T-CIL: Temperature Scaling using Adversarial Perturbation for Calibration in Class-Incremental Learning

Supplementary Material

1. Detailed Algorithms

We provide full algorithms of Algorithm 2 (`MagSearch`) for magnitude search, and Algorithm 3 (`TempOpt`) for temperature optimization.

Algorithm 2: The magnitude search algorithm for perturbation (`MagSearch`).

```
Input: Model parameters $\theta = \{w, v\}$, exemplar set
       from new task $\mathcal{M}_{t,new}$, target temperature
       $T_{target}$, set of feature means $\mu$, tolerance $\delta$
$\epsilon_{low} \leftarrow 0.0$, $\epsilon_{high} \leftarrow 1.0$
while $\epsilon_{high} - \epsilon_{low} > \delta$ do
    $\epsilon = \frac{\epsilon_{low} + \epsilon_{high}}{2.0}$
    $\mathcal{M}^{\epsilon}_{t,new} = \{\ \}$
    for $(\mathbf{x}_e, y_e) \in \mathcal{M}_{t,new}$ do
        $y'_e = \arg\max_{c:\mu_c \in \mu, c \neq y_e} \|\phi_v(\mathbf{x}_e) - \mu_c\|$
        $\mathbf{x}^{adv}_e = \mathbf{x}_e - \epsilon\ \text{sign}(\nabla_{\mathbf{x}_e} \mathcal{L}_{CE}(\mathbf{x}_e, y'_e; \theta))$
        $\mathcal{M}^{\epsilon}_{t,new} \leftarrow \mathcal{M}^{\epsilon}_{t,new} \cup \{(\mathbf{x}^{adv}_e, y_e)\}$
    $T = \texttt{TempOpt}(\mathcal{M}^{\epsilon}_{t,new}, \theta)$
    if $T < T_{target}$ then
        $\epsilon_{low} \leftarrow \epsilon$
    else
        $\epsilon_{high} \leftarrow \epsilon$
Output: $\frac{\epsilon_{low} + \epsilon_{high}}{2.0}$
```

Algorithm 3: The temperature optimization algorithm (`TempOpt`).

```
Input: Dataset $\mathcal{D}$, model parameters $\theta$
$T \leftarrow 1$ ;                          // Initialize $T$
while not converge do
    for $(\mathbf{x}, y) \in \mathcal{D}$ do
        Update $T$ to minimize $\mathcal{L}_{CE}(\mathbf{x}, y; \theta, T)$
Output: $T$
```

2. Computational Complexity Analysis

In this section, we analyze the computational complexity of T-CIL. T-CIL consists of four main components: temperature optimization, feature means calculation, perturbation magnitude search, and memory update. The complexity of temperature optimization is $O(M)$, where M represents the memory size, since we optimize the temperature on the set of perturbed exemplars for memory. Similarly, calculating feature means requires $O(M)$ operations. The adversarial search process takes $O(M)$ time (with a constant factor k for perturbation iterations), and applying perturbations also has a complexity of $O(M)$. Therefore, the overall computational complexity of T-CIL for a single incremental task is also $O(M)$.

As conventional class-incremental learning approaches operate with $O(T(N_{\text{new}}+M))$ complexity, T-CIL maintains a lighter $O(M)$ complexity, where T is the number of tasks, N_{new} is the number of new task data points, and M is the memory size. Consequently, integrating T-CIL with existing class-incremental frameworks preserves the asymptotic computational efficiency, as M remains constant and substantially less than N_{new}. With a small overhead, T-CIL is an efficient approach that can be practically combined with existing class-incremental learning methods.

3. Experiments

3.1. Inapplicability of PerturbTS on the CIFAR-10

The reason PerturbTS [37] is not applicable on the CIFAR-10 in a class-incremental learning setup is that, with only two new classes per task, the model quickly fits to the data, making it impossible to achieve the designated accuracy reduction through perturbation. This overfitting prevents the perturbation magnitude optimization from converging.

3.2. Detailed Experimental Settings

To obtain the best calibration performance of T-CIL with the minimal impact on accuracy, we use a new-task validation set whose size varies depending on the class-incremental learning technique and dataset used. The new-task validation set sizes are listed in Table 4. The effect of the new-task validation set size will be explained later.

Class-incremental learning techniques require specific training parameters. For both EEIL [3] and WA [45], we set the knowledge distillation temperature to 2. EEIL and DER [39] incorporate a balanced fine-tuning phase. For this phase, we train the model for 30 epochs with 10 tasks and 100 epochs with 20 tasks.

After model training, we store a subset of new-task data used for training to the memory by uniformly sampling examples from each class. As the memory size is fixed, we remove some existing exemplars to accommodate the new-task data.

Table 4. The size of the new-task validation set for each class-incremental learning method and dataset used in the main experiments.

Method	CIFAR-10	CIFAR-100	Tiny-ImageNet
ER	500	500	100
EEIL	300	300	200
WA	100	500	100
DER	500	1000	100

We make a fair comparison between post-hoc calibration methods including T-CIL versus vanilla class-incremental learning techniques in terms of training data. In particular, whenever we take a (minimal) validation set from the training data, we only train models on the remaining training data. In comparison, the vanilla techniques always train on the full training set.

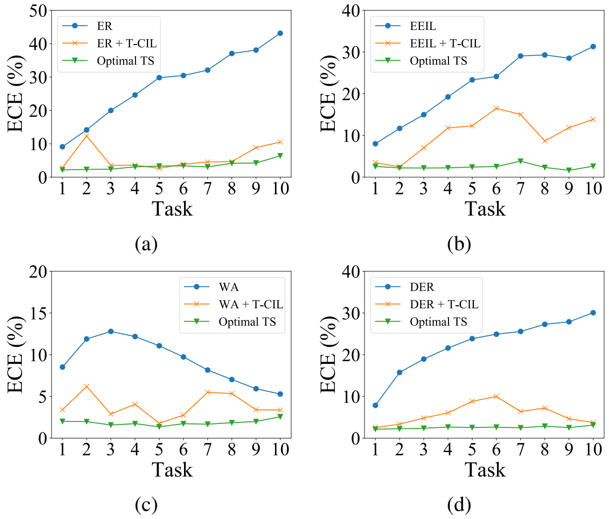
3.3. Additional Experiments

Full experimental results We evaluate T-CIL against five calibration baselines (Cal method) in combination with four class-incremental learning techniques (CIL method) across three datasets. Table 5 presents a comprehensive comparison of all possible combinations between post-hoc calibration methods and class-incremental learning techniques.

Overall, T-CIL outperforms the five calibration baselines when integrated with four existing class-incremental learning techniques across three datasets. Notably, T-CIL consistently shows low calibration errors compared to all the baselines. While PerturbTS achieves the best calibration performances when combined with EEIL and WA on the Tiny-ImageNet, its effectiveness is inconsistent. In addition, PerturbTS is not applicable to the CIFAR-10 dataset and exhibits unusually high calibration errors on the CIFAR-100 dataset. In contrast, T-CIL demonstrates robust and superior performances across all experimental settings, consistently achieving lower calibration errors regardless of the underlying class-incremental learning technique. This comprehensive evaluation validates that T-CIL is a more reliable and versatile approach for addressing calibration challenges in class-incremental learning scenarios.

Expansion of incremental tasks We evaluate the performance when expanding incremental tasks from 10 to 20 tasks, with results presented in Table 6. For all class-incremental learning techniques, we use 100 samples from each new task as a validation set on both CIFAR-100 and Tiny-ImageNet.

The results show that T-CIL outperforms most vanilla class-incremental learning techniques, with WA being the only exception. As explained in Section 6.2, WA scales the output logits corresponding to the new task only after training. This scaling of specific logits may not align with the insight behind T-CIL's perturbation direction policy. Nevertheless, T-CIL significantly improves the calibration performance of poorly calibrated class-incremental learning techniques.



(a) (b) (c) (d)

Figure 6. ECE (%) comparison after training each task on the CIFAR-100 among existing class-incremental learning techniques, their combinations with T-CIL, and the optimal TS. The existing techniques include: (a) ER, (b) EEIL, (c) WA, and (d) DER.

Varying size of memory We analyze how memory size affects the ECE of T-CIL compared to the vanilla method without calibration on the CIFAR-100 and Tiny-ImageNet, as shown in Figure 7. Using ER as our base class-incremental learning technique, we experiment with memory sizes of 500, 1,000, 1,500, and 2,000 samples for CIFAR-100, and 1,000, 2,000, 3,000, and 4,000 samples for Tiny-ImageNet. We use a new-task validation set sized of 100.

Our results demonstrate that T-CIL consistently achieves significantly lower ECE than the vanilla method across all memory sizes. Although smaller memory sizes lead to decreased model accuracy and consequently worse calibration performance, T-CIL with just 500 memory samples still outperforms the vanilla method using 2,000 samples in terms of calibration quality.

Varying size of new-task validation set We vary the size of the new-task validation set and evaluate ECE and accuracy on CIFAR-100, using ER as a base class-incremental learning technique. We present the results in Figure 8 and Table 7.

Figure 8 demonstrates that T-CIL is effective even with a small-sized validation set. As the validation set size increases, ECE decreases. However, increasing validation set size leads to accuracy drop due to the smaller training set size. These trends in ECE and accuracy indicate that T-CIL only requires small new-task validation set for calibrating the model effectively while minimizing the impact on accuracy.

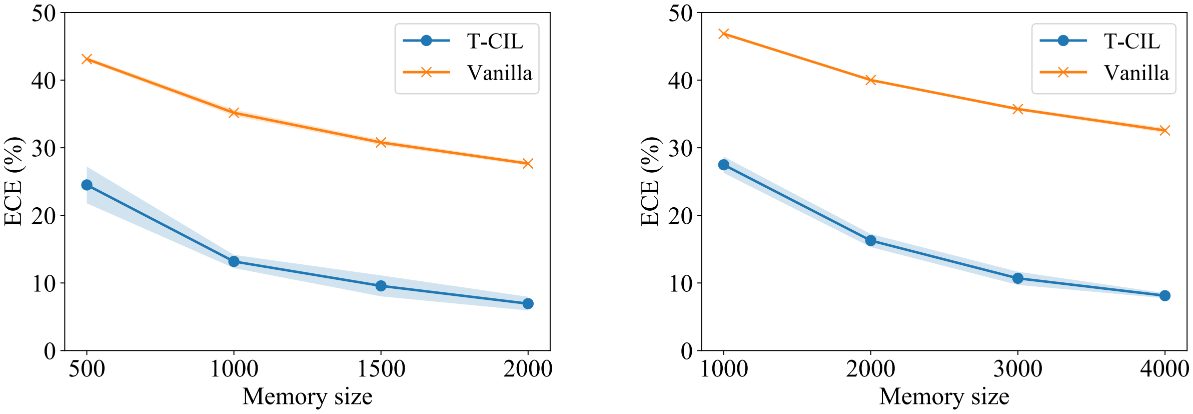
Additional ECE progressions We present the progression of ECE across tasks on CIFAR-100 and Tiny-ImageNet of vanilla, T-CIL, and Optimal TS when combined with four class-incremental learning techniques in Figure 6 and Figure 9. These figures show the progression of ECE throughout tasks, where we compare T-CIL against an ideal scenario where we run *TS* on the test set of both old and new tasks and thus obtain the best achievable calibration performance (called "Optimal TS"). When combined with various class-incremental learning techniques, T-CIL consistently demonstrates strong calibration performance across all tasks, with calibration errors approaching Optimal TS's ideal performance.

Table 5. Performance comparison between T-CIL and five baselines when integrated with four class-incremental learning techniques on three datasets.

		CIFAR-10			CIFAR-100			Tiny-ImageNet		
CIL Method	Cal Method	Acc. (↑)	ECE (↓)	AECE (↓)	Acc. (↑)	ECE (↓)	AECE (↓)	Acc. (↑)	ECE (↓)	AECE (↓)
ER [6]	Vanilla	65.61± 0.49	28.16± 0.33	28.12± 0.32	56.51± 0.37	27.66± 0.29	27.64± 0.28	31.95± 0.54	32.56± 0.36	32.55± 0.35
	TS [13]	65.86± 0.09	23.97± 0.82	23.92± 0.81	56.25± 0.62	16.28± 0.32	16.22± 0.37	31.48± 0.39	19.44± 0.75	19.44± 0.75
	ETS [44]	65.86± 0.09	22.46± 1.20	22.39± 1.21	56.25± 0.62	16.83± 0.47	16.80± 0.49	31.48± 0.39	19.81± 0.65	19.83± 0.65
	IRM [44]	65.86± 0.09	22.09± 1.73	21.83± 1.73	56.25± 0.62	17.50± 0.64	17.39± 0.61	31.48± 0.39	19.77± 0.72	19.73± 0.78
	PerturbTS [37]	n/a	n/a	n/a	56.25± 0.62	16.49± 2.19	16.49± 2.18	31.48± 0.39	10.60± 0.60	10.58± 0.61
	T-CIL	65.86± 0.09	**17.70± 2.60**	**17.64± 2.59**	56.25± 0.62	**5.74± 0.53**	**5.75± 0.50**	31.48± 0.39	**8.12± 0.38**	**8.12± 0.41**
EEIL [3]	Vanilla	77.67± 0.74	15.48± 0.75	15.45± 0.74	60.61± 0.33	21.96± 0.29	21.94± 0.28	37.44± 0.85	29.69± 0.36	29.68± 0.36
	TS	76.91± 1.27	**10.20± 0.94**	**10.16± 0.93**	60.74± 0.37	13.25± 0.60	13.14± 0.55	37.16± 0.91	13.16± 0.76	13.15± 0.73
	ETS	76.91± 1.27	10.31± 0.71	10.29± 0.72	60.74± 0.37	13.89± 0.47	13.83± 0.42	37.16± 0.91	13.52± 0.73	13.51± 0.71
	IRM	76.91± 1.27	10.51± 0.59	10.32± 0.53	60.74± 0.37	15.20± 0.48	15.09± 0.51	37.16± 0.91	14.66± 0.76	14.69± 0.76
	PerturbTS	n/a	n/a	n/a	60.74± 0.37	40.34± 3.84	40.34± 3.84	37.16± 0.91	**8.81± 0.59**	**8.81± 0.61**
	T-CIL	76.91± 1.27	10.49± 2.34	10.46± 2.32	60.74± 0.37	**10.30± 1.10**	**10.22± 1.06**	37.16± 0.91	15.58± 0.76	15.56± 0.76
WA [45]	Vanilla	73.06± 0.57	19.10± 0.50	19.07± 0.51	64.34± 0.40	8.89± 0.64	8.86± 0.63	39.66± 0.88	10.97± 0.41	10.96± 0.43
	TS	72.75± 0.47	18.22± 0.69	18.19± 0.69	64.02± 0.06	5.93± 0.24	5.88± 0.28	38.59± 0.44	13.24± 1.08	13.28± 1.09
	ETS	72.75± 0.47	17.96± 0.67	17.92± 0.69	64.02± 0.06	5.77± 0.39	5.75± 0.42	38.59± 0.44	13.01± 1.12	13.06± 1.12
	IRM	72.75± 0.47	18.03± 0.94	17.58± 0.97	64.02± 0.06	6.61± 0.47	6.51± 0.46	38.59± 0.44	10.88± 0.46	10.87± 0.48
	PerturbTS	n/a	n/a	n/a	64.02± 0.06	46.55± 2.20	46.54± 2.20	38.59± 0.44	**8.63± 1.07**	**8.61± 1.10**
	T-CIL	72.75± 0.47	**15.61± 0.23**	**15.58± 0.22**	64.02± 0.06	**3.87± 0.52**	**3.84± 0.55**	38.59± 0.44	11.43± 1.00	11.46± 1.04
DER [39]	Vanilla	74.53± 0.48	21.81± 0.46	21.78± 0.47	69.98± 0.69	22.38± 0.37	22.35± 0.35	46.62± 2.84	39.00± 1.72	38.99± 1.72
	TS	74.93± 0.35	17.27± 0.37	17.25± 0.37	69.98± 0.58	6.16± 0.25	6.04± 0.26	47.79± 0.47	11.29± 0.66	11.26± 0.66
	ETS	74.93± 0.35	16.82± 0.28	16.79± 0.29	69.98± 0.58	6.12± 0.36	6.03± 0.37	47.79± 0.47	7.83± 0.63	7.86± 0.58
	IRM	74.93± 0.35	17.04± 0.45	16.80± 0.48	69.98± 0.58	7.88± 0.33	8.03± 0.40	47.79± 0.47	10.47± 0.61	11.01± 0.61
	PerturbTS	n/a	n/a	n/a	69.98± 0.58	52.21± 5.97	52.21± 5.97	47.79± 0.47	15.56± 1.18	15.55± 1.18
	T-CIL	74.93± 0.35	**12.70± 1.35**	**12.68± 1.34**	69.98± 0.58	**4.37± 0.59**	**4.34± 0.60**	47.79± 0.47	**6.91± 0.86**	**6.90± 0.84**

Table 6. T-CIL performance combined with four existing class-incremental learning techniques on two datasets, each containing 20 incremental tasks.

	CIFAR-100			Tiny-ImageNet		
Method	Acc. (↑)	ECE (↓)	AECE (↓)	Acc. (↑)	ECE (↓)	AECE (↓)
ER	54.81± 0.70	28.95± 0.84	28.93± 0.84	30.43± 0.51	34.90± 0.16	34.88± 0.16
ER+T-CIL	54.48± 1.95	**7.08± 0.49**	**7.07± 0.46**	30.62± 0.29	**10.72± 0.88**	**10.79± 0.86**
EEIL	55.36± 1.19	25.17± 0.84	25.15± 0.84	33.54± 0.60	28.89± 0.27	28.89± 0.27
EEIL+T-CIL	55.86± 0.93	**14.38±1.34**	**14.35± 1.35**	34.22± 0.18	**24.45± 0.67**	**24.46± 0.67**
WA	59.30± 0.64	**7.39± 0.35**	**7.37± 0.35**	35.99± 0.97	**10.06± 0.95**	**10.06± 0.99**
WA+T-CIL	58.89± 0.67	**7.45± 0.72**	**7.47± 0.72**	36.80± 0.42	19.56± 0.47	19.57± 0.48
DER	68.90± 0.31	24.05± 0.45	24.02± 0.45	48.63± 0.75	42.44± 0.56	42.43± 0.56
DER+T-CIL	68.84± 0.61	**5.78± 0.92**	**5.75± 0.91**	48.33± 1.31	**5.90± 0.97**	**5.91± 0.96**



(a)

(b)

Figure 7. T-CIL performance when varying the memory size on (a) CIFAR-100 and (b) Tiny-ImageNet.

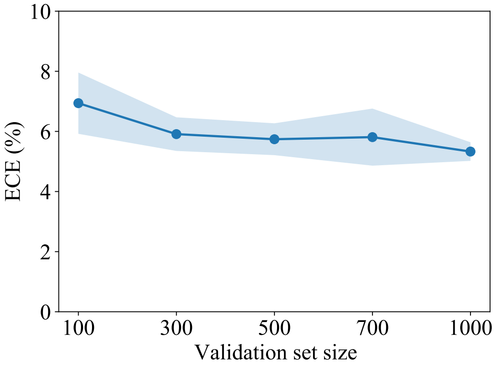
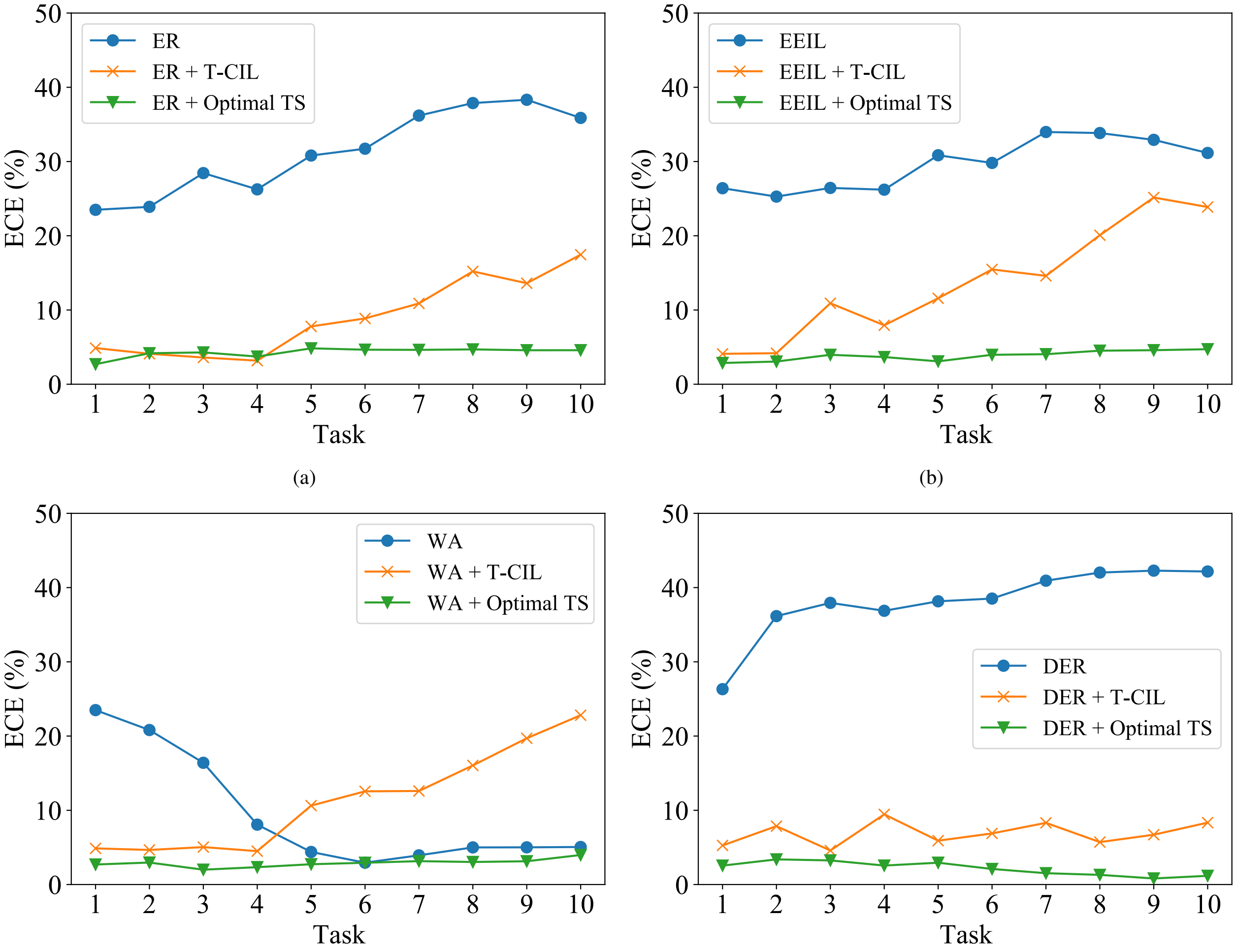


Figure 8. T-CIL performance when varying the size of the new-task validation set on the CIFAR-100.

Table 7. T-CIL performance when varying the size of the new-task validation set on the CIFAR-100.

Val size	100	300	500	700	1000
ECE (%)	6.94± 1.02	5.91± 0.56	5.74± 0.53	5.81± 0.95	5.33± 0.31
Acc (%)	56.97± 0.65	56.95± 0.34	56.25± 0.62	56.55± 0.80	56.08± 0.57



(a)

(b)

(c)

(d)

Figure 9. ECE progression after training each task on the Tiny-ImageNet among existing class-incremental learning techniques, their combinations with T-CIL, and the optimal TS. The existing techniques include: (a) ER, (b) EEIL, (c) WA, and (d) DER.