A Strong Baseline for Semi-Supervised Incremental FSL-Supplementary Material

Linglan Zhao*[1]
llzhao@sjtu.edu.cn
Dashan Guo[2], Yunlu Xu†[2], Liang Qiao[2]
{guodashan,xuyunlu,qiaoliang6}@hikvision.com
Zhanzhan Cheng[2], Shiliang Pu[2], Yi Niu[2]
{chengzhanzhan,pushiliang,niuyi}@hikvision.com
Xiangzhong Fang[1]
xzfang@sjtu.edu.cn

[1] Dept. of Electronic Engineering
Shanghai Jiao Tong University
Shanghai, China

[2] Hikvision Research Institute
Hangzhou, China

Appendix

A Dataset Statistics

In this section, we introduce more details of the dataset statistics used in experiments.

To evaluate the performance of the model on both base and novel classes, as stated in Section 3, we follow the data splits used in prior incremental FSL works [9, 27, 40]. Splits and dataset statistics for *mini*-ImageNet and *tiered*-ImageNet are provided in Table 1 and Table 2, respectively. As in standard FSL, novel class training ($\mathcal{D}_{\text{novel/train}}$), validation ($\mathcal{D}_{\text{novel/val}}$), and test set ($\mathcal{D}_{\text{novel/test}}$) have disjoint sets of object classes. However, in incremental FSL and S^{2}I-FSL, the performance on the base class predictions is also evaluated. As a result, additional splits of base class training ($\mathcal{D}_{\text{base/train}}$), validation ($\mathcal{D}_{\text{base/val}}$) and test set ($\mathcal{D}_{\text{base/test}}$) are required. Concretely, in the pre-training phase, $\mathcal{D}_{\text{base/train}}$ is used to train backbone f_θ and base class classification weights W_b. Then in the meta-training phase, $\mathcal{D}_{\text{base/train}}$ and $\mathcal{D}_{\text{novel/train}}$ are used to simulate the incremental scenario during testing. Models are selected using $\mathcal{D}_{\text{base/val}}$ and $\mathcal{D}_{\text{novel/val}}$, and the final performance is evaluated using $\mathcal{D}_{\text{base/test}}$ and $\mathcal{D}_{\text{novel/test}}$. Each image in both datasets is of size 84×84.

Classes	Purpose	Split	N_{classes}	N_{samples}
Base ($\mathcal{D}_{\text{base}}$)	Train	Train-Train ($\mathcal{D}_{\text{base/train}}$)	64	38,400
	Validate	Train-Val ($\mathcal{D}_{\text{base/val}}$)	64	18,748
	Test	Train-Test ($\mathcal{D}_{\text{base/test}}$)	64	19,200
Novel ($\mathcal{D}_{\text{novel}}$)	Train	Train-Train ($\mathcal{D}_{\text{novel/train}} \leftarrow \mathcal{D}_{\text{base/train}}$)	64	38,400
	Validate	Val ($\mathcal{D}_{\text{novel/val}}$)	16	9,600
	Test	Test ($\mathcal{D}_{\text{novel/test}}$)	20	12,000

Table 1: Dataset statistics of *mini*-ImageNet for incremental FSL and S^{2}I-FSL.

For meta-training on *mini*-ImageNet, since $\mathcal{D}_{\text{novel/train}}$ is not provided explicitly, $\mathcal{D}_{\text{base/train}}$ is reused as $\mathcal{D}_{\text{novel/train}}$ for *fake novel training* [9]. In fake novel training, N sampled classes from $\mathcal{D}_{\text{base/train}}$ are regarded as fake novel classes while the remaining $N_b - N$ classes are viewed as base classes at each episode. Concretely, the model is meta-trained with 59+5-way

* Linglan Zhao did this work when he was an intern at Hikvision Research Institute. †: corresponding author.

incremental FSL episodes, and the final model is evaluated with 64+5-way incremental FSL episodes ($N_b = 64$, $N = 5$). As for *tiered*-ImageNet, $\mathcal{D}_{\text{base/train}}$ and $\mathcal{D}_{\text{novel/train}}$ are different sets. The original training split of *tiered*-ImageNet [26] which contains 351 classes is divided into two splits of "Train-A" (200 classes) and "Train-B" (151 classes) containing disjoint classes. The "Train-A" is then further divided into base class training, validation, test sets of "Train-A-Train", "Train-A-Val" and "Train-A-Test" with disjoint samples from the same 200 base classes ($N_b = 200$, $N = 5$). It is noteworthy that original training data from 'Train-A-Val" and "Train-A-Test" cannot be used during training. Namely, less than 90% of the *tiered*-ImageNet training data is used due to the dataset split for incremental FSL. As a result, our method can get higher accuracy if the whole training set is given (Table 4 & 6).

Classes	Purpose	Split	N_{classes}	N_{samples}
Base ($\mathcal{D}_{\text{base}}$)	Train	Train-A-Train ($\mathcal{D}_{\text{base/train}}$)	200	203,751
	Validate	Train-A-Val ($\mathcal{D}_{\text{base/val}}$)	200	25,460
	Test	Train-A-Test ($\mathcal{D}_{\text{base/test}}$)	200	25,488
Novel ($\mathcal{D}_{\text{novel}}$)	Train	Train-B ($\mathcal{D}_{\text{novel/train}}$)	151	193,996
	Validate	Val ($\mathcal{D}_{\text{novel/val}}$)	97	124,261
	Test	Test ($\mathcal{D}_{\text{novel/test}}$)	160	206,209

Table 2: Dataset statistics of *tiered*-ImageNet for incremental FSL and S^2I-FSL.

B Details of Network Architecture

ResNet12. To have fair comparisons between other standard FSL, transductive FSL, semi-supervised FSL and incremental FSL methods, ResNet12 [11, 24] is adopted as feature extractor f_θ in most of the experiments. ResNet12 contains four residual blocks where each block is comprised of three 3×3 convolutional layers followed by a 2×2 max-pooling layer. Each convolutional layer consists of a 3×3 kernel, followed by Batch Normalization [15] and leaky ReLU of 0.1. The first block includes 64 feature channels which are doubled at each subsequent block, and the final output feature maps have 512 channels. For a $3 \times 84 \times 84$ input, the output feature maps have a size of $512 \times 5 \times 5$. It is worth noting that we do not use the $1.25\times$ wider ResNet12 with DropBlock [8] as in some current papers [18, 22, 33]. Although using this more sophisticated architecture should further improve the performance, we contend that the design of the feature extractor is not the focus of this paper.

C Hyperparameter Settings

For all the experiments, we use the SGD optimizer with the Nesterov momentum 0.9, where a weight decay of 5×10^{-4} is applied to the model parameters. The learnable scalar γ in cosine classifier is initialized to 10. In addition, standard data augmentation operations including random crop, left-right flip, and color jitter are applied.

In the pre-training phase, the learning rate is initialized with 0.1 and applied by an exponential rate decay schedule of the form $10^{-\lambda \cdot \frac{t}{T}}$, where $\lambda = 4$, t is epoch index, and T is the total number of training epochs. We train 400/100 epochs for *mini*-ImageNet/*tiered*-ImageNet with a batch size of 64. As for the meta-training phase, the model is further optimized in the episodic training [34] manner with learning rate initialized with $\eta_1 = 0.005$ and $\eta_2 = 0.05$ for f_θ and W_b, respectively. η_1 and η_2 are dropped by 0.5 every 10 epochs. We train 40/120 epochs for *mini*-ImageNet/*tiered*-ImageNet where each epoch contains 400 randomly sampled few-shot classification tasks. When the proposed meta-training Algorithm 2 is applied in the meta-training phase, for effective gradient backpropagation, we set the prototype refinement learning rate $\alpha = 1.0$ and the number of refinement steps $n_{\text{steps}} = 1$.

When the proposed model adaptation mechanism is used in the testing phase, we set batch size $B = 100$, temperature $\tau_1 = 0.07$ and $\tau_2 = 4$ for loss function $\mathcal{L}_{ctr}$ and $\mathcal{L}_{dst}$. Weights $w_{cls}, w_{ctr}, w_{dst}$ of different loss functions are selected from $\{0.1, 1.0\}$, $\{0.1, 1.0\}$, $\{1.0, 10.0\}$ according to the performance on validation set. We set the prototype refinement learning rate $\alpha = 0.2$ and the number of steps $n_{\text{steps}} = 20$ during testing.

For experiments on regular semi-supervised/transductive FSL, we directly apply the models trained for S^2I-FSL benchmark (*i.e.*, meta-training with Algorithm 2 + model adaptation during testing) and also use the Sinkhorn-Knopp algorithm [7] during testing for class balancing to fully explore novel class distribution [1, 14, 17].

D Standard Few-Shot Learning

Besides the superiority of the proposed Meta-Inc-Baseline in incremental FSL (Section 5.2), we also evaluate the standard FSL classification performance of Meta-Inc-Baseline. As shown in the upper portions of Table 3 and Table 4, Meta-Inc-Baseline is able to achieve state-of-the-art performance and perform on par with other current FSL methods on both datasets. We emphasize that obtaining top performance on the standard FSL benchmark is not the focus of this work since Meta-Inc-Baseline is trained as an incremental few-shot learner supposed to handle both base and novel classes. Moreover, Meta-Inc-Baseline does not require any extra modules [12, 19] or auxiliary tasks [22, 33] specialized to standard FSL classification. Due to the dataset split of *tiered*-ImageNet for incremental FSL (Appendix A), more than 10% of the training data in standard FSL are used for the base class val/test set. Thus, Meta-Inc-Baseline should yield higher accuracy with the full training set.

Method	Backbone	Purpose	*mini*-ImageNet 5-way	
			1-shot	5-shot
MatchNet§ [34]	ResNet18	standard FSL	52.91±0.88	68.88±0.69
ProtoNet§ [31]	ResNet18		54.16±0.82	73.68±0.65
AFHN [19]	ResNet18		62.38±0.72	78.16±0.56
TADAM [24]	ResNet12		58.50±0.30	76.70±0.30
MetaOpt [18]	ResNet12		62.64±0.62	78.63±0.46
CAN [13]	ResNet12		63.85±0.48	79.44±0.34
DSN [30]	ResNet12		62.64±0.66	78.83±0.45
Neg-Cos [23]	ResNet12		63.85±0.81	81.57±0.56
Meta-Baseline [6]	ResNet12		63.17±0.23	79.26±0.17
RFS-simple [33]	ResNet12		62.02±0.63	79.64±0.44
RFS-distill [33]	ResNet12		**64.82±0.60**	**82.14±0.43**
LwoF♢ [9]	ResNet12	incremental FSL	55.45±0.89	70.92±0.35
Attractor [27]	ResNet12		55.75±0.51	70.14±0.44
XtarNet [40]	ResNet12		60.03±0.30	75.03±0.30
Meta-Inc-Baseline	ResNet12		**62.81±0.63**	**80.18±0.46**

§: results from [6]. ♢: results from [40]. Gray: Meta-Inc-Baseline is modified from [6].

Table 3: Standard few-shot classification results on *mini*-ImageNet. The two emphasized rows mean that our Meta-Inc-Baseline is modified from the simple but strong FSL baseline Meta-Baseline [6] and performs on par with their results. This accounts for the superiority of our approach compared to other incremental FSL methods on standard & incremental FSL.

In addition, it can be seen in the lower portions of the tables that Meta-Inc-Baseline outperforms other incremental FSL methods by a significant margin. It demonstrates the effectiveness of Algorithm 2 to further meta-train the feature extractor instead of training extra modules with the pre-trained backbone fixed (for preserving base knowledge) as in [9, 27, 40]. As verified in previous works [6, 22], there is a trade-off between the transferability of learned features to novel classes which is the only focus of standard FSL and the discriminability of base classes. By meta-training f_θ in Algorithm 2, learned features

can trade-off automatically between discriminability of base classes and transferability of novel classes to achieve higher classification accuracy. This observation is also consistent with the current finding in standard FSL that a well-trained feature extractor instead of other complicated extra modules is the key component for good performance [5, 10, 33].

Method	Backbone	Purpose	*tiered*-ImageNet 5-way	
			1-shot	5-shot
LEO [29]	WRN28	standard FSL	66.33±0.05	81.44±0.09
MetaOpt [18]	ResNet12		65.99±0.72	81.56±0.53
TapNet [39]	ResNet12		63.08±0.15	80.26±0.12
CAN [14]	ResNet12		69.89±0.51	84.23±0.37
DSN [30]	ResNet12		66.22±0.75	82.79±0.48
Meta-Baseline [6]	ResNet12		68.62±0.27	83.29±0.18
RFS-simple [33]	ResNet12		69.74±0.72	84.41±0.55
RFS-distill [33]	ResNet12		**71.52±0.69**	**86.03±0.49**
LwoF$^{\diamond\dagger}$ [9]	ResNet12	incremental FSL	59.79±0.68	75.77±0.54
XtarNet† [40]	ResNet12		63.08±0.30	79.20±0.30
Meta-Inc-Baseline†	ResNet12		**68.33±0.72**	**83.91±0.50**

$^{\diamond}$: results from [40]. Gray: Meta-Inc-Baseline is modified from Meta-Baseline [6].
†: less than 90% of the training data is used due to the dataset split for incremental FSL.

Table 4: Standard few-shot classification results on *tiered*-ImageNet. The two emphasized rows mean that our Meta-Inc-Baseline is modified from the simple but strong FSL baseline Meta-Baseline [6] and performs on par with their results. This accounts for the superiority of our approach compared to other incremental FSL methods on standard & incremental FSL.

E Incremental Few-shot Learning

To verify the extensiveness of our proposed method from the first setting of incremental FSL [9, 27, 40] (two-stage) to the second setting [23, 32, 43] with multiple incremental sessions, we conduct experiments on the latter. Since the second setting does not provide additional unlabeled data, our proposed S^{2}I-FSL benchmark can not be directly evaluated on this setting. However, our baseline method Meta-Inc-Baseline can be used to compare the result with other state-of-the-art methods on this benchmark without utilizing unlabeled samples. It is noteworthy that generalizing Meta-Inc-Baseline from the two-stage incremental learning to the one with multiple stages is straightforward: (1) We first pre-train and meta-train (fake novel training is used as in the first setting of incremental FSL on *mini*-ImageNet) our model as discussed in Section 4.1 of the main paper. (2) When each incremental session comes, corresponding novel class weights are generated using Eq. 2 in Section 4.1 sequentially. We follow all the settings of previous works [23, 32, 43] including data splits and backbones, and conduct experiments on *mini*-ImageNet , CIFAR100 and CUB200 [35]. As shown in Table 5 and Fig. 1, our Meta-Inc-Baseline outperforms previous works significantly on all the datasets, which validates that Meta-Inc-Baseline is generalizable and effective. Since our proposed components for S^{2}I-FSL are designed based on Meta-Inc-Baseline, they can also generalize to incremental FSL with multi sessions given additional unlabeled data.

F Semi-Supervised/Transductive Few-Shot Learning

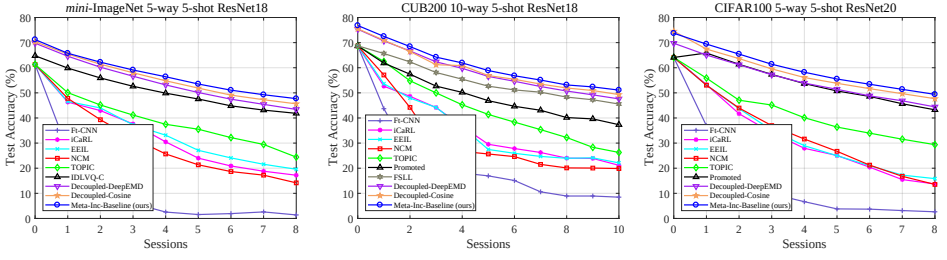
Besides semi-supervised/transductive FSL results on *mini*-ImageNet in Section 5.2, we provide additional experimental results on *tiered*-ImageNet. Following the most common setting in semi-supervised FSL [20, 21, 36], the unlabeled set $\mathcal{U}$ in semi-supervised FSL contains 5×30 / 5×50 unlabeled novel class samples for 1/5-shot setting.

As shown in Table 6, our model trained for S^{2}I-FSL is directly applied to regular transductive and semi-supervised FSL benchmarks on *tiered*-ImageNet, respectively. It is obvious

Method	Acc. in each session (%)								
	0	1	2	3	4	5	6	7	8
Ft-CNN[♯]	61.31	27.22	16.37	6.08	2.54	1.56	1.93	2.60	1.40
iCaRL[♯] [23]	61.31	46.32	42.94	37.63	30.49	24.00	20.89	18.80	17.21
EEIL[♯] [6]	61.31	46.58	44.00	37.29	33.14	27.12	24.10	21.57	19.58
NCM[♯] [23]	61.31	47.80	39.31	31.91	25.68	21.35	18.67	17.24	14.17
TOPIC [32]	61.31	50.09	45.17	41.16	37.48	35.52	32.19	29.46	24.42
IDLVQ-C [6]	64.77	59.87	55.93	52.62	49.88	47.55	44.83	43.14	41.84
Decoupled-DeepEMD[◇] [43]	69.77	64.59	60.21	56.63	53.16	50.13	47.49	45.42	43.41
Decoupled-Cosine [43]	70.37	65.45	61.41	58.00	54.81	51.89	49.10	47.27	45.63
Meta-Inc-Baseline (ours)	**71.20**	**65.77**	**62.21**	**59.11**	**56.41**	**53.52**	**51.07**	**49.26**	**47.70**

[♯]: results from [32]. [◇]: results from [43]. Following [32] ResNet18 is used as backbone model on *mini*-ImageNet.

Table 5: Results of 5-way 5-shot incremental FSL with multi sessions on *mini*-ImageNet.



(a) Results on *mini*-ImageNet (b) Results on CUB200 (c) Results on CIFAR100

Figure 1: Comparison with other SOTA methods on: (a) *mini*-ImageNet (b) CUB200 and (c) CIFAR100. Our Meta-Inc-Baseline outperforms previous works significantly.

that the proposed approach consistently outperforms or performs on par with other state-of-the-art methods using the same backbone. It is noteworthy that, due to the dataset split of *tiered*-ImageNet for incremental FSL (Appendix A), more than 10% of the training data in standard FSL are used for the base class val/test set. Thus, our proposed approach should yield higher accuracy with the full training set. These results further verify the effectiveness of the proposed Algorithm 2 for effectively utilizing unlabeled samples and the model adaptation mechanism for learning discriminative features of novel classes.

Method	f_θ	Transductive FSL 1-shot	Transductive FSL 5-shot	Method	f_θ	Semi-supervised FSL 1-shot	Semi-supervised FSL 5-shot
CAN+Top-k [22]	Res12	73.21±0.58	84.93±0.38	Soft k-M[*] [28]	Res12	68.60±N/A	81.00±N/A
DPGN [37]	Res12	72.45±0.51	87.24±0.39	LST [20]	Res12	77.70±1.60	85.20±0.80
EPNet [28]	Res12	76.53±0.87	87.32±0.64	TACO [33]	Res12	75.53±N/A	85.72±N/A
ECKPN [8]	Res12	73.59±0.45	88.13±0.28	MCT [14]	Res12	76.90±0.70	86.30±0.50
Completion [41]	Res12	81.04±0.89	87.42±0.57	EPNet [28]	Res12	81.79±0.97	88.45±0.61
LR + ICI [34]	Res12	80.79±N/A	87.92±N/A	LR + ICI [34]	Res12	84.01±N/A	89.00±N/A
Our proposed[†]	Res12	**82.42±0.80**	**88.25±0.47**	Our proposed[†]	Res12	**84.21±0.76**	**89.17±0.44**

[*]: results from [20]. [†]: less than 90% of the training data is used due to the dataset split for incremental FSL.

Table 6: Regular semi-supervised/transductive FSL results on *tiered*-ImageNet.

G More Experiments on S^{2}I-FSL

G.1 Additional Ablation Studies

For further verifying the effectiveness of each proposed component and providing the method without adaptation in regular transductive setting, we conduct additional ablation studies on

each component proposed for S^{2}I-FSL on *tiered*-ImageNet (transductive setting) in addition to Table 5 in the main paper (semi-supervised setting).

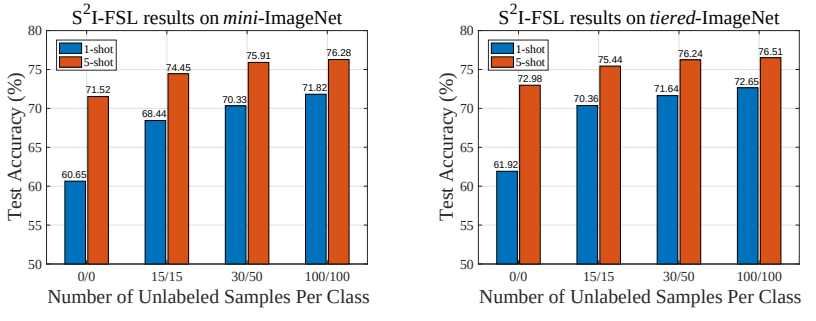
Method (Transductive)	Proto refine	Fake unlabel	Model adapt	200+5-way 1-shot Acc.	Δ	$Acc_{n/n}$	200+5-way 5-shot Acc.	Δ	$Acc_{n/n}$
Meta-Inc-Baseline				61.92	-7.87	67.79	72.98	-5.39	83.39
Meta-Inc-Baseline + PR	✓			57.99	-15.58	74.75	72.08	-7.47	85.74
Our method w/o adaptation	✓	✓		68.12	-7.02	77.97	74.95	-4.87	86.41
Our full method	✓	✓	✓	**70.03**	**-5.41**	**78.37**	**75.73**	**-4.53**	**87.12**

Table 7: Ablation study of each component proposed for S^{2}I-FSL on *tiered*-ImageNet.

As shown in Table 7, our proposed meta-training alleviates the severe confusion of Meta-Inc-Baseline + PR and gets 10.1%/2.9% absolute performance gains in 1/5-shot cases. Moreover, model adaptation mechanism that learns discriminative features for novel classes yields another 1.9%/0.8% improvement. The above observations are consistent with those in Table 5 of the main paper, which further validates the superiority of the proposed algorithms.

G.2 Studies on Unlabeled Set

Experiments on the number of samples in $\mathcal{U}$. As shown in Fig. 2, we vary the cardinality of the unlabeled set $\mathcal{U}$ on both *mini*-ImageNet and *tiered*-ImageNet. In this experiment, unlabeled base and novel class samples in $\mathcal{U}$ are maintained in equal proportion. For example, in the $N_b + N$-way 1/5-shot classification task ($N_b = 64$ or 200, $N = 5$), if unlabeled set $\mathcal{U} = \mathcal{U}_b \cup \mathcal{U}_n$ in S^{2}I-FSL contains 5×30 / 5×50 (x-axis) novel class samples ($\mathcal{U}_n$) from $\mathcal{D}_{\text{novel/test}}$, 150 / 250 additional base class samples ($\mathcal{U}_b$) are sampled from $\mathcal{D}_{\text{base/test}}$ uniformly. It can be observed that our proposed approach can achieve better performance with more unlabeled samples in $\mathcal{U}$, which indicates the effectiveness of the method in mining auxiliary information from the unlabeled set for semi-supervised incremental few-shot learning.



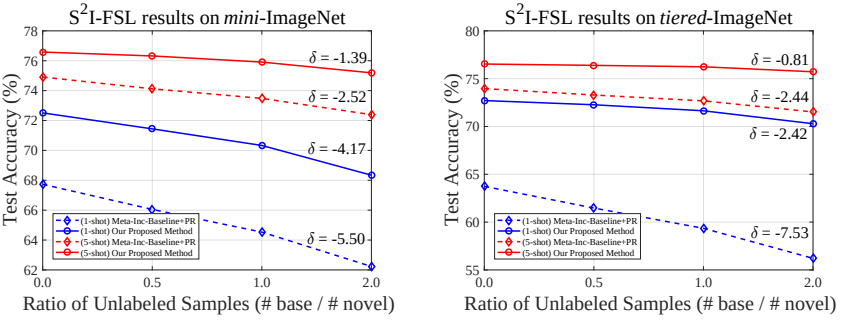
(a) Results on *mini*-ImageNet (b) Results on *tiered*-ImageNet

Figure 2: Studies on the number of samples in the unlabeled set $\mathcal{U}$. (a) Results on *mini*-ImageNet; (b) Results on *tiered*-ImageNet.

Experiments on the ratio of base and novel class samples in $\mathcal{U}$. In the previous settings, unlabeled base and novel class samples in $\mathcal{U}$ are maintained in equal proportion, which may not always be the case in real applications. As mentioned in Section 1, one of the main challenges in S^{2}I-FSL is that the unbalanced sample number of different classes in unlabeled set (*e.g.*, relatively large amount of base class samples and few novel ones) instead of a balanced number in each novel class makes it harder to learn the classifier. Hence, we change the ratio ρ of the base and novel class samples in $\mathcal{U}$ and ρ is defined as follows:

$$\rho = \frac{|\mathcal{U}_b|}{|\mathcal{U}_n|} = \frac{\text{number of base class samples in } \mathcal{U}}{\text{number of novel class samples in } \mathcal{U}}, \tag{1}$$

where the total number of samples in $\mathcal{U}$ is fixed, *i.e.*, $|\mathcal{U}| = 300/500$ for 1/5-shot setting. As shown in Fig. 3, the accuracy decreases when fewer novel class samples are included in $\mathcal{U}$. It is because that the more unbalanced samples between base and novel classes, the more challenging the task becomes. Compared to Meta-Inc-Baseline + PR, our method is more robust to changes in sample ratio, as less degradation (δ) is observed on both datasets. The robustness of our method can be mainly attributed to the proposed meta-training Algorithm 2 with fake unlabeled data for effectively utilizing unlabeled samples in $\mathcal{U}$.



(a) Results on *mini*-ImageNet (b) Results on *tiered*-ImageNet

Figure 3: Studies on the sample ratio ρ in the unlabeled set $\mathcal{U}$. (a) Results on *mini*-ImageNet; (b) Results on *tiered*-ImageNet. In this experiment, the number of samples in $\mathcal{U}$ is fixed. Concretely, the degradation metric (δ) denotes the performance gap between $\rho = 0.0$ and $\rho = 2.0$, *i.e.*, $\delta = \text{Acc}|_{(\rho=2.0)} - \text{Acc}|_{(\rho=0.0)}$.

Experiments on the hard testing scenario. We empirically find that confusion between novel classes is also an issue. We evaluate on *tiered*-ImageNet S^{2}I-FSL (semi-supervised 1/5-shot) where novel classes consist of classes that are similar to each other. Degraded performance on joint accuracy is observed: 68.73/74.05% v.s. original 71.64/76.24%.

References

[1] Malik Boudiaf, Ziko Imtiaz Masud, Jérôme Rony, José Dolz, and et al. Transductive information maximization for few-shot learning. In *NeurIPS*, 2020.

[2] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *ECCV*, 2018.

[3] Chaofan Chen, Xiaoshan Yang, Changsheng Xu, and et al. Eckpn: Explicit class knowledge propagation network for transductive few-shot learning. In *CVPR*, 2021.

[4] Kuilin Chen and Chi-Guhn Lee. Incremental few-shot learning via vector quantization in deep embedded space. In *ICLR*, 2021.

[5] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019.

[6] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *ArXiv*, abs/2003.04390, 2020.

[7] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *NeurIPS*, 2013.

[8] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Dropblock: A regularization method for convolutional networks. In *NeurIPS*, 2018.

[9] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018.

[10] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Perez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *ICCV*, 2019.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[12] Ruibing Hou, Hong Chang, MA Bingpeng, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. In *NeurIPS*, 2019.

[13] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *CVPR*, 2019.

[14] Yuqing Hu, Vincent Gripon, and Stéphane Pateux. Leveraging the feature distribution in transfer-based few-shot learning. *arXiv preprint arXiv:2006.03806*, 2020.

[15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[16] Seong Min Kye, Hae Beom Lee, Hoirin Kim, and Sung Ju Hwang. Meta-learned confidence for few-shot learning, 2020.

[17] Michalis Lazarou, Yannis Avrithis, and Tania Stathaki. Iterative label cleaning for transductive and semi-supervised few-shot learning, 2020.

[18] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *CVPR*, 2019.

[19] Kai Li, Yulun Zhang, Kunpeng Li, and Yun Fu. Adversarial feature hallucination networks for few-shot learning. In *CVPR*, 2020.

[20] Xinzhe Li, Qianru Sun, Yaoyao Liu, and et al. Learning to self-train for semi-supervised few-shot classification. *NeurIPS*, 2019.

[21] Moshe Lichtenstein, Prasanna Sattigeri, Rogerio Feris, and et al. Tafssl: Task-adaptive feature sub-space learning for few-shot classification. In *ECCV*, 2020.

[22] Bin Liu, Yue Cao, Yutong Lin, and et al. Negative margin matters: Understanding margin in few-shot classification. In *ECCV*, 2020.

[23] Pratik Mazumder, Pravendra Singh, and Piyush Rai. Few-shot lifelong learning. In *AAAI*, 2021.

[24] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *NeurIPS*, 2018.

[25] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, 2017.

[26] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018.

[27] Mengye Ren, Renjie Liao, Ethan Fetaya, and Richard Zemel. Incremental few-shot learning with attention attractor networks. In *NeurIPS*, 2019.

[28] Pau Rodríguez, Issam Laradji, Alexandre Drouin, and Alexandre Lacoste. Embedding propagation: Smoother manifold for few-shot classification. In *ECCV*, 2020.

[29] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, and et al. Meta-learning with latent embedding optimization. In *ICLR*, 2019.

[30] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. Adaptive subspaces for few-shot learning. In *CVPR*, 2020.

[31] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[32] Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *CVPR*, 2020.

[33] Yonglong Tian, Yue Wang, Dilip Krishnan, and et al. Rethinking few-shot image classification: a good embedding is all you need? In *ECCV*, 2020.

[34] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NeurIPS*, 2016.

[35] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. *Technical report*, 2011.

[36] Yikai Wang, Chengming Xu, Chen Liu, Li Zhang, and Yanwei Fu. Instance credibility inference for few-shot learning. In *CVPR*, 2020.

[37] Ling Yang, Liangliang Li, Zilun Zhang, Xinyu Zhou, Erjin Zhou, and Yu Liu. Dpgn: Distribution propagation graph network for few-shot learning. In *CVPR*, 2020.

[38] Han-Jia Ye, Xin-Chun Li, and De-Chuan Zhan. Task cooperation for semi-supervised few-shot learning. In *AAAI*, 2021.

[39] Sung Whan Yoon, Jun Seo, and Jaekyun Moon. TapNet: Neural network augmented with task-adaptive projection for few-shot learning. In *ICML*, 2019.

[40] Sung Whan Yoon, Do-Yeon Kim, Jun Seo, and Jaekyun Moon. Xtarnet: Learning to extract task-adaptive representation for incremental few-shot learning. In *ICML*, 2020.

[41] Baoquan Zhang, Xutao Li, Yunming Ye, Zhichao Huang, and Lisai Zhang. Prototype completion with primitive knowledge for few-shot learning. In *CVPR*, 2021.

[42] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *CVPR*, 2020.

[43] Chi Zhang, Nan Song, Guosheng Lin, Yun Zheng, Pan Pan, and Yinghui Xu. Few-shot incremental learning with continually evolved classifiers. In *CVPR*, 2021.