



南京航空航天大学

Nanjing University of Aeronautics and Astronautics

On the Pitfall of Mixup for Uncertainty Calibration

Deng-Bao Wang¹, Lanqing Li^{2,3,4*}, Peilin Zhao², Pheng-Ann Heng⁴, Min-Ling Zhang^{1*}

¹School of Computer Science and Engineering, Southeast University, Nanjing, China

²Tencent AI Lab ³Zhejiang Lab ⁴The Chinese University of Hong Kong

{wangdb, zhangml}@seu.edu.cn, lanqingli1993@gmail.com

masonzhao@tencent.com, pheng@cse.cuhk.edu.hk

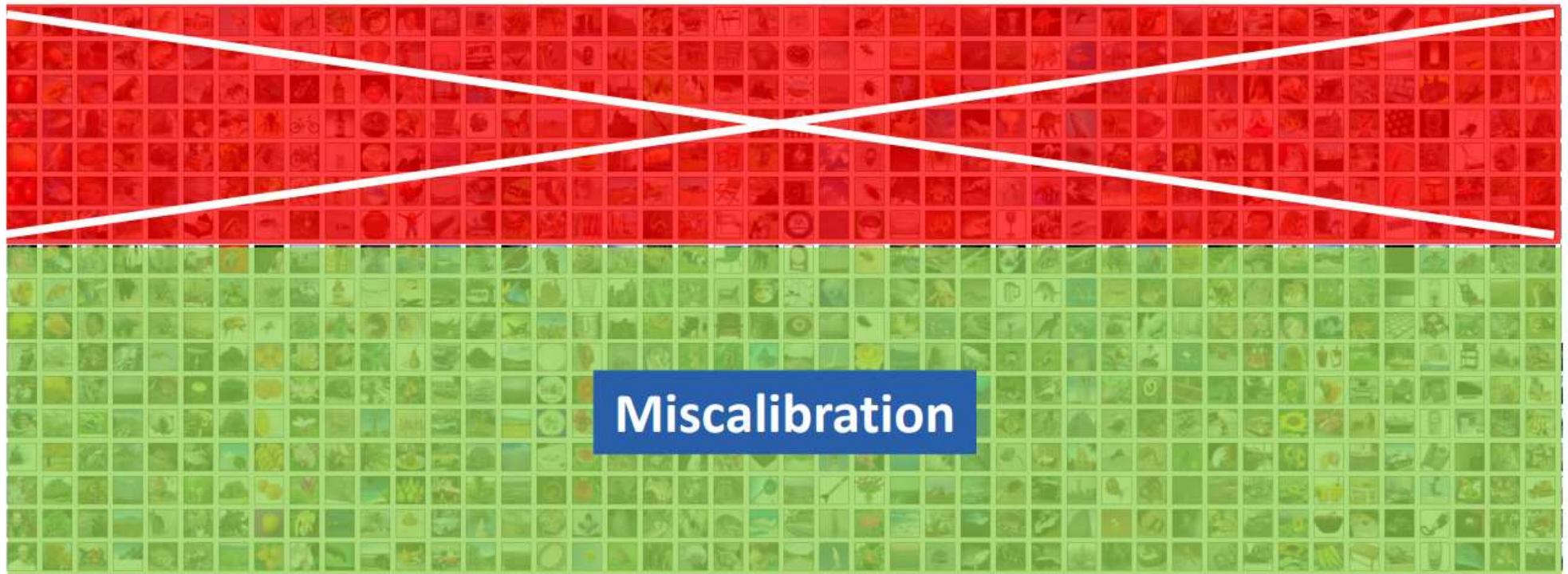
CVPR 2023

Background: Why do we need model calibration?

The ResNet's accuracy is better but not match its confidence.

ResNet 101, Cifar 100
Samples with 80%-85% confidence

Accuracy 70%



Background: Why do we need model calibration?

The problem modern neural networks: overconfidence.

What happens if the confidence is 90% ?

Neural network

Plastic bag

50% confidence

Other sensors

Person

90% confidence

Uncertainty calibration is important for many safety-critical applications

Background: What kind of model is well calibrated?

Intuitive explanation:

- Its average prediction confidence should reflect its actual accuracy.

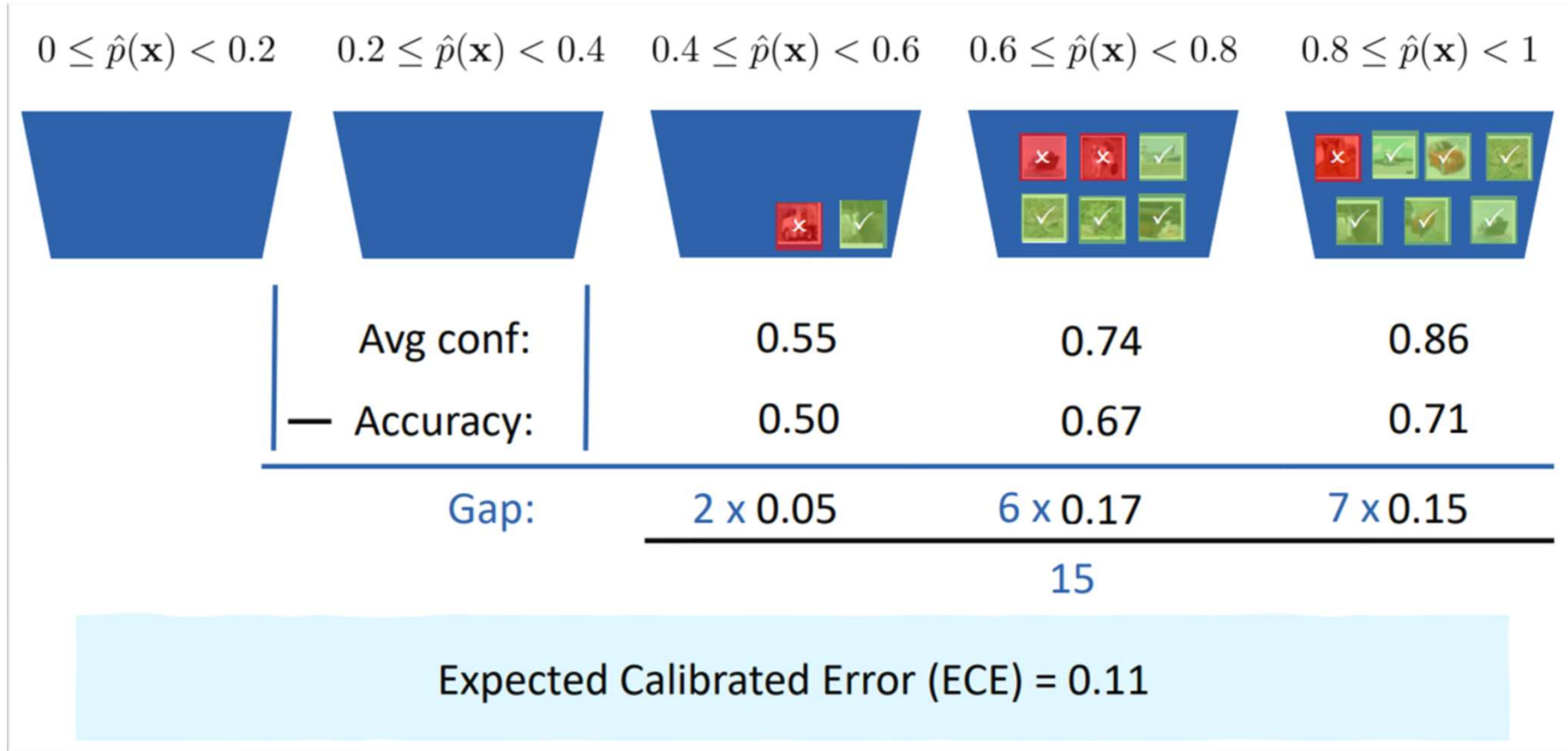
Input		Pred.	Conf.		Input		Pred.	Conf.	
x_1	→	CAT	70%	✓	x_6	→	CAT	70%	✗
x_2	→	DOG	70%	✗	x_7	→	CAT	70%	✓
x_3	→	CAT	70%	✓	x_8	→	DOG	70%	✗
x_4	→	CAT	70%	✓	x_9	→	CAT	70%	✓
x_5	→	DOG	70%	✓	x_{10}	→	DOG	70%	✓

Formally:

- A perfect classifier satisfies: $\mathbb{P}(\hat{Y} = Y \mid \hat{P} = p) = p, \quad \forall p \in [0, 1]$

Evaluation metric:
$$\text{ECE} = \sum_{m=1}^M \frac{|B_m|}{n} \left| \text{acc}(B_m) - \text{conf}(B_m) \right|$$

Background: Expected Calibrated Error (ECE)



$$\text{ECE} = \sum_{m=1}^M \frac{|B_m|}{n} \left| \text{acc}(B_m) - \text{conf}(B_m) \right|$$

Background: Post-hoc calibration strategy


Temperature Scaling(TS):

$$\hat{p} = \max_{i \in [K]} \frac{\exp(\delta_i / \tau)}{\sum_{k=1}^K \exp(\delta_k / \tau)}$$

In practice,

- find the temperature that yields the best ECE on a validation set,
- then apply this temperature to the softmax layer.

Motivation: Most of existing work investigates mixup for calibration **without the consideration of post-calibration**, which is unfair.

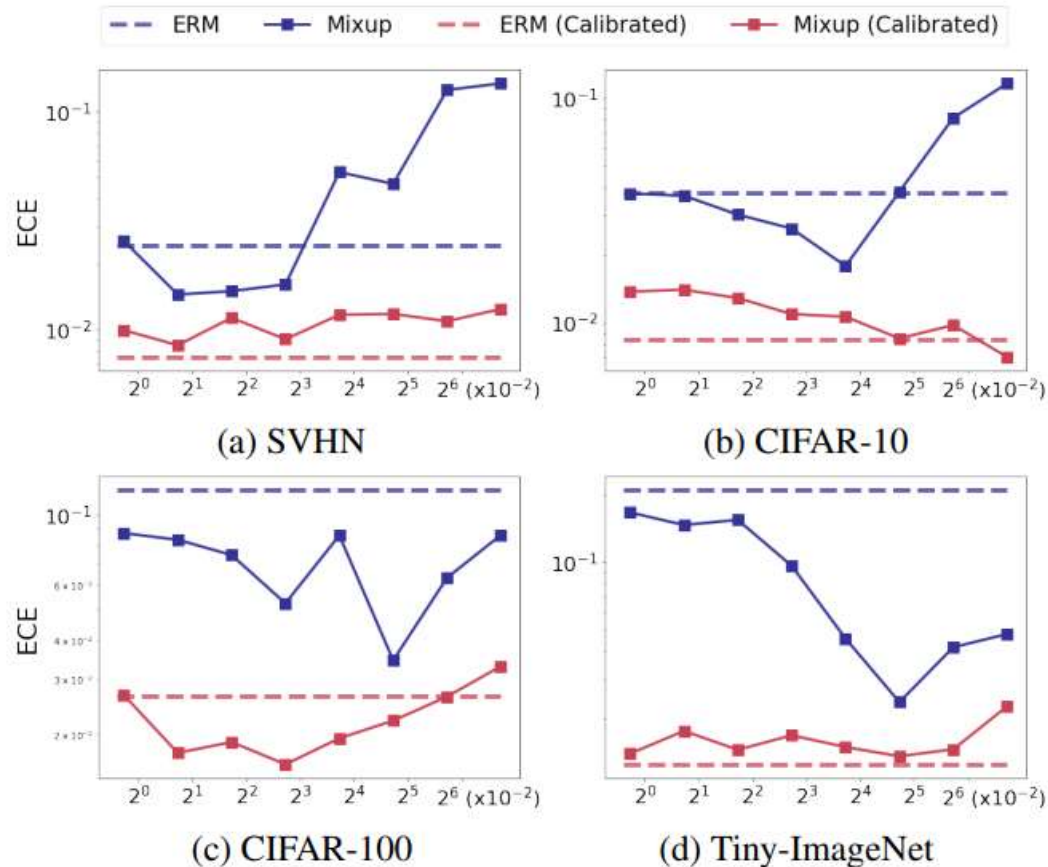
- ❑ Does mixup really help calibration?
 - ❑ If it does not, what leads to the failure?
 - ❑ How can we mitigate the pitfall of mixup on calibration?
-
- 

Does mixup really help calibration?

Table 1. Comparison between mixup and ERM in terms of **uncalibrated ECE**, **calibrated ECE** and the **optimal ECE**. ▲/▼ indicates that mixup outperforms/underperforms the vanilla ERM. The values reported in each entry are the results of different backbones: ResNet-18, ResNet-50, ResNet-110 and ResNet-152.

Datasets	Metrics	ERM				mixup ($\alpha = 0.1$)				mixup ($\alpha = 0.5$)				mixup ($\alpha = 1.0$)			
SVHN	ECE	2.15	2.67	2.43	2.56	3.96	1.89▲	2.46	1.38▲	11.2	9.48	8.04	9.37	14.8	13.7	13.8	12.9
	Calibrated ECE	0.50	0.87	0.75	0.90	0.99▼	1.03▼	1.08▼	1.05▼	1.23▼	1.21▼	1.28▼	1.21▼	1.12▼	1.18▼	1.14▼	1.04▼
	Optimal ECE	0.24	0.56	0.45	0.58	0.75▼	0.74▼	0.85▼	0.68▼	1.12▼	0.95▼	0.95▼	0.88▼	1.04▼	0.98▼	0.88▼	0.78▼
CIFAR-10	ECE	3.33	3.99	3.78	3.47	2.57▲	2.22▲	2.55▲	2.53▲	6.87	6.25	6.55	6.20	12.1	11.5	10.5	11.2
	Calibrated ECE	0.65	0.79	0.83	0.65	1.04▼	1.07▼	1.08▼	1.12▼	1.15▼	1.15▼	0.95▼	1.05▼	0.94▼	0.91▼	0.83	0.76▼
	Optimal ECE	0.59	0.63	0.61	0.52	0.97▼	0.98▼	1.01▼	1.01▼	0.97▼	1.03▼	0.88▼	0.88▼	0.85▼	0.80▼	0.71▼	0.65▼
CIFAR-100	ECE	10.9	12.5	11.9	11.7	2.43▲	6.63▲	5.95▲	5.59▲	10.8▲	3.89▲	3.91▲	3.85▲	13.0	7.44▲	7.50▲	7.55▲
	Calibrated ECE	2.56	2.41	2.64	2.42	1.76	1.87	1.37	1.67	1.22	2.63▼	3.21▼	2.57▼	1.25	2.66▼	3.02▼	3.52▼
	Optimal ECE	2.45	2.29	2.44	2.31	1.60	1.59	1.23	1.45	0.98	2.46▼	3.04▼	2.39▼	1.09	2.54▼	2.85▼	3.38▼
Tiny-ImageNet	ECE	23.2	20.5	20.7	21.6	8.57▲	7.51▲	9.76▲	10.4▲	3.98▲	3.92▲	2.16▲	3.29▲	6.85▲	7.44▲	4.98▲	5.93▲
	Calibrated ECE	1.33	1.23	1.36	1.33	1.32	1.28▼	1.55▼	2.08▼	1.33	1.46▼	1.52▼	1.82▼	1.49▼	1.65▼	2.26▼	2.00▼
	Optimal ECE	1.14	1.00	1.16	1.16	1.02	1.05▼	1.40▼	1.93▼	1.08	1.21▼	1.23▼	1.60▼	1.20▼	1.30▼	1.91▼	1.69▼

Does mixup really help calibration?



- The best α varies across datasets;
 - there is a large margin between uncalibrated and calibrated ECE.
- ✓ There exists **a dilemma between accuracy and calibration** in using mixup when considering training and post-calibration as a unified system.

Figure 1. The comparison of mixup with different choices of α . The experiments are conducted with ResNet110.

Why mixup fails on calibration?

Remark 1. [3] Let $\lambda \sim \text{Beta}[\frac{1}{2}, 1](\alpha, \alpha)$ and $j \sim \text{Uniform}([n])$ be two random variables with $\alpha > 0$, $n > 0$ and let $\bar{\lambda} = \mathbb{E}_{\lambda} \lambda$. The mixed sample $(\tilde{x}_i, \tilde{y}_i)$ as in Equation (1) for any $i \in [n]$ can be reformulated as:

$$\begin{aligned} \tilde{x}_i &= \bar{x} + \bar{\lambda}(x_i - \bar{x}) + (\lambda - \bar{\lambda})x_i + (1 - \lambda)x_j - (1 - \bar{\lambda})\bar{x}, \\ \tilde{y}_i &= \bar{y} + \bar{\lambda}(y_i - \bar{y}) + (\lambda - \bar{\lambda})y_i + (1 - \lambda)y_j - (1 - \bar{\lambda})\bar{y}, \end{aligned} \quad (2)$$

Data Transformation x'_i, y'_i
Random Perturbation $\epsilon_i^x, \epsilon_i^y$

where \bar{x}, \bar{y} are the mean of inputs and labels of all training samples, and the perturbation terms satisfy $\mathbb{E}_{\lambda, j} \epsilon_i^x = \mathbb{E}_{\lambda, j} \epsilon_i^y = 0$.

The authors conjecture that the **label transformation part** of the second Equation in (2) **leads to the failure on calibration.**

Why mixup fails on calibration?

Table 2. The inputs and targets used by 4 ablated variants.

Variants	Inputs	Targets	One-hot?
Mixup-DT	$\{\bar{x} + \bar{\lambda}(x_i - \bar{x})\}$	$\{\bar{y} + \bar{\lambda}(y_i - \bar{y})\}$	✗
Mixup-TO	$\{x_i\}$	$\{\lambda y_i + (1 - \lambda)y_j\}$	✗
Mixup-SC	$\{\lambda x_i + (1 - \lambda)x_j y_i = y_j\}$	$\{y_i\}$	✓
Mixup-IO	$\{\lambda x_i + (1 - \lambda)x_j\}$	$\{y_i\}$	✓

Why mixup fails on calibration?

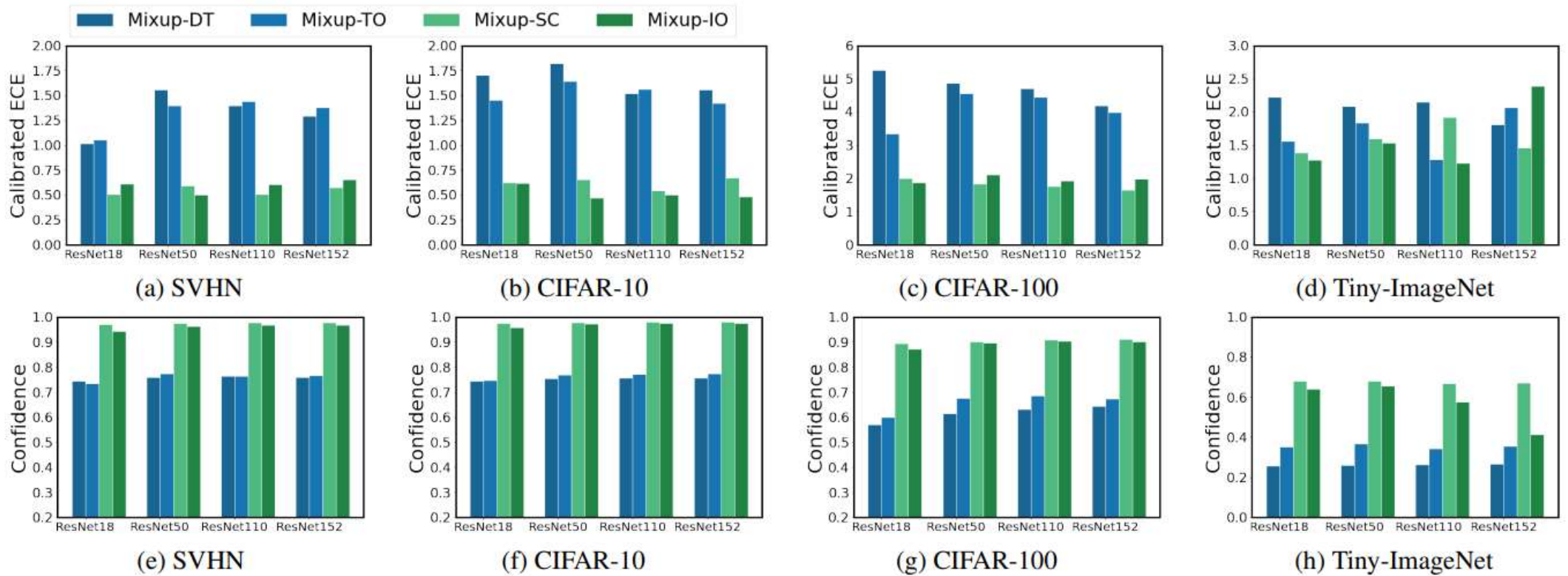


Figure 2. The top row shows the comparison of **Calibrated ECE** between four ablated variants of mixup, where the variants with blue color use the transformed labels while the variants with green color use the original one-hot labels. The bottom row shows the comparison of **average confidence** between four ablated variants.

Confidence penalty hurts calibration

Why mixup fails on calibration?

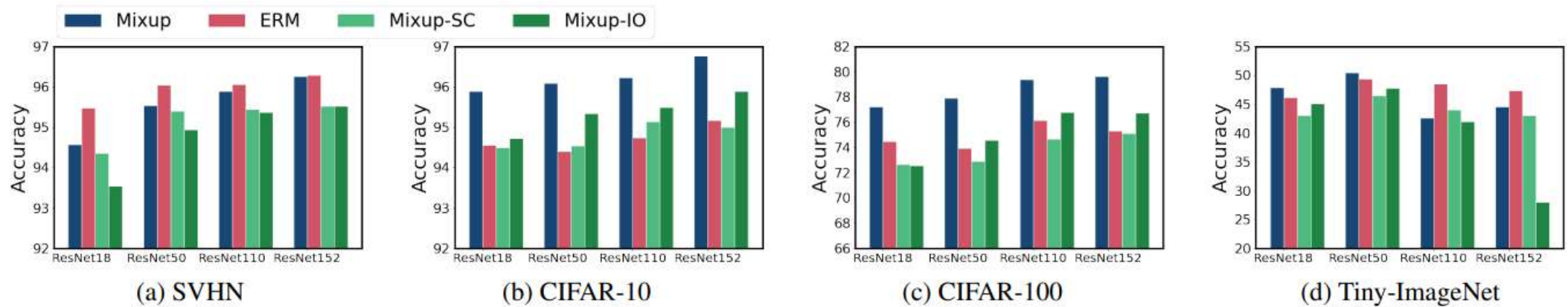


Figure 3. The predictive **accuracy** of Mixup-IO, Mixup-SC by comparing with that of vanilla mixup and ERM. The detailed results could be found in Table 4 and Table 5.

Trivial confidence promotion hurts accuracy

Mitigating the pitfall of mixup.

Remark 2. Recall the basic idea of mixup: linear interpolations of feature vectors should lead to linear interpolations of the output space. Based on this assumption, by mixing two samples twice with $\lambda_1 \neq \lambda_2 \in (0, 1)$, as is

$$\begin{aligned}\tilde{x}_1 &= \lambda_1 x_a + (1 - \lambda_1) x_b, \\ \tilde{x}_2 &= \lambda_2 x_a + (1 - \lambda_2) x_b,\end{aligned}\tag{3}$$

we can decouple these two samples in outputs space:

$$\begin{aligned}\hat{y}_a &= \frac{f(\tilde{x}_1) - f(\tilde{x}_2)(1 - \lambda_1)/(1 - \lambda_2)}{\lambda_1 - \lambda_2(1 - \lambda_1)/(1 - \lambda_2)}, \\ \hat{y}_b &= \frac{f(\tilde{x}_1) - f(\tilde{x}_2)\lambda_2/\lambda_1}{1 - \lambda_2 - (1 - \lambda_1)\lambda_2/\lambda_1}.\end{aligned}\tag{4}$$

Mitigating the pitfall of mixup.

```
1 # Sampling  $\lambda_1$  and  $\lambda_2$  for mixup
2 l1 = np.random.beta(alpha, alpha)
3 l1 = max(l1, 1 - l1)
4 l2 = np.random.beta(alpha, alpha)
5 l2 = min(l2, 1 - l2)
6 while constraint == True and abs(l1 - l2) < margin:
7     l1 = np.random.beta(alpha, alpha)
8     l1 = max(l1, 1 - l1)
9     l2 = np.random.beta(alpha, alpha)
10    l2 = min(l2, 1 - l2)
11 # Mixing inputs by Equation (3)
12 mixed_inputs1 = l1 * inputs_a + (1 - l1) * inputs_b
13 mixed_inputs2 = l2 * inputs_a + (1 - l2) * inputs_b
14 # Forward Propagation
15 mixed_outputs1 = model(mixed_inputs1)
16 mixed_outputs2 = model(mixed_inputs2)
17 # Decoupling outputs by Equation (4)
18 outputs_a = (mixed_outputs1 - (1 - l1) / (1 - l2) * mixed_outputs2) / (l1 - l2 * (1 - l1) / (1 - l2))
19 outputs_b = (mixed_outputs2 - l2 / l1 * mixed_outputs1) / (1 - l2 - (1 - l1) * l2 / l1)
20 # Calculating Softmax Cross-Entropy Loss
21 loss1 = criterion(outputs_a, targets_a)
22 loss2 = criterion(outputs_b, targets_b)
23 loss = 0.5 * loss1 + 0.5 * loss2
24 # Updating model's weights
25 optimizer.zero_grad()
26 loss.backward()
27 optimizer.step()
```

Mitigating the pitfall of mixup.

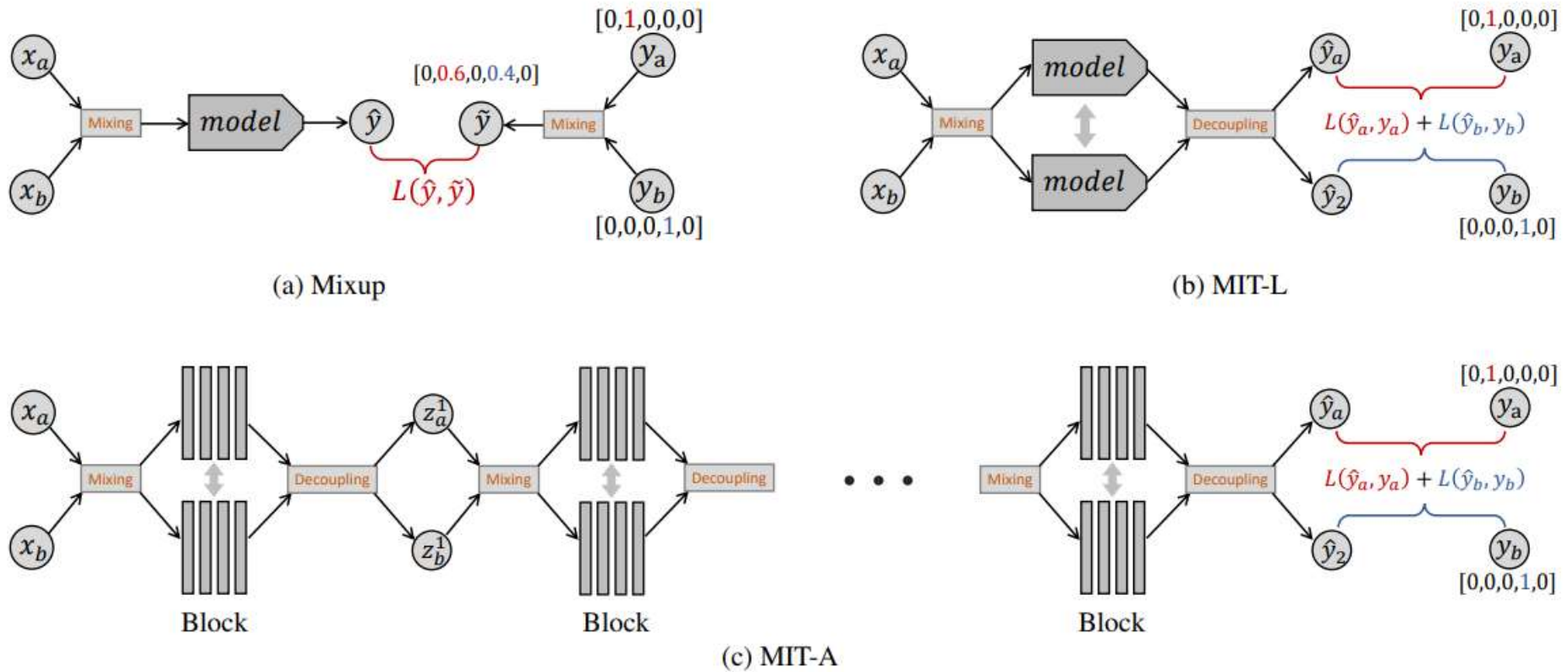


Figure A. (a), (b) and (c) show the pipelines of vanilla mixup, our MIT-L and MIT-A respectively, where where loss function L is the widely used softmax cross-entropy loss. For MIT-A, we simply apply the mix-then-decouple process in every block of ResNets. It should be noted that this process can also be applied in each latent layer.

Mitigating the pitfall of mixup.

Table 5. The overall comparative results in terms of the predictive **accuracy**. The number in each bracket indicates the ranking across all methods. The **orange/blue** color indicates that a method outperforms/underperforms ERM in average. The **boldface** and underline denote the best and the second best results of each row. The marker † means the backbone is pretrained.

	Backbones	ERM	Mixup (0.1)	Mixup (0.5)	Mixup (1.0)	Mixup (DT)	Mixup (TO)	Mixup (SC)	Mixup (IO)	MIT-A ($\Delta\lambda > \frac{1}{2}$)	MIT-L ($\Delta\lambda > \frac{1}{2}$)	MIT-A ($\Delta\lambda > \frac{1}{2}$)
SVHN	ResNet18	95.4 (5)	95.5 (4)	94.8 (7)	94.5 (8)	95.6 (3)	96.0 (1)	94.3 (9)	93.5 (10)	95.0 (6)	93.2 (11)	95.7 (2)
	ResNet50	96.0 (4)	96.0 (3)	95.8 (5)	95.5 (8)	95.5 (7)	95.7 (6)	95.3 (9)	94.9 (10)	<u>96.2 (2)</u>	94.3 (11)	96.2 (1)
	ResNet110	96.0 (5)	96.1 (4)	96.3 (3)	95.8 (7)	95.6 (8)	95.9 (6)	95.4 (9)	95.3 (10)	<u>96.5 (2)</u>	95.0 (11)	96.7 (1)
	ResNet152	96.2 (5)	<u>96.6 (2)</u>	96.4 (4)	96.2 (6)	95.6 (8)	95.9 (7)	95.5 (9)	95.5 (10)	96.5 (3)	94.9 (11)	96.7 (1)
	Avg. gain	—	+0.11	-0.10	-0.40	-0.32	-0.06	-0.78	-1.12	+0.14	-1.55	+0.41
CIFAR-10	ResNet18	94.5 (9)	95.1 (6)	95.7 (3)	<u>95.8 (2)</u>	93.9 (11)	94.5 (8)	94.4 (10)	94.7 (7)	95.5 (4)	95.2 (5)	95.9 (1)
	ResNet50	94.4 (9)	95.3 (7)	95.8 (3)	96.0 (1)	93.1 (11)	94.2 (10)	94.5 (8)	95.3 (6)	95.8 (4)	95.7 (5)	<u>96.0 (2)</u>
	ResNet110	94.7 (9)	95.7 (6)	96.3 (1)	<u>96.2 (2)</u>	93.7 (11)	94.3 (10)	95.1 (8)	95.4 (7)	96.1 (4)	96.0 (5)	96.1 (3)
	ResNet152	95.1 (8)	95.8 (7)	<u>96.4 (2)</u>	96.7 (1)	93.9 (11)	94.8 (10)	95.0 (9)	95.8 (6)	96.3 (4)	96.2 (5)	96.4 (3)
	Avg. gain	—	+0.78	+1.36	+1.53	-1.01	-0.21	+0.08	+0.64	+1.27	+1.12	+1.41
CIFAR-100	ResNet18	74.4 (8)	75.3 (7)	<u>76.8 (2)</u>	77.2 (1)	72.4 (11)	76.4 (4)	72.6 (9)	72.5 (10)	76.2 (5)	75.9 (6)	76.6 (3)
	ResNet50	73.9 (9)	76.4 (6)	<u>78.3 (2)</u>	77.8 (3)	68.2 (11)	75.1 (7)	72.9 (10)	74.5 (8)	78.3 (1)	76.6 (5)	77.7 (4)
	ResNet110	76.1 (9)	77.9 (6)	80.1 (1)	<u>79.3 (2)</u>	70.9 (11)	77.3 (7)	74.6 (10)	76.7 (8)	78.7 (4)	77.9 (5)	79.1 (3)
	ResNet152	75.3 (9)	78.2 (6)	<u>79.7 (2)</u>	79.6 (3)	72.5 (11)	76.9 (7)	75.1 (10)	76.7 (8)	79.1 (4)	78.2 (5)	79.8 (1)
	Avg. gain	—	+2.01	+3.79	+3.55	-3.92	+1.50	-1.12	+0.18	+3.14	+2.24	+3.38
Tiny-ImageNet	ResNet18	46.1 (9)	46.6 (7)	47.4 (5)	47.8 (4)	36.5 (11)	47.1 (6)	43.0 (10)	46.6 (8)	49.5 (1)	48.5 (3)	<u>49.3 (2)</u>
	ResNet50	49.3 (7)	49.5 (6)	50.0 (5)	50.4 (4)	37.5 (11)	49.0 (8)	46.4 (10)	48.8 (9)	<u>51.4 (2)</u>	51.0 (3)	51.8 (1)
	ResNet110	48.5 (3)	43.6 (7)	42.7 (9)	42.6 (10)	35.6 (11)	44.6 (4)	43.9 (6)	43.5 (8)	<u>48.6 (2)</u>	44.4 (5)	50.8 (1)
	ResNet152	47.3 (2)	44.7 (5)	42.3 (9)	44.6 (6)	34.5 (11)	45.5 (4)	43.0 (8)	39.7 (10)	46.1 (3)	43.8 (7)	50.0 (1)
	ResNet18†	53.6 (6)	53.5 (8)	54.0 (5)	53.5 (7)	44.1 (11)	54.7 (1)	49.7 (10)	50.5 (9)	54.5 (3)	54.4 (4)	<u>54.7 (2)</u>
	ResNet152†	62.4 (6)	<u>63.2 (2)</u>	63.7 (1)	63.0 (3)	49.6 (11)	62.6 (4)	58.8 (10)	59.9 (9)	61.9 (7)	62.5 (5)	61.6 (8)
Avg. gain	—	-1.01	-1.18	-0.87	-11.5	-0.60	-3.70	-3.04	+0.81	-0.41	+1.83	

Mitigating the pitfall of mixup.

Table 4. The overall comparative results in terms of **calibrated ECE**. The number in each bracket indicates the ranking across all methods. The **orange/blue** color indicates that a method outperforms/underperforms ERM in average. The **boldface** and underline denote the best and the second best results of each row. The marker † means the backbone is pretrained.

	Backbones	ERM	Mixup (0.1)	Mixup (0.5)	Mixup (1.0)	Mixup (DT)	Mixup (TO)	Mixup (SC)	Mixup (IO)	MIT-A	MIT-L	MIT-A
										$(\Delta\lambda > \frac{1}{2})$	$(\Delta\lambda > \frac{1}{2})$	$(\Delta\lambda > \frac{1}{2})$
SVHN	ResNet18	<u>0.50 (2)</u>	0.99 (7)	1.23 (11)	1.12 (10)	1.01 (8)	1.05 (9)	0.50 (3)	0.61 (5)	0.47 (1)	0.65 (6)	0.53 (4)
	ResNet50	0.87 (6)	1.03 (7)	1.21 (9)	1.18 (8)	1.55 (11)	1.39 (10)	0.59 (4)	<u>0.50 (2)</u>	0.49 (1)	0.52 (3)	0.66 (5)
	ResNet110	0.75 (6)	1.08 (7)	1.28 (9)	1.14 (8)	1.39 (10)	1.43 (11)	<u>0.50 (2)</u>	<u>0.60 (4)</u>	0.48 (1)	0.53 (3)	0.70 (5)
	ResNet152	0.90 (6)	1.05 (8)	1.21 (9)	1.04 (7)	1.28 (10)	1.37 (11)	<u>0.57 (2)</u>	0.65 (4)	0.61 (3)	0.53 (1)	0.67 (5)
	Avg. gain	—	+0.28	+0.47	+0.36	+0.55	+0.55	-0.21	-0.16	-0.24	-0.19	-0.11
CIFAR-10	ResNet18	0.65 (6)	1.04 (8)	1.15 (9)	0.94 (7)	1.70 (11)	1.45 (10)	0.62 (4)	0.61 (3)	0.56 (1)	<u>0.59 (2)</u>	0.62 (5)
	ResNet50	0.79 (6)	1.07 (8)	1.15 (9)	0.91 (7)	1.81 (11)	1.64 (10)	0.65 (4)	0.46 (1)	0.63 (3)	<u>0.59 (2)</u>	0.68 (5)
	ResNet110	0.83 (7)	1.08 (9)	0.95 (8)	0.83 (6)	1.52 (10)	1.56 (11)	0.54 (3)	0.50 (1)	<u>0.52 (2)</u>	0.54 (4)	0.78 (5)
	ResNet152	0.65 (4)	1.12 (9)	1.05 (8)	0.76 (7)	1.55 (11)	1.42 (10)	0.67 (5)	0.48 (1)	0.57 (3)	<u>0.50 (2)</u>	0.67 (6)
	Avg. gain	—	+0.34	+0.34	+0.13	+0.91	+0.78	-0.11	-0.21	-0.15	-0.17	-0.04
CIFAR-100	ResNet18	2.56 (9)	1.76 (5)	1.22 (1)	<u>1.25 (2)</u>	5.24 (11)	3.33 (10)	2.00 (7)	1.87 (6)	1.44 (3)	2.18 (8)	1.75 (4)
	ResNet50	2.41 (7)	<u>1.87 (2)</u>	2.63 (8)	2.66 (9)	4.86 (11)	4.55 (10)	1.82 (1)	2.10 (5)	1.90 (3)	2.15 (6)	1.97 (4)
	ResNet110	2.64 (7)	1.37 (1)	3.21 (9)	3.02 (8)	4.70 (11)	4.45 (10)	<u>1.76 (2)</u>	1.93 (3)	1.98 (4)	2.25 (6)	2.00 (5)
	ResNet152	2.42 (6)	<u>1.67 (2)</u>	2.57 (8)	3.52 (9)	4.19 (11)	3.97 (10)	1.65 (1)	1.98 (4)	1.71 (3)	2.47 (7)	2.17 (5)
	Avg. gain	—	-0.84	-0.10	+0.10	+2.24	+1.56	-0.69	-0.53	-0.74	-0.24	-0.53
Tiny-ImageNet	ResNet18	1.33 (4)	<u>1.32 (2)</u>	1.33 (3)	1.49 (8)	2.22 (11)	1.55 (10)	1.38 (5)	1.30 (1)	1.47 (7)	1.54 (9)	1.41 (6)
	ResNet50	1.23 (3)	1.28 (4)	1.46 (6)	1.65 (9)	2.08 (11)	1.83 (10)	1.59 (8)	1.58 (7)	1.18 (1)	<u>1.23 (2)</u>	1.36 (5)
	ResNet110	1.36 (4)	1.55 (8)	1.52 (7)	2.26 (11)	2.14 (10)	1.28 (1)	1.92 (9)	1.49 (6)	1.35 (3)	<u>1.29 (2)</u>	1.39 (5)
	ResNet152	1.33 (3)	2.08 (10)	1.82 (7)	2.00 (8)	1.81 (6)	2.06 (9)	1.46 (5)	2.19 (11)	1.43 (4)	1.17 (1)	<u>1.30 (2)</u>
	ResNet18†	<u>1.12 (2)</u>	1.43 (5)	1.22 (3)	1.31 (4)	2.83 (11)	1.90 (10)	1.58 (7)	1.72 (8)	1.56 (6)	1.79 (9)	1.11 (1)
	ResNet152†	1.96 (6)	1.57 (4)	2.75 (9)	2.74 (8)	4.83 (10)	6.60 (11)	1.19 (1)	1.68 (5)	1.37 (3)	2.58 (7)	1.26 (2)
Avg. gain	—	+0.14	+0.29	+0.51	+1.26	+1.14	+0.13	+0.27	0.00	+0.21	-0.08	

Mitigating the pitfall of mixup.

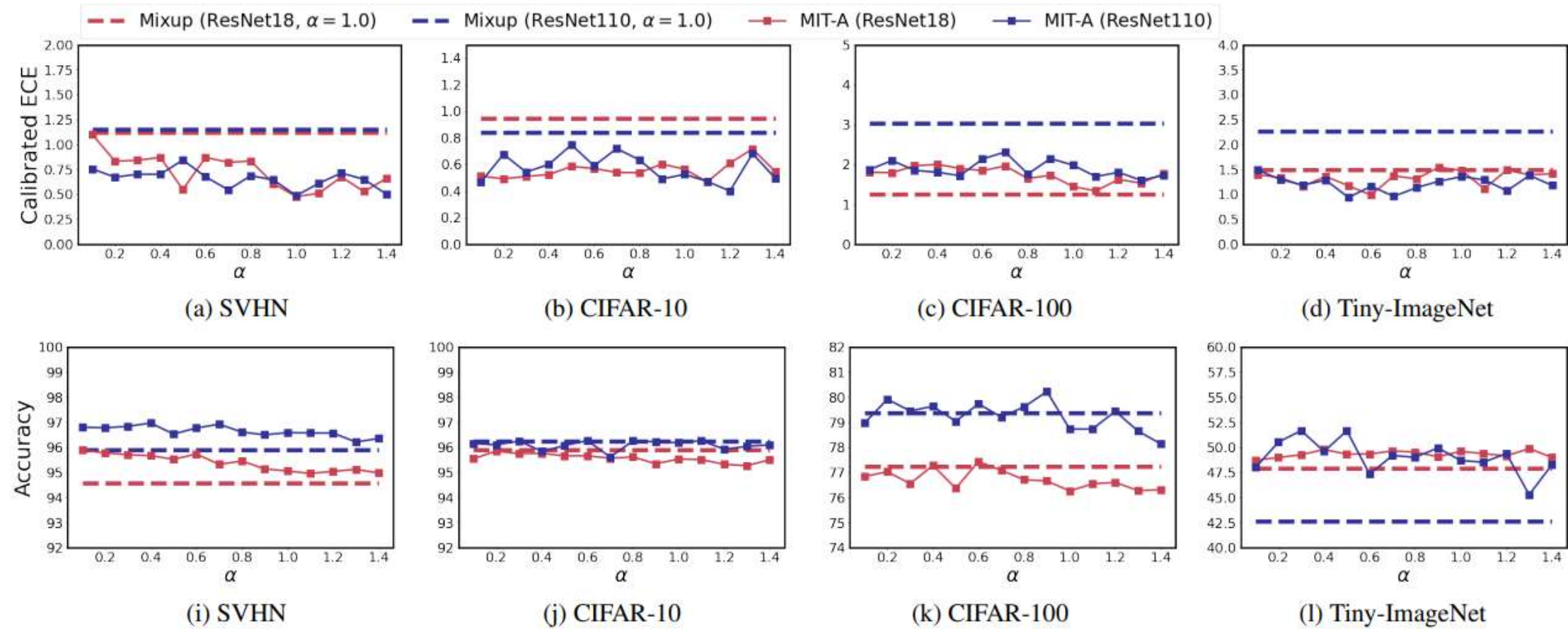


Figure C. The results of MIT-A on **Calibrated ECE** and **Accuracy** with different α on ResNet18 and ResNet110.

THANKS