Unsupervised Continual Learning of Image Representation via Rememory-based SimSiam

Feifei Fu, Yizhao Gao, Zhiwu Lu, Haoran Wu, Shiqi Zhao

RenMin University of China

Contents

Motivation

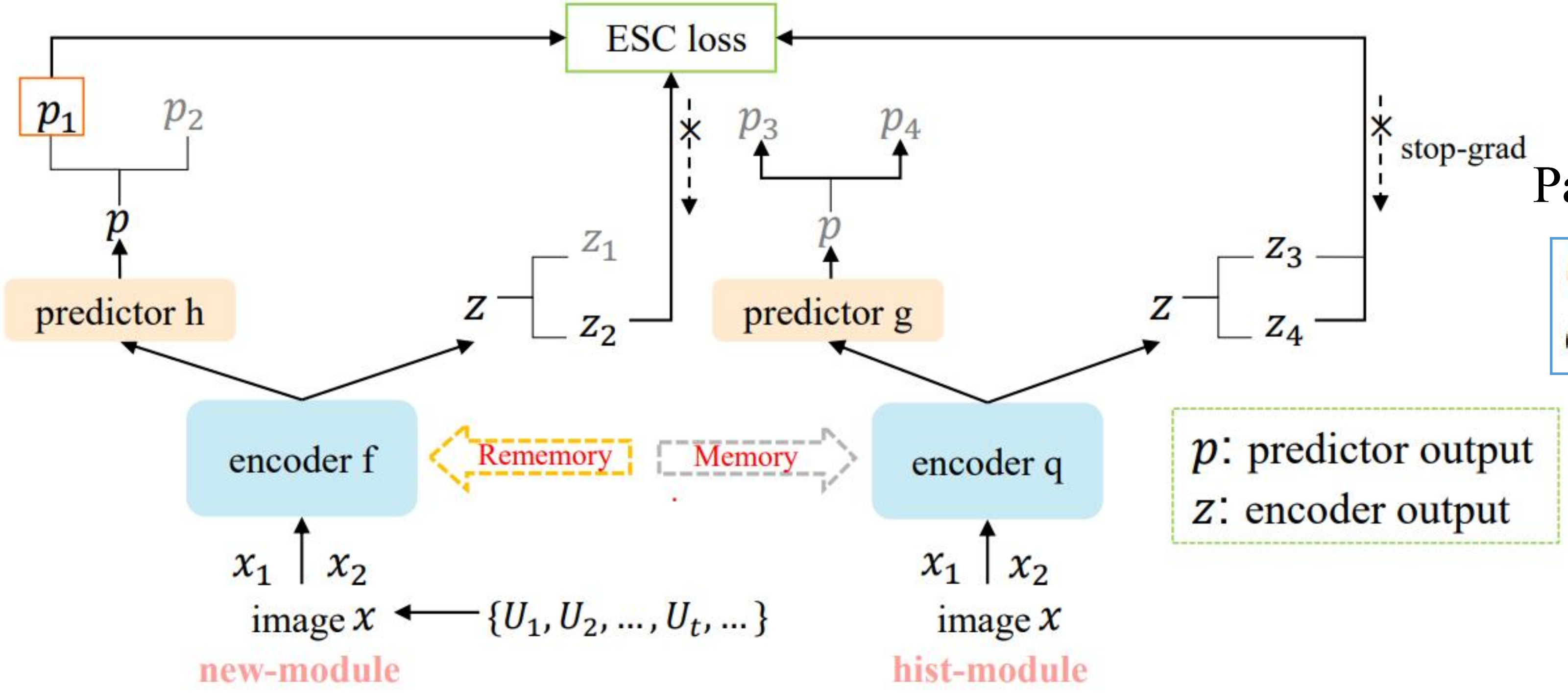
- The pratical need of representation learning with unlabled data on sequential tasks -> Unsupervised Continual Learning, UCL.
- Recent UCL methods focus on mitigating the catastrophic forgetting problem with a replay buffer to store previous data (i.e., rehearsal-based strategy), which needs much extra storage and thus limits their practical applications.

Idea

Based on contrastive learning via SimSiam, we propose a novel rememory-based SimSiam (**RM-SimSiam**) method to reduce the dependency on replay buffer under the UCL setting. The **core idea** of our RM-SimSiam is to store and remember the old knowledge with a data-free historical module instead of replay buffer.

Method (RM-SimSiam)

- Rememory Mechanism: consolidate (memory) and remember (rememory) old knowledge
  - **Memory process**: The hist-module is designed to retain old knowledge by storing the historical average model of all previous models.
  - **Rememory process**: The hist-module is designed to then transfer the knowledge of the historical average model to the new-module.



Parameter transfer process (EMA strategy):

$$\theta_i^e = m \cdot \theta_i^e + (1-m) \cdot \theta_j^e, \quad i, j \in \{f, q\}, \; i \neq j,$$
$$\theta_i^p = m \cdot \theta_i^p + (1-m) \cdot \theta_j^p, \quad i, j \in \{h, g\}, \; i \neq j,$$

θ_q^e, θ_g^p of the hist-module ->no gradient back-propagation.

Method (RM-SimSiam)

- Enhanced SimSiam-based Contrastive Loss

  - To improve the rememory ability of RM-SimSiam, by aligning the feature representations outputted by the historical and new models.

$$\boxed{L_{sim}} = \frac{1}{2}D(p_1, z_2) + \frac{1}{2}D(p_2, z_1),$$

(1) Original SimSiam

cosine-distance

$$\underline{D(p_1, z_2)} = -\frac{p_1}{\|p_1\|_2} \cdot \frac{z_2}{\|z_2\|_2},$$

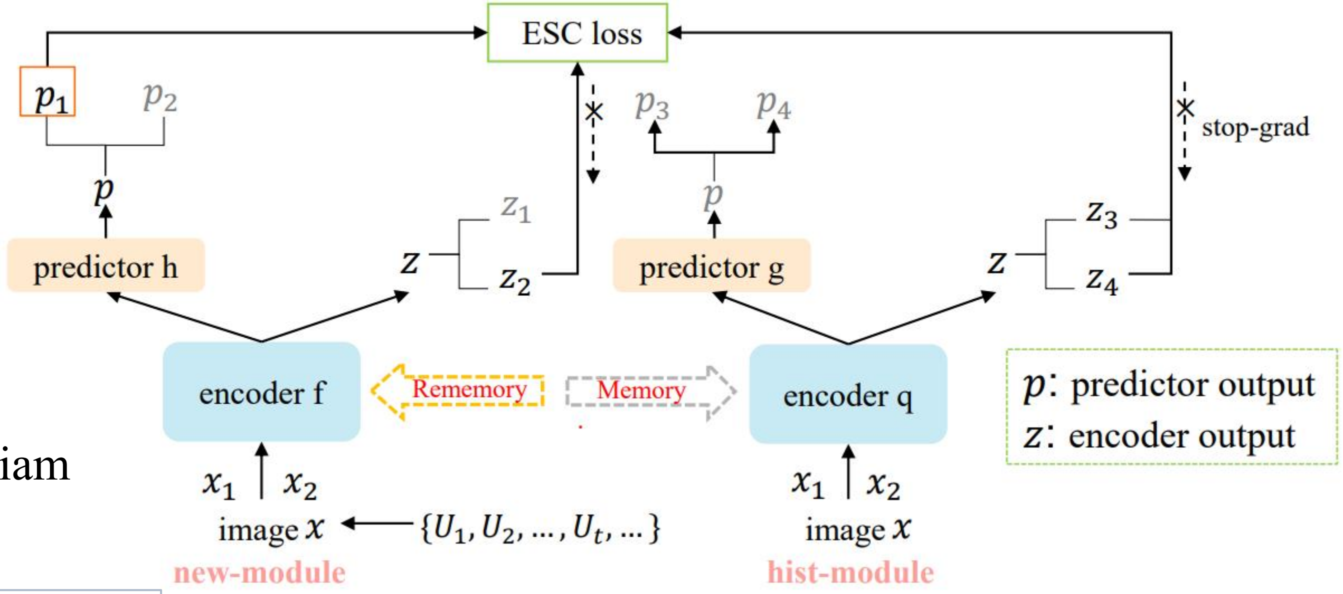
$$\begin{aligned}\boxed{L_{hist}} = & \frac{1}{2}D(p^1_{i,t}, z^3_{i,t}) + \frac{1}{2}D(p^3_{i,t}, z^1_{i,t}) + \frac{1}{2}D(p^1_{i,t}, z^4_{i,t}) \\ & + \frac{1}{2}D(p^4_{i,t}, z^1_{i,t}) + \frac{1}{2}D(p^2_{i,t}, z^3_{i,t}) + \frac{1}{2}D(p^3_{i,t}, z^2_{i,t}) \\ & + \frac{1}{2}D(p^2_{i,t}, z^4_{i,t}) + \frac{1}{2}D(p^4_{i,t}, z^2_{i,t}) + \frac{1}{2}D(p^3_{i,t}, z^4_{i,t}) \\ & + \frac{1}{2}D(p^4_{i,t}, z^3_{i,t}).\end{aligned}$$

(2) RM-SimSiam

(3) Simplified version

non-gradient property of the hist-module, impose the stop-gradient operation $sg(\cdot)$ on z.

$$\begin{aligned}\boxed{L_{hist}} \triangleq & \frac{1}{2}D(p^1_{i,t}, \mathrm{sg}(z^3_{i,t})) + \frac{1}{2}D(p^2_{i,t}, \mathrm{sg}(z^3_{i,t})) \\ & + \frac{1}{2}D(p^1_{i,t}, \mathrm{sg}(z^4_{i,t})) + \frac{1}{2}D(p^2_{i,t}, \mathrm{sg}(z^4_{i,t})).\end{aligned}$$



Total loss: $\boxed{L_{esc} = L_{sim} + \gamma L_{hist},}$

Method (RM-SimSiam)

- Algorithm

Algorithm 1 Unsupervised Continual Learning with RM-SimSiam

Input: the memory buffer M, the dataset U
the new-module with parameters θ_{new}
the hist-module with parameters θ_{hist}
hyperparameters α and m, the learning rate η

Output: the learned θ^*_{new}

$M \longleftarrow \{\}$
for x in U **do**
$\theta_{hist} \longleftarrow \theta_{new}$ ▷ Initialize the hist-module
$(x_1^m, x_2^m) \longleftarrow \text{sample}(M)$
$x_1, x_2 \longleftarrow \text{augment}(x)$
$\lambda \longleftarrow \text{numpy.random.beta}(\alpha, \alpha)$
$\hat{x}_1 \longleftarrow \lambda \cdot x_1 + (1-\lambda) \cdot x_1^m$
$\hat{x}_2 \longleftarrow \lambda \cdot x_2 + (1-\lambda) \cdot x_2^m$
$z_1, z_2 \longleftarrow f_\theta(\hat{x}_1), f_\theta(\hat{x}_2)$ ▷ Compute the outputs of the new-module
$p_1, p_2 \longleftarrow h_\theta(z_1), h_\theta(z_2)$
$z_3, z_4 \longleftarrow q_\theta(\hat{x}_1), q_\theta(\hat{x}_2)$ ▷ Compute the outputs of the hist-module
$p_3, p_4 \longleftarrow g_\theta(z_3), g_\theta(z_4)$
$\theta_{hist} \longleftarrow m \cdot \theta_{hist} + (1-m) \cdot \theta_{new}$ ▷ Update the hist-module
$\theta_{new} \longleftarrow m \cdot \theta_{new} + (1-m) \cdot \theta_{hist}$ ▷ Reverse update the new-module
$\theta_{new} = \theta_{new} - \eta \cdot \nabla_{\theta_{new}} L_{esc}$
$M \longleftarrow \text{reservoir}(x, \hat{x}_2)$
end for
return the found best θ^*_{new}

Experiments

- Experimental Setup

  - Datasets

Datasets	Classes	Class/Task	Resolution
Split CIFAR-10	10	2 classes/5 tasks	32*32
Split CIFAR-100	100	5 classes/20 tasks	32*32
Split Tiny-ImageNet	100	5 classes/20 tasks	64*64

  - Evaluation Metrics

Average accuracy: $A_t = \frac{1}{t}\sum_{i=1}^{t} a_{t,i}$

Average forgetting: $F_t = \frac{1}{t-1}\sum_{i=1}^{t-1} \max_{t' \in \{1,\ldots,t\}} (a_{t',i} - a_{t,i})$

Experiments

- Main Results

Table 1. Comparison to the state-of-the-arts under the UCL setting in terms of average accuracy and average forgetting over three independent runs. 'acc' and 'fg' refer to average accuracy and average forgetting respectively. The standard deviation is given in brackets. All UCL methods (with the same backbone ResNet18) are trained from scratch. * denotes our RM-SimSiam without buffer.

Method	S-CIFAR-10		S-CIFAR-100		S-Tiny-IMAGENET	
	acc (↑)	fg (↓)	acc (↑)	fg (↓)	acc (↑)	fg (↓)
FINETUNE	90.11 (±0.12)	5.42 (±0.08)	75.42 (±0.78)	10.19 (±0.37)	71.07 (±0.20)	9.48 (±0.56)
PNN [33]	90.93 (±0.22)	–	66.58 (±1.00)	–	62.15 (±1.35)	–
SI [43]	92.75 (±0.06)	1.81 (±0.21)	80.08 (±1.30)	5.54 (±1.30)	72.34 (±0.42)	8.26 (±0.64)
DER [4]	91.22 (±0.30)	4.63 (±0.26)	77.27 (±0.30)	9.31 (±0.09)	71.90 (±1.44)	8.36 (±2.06)
LUMP [22]	91.00 (±0.40)	2.92 (±0.53)	82.30 (±1.35)	4.71 (±1.52)	76.66 (±2.39)	3.54 (±1.04)
Cassle [10]	90.84 (±0.13)	2.29 (±0.23)	76.46 (±1.02)	3.05 (±0.87)	71.99 (±0.46)	3.34 (±0.52)
RM-SimSiam* (ours)	91.22 (±0.12)	4.15 (±0.18)	78.48 (±0.31)	4.09 (±0.99)	72.25 (±0.06)	4.51 (±0.04)
RM-SimSiam (ours)	**93.07 (±0.13)**	**1.36 (±0.10)**	**83.26 (±0.30)**	**2.73 (±0.42)**	**77.10 (±0.16)**	**2.67 (±0.01)**
MULTITASK	95.76 (±0.08)	–	86.31 (±0.38)	–	82.89 (±0.49)	–

- The effectiveness of our RM-SimSiam.
- Our RM-SimSiam is indeed complementary to the rehearsal-based strategy and provides a new perspective to mitigate forgetting in UCL.

Table 2. Comparison to the state-of-the-arts on the out-of-distribution (OOD) datasets.

IN-CLASS	S-CIFAR-100			
OUT-OF-CLASS	MNIST	FMNIST	SVHN	CIFAR-10
FINETUNE	85.99 (±0.86)	76.90 (±0.11)	50.09 (±1.41)	57.15 (± 0.96)
SI [2]	91.50 (±1.26)	80.57 (±0.93)	54.07 (±2.73)	60.55 (±2.54)
DER [3]	87.96 (±2.04)	76.21 (±0.63)	47.70 (±0.94)	56.26 (±0.16)
LUMP [6]	91.76 (±1.17)	81.61 (±0.45)	50.13 (±0.71)	63.00 (±0.53)
Cassle [7]	88.87 (±0.45)	81.30 (±0.45)	51.04 (±0.01)	59.46 (±1.62)
RM-SimSiam (ours)	**94.96 (±0.21)**	**83.29 (±0.19)**	**60.37 (±1.72)**	**69.16 (±0.17)**
MULTITASK	90.35 (±0.24)	81.11 (±1.86)	52.20 (±0.61)	70.19 (±0.15)

- The obtained improvements on the OOD datasets show the superior generalization ability of our RM-SimSiam when unseen data distributions are encountered.

Experiments

- Ablation Study

Table 3. Ablation study results for our full RM-SimSiam on S-CIFAR-10 and S-CIFAR-100 under the UCL setting. Notations: RM – the rememory mechanism; Hist – the extra contrastive loss L_{hist} defined based on the historical module (hist-module).

Method	S-CIFAR-10		S-CIFAR-100	
	acc (↑)	fg (↓)	acc (↑)	fg (↓)
Base (SimSiam)	90.16 (±0.24)	5.85 (±0.32)	75.51 (±0.70)	10.70 (±0.83)
Base+Mixup	90.40 (±0.18)	2.47 (±0.08)	77.89 (±0.77)	6.97 (±0.67)
Base+Mixup+RM	91.10 (±0.21)	1.67 (±0.41)	80.29 (±0.19)	4.24 (±0.45)
Base+Mixup+Hist	92.49 (±0.19)	1.96 (±0.26)	82.26 (±0.22)	3.91 (±0.26)
Base+Mixup+RM+Hist (full)	**93.07 (±0.13)**	**1.36 (±0.10)**	**83.26 (±0.30)**	**2.73 (±0.42)**

➢ Our full RM-SimSiam achieves significant improvements over Base+Mixup, which means that we have made sufficient contributions by devising new rememory mechanism and enhanced SimSiam-based contrastive loss for UCL.

Experiments

- Visualization Results

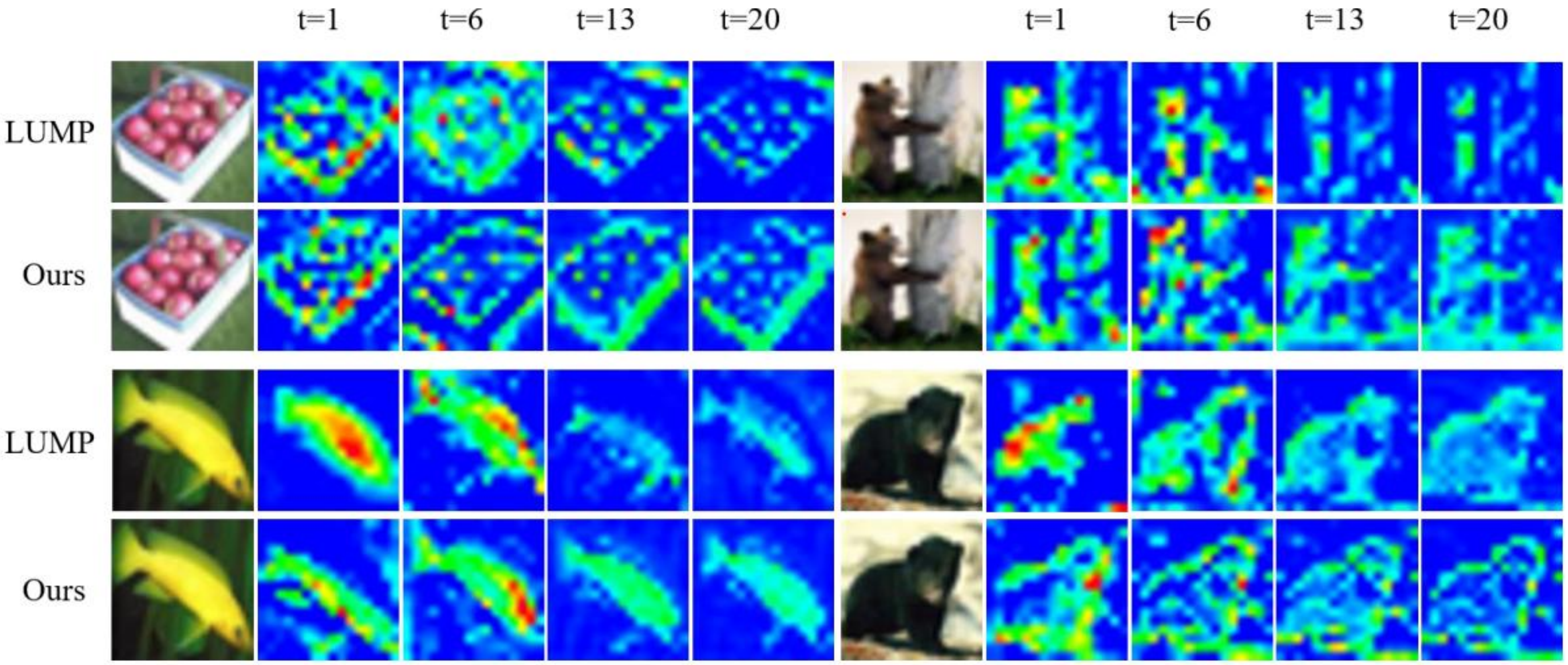


Figure 3. Visualization examples of feature maps from (the last layer of) the second block of the backbone ResNet18 when LUMP and our RM-SimSiam are being trained sequentially across all 20 tasks of S-CIFAR-100 (but only task 1, 6, 13 and 20 are shown for conciseness). The input images are randomly selected from the test set of task 1.

- We can clearly observe that our RM-SimSiam can better locate the important areas of the objects and represent the key visual features more stably across sequential tasks as compared with LUMP.

Conclusion

- We propose a novel rememory-based method termed **RM-SimSiam** for unsupervised continual learning by storing and remembering the old knowledge with a **data-free historical module** instead of replay buffer.
- To effectively rememory the knowledge of previous tasks, we design a **hist-module** by **storing** the knowledge of previous models and **transferring** the knowledge of previous models to the new model.
- To further improve the rememory ability of our RM-SimSiam, we devise an **enhanced SimSiam-based contrastive loss** by aligning the representations outputted by the historical and new models.
- Extensive experiments on three benchmarks show that our RM-SimSiam achieves new state-of-the-art under the UCL setting.

Thank you!