_t. Based on the results, we set $\alpha = 2.65$ and $\beta = 2$.

In our method, instead of changing the weight of the $\mathcal{L}_{ppe}$, which is designed for prototype optimization, we implement progressive prototypes with varying learning rates to control their optimization. Results for different learning rates are shown in Table 4. Our method demonstrates robustness across different learning rates for prototypes, with a marginal improvement observed at a learning rate of 60. Consequently, we select a learning rate of 60 for prototypes.

Furthermore, we explore the influence of weight parameters γ, μ in Table 5 and Table 6. γ represents the weight of $\mathcal{L}_{pce}$, which is designed to mitigate intra-stage forgetting, while μ is the weight of $\mathcal{L}_{psp}$, which is designed to mitigate inter-stage forgetting. When

Table 1: Average Forgetting (lower is better) of different methods on various datasets. All experiment results are the average results across 15 runs.

Dataset		CIFAR-10			CIFAR-100			MiniImageNet		
Memory size		10	20	100	100	200	500	100	200	500
SCR	CVPR-W2021	51.9±4.7	56.9±2.0	46.8±1.4	26.8±0.9	26.7±0.8	24.1±0.6	16.4±1.0	17.8±1.0	15.5±0.7
RAR	NeurIPS2022	57.2±4.4	61.3±3.0	54.2±1.2	40.0±1.0	41.9±0.8	38.7±0.8	22.4±1.4	24.9±1.1	24.0±1.0
DVC	CVPR2022	50.9±2.7	48.1±2.8	36.3±2.5	42.0±1.0	39.0±0.9	34.2±0.8	36.5±1.1	34.2±0.9	29.2±0.9
GSA	CVPR2023	62.4±1.4	57.3±1.4	38.3±1.2	51.6±0.6	48.4±0.6	40.5±0.5	40.9±0.4	38.8±0.6	32.4±0.7
PCR	CVPR2023	45.0±4.1	37.0±4.2	23.8±3.2	36.7±0.8	31.3±0.8	21.0±1.2	35.4±0.5	30.5±0.9	20.5±1.2
SSD	AAAI2024	65.3±1.0	65.0±1.0	48.4±1.4	36.0±0.6	32.7±0.7	24.7±0.6	19.5±0.6	18.0±0.8	16.0±0.5
PPE(Ours)	**This Paper**	**23.3±0.7**	**23.3±0.7**	**23.3±0.7**	**14.8±0.6**	**14.8±0.6**	**14.8±0.6**	**10.8±0.5**	**10.8±0.5**	**10.8±0.5**

Table 2: Ablation study of α in Eq.9 in the main paper.

α	2.55	2.60	**2.65**	2.70	2.75
acc(%)	21.5±0.4	21.9±0.4	**22.0±0.4**	21.6±0.3	20.6±0.3

Table 3: Ablation study of β in Eq.9 in the main paper.

β	1.8	1.9	**2.0**	2.1	2.2
acc(%)	17.2±0.2	20.7±0.3	**22.0±0.4**	21.6±0.4	20.3±0.4

Table 4: Ablation study of the learning rate (lr) of progressive prototypes.

lr	40	50	**60**	70	80
acc(%)	21.4±0.3	**22.0±0.3**	**22.0±0.4**	21.9±0.3	**22.0±0.3**

Table 5: Ablation study of γ, the weight of $\mathcal{L}_{pce}$.

γ	0.3	**0.5**	1	1.5	2
acc(%)	21.8±0.3	**22.0±0.4**	21.9±0.3	21.7±0.3	21.4±0.4

Table 6: Ablation study of μ, the weight of $\mathcal{L}_{psp}$.

μ	6	8	**10**	12	14
acc(%)	21.9±0.3	21.9±0.4	**22.0±0.4**	**22.0±0.4**	21.9±0.4

these parameters are too small, the model tends to forget previous knowledge, while excessively large values lead the model to primarily focus on retaining prior knowledge, hindering the acquisition of new knowledge. Based on experiment results, we set $\gamma = 0.5$ and $\mu = 10$ to achieve an optimal balance.

4 DETAILS ABOUT GRADIENT PROJECTION

4.1 Relation between input and gradient space

According to [2, 3], the gradient update of each layer lies in the span of input. The linear layer without bias can be formed as:

$$x_{out} = \sigma(W^T x_{in}), \tag{3}$$

where $x_{in} \in \mathbb{R}^{d_{in}}$, $x_{out} \in \mathbb{R}^{d_{out}}$ denote the input and output, d_{in}, d_{out} denote the input and output dimension, σ denotes the activation function, $W \in \mathbb{R}^{d_{in} \times d_{out}}$ denote the weight. Given the loss function L, the gradient of W can be computed by chain rule:

$$\frac{\partial L}{\partial W} = \frac{\partial L}{\partial x_{out}} \frac{\partial x_{out}}{\partial W} = (\frac{\partial L}{\partial x_{out}} \sigma') x_{in}^T. \tag{4}$$

Thus Equation (4) shows that the the gradient of W lies in the span of input.

The convolution layer without bias can be formed as:

$$x_{out} = \sigma(W * x_{in}) \tag{5}$$

where $x_{in} \in \mathbb{R}^{C_{in} \times h_{in} \times w_{in}}$, $x_{out} \in \mathbb{R}^{C_{out} \times h_{out} \times w_{out}}$ denote the input and output, C_{in}, C_{out} denote the input and output channel, h_{in}, w_{in}, h_{out}, w_{out} denote the size of the input and output, σ denotes the activation function, $W \in \mathbb{R}^{C_{out} \times C_{in} \times k \times k}$ denote the weight. We can reshape x_{in} into $\hat{x}_{in} \in \mathbb{R}^{(C_{in} \times k \times k) \times h_{out} \times w_{out}}$ and reshape W into $\hat{W} \in \mathbb{R}^{(C_{in} \times k \times k) \times C_{out}}$, then the convolution can be reformed as matrix multiplication:

$$\hat{x}_{out} = \sigma(\hat{W}^T \hat{x}_{in}), \tag{6}$$

where $\hat{x}_{out} \in \mathbb{R}^{C_{out} \times (h_{out} \times w_{out})}$. Similar to the chain rule in Equation (4), the gradient of the weight of the convolution layer also lies in the span of input.

4.2 Construction of input matrix R_t

As mentioned after Equation (5) in the main paper, existing offline continual learning methods [2, 3] construct the input matrix R_t by sampling data from the entire dataset which is impractical in NEOCL due to the one-pass data stream. Therefore, we handle this constraint by utilizing the latest batch of a learning stage to obtain R_t. To validate the adequacy of R_t constructed from the latest batch in representing the input space, we conduct experiments where R_t is replaced with data sampled from the entire dataset. The results presented in Table 7 indicate that constructing R_t with the latest batch achieves superior performance. This outcome is attributed to the nature of the online setting, where the model continuously updates based on the single-pass data stream. Consequently, features extracted from later encountered samples may better represent the input space of the model.

Table 7: Results of differnt R_t on CIFAR-100.

R_t	Latest Batch	Entire Dataset
acc(%)	**22.0±0.4**	21.9±0.4

REFERENCES

[1] Yanan Gu, Xu Yang, Kun Wei, and Cheng Deng. 2022. Not just selection, but exploration: Online class-incremental continual learning via dual view consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[2] Yan-Shuo Liang and Wu-Jun Li. 2023. Adaptive Plasticity Improvement for Continual Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[3] Gobinda Saha, Isha Garg, and Kaushik Roy. 2021. Gradient Projection Memory for Continual Learning. In *International Conference on Learning Representations*. https://openreview.net/forum?id=3AOj0RCNC2

[4] Dongsub Shim, Zheda Mai, Jihwan Jeong, Scott Sanner, Hyunwoo Kim, and Jongseong Jang. 2021. Online class-incremental continual learning with adversarial shapley value. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35.

[5] Yujie Wei, Jiaxin Ye, Zhizhong Huang, Junping Zhang, and Hongming Shan. 2023. Online Prototype Learning for Online Continual Learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.