Supplementary Material for Learning Dynamic Alignment via Meta-filter for Few-shot Learning

Chengming Xu*[1] Yanwei Fu*[1] Chen Liu[1]
Chengjie Wang[2] Jilin Li[2] Feiyue Huang[2] Li Zhang[1] Xiangyang Xue[1]
[1]Fudan University [2]Youtu Lab, Tencent

1. Model Detail

Dynamic Meta-filter Here we give the coding-level technical explanations of Eq. 2 and Eq. 3 in the main paper. Particularly, both equations consider operation on each position. (1) The procedure of Eq.2 is similar to normal convolution. A conv layer (kernel size 3, input channel c, output channel c*k*k) is applied to support feature, leading to output size (c*k*k)*w*h. The feature vector of position (i, j) with size c*k*k can then be written as in Eq.2. (2) For Eq.3, we first reshape $f_d(i,j)$ into size c*1*k*k, which is a convolution kernel with size k, group c, output channel c. It is applied to the corresponding query region, i.e., $\mathcal{B}_k(X^q_{:,i,j})$. The same procedure is applied to all (i,j). (3) The realization of Eq.3 follows idea of sliding window, where all windows with size k*k are extracted and convolved with corresponding DMF in form of inner production.

Adaptive Alignment The Eq. 7 in main paper can be reformulated as $\frac{x^q_{t+1}-X^q_t}{a} = F(X^q_t, f_d)$, where $a=1$. By gradually taking smaller a, the LHS can be seen as derivative of X^q and the whole equation can be seen as a ODE (Eq.9) in the limit. If we take the procedure of alignment as a curve with X-axis as number of alignment, Y-axis as query feature, the solution can then be explained as the query feature after the alignment. Euler's method is the most direct ODE solver. Other adaptive solvers used by us can have adaptive step size and estimation method, thus being able to solve inflexible alignment.

* indicates equal contributions. Li Zhang is the corresponding author.
Chengming Xu, Yanwei Fu, Chen Liu, Li Zhang and Xiangyang Xue are with School of Data Science, MOE Frontiers Center for Brain Science, and Shanghai Key Lab of Intelligent Information Processing, Fudan University. Email: {cmxu18, yanweifu, chenliu18, lizhangfd, xyxue}@fudan.edu.cn
Chengjie Wang, Jilin Li and Feiyue Huang are with Youtu Lab, Tencent. Email: {jasoncjwang, jerolinli, garyhuang}@tencent.com

2. Additional Experiment Results

2.1. Comparison with state-of-the-arts using WRN

To further verify the effectiveness of our proposed method, we train our method with the WRN-28-10 backbone network which is widely used in few-shot learning literature. The results on *mini*ImageNet are listed in Tab. 1. It is noteworthy that using WRN on FEAT [6] will lead to significant performance drop compared to model using Res12 (1.68% on 1-shot and 1.30% on 5-shot), while such drop in our model is less. Specifically, our method beats the nearest rival FEAT by 2.42% on 1-shot and Robust dist++ by 1.14% on 5-shot tasks.

2.2. Model Efficiency

As we have discussed in our paper, the implementation of our dynamic meta-filter can be highly efficient. To show this, we compare the training and inference time for one 5-way 1-shot episode with the state-of-the-art method DeepEMD [7]. Both methods use Res12 as backbone network. The results in Tab. 2 reflects that while both methods have similar inference speed, DeepEMD is far slower than our model when training, with a gap of 22.608 seconds between two models. Such results validates our model is efficient in practical usage while also achieving remarkable performance.

Model	*mini*ImageNet. 1-shot	5-shot
LEO [5]	61.76±0.08	77.59±0.12
PPA [4]	59.60±0.41	73.74±0.19
Robust dist++ [1]	63.28±0.62	81.17±0.43
wDAE [3]	61.07±0.15	76.75±0.11
CC+rot [2]	62.93±0.45	79.87±0.33
FEAT [6]	65.10±0.20	81.11±0.14
Ours	**67.52±0.47**	**82.25±0.32**

Table 1. 5-way few-shot accuracy with 95% confidence interval on *mini*ImageNet. All methods use WRN-28-10 as backbone network.

Model	train(s)	inference(s)	Acc.
DeepEMD [7]	23.776	**0.362**	65.91
Ours	**1.168**	0.416	**67.76**

Table 2. Comparison of training and inference speed between our method and DeepEMD on 1-shot *mini*ImageNet tasks.

References

[1] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *Int. Conf. Comput. Vis.*, 2019. 1

[2] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *Int. Conf. Comput. Vis.*, 2019. 1

[3] Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1

[4] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille. Few-shot image recognition by predicting parameters from activations. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 1

[5] 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. 1

[6] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8808–8817, 2020. 1

[7] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12203–12213, 2020. 1, 2