Supplementary Material for MICS: Midpoint Interpolation to Learn Compact and Separated Representations for Few-Shot Class-Incremental Learning

1. Implementation Details

Similarity metric: For all experiments, we used cosine similarity as a similarity metric. During base training stage, τ was set as 1/32 for miniImageNet and 1/16 for the others. During incremental session, τ is set as 1/16 for all datasets.

Optimizer and learning rate: We used SGD with momentum 0.9 and weight decay 5×10^{-4} for all sessions. For the miniImageNet case, we used a learning rate of 0.1 with a cosine annealing scheduler and a batch size of 128 for 700 epochs during the base training stage. During the incremental sessions, we fixed the learning rate to 0.5 and trained for 5 epochs with $\epsilon = 0.3$ for each session. For the CIFAR-100 dataset, we used a learning rate of 0.1 with a cosine annealing scheduler and a batch size of 256 for 600 epochs during the base training stage. During the incremental sessions, we fixed the learning rate to 0.0005 and trained for 10 epochs with $\epsilon = 0.01$ for each session. For the CUB-200-2011 case, we used a learning rate of 0.001 for the feature extractor and 0.01 for the base classifiers. Because we used the ImageNet pre-trained model for the CUB-200-2011 dataset, we had to choose a small learning rate for the feature extractor. We trained the model for 2,000 epochs with 256 batch-size. For the CUB-200-2011 case, it is shown to be helpful to train the model with conventional supervised training based on cross-entropy loss for the first 150 epochs, i.e., a warm-up process. After the warm-up process, we train the model with the MICS objective. We used a step-wise learning rate scheduler and learning rate decaying at [1000, 1500] epochs with a decaying factor of 0.1 during the base training stage. During the incremental sessions, we fixed the learning rate to 0.1 and trained for 20 epochs with $\epsilon = 0.3$ for each session.

Selection of mixup layers: For the selection of the mixup layer l, we borrowed the strategy of [10], which firstly proposes Manifold mixup. Specifically, we set the eligible layers, i.e., where the manifold mixup can take place, and then selected one layer randomly from the eligible layers for each optimization step. We select the input layer as an eligible layer. Also, the input of the down-sample layer for each residual block is selected as an eligible layer.

Implementation of FACT without AutoAugment: For the re-implemenation of FACT without strong autmentation (denoted as FACT*), we referred to the experimental options specified in the original paper [15]. In all experiments, we used cosine similarity with $\tau = 1/16$ and trained the model for 600 epochs with 256 batch-size. We optimized the model using SGD with momentum 0.9 and weight decay 5×10^{-4}. We changed η from $\{0.5, 0.9, 1.0\}$ for incremental inference and α from $\{0.5, 2\}$ for sampling λ. In the case of $\eta = 1.0, \alpha = 0.5$, and the trade-off parameter 0.01, we obtained the best results. For the CIFAR-100 and miniImageNet datasets, we used a cosine annealing scheduler with a learning rate 0.1. For CUB-200-2011, it was helpful to use a small learning rate, so we used a step-wise learning rate scheduler with a learning rate 0.005 and learning rate decaying at [50, 100, 150, 200, 250, 300] epochs with decaying factor 0.25.

Additional implementation details: There is another experimental option that is necessary for the data loader: *'drop last'*. The data loader with the 'drop last' option drops the last non-full batch of the dataset so that we can get exactly the same batch size for all optimization steps. If we do not use this option for MICS, N_o of the last iteration of each epoch greatly differs from N_o of the other iterations so that training becomes unstable. We present all performance results on three datasets, i.e., miniImageNet, CIFAR-100, and CUB-200-2011.

Performance Comparisions: We evaluated our method MICS for all three datasets as shown in Tables 6, 7, and 8 for CIFAR-100, miniImageNet, and CUB-200-2011 benchmark, respectively. The results confirm that MICS achieves outperforming performance. In miniImageNet and CUB-200-2011, MICS shows remarkable gains beyond the runner-up methods, i.e., +2.43% and +1.27%, respectively. For CUB-200-2011, MICS is slightly lower than

			CIFAR-100		miniImageNet		CUB-200-2011	
Mixup	**Soft Labeling**	**Midpoint**	Final accuracy	nVAR ↓	Final accuracy	nVAR ↓	Final accuracy	nVAR ↓
✓	✗	✗	51.59%	479.3	2.07%	1.691×10^4	60.98%	100.7
✓	✓	✗	51.59%	476.4	1.34%	1.746×10^5	61.24%	101.911
✓	✓	✓	**52.94%**	**463.5**	**60.74%**	**421.4**	**61.37%**	**100**

Table 1. Ablations for the components of MICS

	CIFAR-100		miniImageNet		CUB-200-2011	
Mixup Method	Final accuracy	nVAR ↓	Final accuracy	nVAR ↓	Final accuracy	nVAR ↓
Mixup [14]	51.21%	468.4	58.77%	456.6	60.99%	101.4
CutMix [12]	50.64%	**435.5**	56.74%	491.5	61.01%	101.5
Manifold mixup (MICS)	**52.94%**	463.5	**60.74%**	**421.4**	**61.37%**	**100**

Table 2. MICS with various mixup methods.

	miniImageNet	
Labeling Policy	$\epsilon = 0$	$\epsilon = 0.3$
Gaussian (with 1(b))	58.82%	40.66%
Exponential (with 1(c))	59.91%	4.62%
MICS (with 1(a))	**60.54%**	**60.74%**

Table 3. MICS with various labeling policies

NC-FSCIL but it shows the best PD performance.

Effectiveness of MICS Components: To evaluate the effectiveness of MICS components, i.e., Manifold mixup, Soft labeling policy, and Midpoint classifier, we conducted ablation experiments on all three datasets. Table 1 presents a comparative analysis of the results obtained by adding these components one by one. The results indicate that MICS with all components show the best performance in accuracy and nVAR.

Mixup Methods: We evaluated the performance of the Mixup Method, including Mixup of [14], Cutmix of [12], and Manifold mixup of [10], on all three datasets. Table 2 shows that MICS performs best when combined with the Manifold mixup method.

Label Mixing Policy: Mixup adopts a particular labeling policy presented in Fig. 1(a). We test other possible labeling policies by using a smooth function. Fig. 1 shows labeling functions based on (b) Gaussian and (c) Exponential functions. We test these two smooth labeling functions by substituting the original labeling function of MICS. As shown in Table 3, two variants denoted as 'Gaussian' and 'Exponential' perform worse than MICS. Specifically, they are slightly inferior to the MICS case when the feature extractor is frozen during the incremental sessions, i.e., $\epsilon = 0$ case. However, they suffer from severe catastrophic forgetting with fine-tuning, $\epsilon = 0.3$. We conjecture that the bell-like and concave shapes of the two labeling functions assign an excessive probability value to the virtual class so that it results in the forgetting of the past classes

2. MICS with Strong Augmentations

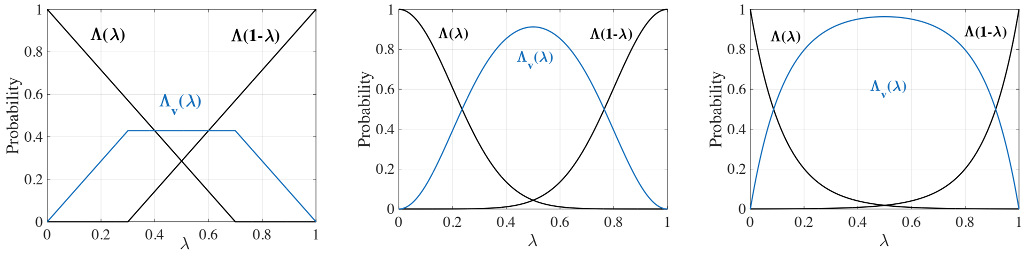
MICS with AutoAugment: AutoAugment of [1] is a learning based augmentation policy searching algorithm. It requires thousands of GPU time, even for a small dataset such as CIFAR-100, but it is powerful for image recognition tasks. The original experiments for FACT of [15] user AutoAugment as a default. Therefore, we drop the augmentation for a fair comparison in the main tables. We also tested MICS with AutoAugment. In Table 5, MICS with AutoAugment, which is denoted as MICS+AA, shows better performance than FACT with AutoAugment. The results confirm that our method still performs FACT regardless of the AutoAugment.

3. Influence of Backbone Architecture

Existing FSCIL methods, including TOPIC, F2M, CEC, and FACT, utilize the ResNet architecture proposed in [2]. However, there are two key differences in the ResNet architectures used by ALICE of [6] compared to these FSCIL methods.

For the CIFAR-100 experiment, ALICE in [6] uses ResNet-18 (11M) instead of ResNet-20 (0.27M), resulting in 40 times more parameters than the existing FSCIL methods. For a fair comparison of the CIFAR-100 case, we tried to re-implement ALICE with ResNet-20, but it deteriorated the FSCIL performance significantly. Thus, we exclude the results of ALICE in Table 6.

Additionally, for CIFAR-100 and miniImageNet, ALICE selects different ResNet-18 hyper-parameters than those proposed in [6]. Specifically, ALICE modifies the kernel size and stride of the first convolution layer from 7 to 3 and from 2 to 1, respectively, compared to the original ResNet-18, and removes the max-pooling layer. Surprisingly, the



(a) MICS (b) Gaussian (c) Exponential

Figure 1. Label mixing policies

Method	Accuracy in each session (%)									PD ↓
	0	1	2	3	4	5	6	7	8	
ALICE† [6]	71.65	65.48	61.74	58.09	55.16	52.58	49.90	48.25	47.05	**24.60**
MICS (Ours)	**78.72**	**73.75**	**69.41**	**65.91**	**62.59**	**59.19**	**56.26**	**54.18**	**52.24**	26.58

Table 4. Comparision results on miniImageNet using the original ResNet-18 architecture proposed in [2]. We re-implementation ALICE (denoted as ALICE†) because the results with the original ResNet-18 on miniImageNet are not givien in [6]

Method	CIFAR-100	miniImageNet	CUB-200-2011
FACT [15]	52.10%	50.49%	56.94%
MICS+AA	**53.80%**	**52.91%**	**58.22%**

Table 5. Last session accuracies with AutoAugment(AA)

FSCIL performance on miniImageNet improves when using this modified ResNet-18, even though it contains fewer parameters than the original ResNet-18. Thus, we also adopted the modified ResNet-18 architecture as ALICE for the miniImageNet experiments. We followed the experimental options specified in [6], but modified the learning rate, scale factor, cosine margin, and testing method based on our findings. Specifically, we obtained the best results with a learning rate of 0.1, a scale factor of $s = 16$, a cosine margin of $m = 0$, and without balanced testing, i.e., using all base training images for calculating base class classifiers (or prototypes). For a fair comparison, we re-implemented ALICE using the original ResNet-18 in Table 4 (denoted as ALICE†) to exclude any effects of changed kernel size, stride, and the existence of max-pooling layers. Despite these modifications, the results in Table 4 show that MICS still outperforms ALICE for the last session accuracies on miniImageNet.

4. Boundary Thickness

Let us remind the definition of the normalized boundary thickness in representation space For a C-way classification task with input, output and representation space, i.e., $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{H}$, respectively, let us denote the embedding function as $g(x) : \mathcal{X} \rightarrow \mathcal{H}$ and the prediction function as $f(h) : \mathcal{H} \rightarrow [0, 1]^C$.

Definition 1. *(Normalized Boundary Thickness in Representation Space) For $\alpha \in (0, 1)$, the boundary thickness $\Theta(f, \alpha)$ in representation space $\mathcal{H}$ is defined as follows:*

$$\Theta(f, \alpha) = \mathbb{E}_{(x_i, x_j)} \left[\int_0^1 \mathbf{I}\{|\Delta_{ij} f(h^*_{ij})| < \alpha\} d\lambda \right], \quad (1)$$

where $\mathbf{I}\{\cdot\}$ is an indicator function and $\Delta_{ij} f(h) = f(h)_i - f(h)_j$ is the gap between the probabilities for classifying embedded feature h to class i and j. Also, $h^*_{ij} = \lambda h_i + (1 - \lambda) h_j$, where $h_i = g(x_i)$ and $h_j = g(x_j)$.

Based on the definition, our theorem is as follows:

Theorem 1. *For all $\alpha \in (0, 1)$, MICS achieves larger normalized boundary thickness than Manifold mixup, i.e., $\Theta(f_{Mixup}, \alpha) \leq \Theta(f_{MICS}, \alpha, \Lambda)$, when the following holds:*

$$\lambda - \Lambda(1 - \lambda) \geq 1 - \lambda - \Lambda(\lambda). \quad (2)$$

▷ ***Proof:*** Let us assume that a representation for the Manifold mixup method with an arbitrarily small loss. For the learned Manifold mixup-based representation, the normalized boundary thickness in the feature space, i.e.,

$\Theta(f_{\text{Mixup}}, \alpha)$ can be computed as follows:

$$\Theta(f_{\text{Mixup}}, \alpha) = \mathbb{E}_{(x_i,x_j)} \left[\int_0^1 \mathbf{I}\{|\Delta_{ij} f(h_{ij}^*)| < \alpha\} d\lambda \right] \tag{3}$$

$$= \mathbb{E}_{(x_i,x_j)} \left[\int_0^1 \mathbf{I}\{|f_i(h_{ij}^*) - f_j(h_{ij}^*)| < \alpha\} d\lambda \right] \tag{4}$$

$$\overset{(a)}{=} \mathbb{E}_{(x_i,x_j)} \left[\int_0^1 \mathbf{I}\{|(1-\lambda) - \lambda| < \alpha\} d\lambda \right] \tag{5}$$

$$= \mathbb{E}_{(x_i,x_j)} \int_{\frac{1-\alpha}{2}}^{\frac{1+\alpha}{2}} 1 d\lambda = \alpha. \tag{6}$$

The equality (a) is from the definition of soft-labeling policy of Manifold mixup with assuming the number of layers of the representation that maps input space $\mathcal{X}$ to the representation space $\mathcal{H}$ is asymptotically increased, i.e., arbitrary small loss can be achieved by Manifold mixup.

Let us assume a representation with an arbitrarily small loss for the MICS-based labeling function. For MICS, which considers a virtual class from class mixup between a pair of original classes, the soft-labeling of the original class becomes $\Lambda(\lambda)$ and $\Lambda(1-\lambda)$ for $0 \leq \lambda \leq 1$. Therefore, the class with probability λ for Manifold mixup decreases to $\Lambda(1-\lambda)$ for MICS, where the gap is $\Delta_u = \lambda - \Lambda(1-\lambda)$, and another class with probability $1-\lambda$ for Manifold mixup decreases to $\Lambda(\lambda)$ for MICS, where the gap is $\Delta_v = 1 - \lambda - \Lambda(\lambda)$. When $\Delta_u \geq \Delta_v$, the probability gap between two classes always decreases than MICS, which completes the proof. In this case, we do not consider the probability of the virtual class in computing the boundary thickness which strongly makes the thickness of MICS to be larger than that of Manifold mixup.

Corollary 1. *For all $\alpha \in (0,1)$, MICS with linear function $\Lambda(\cdot)$ shows larger normalized boundary thickness than Manifold mixup when the following holds:* $\gamma \geq 0.25$.
▷ ***Proof:*** For MICS, $\Delta_{ij} f(h_{ij}^*)$ is formulated as follows:

$$\Delta_{ij} f(h_{ij}^*) = \Lambda(\lambda) - \max\left[\Lambda(1-\lambda), 1 - \Lambda(\lambda) - \Lambda(1-\lambda)\right]. \tag{7}$$

By taking the definition of $\Lambda(\lambda)$ for MICS in the main paper,

$$\Delta_{ij} f(h_{ij}^*) = \max\big(\frac{(1-\lambda-\gamma)}{(1-\gamma)}, 0\big) \\ - \max\Big(\max\big(\frac{(\lambda-\gamma)}{(1-\gamma)}, 0\big), 1 - \max\big(\frac{(1-\lambda-\gamma)}{(1-\gamma)}, 0\big) - \max\big(\frac{(\lambda-\gamma)}{(1-\gamma)}, 0\big). \tag{8}$$

On the other hand, Manifold mixup shows $\Delta_{ij} f(h_{ij}^*) = 1 - 2\lambda$. When the MICS's eq. (8) become smaller than that of Manifold mixup, MICS shows enlarged boundary thickness. Here, we take the probability of virtual class into account when computing the confidence difference, which is for achieving the weak condition for the larger thickness. When focusing a single side with $\lambda < 0.5$ due to the symmetricity, to making eq. (8) smaller than Manifold mixup with $1 - 2\lambda$ value, γ should be larger than 0.25.

Method	**Accuracy in each session (%)**									**PD ↓**
	0	1	2	3	4	5	6	7	8	
Finetune	69.05	41.86	3.79	4.19	3.85	3.33	2.14	2.31	2.06	66.99
Baseline	69.05	65.52	61.66	57.84	55.04	52.53	50.97	49.16	47.35	21.7
Rebalance [3]	64.10	53.05	43.96	36.97	31.61	26.73	21.23	16.78	13.54	50.56
iCaRL [7]	64.10	53.28	41.69	34.13	27.93	25.06	20.41	15.48	13.73	50.37
TOPIC [9]	64.10	55.88	47.07	45.16	40.11	36.38	33.96	31.55	29.37	34.73
FSLL [5]	64.10	55.85	51.71	48.59	45.34	43.25	41.52	39.81	38.16	25.94
CEC [13]	73.07	68.88	65.26	61.19	58.09	55.57	53.22	51.34	49.14	23.93
F2M [8]	71.45	68.1	64.43	60.8	57.76	55.26	53.53	51.57	49.35	**22.10**
FACT* [15]	73.68	68.39	63.91	59.94	56.17	53.24	50.6	48.14	45.91	27.77
CLOM [16]	74.20	69.83	66.17	62.39	59.26	56.48	54.36	52.16	50.25	23.95
NC-FSCIL [11]	**82.52**	**76.82**	**73.34**	**69.68**	**66.19**	**62.85**	**60.96**	**59.02**	**56.11**	26.41
WaRP [4]	80.31	75.86	71.87	67.58	64.39	61.34	59.15	57.10	54.74	25.57
MICS (Ours)	78.18	73.49	68.97	65.01	62.25	59.34	57.31	55.11	52.94	25.24

Table 6. The evaluation for the FSCIL benchmark with the CIFAR-100 for 5-way 5-shot setting. MICS uses $\epsilon = 0.01$. * indicates FACT of [15] without AutoAugment of [1] for a fair comparison.

Method	**Accuracy in each session (%)**									**PD ↓**
	0	1	2	3	4	5	6	7	8	
Finetune	69.37	47.41	11.00	8.48	4.79	6.00	4.59	5.41	6.72	62.65
Baseline	69.37	64.34	60.33	57.23	54.18	51.35	48.87	47.08	45.56	23.81
Rebalance [3]	61.31	47.80	39.31	31.91	25.68	21.35	18.67	17.24	14.17	47.14
iCaRL [7]	61.31	46.32	42.94	37.63	30.49	24.00	20.89	18.80	17.21	44.10
TOPIC [9]	61.31	50.09	45.17	41.16	37.48	35.52	32.19	29.46	24.42	36.89
FSLL [5]	66.48	61.75	58.16	54.16	51.10	48.53	46.54	44.20	42.28	24.20
CEC [13]	72.00	66.83	62.97	59.43	56.70	53.73	51.19	49.24	47.63	24.37
F2M [8]	72.05	67.47	63.16	59.70	56.71	53.77	51.11	49.21	47.84	24.21
FACT* [15]	71.78	66.54	62.39	58.96	55.80	52.65	49.82	47.78	45.80	25.98
CLOM [16]	73.08	68.09	64.16	60.41	57.41	54.29	51.54	49.37	48.00	25.08
NC-FSCIL [11]	84.02	76.80	72.00	67.83	66.35	64.04	61.46	59.54	58.31	25.71
WaRP [4]	72.99	68.10	64.31	61.30	58.64	56.08	53.40	51.72	50.65	**22.34**
ALICE [6]	80.6	70.6	67.4	64.5	62.5	60.0	57.8	56.8	55.7	24.9
MICS (Ours)	**84.40**	**79.48**	**75.09**	**71.40**	**68.89**	**66.16**	**63.57**	**61.79**	**60.74**	23.66

Table 7. The evaluation for the FSCIL benchmark with the miniImageNet dataset for 5-way 5-shot setting. MICS uses $\epsilon = 0.3$. * indicates FACT of [15] without AutoAugment of [1] for a fair comparison.

Method	**Accuracy in each session (%)**											**PD ↓**
	0	1	2	3	4	5	6	7	8	9	10	
Finetune	77.55	8.21	10.58	6.06	5.49	5.39	4.81	3.69	2.77	2.68	2.64	74.91
Baseline	68.68	52.65	48.61	44.16	36.62	29.52	27.83	26.26	24.01	23.89	21.16	47.52
Rebalance [3]	68.68	57.12	44.21	28.78	26.71	25.66	24.62	21.52	20.12	20.06	19.87	48.81
iCaRL [7]	68.68	52.65	48.61	44.16	36.62	29.52	27.83	26.26	24.01	23.89	21.16	47.52
TOPIC [9]	68.88	62.49	54.81	49.99	45.25	41.40	38.35	35.36	32.22	28.31	26.28	42.6
FSLL [5]	72.77	69.33	65.51	62.66	61.1	58.65	57.78	57.26	55.59	55.39	54.21	18.56
CEC [13]	75.85	71.94	68.50	63.50	62.43	58.27	57.73	55.81	54.83	53.52	52.28	23.57
F2M [8]	77.13	73.92	70.27	66.37	64.34	61.69	60.52	59.38	57.15	56.94	55.89	21.24
FACT* [15]	**79.83**	74.59	71.10	66.26	66.33	62.93	62.09	61.21	58.88	58.33	57.24	22.59
CLOM [16]	79.57	76.07	72.94	**69.82**	67.8	65.56	63.94	62.59	60.62	60.34	59.58	19.99
NC-FSCIL [11]	80.45	75.98	72.30	70.28	**68.17**	65.16	64.43	63.25	60.66	60.01	59.44	21.01
WaRP [4]	77.74	74.15	70.82	66.90	65.01	62.64	61.40	59.86	57.95	57.77	57.01	20.73
ALICE [6]	77.4	72.7	70.6	67.2	65.9	63.4	62.9	61.9	60.5	60.6	60.1	**17.3**
MICS (Ours)	78.77	**75.37**	**72.30**	68.72	67.45	**65.40**	**64.72**	**63.39**	**61.89**	**61.89**	**61.37**	17.40

Table 8. The evaluation for the FSCIL benchmark with the CUB-200-2011 for 10-way 5-shot setting. MICS uses $\epsilon = 0.3$. * indicates FACT of [15] without AutoAugment of [1] for a fair comparison.

References

[1] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019. 2, 5, 6

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 2, 3

[3] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019. 5, 6

[4] Do-Yeon Kim, Dong-Jun Han, Jun Seo, and Jaekyun Moon. Warping the space: Weight space rotation for class-incremental few-shot learning. In *The Eleventh International Conference on Learning Representations*, 2023. 5, 6

[5] Pratik Mazumder, Pravendra Singh, and Piyush Rai. Few-shot lifelong learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2337–2345, 2021. 5, 6

[6] Can Peng, Kun Zhao, Tianren Wang, Meng Li, and Brian C Lovell. Few-shot class-incremental learning from an open-set perspective. In *European Conference on Computer Vision*, pages 382–397, 2022. 2, 3, 5, 6

[7] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 5, 6

[8] Guangyuan Shi, Jiaxin Chen, Wenlong Zhang, Li-Ming Zhan, and Xiao-Ming Wu. Overcoming catastrophic forgetting in incremental few-shot learning by finding flat minima. *Advances in Neural Information Processing Systems*, 34:6747–6761, 2021. 5, 6

[9] Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12183–12192, 2020. 5, 6

[10] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR, 2019. 1, 2

[11] Yibo Yang, Haobo Yuan, Xiangtai Li, Zhouchen Lin, Philip Torr, and Dacheng Tao. Neural collapse inspired feature-classifier alignment for few-shot class incremental learning. *arXiv preprint arXiv:2302.03004*, 2023. 5, 6

[12] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019. 2

[13] Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12455–12464, 2021. 5, 6

[14] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 2

[15] Da-Wei Zhou, Fu-Yun Wang, Han-Jia Ye, Liang Ma, Shiliang Pu, and De-Chuan Zhan. Forward compatible few-shot class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9046–9056, 2022. 1, 2, 3, 5, 6

[16] Yixiong Zou, Shanghang Zhang, Yuhua Li, and Ruixuan Li. Margin-based few-shot class-incremental learning with class-level overfitting mitigation. *arXiv preprint arXiv:2210.04524*, 2022. 5, 6