Exploring Data Geometry for Continual Learning
Supplementary Materials

Zhi Gao[1], Chen Xu[2*], Feng Li, Yunde Jia[2,1], Mehrtash Harandi[3], Yuwei Wu[1,2*]
[1]Beijing Key Laboratory of Intelligent Information Technology,
School of Computer Science & Technology, Beijing Institute of Technology, China
[2]Guangdong Laboratory of Machine Perception and Intelligent Computing,
Shenzhen MSU-BIT University, China
[3]Department of Electrical and Computer Systems Eng., Monash University, and Data61, Australia
{gaozhi_2017,jiayunde,wuyuwei}@bit.edu.cn
xuchen@smbu.edu.cn, lifeng.passion@gmail.com, mehrtash.harandi@monash.edu

Size of $\mathcal{B}$	Method	T200-B0-S10
200	ER [8]	8.79 ± 0.21
	AGEM [5]	8.28 ± 0.15
	iCaRL [7]	8.64 ± 0.78
	FDR [2]	8.77 ± 0.82
	DER++ [3]	11.16 ± 0.95
	ERT [4]	10.85 ± 0.24
	RM [1]	13.58 ± 1.07
	Ours	$\mathbf{14.45 \pm 0.85}$

Table 1. Accuracy (%) of the C100-B0-S5, C100-B0-S10, C100-B0-S20, and T200-B0-S10 settings.

1. More Comparison

We further evaluate our method the T200-B0-S10 setting using the Tiny-ImageNet dataset. We set the size of the memory buffer as 200. Results are shown in Tab. 1. Our method achieves good performance again.

2. Anaylsis of Submanifold Pool

In our method, we constructed the submanifold pool before training. We first pre-define the dimension of constant curvature spaces (CCSs). For simplicity, we then sequentially sample CCSs for the submanifold pool. Take CCSs with the dimension of 16 as an example, the first CCS uses dimensions $1-16$, and the second one uses dimensions $17-32$. Here, we evaluate the performance of our method with different numbers of CCSs for the submanifold pool. Concretely, we conduct the following settings.

(1) We sample CCSs with the dimension of 16. Thus, there are $\frac{512}{16}=32$ CCSs in the submanifold pool totally.

(2) We sample CCSs with the dimension of 32. Thus, there are $\frac{512}{32}=16$ CCSs in the submanifold pool totally.

(3) We sample CCSs with the dimension of 64. Thus, there are $\frac{512}{64}=8$ CCSs in the submanifold pool totally.

(4) We sample CCSs with the dimension of128. Thus, there are $\frac{512}{128}=4$ CCSs in the submanifold pool totally.

(5) We sample CCSs with the dimension of 256. Thus, there are $\frac{512}{256}=2$ CCSs in the submanifold pool totally.

(6) We sample CCSs with the dimension of 128 and 256. Thus, there are $\frac{512}{128}+\frac{512}{256}=6$ CCSs in the submanifold pool totally.

(7) We sample CCSs with the dimension of 64, 128, and 256. Thus, there are $\frac{512}{64}+\frac{512}{128}+\frac{512}{256}=14$ CCSs in the submanifold pool totally.

(8) We sample CCSs with the dimension of 32, 64, 128, and 256. Thus, there are $\frac{512}{32}+\frac{512}{64}+\frac{512}{128}+\frac{512}{256}=30$ CCSs in the submanifold pool totally.

(9) We sample CCSs with the dimension of 16, 32, 64, 128, and 256. Thus, there are $\frac{512}{16}+\frac{512}{32}+\frac{512}{64}+\frac{512}{128}+\frac{512}{256}=62$ CCSs in the submanifold pool totally.

We conduct experiments using the CIFAR-100 dataset, on the C100-B50-S5, C100-B50-S10, and C100-B40-S20 settings. Results are shown in Tab. 2. d' denotes the used dimensions of CSSs, and m means the number of CSSs in the submanifold pool. We observe that diverse CCSs in the submanifold pool lead to better performance.

3. Hyperparameter Analysis

In this section, we further evaluate the trade-off hyperparameters λ_1 and λ_2 for the angular-regularization loss and the neighbor-robustness loss. Due to the page limitation, we simply set $\lambda_1=\lambda_2$ in the manuscript. Here, we conduct more experiments to evaluate λ_1 and λ_2 in the range of

* Corresponding authors: Chen Xu and Yuwei Wu.

Method	C100-B50-S5	C100-B50-S10	C100-B40-S20
$d' = 16, m = 32$	45.87	52.68	53.34
$d' = 32, m = 16$	44.8	53.07	54.17
$d' = 64, m = 8$	44.54	52.27	52.03
$d' = 128, m = 4$	44.52	53.61	53.48
$d' = 256, m = 2$	45.74	51.95	53.22
$d' = 128, 256, m = 6$	46.77	53.5	55.10
$d' = 64, 128, 256, m = 14$	47.26	53.86	54.75
$d' = 32, 64, 128, 256, m = 30$	48.41	53.97	55.46
$d' = 16, 32, 64, 128, 256, m = 62$	**56.03**	**54.31**	**49.32**

Table 2. Anaylsis of submanifold pool on the C100-B50-S5, C100-B50-S10, and C100-B40-S20 settings.

λ_1 \ λ_2	0.001	0.01	0.1	1	10
0.001	51.8	52.37	52.6	51.76	47.51
0.01	52.53	52.87	52.70	53.20	51.34
0.1	52.72	53.41	52.70	53.50	53.28
1	55.19	54.64	**56.03**	54.34	50.59
10	52.23	53.25	53.01	52.90	52.16

Table 3. Accuracy (%) of the C100-B50-S5 setting.

λ_1 \ λ_2	0.001	0.01	0.1	1	10
0.001	49.51	50.42	50.44	50.86	48.46
0.01	51.4	51.43	50.16	50.81	48.74
0.1	53.07	51.82	51.94	52.47	52.09
1	53.74	53.89	**54.31**	54.29	51.99
10	50.95	51.53	51.38	51.30	51.37

Table 4. Accuracy (%) of the C100-B50-S10 setting.

$[0.001, 0.01, 0.1, 1, 10]$ on the C100-B50-S10 setting. Results are shown in Tab. 3 and Tab. 4. The best performance is achieved when $\lambda_1 = 1$ and $\lambda_2 = 0.1$.

4. Visualization

In this section, we visualize the geometric structures and the mixed-curvature space in the continual learning process. Some examples are shown in Fig. 1, Fig. 2, and Fig. 3. We observe that our method can well preserve geometric structures of data, and prevent forgetting of old data.

5. Selected Submanifolds

In this section, we show selected submanifolds in the C100-B50-S10 setting, including the number, curvature, and dimension of selected submanifolds. As shown in Section 5.1 of the manuscript, the dimensions of CCSs in the submanifold pool are 16, 32, 64, 128, and 256, and there are 62 submanifolds totally. In the 11 steps of the C100-B50-S10 setting (the first step is used to pre-train the model, and there are 10 following steps), the mixed-curvature spaces in our method are constructed in Tab. 5, where K^+ denotes a positive curvature and K^- denotes a negative curvature. We observe an interesting phenomenon. In the beginning, few data is provided, and CCSs of low-dimension with positive curvatures are selected. Then, when more and more data comes, CCSs of high-dimension with negative curvatures are selected. This may show that few data tends to have cyclical structures, while more data tends to have complex hierarchical structures.

6. Hyperparameter Analysis

We conduct experiments to further evaluate the proposed two loss functions that alleviates catastrophic forgetting by preserving geometric structures. They have two trade-off hyperparameters that need to be tuned: λ_1 for the angular-regularization loss and λ_2 for the neighbor-robustness loss. In implementation, we set $\lambda_1 = \lambda_2$ and tune them in the range of $[0.001, 0.01, 0.1, 1, 10]$. We conduct the experiment on the CIFAR-100 dataset, and report the mean of accuracies of all data, old data, and new data over all steps in the data stream. Results are shown in Tab. 6. We observe that with the increase of λ_1 and λ_2, the accuracy of total data first increases and then decreases, and a good balance is achieved when λ_1 and λ_2 are set around 1. Larger λ_1 and λ_2 lead to higher accuracy on the old data, while they may decrease the performance of new data.

We compare the scheme of preserving geometric structures with conventional used regularization schemes, *i.e.*, preserving the representation unchanged [6,9]. In doing so, we replace the two geometric structure preserving loss functions with a representation preserving loss function. We apply a trade-off hyperparameter λ to the loss function and tune λ to achieve the best performance. Experimental results in Tab. 7 shows the effectiveness of the proposed two geometric structure preserving loss functions.

References

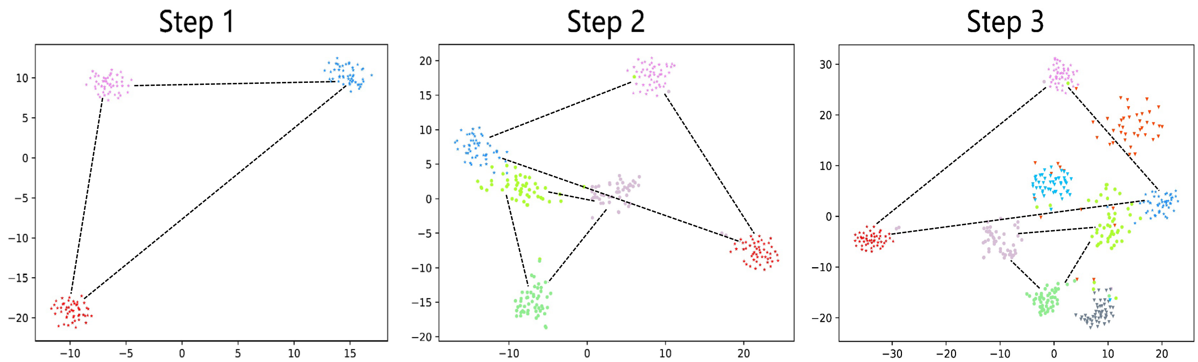
[1] Jihwan Bang, Heesu Kim, Young Joon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8214–8223, 2021. 1

[2] Ari S. Benjamin, David Rolnick, and Konrad P. Körding. Measuring and regularizing networks in function space. In *International Conference on Learning Representations (ICLR)*, 2019. 1

[3] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1

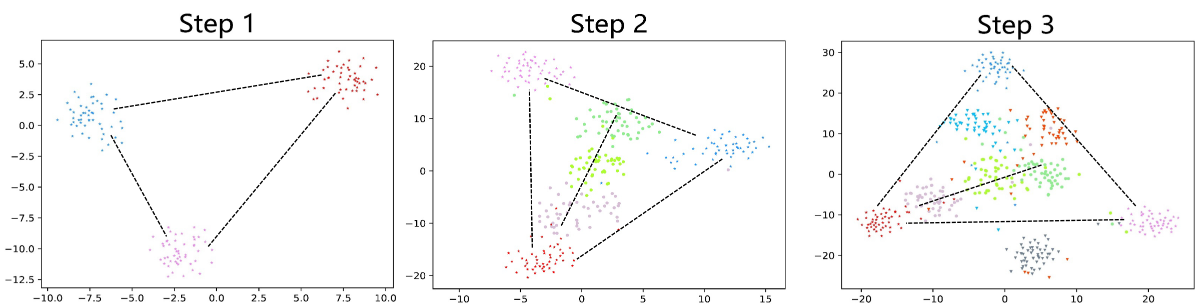
[4] Pietro Buzzega, Matteo Boschini, Angelo Porrello, and Simone Calderara. Rethinking experience replay: a bag of tricks for continual learning. *International Conference on Pattern Recognition (ICPR)*, pages 2180–2187, 2021. 1

[5] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-



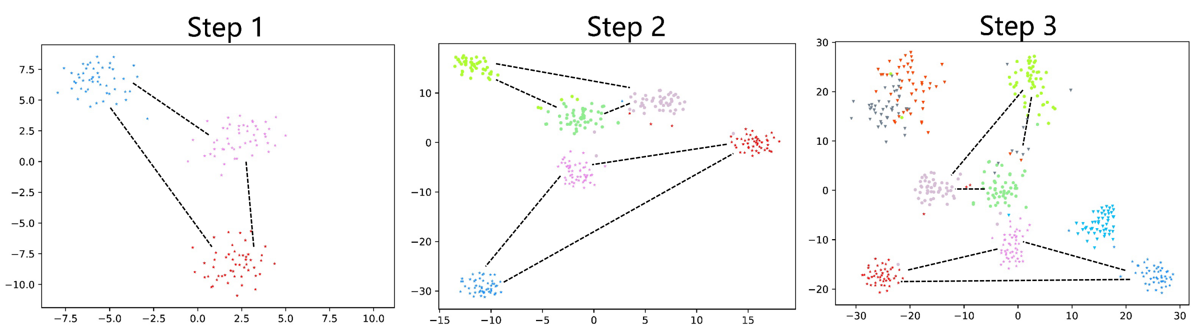
(a) Geometric structure of the first task. (b) Geometric structure of the second task. (c) Geometric structure of the third task.

Figure 1. An example of structures of data in the CIFAR-100 dataset.



(a) Geometric structure of the first task. (b) Geometric structure of the second task. (c) Geometric structure of the third task.

Figure 2. An example of structures of data in the CIFAR-100 dataset.



(a) Geometric structure of the first task. (b) Geometric structure of the second task. (c) Geometric structure of the third task.

Figure 3. An example of structures of data in the CIFAR-100 dataset.

Step	Selected constant curvature spaces
Step 1	$30 \times \mathbb{C}_{K+}^{16}$
Step 2	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 3 \times \mathbb{C}_{K-}^{32}$
Step 3	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 1 \times \mathbb{C}_{K+}^{32}, 7 \times \mathbb{C}_{K-}^{32}$
Step 4	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 1 \times \mathbb{C}_{K+}^{32}, 9 \times \mathbb{C}_{K-}^{32}$
Step 5	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 1 \times \mathbb{C}_{K+}^{32}, 9 \times \mathbb{C}_{K-}^{32}, 2 \times \mathbb{C}_{K-}^{64}, 1 \times \mathbb{C}_{K+}^{128}$
Step 6	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 1 \times \mathbb{C}_{K+}^{32}, 9 \times \mathbb{C}_{K-}^{32}, 2 \times \mathbb{C}_{K-}^{64}, 1 \times \mathbb{C}_{K+}^{128}, 2 \times \mathbb{C}_{K+}^{256}$
Step 7	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 1 \times \mathbb{C}_{K+}^{32}, 9 \times \mathbb{C}_{K-}^{32}, 1 \times \mathbb{C}_{K+}^{64}, 2 \times \mathbb{C}_{K-}^{64}, 1 \times \mathbb{C}_{K+}^{128}, 2 \times \mathbb{C}_{K+}^{256}$
Step 8	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 2 \times \mathbb{C}_{K+}^{32}, 9 \times \mathbb{C}_{K-}^{32}, 1 \times \mathbb{C}_{K+}^{64}, 3 \times \mathbb{C}_{K-}^{64}, 1 \times \mathbb{C}_{K+}^{128}, 2 \times \mathbb{C}_{K+}^{256}$
Step 9	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 2 \times \mathbb{C}_{K+}^{32}, 10 \times \mathbb{C}_{K-}^{32}, 1 \times \mathbb{C}_{K+}^{64}, 3 \times \mathbb{C}_{K-}^{64}, 1 \times \mathbb{C}_{K+}^{128}, 2 \times \mathbb{C}_{K+}^{256}$
Step 10	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 2 \times \mathbb{C}_{K+}^{32}, 10 \times \mathbb{C}_{K-}^{32}, 1 \times \mathbb{C}_{K+}^{64}, 4 \times \mathbb{C}_{K-}^{64}, 1 \times \mathbb{C}_{K+}^{128}, 2 \times \mathbb{C}_{K+}^{256}$
Step 11	$31 \times \mathbb{C}_{K+}^{16}, 1 \times \mathbb{C}_{K-}^{16}, 2 \times \mathbb{C}_{K+}^{32}, 10 \times \mathbb{C}_{K-}^{32}, 1 \times \mathbb{C}_{K+}^{64}, 4 \times \mathbb{C}_{K-}^{64}, 1 \times \mathbb{C}_{K+}^{128}, 2 \times \mathbb{C}_{K+}^{256}$

Table 5. Selected CCSs of 11 steps in the C100-B50-S10 setting. $30 \times \mathbb{C}_{K+}^{16}$ means 30 8-dimension CCS with positive curvatures.

Method	C100-B50-S5			C100-B50-S10			C100-B40-S20		
	Total acc	Old acc	New acc	Total acc	Old acc	New acc	Total acc	Old acc	New acc
$\lambda_1 = \lambda_2 = 0.001$	61.35	62.92	50.70	60.81	61.26	55.18	58.62	58.20	64.22
$\lambda_1 = \lambda_2 = 0.01$	61.52	63.45	49.02	61.08	61.68	53.54	59.22	59.05	63.20
$\lambda_1 = \lambda_2 = 0.1$	62.04	64.08	45.98	62.92	63.71	51.88	60.44	60.45	61.30
$\lambda_1 = \lambda_2 = 1$	63.18	66.03	44.06	64.01	65.82	39.78	59.75	61.08	35.25
$\lambda_1 = \lambda_2 = 10$	61.56	65.56	35.90	63.43	66.10	28.64	58.66	60.24	27.73

Table 6. Hyperparameter analysis (λ_1 and λ_2) of the proposed two loss functions. 'Total acc', 'Old acc', and 'New acc' denote the accuracies of all data, old data, and new data, respectively. Results are obtained by averaging accuracies of all steps in the data stream.

Method	C100-B50-S5	C100-B50-S10	C100-B40-S20
PR	53.87	52.86	47.63
PGS	**56.03**	**54.31**	**49.32**

Table 7. Comparisons between loss functions for preserving geometric structures (PGS) and preserving representation (PR).

gem. In *International Conference on Learning Representations (ICLR)*, 2019. 1

[6] Arthur Douillard, Alexandre Ramé, Guillaume Couairon, and Matthieu Cord. Dytox: Transformers for continual learning with dynamic token expansion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9275–9285, 2022. 2

[7] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, G. Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5533–5542, 2017. 1

[8] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *International Conference on Learning Representations (ICLR)*, 2019. 1

[9] Fei Zhu, Xu-Yao Zhang, Chuan Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5867–5876, 2021. 2