

SELF-SUPERVISED LEARNING FOR FEW-SHOT IMAGE CLASSIFICATION

Da Chen*, Yuefeng Chen*, Yuhong Li, Feng Mao, Yuan He, Hui Xue

Alibaba Group, China

{chen.cd, yuefeng.chenyf, daniel.lyh, maofeng.mf, heyuan.hy, hui.xueh@alibaba-inc.com}

ABSTRACT

Few-shot image classification aims to classify unseen classes with limited labeled samples. Recent works benefit from the meta-learning process with episodic tasks and can fast adapt to class from training to testing. Due to the limited number of samples for each task, the initial embedding network for meta learning becomes an essential component and can largely affects the performance in practice. To this end, many pre-trained methods have been proposed, and most of them are trained in supervised way with limited transfer ability for unseen classes. In this paper, we proposed to train a more generalized embedding network with self-supervised learning (SSL) which can provide slow and robust representation for downstream tasks by learning from the data itself. We evaluate our work by extensive comparisons with previous baseline methods on two few-shot classification datasets (*i.e.*, MiniImageNet and CUB). Based on the evaluation results, the proposed method achieves significantly better performance, *i.e.*, improve 1-shot and 5-shot tasks by nearly **3%** and **4%** on MiniImageNet, by nearly **9%** and **3%** on CUB. Moreover, the proposed method can gain the improvement of (**15%**, **13%**) on MiniImageNet and (**15%**, **8%**) on CUB by pre-training using more unlabeled data. Our code will be available at <https://github.com/phcey/ssl-few-shot>.

Index Terms— Self-supervised learning, Few-shot learning, Embedding network, Metric learning, Image classification

1. INTRODUCTION

Recent advances of deep learning techniques have made significant progresses in many areas. The main reason for such success is the ability to train a deep model that can retain profound knowledge from large scale labeled dataset [1, 2]. This is somehow against the human learning behavior - one can easily classify objects from just a few examples with limited prior knowledge. How to computationally model such behavior motivates the recent researches in few-shot learning, where the focus is how to adapt the model to new data or tasks with restricted number of instances.

One popular solution for few-shot classification is to apply a meta-learning process, in which the dataset is divided into subsets for different meta tasks, in order to learn how to adapt the model according to the task change. The main challenge here is that the meta-learning could easily lead to over fitting, as only a few samples for each class are available. To solve this problem, an attention mechanism that achieves a good classification for the unlabeled samples by learning an embedding of the labeled ones has been proposed [3]. By sub-sampling classes and the associated data therein to simulate few-shot tasks [4, 5, 6], the so called episodes can further benefit meta-learning by constructing a probability model to predict a decision

boundary between classes. Recent meta-learning methods [7, 8] focus on retrieving transferable embedding from the dataset along with the relation between images and their class descriptions. This is done by decomposing the training into two stages, *i.e.*, i). learning a robust and transferable embedding, and ii). fine-tuning the learned embedding for downstream classification task. All these works demonstrate that a robust pre-trained embedding network is essential to the performance of the few-shot image classification task.

Current methods [9, 10, 8, 11] with good performance mostly apply a ResNet12 [12] or a wide ResNet [13] as the embedding network and surpass the methods [14, 15] with deeper network. We argue that the abandon of large network are mainly because that all these methods are trained in a supervised way with limited labeled samples. In this paper, we propose to apply a much more larger embedding network with self-supervised learning (SSL) to incorporate with episodic task based meta-learning. According to the evaluation presented in Section 4, the proposed method can significantly improve few-shot image classification performance over baseline methods in two common datasets. As a remark, under the same experimnt setting, the proposed method improves 1-shot and 5-shot tasks by nearly **3%** and **4%** on MiniImageNet, by nearly **9%** and **3%** on CUB. Moreover, the proposed method can gain the improvement of **15%**, **13%** and **15%**, **8%** in two tasks on MiniImageNet and CUB dataset by pretraining using more unlabeled data. We also observe that self-supervised learning pre-train model can be robustly transferred to other dataset.

2. RELATED WORK

Few-shot learning as an active research topic has been extensively studied. In this paper, we will primarily review recent deep-learning based approaches that are more relevant to our work. A number of works aim to improve the robustness of the training process. Garcia *et al.* [16] propose a graph neural network according to the generic message passing inference method. Zhao *et al.* [17] split the features to three orthogonal parts to improve the classification performance for few-shot learning, allowing simultaneous feature selection and dense estimation. Chen *et al.* [15] propose a Self-Jig algorithm to augment the input data in few-shot learning by synthesizing new images that are either labeled or unlabeled.

A popular strategy for few-shot learning is through meta-learning (also called learning-to-learn) with multi-auxiliary tasks [3, 18, 6, 8, 9]. The key is how to robustly accelerate the learning progress of the network without suffering from over-fitting with limited training data. Finn *et al.* propose MAML [18] to search the best initial weights through gradient decent for network training, making the fine-tuning easier. REPTILE [19] simplifies the complex computation of MAML by incorporating an L_2 loss, but still performs in high dimension space. To reduce the complexity, Rusu *et al.* propose a network called LEO [11] to learn a low dimension latent embed-

* Equal Contribution.

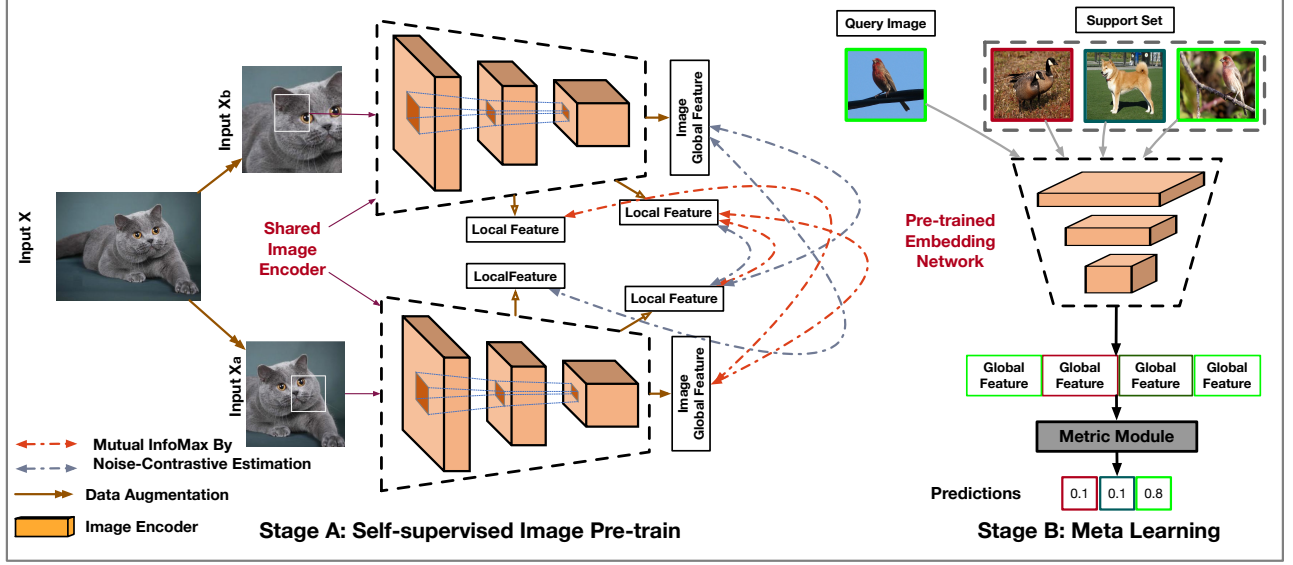


Fig. 1. The overall architecture of our approach. LEFT: Train embedding network by Self-Supervised learning (AMDIM). The pretext task is designed to maximize the mutual information between two views $\langle x_a, x_b \rangle$ generated from the same image x by data augmentation. Right: Meta-learning with episodic task (3-way, 1-shot example). For each task, the training samples and query samples are encoded by the embedding network. Query sample embeddings are compared with centroid of training sample embeddings and make further prediction.

ding of the model. CAML [9] extends MAML by partitioning the model parameters into context parameters and shared parameters, enabling a bigger network without over-fitting. Another stream of meta-learning based approaches [5, 10, 3, 6] attempt to learn a deep embedding model that can effectively project the input samples to a specific feature space. Then the samples can be classified by the nearest neighbour (NN) criterion using a distance function such as Cosine distance or Euclidean distance, *etc.* Koch *et al.* [20] propose the Siamese network to extract embedding features from input images and converge images from the same class. Matching Network [3] utilizes an augment neural network for feature embedding, forming the basis for metric learning.

Self-supervised learning (SSL) aims to learn robust representations from the data itself without class labels. The main challenge here is how to design the pretext tasks that are complex enough to exploit high-level compact semantic visual representations that are useful for solving downstream tasks. This is consistent with the mission of the pre-trained embedding network in few shot learning. The work of [21] revisited some state-of-the-art methods based on various classification based pretext tasks (*e.g.*, Rotation, Exemplar, RelPatchLoc, Jigsaw). Recently, by maximizing mutual information between features extracted from multiple views of a shared context [22, 23, 24, 25], SSL archive comparable performance to supervised learning. Among them, Contrastive Predictive Coding [22, 23] learns from two views (the past and future) and is applicable to sequential data. Contrastive Multiview Coding [24] extends the framework to learn representations from multiple views of a dataset. AMDIM [25] learns features extracted from multiples views which produced by repeatedly applying data augmentation on the input image, achieve 68.1% accuracy on ImageNet.

3. METHOD

Few-shot learning is a challenging problem as it has only limited data for training and need to verify the performance on the data for

unseen classes. One popular solution for few-shot learning classification problem is to apply a meta learning on top of a pre-trained embedding network. Most of current methods are mainly focusing on the second stage *i.e.*, *meta learning stage*. In this work, we follow this two stages paradigm but utilize self-supervised learning to train a large embedding network as our strong base.

3.1. Self-supervised learning stage

Our goal is to learn representations that enhance the feature’s generalization. In our approach, we use Augmented Multiscale Deep InfoMax (AMDIM) [25] as our self-supervised model. The pretext task is designed to maximize the mutual information between features extracted from multiple views of a shared context.

The mutual information (MI) measures the shared information between two random variables X and Y which is defined as the KullbackLeibler (KL) divergence between the joint and the product of the marginals.

$$\begin{aligned} I(X, Y) &= D_{KL}(p(x, y) || p(x)p(y)) \\ &= \sum \sum p(x, y) \log \frac{p(x|y)}{p(x)} \end{aligned} \quad (1)$$

where $P(x, y)$ is the joint distribution and $P(x)$ and $P(y)$ are the marginal distributions of X and Y . Estimating MI is challenging as we just have samples but not direct access to the underlying distribution. [22] proved that we can maximize a lower bound on mutual information by minimizing the Noise Contrastive Estimation (NCE) loss based on negative sampling.

The core concept of AMDIM is to maximize mutual information between global features and local features from two views (x_a, x_b) of the same image. Specifically, maximize mutual information between $\langle f_g(x_a), f_5(x_b) \rangle$, $\langle f_g(x_a), f_7(x_b) \rangle$ and $\langle f_5(x_a), f_5(x_b) \rangle$. Where f_g is the global feature, f_5 is encoder’s 5×5 local feature map as well as f_7 as the encoder’s 7×7 feature map. For example,

the NCE loss between $f_g(x_a)$ and $f_5(x_b)$ is defined as below:

$$\mathcal{L}_{amdim}(f_g(x_a), f_5(x_b)) = -\log \frac{\exp\{\phi(f_g(x_a), f_5(x_b))\}}{\sum_{\tilde{x}_b \in \mathcal{N}_x \cup x_b} \exp\{\phi(f_g(x_a), f_5(\tilde{x}_b))\}} \quad (2)$$

\mathcal{N}_x are the negative samples of image x , ϕ is the distance metric function. At last, the overall loss between x_a and x_b is as follows:

$$\mathcal{L}_{amdim}(x_a, x_b) = \mathcal{L}_{amdim}(f_g(x_a), f_5(x_b)) + \mathcal{L}_{amdim}(f_g(x_a), f_7(x_b)) + \mathcal{L}_{amdim}(f_5(x_a), f_5(x_b)) \quad (3)$$

The stage as shown in the Figure 1 gives a overview of the AMDIM self supervised learning method. The red and blue lines shows the local and global feature between two view x_a and x_b . The detail of encoder network is defined in Table 1.

Layers	Output Size	ConvBlocks (kernel, output_channels, stride, pad)
conv1	62 × 62	[5 × 5, ndf, 2, 2] [3 × 3, ndf, 1, 0]
conv2_x	30 × 30	[4 × 4, 2 × ndf, 2, 0] [1 × 1, 2 × ndf, 1, 0] × (2 × ndepth − 1)
conv3_x	14 × 14	[4 × 4, 4 × ndf, 2, 0] [1 × 1, 4 × ndf, 1, 0] × (2 × ndepth − 1)
conv4_x	7 × 7	[2 × 2, 8 × ndf, 2, 0] [1 × 1, 8 × ndf, 1, 0] × (2 × ndepth − 1)
conv5_x	5 × 5	[3 × 3, 8 × ndf, 1, 0] [1 × 1, 8 × ndf, 1, 0] × (2 × ndepth − 1)
conv6_x	5 × 5	[3 × 3, 8 × ndf, 1, 0] [1 × 1, 8 × ndf, 1, 0] × (2 × ndepth − 1)
conv7	1 × 1	[3 × 3, nrkhs, 1, 0] [1 × 1, nrkhs, 1, 0]
# params.		198M(ndf=192, nrkhs=1536, ndepth=8)
# FLOPs.		10.96 GFLOPs(ndf=192, nrkhs=1536, ndepth=8)

Table 1. Model Architecture of AmdimNet. **ndf** is the output channels’ parameter of the network. **ndepth** controls the model’s depth. **nrkhs** is the embedding dimension. Each convolution blocks in conv2_x to conv6_x contains 2 × **ndepth** convolution layers.

3.2. Meta-learning stage

Given an embedding network, meta-learning is applied to fine-tune it to fit the class changes requirement of few-shot classification. A typical meta learning can be considered as a K -way C -shot episodic classification problem with multi-tasks [3]. For each classification task T , we have K classes with C samples from each class. The entire training dataset can be presented by $D = \{(x_1, y_1), \dots, (x_N, y_N)\}$ where N is the total number of classes in D . For a specific task T , $V = \{y_i | i = 1, \dots, K\}$ denotes the class labels associated therein. Here K is the number of classes in support set for a single training task. The support set and query set can often be randomly selected from D : (a) the support set for task T is denoted by $S = \{(x_i, y_i) | i = 1, \dots, m\}$, where $m = C \times K$ (K -way C -shot); (b) the query set is $Q = \{(x_j, y_j) | j = 1, \dots, n\}$ where n is the number of samples selected for meta testing.

As mentioned in Section 2, the recent popular frameworks such as Snell *et al.* [5], are able to learn an embedding function to map all input samples to a mean vector c in a description space to represent each class. For class k , it is represented by the centroid of embedding features of training samples and can be obtained as:

$$c_k = \frac{1}{|S|} \sum_{(x_i, y_i) \in S} f_g(x_i), \quad (4)$$

where $f_g(x_i)$ is the embedding function.

As a metric learning based method, we employ a distance function d and produce a distribution over all classes given a query sample q from query set Q :

$$p(y = k | q) = \frac{\exp(-d(f_g(q), c_k))}{\sum_{k'} \exp(-d(f_g(q), c_{k'}))} \quad (5)$$

In this paper, Euclidean distance is chosen as distance function d . As shown in Eq. 5, the distribution is based on a softmax over distance between the embedding of the samples (in the query set) and the reconstructed features of the class. The loss in meta learning stage can then read:

$$\mathcal{L}_{meta} = d(f_g(q), c_k) + \log \sum_{k'} d(f_g(q), c_{k'}) \quad (6)$$

4. EXPERIMENTAL RESULTS

In this section, we first introduce the dataset and training process used in our evaluation, then show quantitative comparisons against other baseline methods, finally we conduct a detailed study to validate the transfer ability of our approach.

4.1. Datasets

MiniImageNet dataset, as proposed in [3], is a benchmark to evaluate the performance of few-shot learning methods. This dataset is a subset randomly selected from ImageNet. MiniImageNet contains 60,000 images from only 100 classes, and each class has 600 images. We follow the data split strategy in [4] to sample images of 64 classes for training, 16 classes for validation, 20 classes for test.

Catech-UCSD Birds-200-2011(CUB-200-2011) dataset, proposed in [26], is a dataset for fine-grained classification. The CUB-200-2011 dataset contains 200 classes of birds with 11788 images in total. For evaluation, we follow the split in [27]. 200 species of birds are randomly split to 100 classes for training, 50 classes for validation, and 50 classes for test.

4.2. Training Details

Several recent works show that a typical training process can include a pre-trained network [8, 11] or employ co-training [10] for feature embedding. This can significantly improve the classification accuracy. In this paper, we adopt the AMDIM [25] SSL training framework to pre-train the feature embedding network. AmdimNet(ndf=192, ndepth=8, nrkhs=1536) is used for all datasets and the embedding dimension is 1536. Adam is chosen as the optimizer with a learning rate of 0.0002. We use 128×128 as the input resolution among these datasets. For MiniImageNet dataset, 3 embedding models are trained. **Mini80-SSL** is self-supervised trained from 48,000 images (80 classes training and validation) without labels. **Mini80-SL** is supervised training using same AmdimNet by cross entropy loss with labels. **Image900-SSL** is SSL trained from all images from ImageNet1K except MiniImageNet. For CUB dataset, **CUB150-SSL** is trained by SSL from 150 classes (training and validation). **CUB150-SL** is the supervised trained model. **Image1K-SSL** is SSL trained from all images from ImageNet1K without label.

Baselines	Embedding Net	1-Shot 5-Way	5-Shot 5-Way
MatchingNet [3]	4 Conv	43.56 \pm 0.84%	55.31 \pm 0.73%
MAML [18]	4 Conv	48.70 \pm 1.84%	63.11 \pm 0.92%
RelationNet [6]	4 Conv	50.44 \pm 0.82%	65.32 \pm 0.70%
REPTILE [19]	4 Conv	49.97 \pm 0.32%	65.99 \pm 0.58%
ProtoNet [5]	4 Conv	49.42 \pm 0.78%	68.20 \pm 0.66%
Baseline* [28]	4 Conv	41.08 \pm 0.70%	54.50% \pm 0.66
Spot&learn [29]	4 Conv	51.03 \pm 0.78%	67.96% \pm 0.71
DN4 [30]	4 Conv	51.24 \pm 0.74%	71.02% \pm 0.64
SNAIL [31]	ResNet12	55.71 \pm 0.99%	68.88 \pm 0.92%
ProtoNet ⁺ [5]	ResNet12	56.50 \pm 0.40%	74.2 \pm 0.20%
CAML [9]	ResNet12	59.23 \pm 0.99%	72.35 \pm 0.71%
TPN [32]	ResNet12	59.46%	75.65%
MTL [33]	ResNet12	61.20 \pm 1.8%	75.50 \pm 0.8%
DN4 [30]	ResNet12	54.37 \pm 0.36%	74.44 \pm 0.29%
TADAM [10]	ResNet12	58.50%	76.70%
Qiao-WRN [8]	Wide-ResNet28	59.60 \pm 0.41%	73.74 \pm 0.19%
LEO [11]	Wide-ResNet28	61.76 \pm 0.08%	77.59 \pm 0.12%
Dis. k-shot [14]	ResNet34	56.30 \pm 0.40%	73.90 \pm 0.30%
Self-Jig(SVM) [15]	ResNet50	58.80 \pm 1.36%	76.71 \pm 0.72%
Ours.Mini80_SL	AmdimNet	43.92 \pm 0.19%	67.13 \pm 0.16%
Ours.Mini80_SSL ⁻	AmdimNet	46.13 \pm 0.17%	70.14 \pm 0.15%
Ours.Mini80_SSL	AmdimNet	64.03 \pm 0.20%	81.15 \pm 0.14%
Ours.Image900_SSL	AmdimNet	76.82 \pm 0.19%	90.98 \pm 0.10%

Table 2. Few-shot classification accuracy results on *MiniImageNet* on 1-shot 5-way, 5-shot 5-way and 10-shot 5-way tasks. All accuracy results are reported with 95% confidence intervals. ‘-’ indicates result without meta-learning.

4.3. Quantitative comparison

For MiniImageNet dataset, we evaluate our method in two common few-shot learning tasks *i.e.*, 1-shot 5-way task and 5-shot 5-way task against 18 baseline methods with different embedding networks including classical ones [5, 3, 18] and recently proposed methods [11, 10, 32]. For CUB dataset, we follow the recent work [28] to evaluate the robustness of the proposed framework with 7 other alternatives on this fine-grained dataset.

As detailed in Table 2, the proposed method outperforms all baselines in the tested tasks. In 1-shot 5-way test, our approach achieves 7.53% and 2.27% improvement over ProtoNet⁺ [5] and LEO [11] respectively. The former is an amended variant of ProtoNet using pre-trained Resnet as embedding network and has same meta-learning stage with the proposed method, the later is the state-of-the-art method. In the experience for 5-Shot 5-Way, we observe a similar improvement in accuracy. Furthermore, we observe that the performance of our proposed method significantly increases when receiving more images/classes as input for pretrain. It gives 81.15% improvement on 5-shot 5-ways test against 64.03% for 1-Shot 5-Way.

Table 3 illustrates our experiment on CUB dataset. Our proposed method yields highest accuracy from all trials. In the 1-shot 5-way test, we have 71.85% gaining a margin of 20.54% increment to the classic ProtoNet [5]. The improvement is more significant for 5-shot 5-way test. Our proposed method results is 84.29% which introduces 2.39% improvement to DN4-Da [30]. Comparing to Baseline++ [28], our method shows a significant improvement, *i.e.*, 11.32% and 4.95% in both tests.

4.4. Ablation Study

As shown in the quantitative evaluation, the proposed method can significantly improve the performance in few-shot classification task by self-supervised pretrain using a large network. One concern may be raised is that if the gain of improvements of proposed network is

Baselines	Embedding Net	1-Shot 5-Way	5-Shot 5-Way
MatchingNet [3]	4 Conv	61.16 \pm 0.89	72.86 \pm 0.70
MAML [18]	4 Conv	55.92 \pm 0.95%	72.09 \pm 0.76%
ProtoNet [5]	4 Conv	51.31 \pm 0.91%	70.77 \pm 0.69%
MACO [27]	4 Conv	60.76%	74.96%
RelationNet [6]	4 Conv	62.45 \pm 0.98%	76.11 \pm 0.69%
Baseline++ [28]	4 Conv	60.53 \pm 0.83%	79.34 \pm 0.61%
DN4-DA [30]	4 Conv	53.15 \pm 0.84%	81.90 \pm 0.60%
Ours.CUB150_SL	AmdimNet	45.10 \pm 0.21%	74.59 \pm 0.16%
Ours.CUB150_SSL ⁻	AmdimNet	40.83 \pm 0.16%	65.27 \pm 0.18%
Ours.CUB150_SSL	AmdimNet	71.85 \pm 0.22%	84.29 \pm 0.15%
Ours.Image1K_SSL	AmdimNet	77.09 \pm 0.21%	89.18 \pm 0.13%

Table 3. Few-shot classification accuracy results on CUB dataset [26] on 1-shot 5-way task, 5-shot 5-way task. All accuracy results are reported with 95% confidence intervals. ‘-’ indicates result without meta-learning. For each task, the best-performing method is highlighted.

simply due to the increment of network’s capacity. To prove the effectiveness of the proposed method, we train the embedding network with labeled data (Mini80-SL and CUB150-SL as detailed in Section 4.2). As shown in Table 2 and Table 3, it performs even worse than the methods with simple 4 Conv blocks embedding networks as such big network under supervised learning with limited data can cause overfitting problem and cannot adjust to new unseen classes during testing. However, with SSL based pre-training a more generalized embedding network can be obtained and improve the results significantly. One may also concern about the effectiveness of the meta learning fine-tuning in the second stage. To test this, the pre-train embedding network is directly applied on the task with nearest neighbourhood(NN) classification. As shown in the test results on both dataset, meta-learning can effectively fine-tune the embedding network and achieve remarkable improvement.

We also include more data without labels during SSL pre-training and observe an more significant improvement of the result. As shown in Table 2, the proposed method can gain the improvement of 15% and 13% in two test tasks. As detailed analyzed in [28], current few-shot learning methods can not efficiently transfer the domain of learning, *i.e.*, the training domain can not have huge gap with the testing set. In this paper, a transferability test is also conducted by pre-training the embedding network on ImageNet and applied on CUB dataset. As shown in Table 3, the proposed method with ImageNet pre-trained embedding network can be efficiently transferred to CUB dataset and gain an improvement of 15%,8% in both test tasks.

5. CONCLUSION

In this paper, we propose to utilizes self-supervised learning to efficiently train a robust embedding network for few-shot image classification. The resulted embedding network is more generalized and more transferable comparing to other baselines. After fine-tuning by meta-learning process, the performance of the proposed method can significantly outperform all baselines based on the quantitative results using two common few-shot classification datasets. The current framework can be extended in several ways in the future. For instance, one direction is to combine these two stage together and develop an end-to-end method for this task. Another direction is to investigate the effectiveness of the proposed method on another few-shot tasks such as few-shot detection, *etc.*

6. REFERENCES

- [1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *CVPR*. IEEE, 2009, pp. 248–255.
- [2] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.
- [3] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al., “Matching networks for one shot learning,” in *NeurIPS*, 2016, pp. 3630–3638.
- [4] Sachin Ravi and Hugo Larochelle, “Optimization as a model for few-shot learning,” in *ICLR*, 2017.
- [5] Jake Snell, Kevin Swersky, and Richard Zemel, “Prototypical networks for few-shot learning,” in *NeurIPS*, 2017, pp. 4077–4087.
- [6] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales, “Learning to compare: Relation network for few-shot learning,” in *CVPR*, 2018, pp. 1199–1208.
- [7] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell, “Meta-learning with latent embedding optimization,” *arXiv preprint arXiv:1807.05960*, 2018.
- [8] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille, “Few-shot image recognition by predicting parameters from activations,” in *CVPR*, 2018, pp. 7229–7238.
- [9] Xiang Jiang, Mohammad Havaei, Farshid Varno, Gabriel Chartrand, Nicolas Chapados, and Stan Matwin, “Learning to learn with conditional class dependencies,” in *ICLR*, 2019.
- [10] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste, “Tadam: Task dependent adaptive metric for improved few-shot learning,” in *NeurIPS*, 2018, pp. 719–729.
- [11] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell, “Meta-learning with latent embedding optimization,” in *ICLR*, 2019.
- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” in *CVPR*. IEEE, 2016, pp. 770–778.
- [13] Sergey Zagoruyko and Nikos Komodakis, “Wide residual networks,” *arXiv preprint arXiv:1605.07146*, 2016.
- [14] Matthias Bauer, Mateo Rojas-Carulla, Jakub Bartłomiej Swiatkowski, Bernhard Scholkopf, and Richard E Turner, “Discriminative k-shot learning using probabilistic models,” *arXiv preprint arXiv:1706.00326*, 2017.
- [15] Zitian Chen, Yanwei Fu, Kaiyu Chen, and Yu-Gang Jiang, “Image block augmentation for one-shot learning,” in *AAAI*, 2019.
- [16] Victor Garcia and Joan Bruna, “Few-shot learning with graph neural networks,” *arXiv preprint arXiv:1711.04043*, 2017.
- [17] Bo Zhao, Xinwei Sun, Yanwei Fu, Yuan Yao, and Yizhou Wang, “Msplit lbi: Realizing feature selection and dense estimation simultaneously in few-shot and zero-shot learning,” *arXiv preprint arXiv:1806.04360*, 2018.
- [18] Chelsea Finn, Pieter Abbeel, and Sergey Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in *ICML*. JMLR, 2017, pp. 1126–1135.
- [19] Alex Nichol and John Schulman, “Reptile: a scalable meta-learning algorithm,” *arXiv preprint arXiv:1803.02999*, vol. 2, 2018.
- [20] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov, “Siamese neural networks for one-shot image recognition,” in *ICML Deep Learning Workshop*, 2015, vol. 2.
- [21] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer, “Revisiting self-supervised visual representation learning,” in *CVPR*, 2019, pp. 1920–1929.
- [22] Aaron van den Oord, Yazhe Li, and Oriol Vinyals, “Representation learning with contrastive predictive coding,” *arXiv preprint arXiv:1807.03748*, 2018.
- [23] Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord, “Data-efficient image recognition with contrastive predictive coding,” *arXiv preprint arXiv:1905.09272*, 2019.
- [24] Yonglong Tian, Dilip Krishnan, and Phillip Isola, “Contrastive multiview coding,” *arXiv preprint arXiv:1906.05849*, 2019.
- [25] Philip Bachman, R Devon Hjelm, and William Buchwalter, “Learning representations by maximizing mutual information across views,” *arXiv preprint arXiv:1906.00910*, 2019.
- [26] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, “The Caltech-UCSD Birds-200-2011 Dataset,” Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.
- [27] Nathan Hilliard, Lawrence Phillips, Scott Howland, Artëm Yankov, Courtney D Corley, and Nathan O Hodas, “Few-shot learning with metric-agnostic conditional embeddings,” *arXiv preprint arXiv:1802.04376*, 2018.
- [28] 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.
- [29] Wen-Hsuan Chu, Yu-Jhe Li, Jing-Cheng Chang, and Yu-Chiang Frank Wang, “Spot and learn: A maximum-entropy patch sampler for few-shot image classification,” in *CVPR*, 2019, pp. 6251–6260.
- [30] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo, “Revisiting local descriptor based image-to-class measure for few-shot learning,” in *CVPR*, 2019, pp. 7260–7268.
- [31] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel, “A simple neural attentive meta-learner,” *arXiv preprint arXiv:1707.03141*, 2017.
- [32] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sungju Hwang, and Yi Yang, “LEARNING TO PROPAGATE LABELS: TRANSDUCTIVE PROPAGATION NETWORK FOR FEW-SHOT LEARNING,” in *ICLR*, 2019.
- [33] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele, “Meta-transfer learning for few-shot learning,” in *CVPR*, 2019, pp. 403–412.