Dual Prototypical Network for Robust Few-shot Image Classification

Qi Song* Zebin Peng*, Luchen Ji*, Xiaochen Yang†, Xiaoxu Li*
* Lanzhou University of Technology, Lanzhou, China
E-mail: lixiaoxu@lut.edu.cn
† School of Mathematics and Statistics, University of Glasgow, UK.
E-mail: xiaochen.yang@glasgow.ac.uk

***Abstract*—Deep neural networks have outperformed humans on some image recognition and classification tasks. However, with the emergence of various novel classes, it remains a challenge to continuously expand the learning capability of such networks from a limited number of labeled samples. Metric-based approaches have been playing a key role in few-shot image classification, but most of them measure the distance between samples in the metric space using only a single metric function. In this paper, we propose a Dual Prototypical Network (DPN) to improve the test-time robustness of the classical prototypical network. The proposed method not only focuses on the distance of the original features, but also adds perturbation noise to the image and calculates the distance of noisy features. By enforcing the model to predict well under both metrics, more representative and robust class prototypes are learned and thus lead to better generalization performance. We validate our method on three fine-grained datasets in both clean and noisy settings.**

I. Introduction

With the development of deep learning, the recognition performance of machines has surpassed that of humans in many large-scale image classification tasks. However, when the amount of data that can be learned from is small, the machine's recognition ability is not satisfactory [1, 2]. Therefore, image classification based on a very small number of labeled samples, often referred to as few-shot classification, has attracted considerable research attention in recent years. Few-shot classification usually includes two types of data with disjoint label spaces, namely, base class data and novel class data. It aims to use the knowledge learned from base class data and a small number of labeled samples from novel class data to learn classification rules and accurately predict the categories of unlabeled samples from the same novel classes.

Few-shot classification is a challenging machine learning task, and common approach to this problem include data augmentation, feature alignment and metric learning. Data augmentation algorithms aim to generate image features to compensate for the problem of insufficient labeled data. For example, Wang et al. [3] proposed to meta-learn a parametric hallucinator network on the base data to generate more training samples. Zhang et al. [4] proposed a saliency network to separate foregrounds and backgrounds of support images (*i.e.*, training samples in the classical machine learning settings) and then mix foregrounds and backgrounds to generate more support-query pairs (*i.e.*, training-test pairs). Feature alignment methods are often used for fine-grained image classification. Huang et al. [5] proposed a novel low-rank pairwise bilinear pooling operation to reduce dimension and a feature alignment layer to capture and match subtle differences between support images and query images. Wertheimer et al. [6] proposed to reconstruct local query features from local support features, thus modelling the spatial details and discarding the location constraints.

Metric-based approaches aim to learn an embedding function that map images to a metric space such that the relevance between a pair of images is determined based on their distance; smaller distances indicate higher probability for the two images belonging to the same class. Vinyals et al. [7] proposed the matching network, which embeds support and query images via two separate networks and then calculates the cosine similarity between the query and each support image. Snell et al. [8] proposed the classical prototypical network, which calculates the squared Euclidean distance between the query image with the prototype of each class. The prototype is calculated as the mean of support images and serve as the class representation, which has been demonstrated to be particularly effective in the case of scarce labeled images. Simon et al. [9] proposed to represent the class by a subspace and use the projection distance as the classification criterion. Wang et al. [10] proposed an ensemble metric learning method, which calculates the similarity by the fusion of multiple metrics. Asagawa et al. [11] used the Euclidean distance and cosine similarity as a measure of high-precision deep network models. Instead of using a fixed distance metric, Sung et al. [12] proposed the relation network which parameterizes the distance function as a neural network and uses it to compute the relation score for the support-query pair.

Prior metric-based methods learn the metric space on the base class data and do not require fine-tune the metric, thus avoiding overfitting to the few labeled novel samples. However, these methods often only seek for a single metric to measure the similarity between features, which may lack generalization ability and robustness to test-time noises. Therefore, it is particularly important to find a metric-based method permitting multiple metrics for different features. In this paper, we propose a Dual Prototypical Network in which we input two sets of images to an embedding module at the same time, one for the original clean images and the other for the noisy images perturbed by Gaussian noise, and calculate the Euclidean

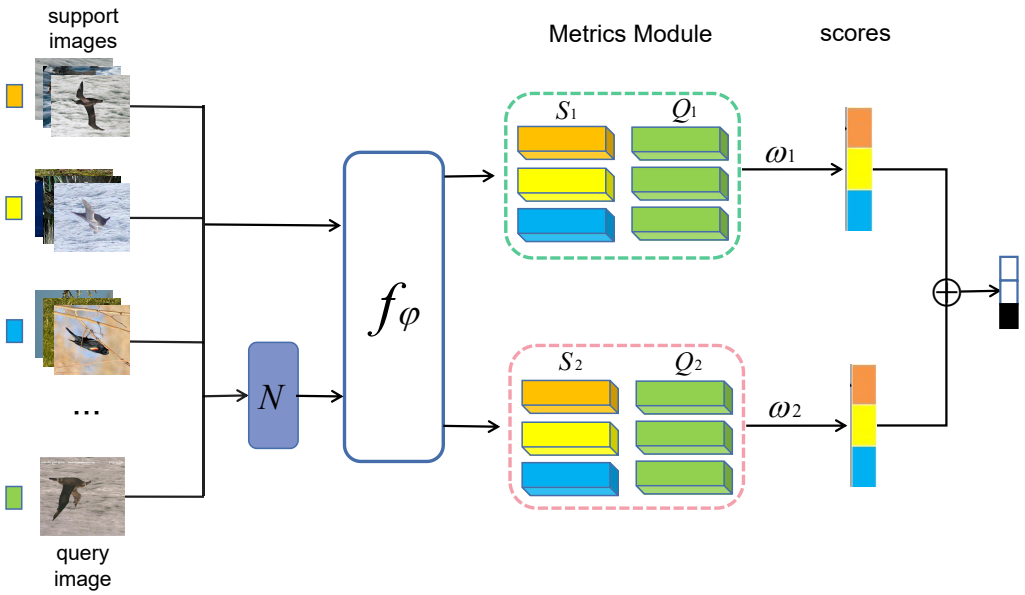

Fig. 1. The proposed Dual Prototypical Network (DPN) framework. Clean images and noisy images, perturbed by Gaussian noise N, are sent to the feature embedding module f_φ. Distances are computed on both clean and noise prototype-query pairs, and the classification decision is made based on the combined distances.

distance between query and prototypes separately for each set of features. Consequently, the final classification is made based on two distance functions, with one of them aiming to improve the network's robustness to noises which may exist in unseen query images. The key benefit of our approach is that the method introduces only one additional parameter compared with the prototypical network [8]. In other words, improvement in robustness is achieved through the designed of the network, rather than training a more complicated model. We evaluate the proposed method on three benchmark fine-grained few-shot classification datasets and results suggest that our method consistently outperforms the prototypical network in both clean and noisy settings, demonstrating its good generalizability and robustness.

II. Methodology

Dual Prototypical Network (DPN) adopts a dual network as the main structure of the network, as shown in Figure 1. The robustness of the model is improved by using noisy feature perturbed by Gaussian noise, and the structure of parallel integration with the prototype network ensures that the accuracy is not compromised. In this section, we will describe the network structure of DPN in detail.

A. Problem Definition

Given a dataset with data-label pairs $D = \{(x_i, y_i), y_i \in C\}$, where C denotes the set of classes, we partition C into base classes C_b, validation classes C_v, and novel classes C_n. Note that in few-shot classification, $C_b \cap C_\nu \cap C_n = \varnothing$. The goal is to train a model on the data from the base classes so that the model can generalize well on tasks sampled from the novel classes. In order to evaluate the fast adaptation ability or the generalization ability of the model, there are only a few labeled samples available for each task τ. We followed the classic N-way K-shot classification setting in few-shot classification. In each task τ, the few available labeled data forms the *support set*, $S = \{(x_i, y_i)\}_{i=1}^{N\times K}$, and the model is evaluated on another *query set*, $Q = \{(x_i, y_i)\}_{i=N\times K+1}^{N\times(K+q)}$, where every class in the task has q test images. The performance of a model is evaluated as the averaged accuracy on (the query set of) multiple tasks sampled from the novel classes.

B. Overview

Prototypical network [8] and many other metric-based approaches classify samples based on a single metric function, which may be incapable of resisting potential noises in query images and enforcing sufficient regularization to learn highly discriminative features. Therefore, we propose a Dual Prototypical Network (DPN) with controllable noise and a parallel structure for bi-similarity calculation.

As shown in Figure 1, our framework first inputs two sets of images into the embedding module f_φ – one clean set and one noisy set which adds Gaussian noise with controllable magnitude to original images. The embedding network, sharing the same set of parameters, then produces two different sets of support features and query features. The two sets of features are used to calculate their respective similarity scores via the Euclidean distance, and finally the scores are summed to get the final score.

C. Disturbance Noise

To reduce the sensitivity of the model to noisy data and improve its generalization from base to novel data, we add Gaussian noise to the original images. Meanwhile, in order

to prevent the distortion of the image due to the inclusion of too large noise, we add a parameter that can control the noise magnitude; the setting of this parameter size will be evaluated in the ablation experiment.

Given a support image x_i and a query image x_j, we can obtain their embedded features as follows:

$$\hat{\mathbf{x}}_i^{\text{clean}} = f_\varphi(x_i) \ , \tag{1}$$

$$\hat{\mathbf{x}}_j^{\text{clean}} = f_\varphi(x_j) \ , \tag{2}$$

$$\hat{\mathbf{x}}_i^{\text{noisy}} = f_\varphi(x_i + \mu N(0,1)) \ , \tag{3}$$

$$\hat{\mathbf{x}}_j^{\text{noisy}} = f_\varphi(x_j + \mu N(0,1)) \ , \tag{4}$$

where $f_\varphi(\cdot)$ denotes the feature embedding function with learnable parameters φ, $\hat{\mathbf{x}}_i^{\text{clean}}$ and $\hat{\mathbf{x}}_j^{\text{clean}}$ are the support and query feature vectors obtained from the original image respectively, and $\hat{\mathbf{x}}_i^{\text{noisy}}$ and $\hat{\mathbf{x}}_j^{\text{noisy}}$ are the support and query feature vectors generated from the image with Gaussian noise respectively. $N(0,1)$ denotes the standard Gaussian noise, and μ is the parameter that controls the magnitude of Gaussian noise. After several comparison studies, we finally set its value to 0.01 in all our experiments.

D. Metrics Module

After feature extraction, we adopt a parallel structure for computing the prototypes and calculating the similarity. In other words, the network consists of two branches, and each branch has its corresponding prototype; clean queries are compared with prototypes constructed from clean support features and noisy queries are compared with prototypes from noisy support features. Following prototypical network [8], the prototype is set as the mean vector of the embedded features of the support set of the class:

$$\begin{aligned} \mathbf{c}_k^{\text{clean}} &= \frac{1}{|K|} \sum_{\{i:y_i=k\}} \hat{\mathbf{x}}_i^{\text{clean}} \\ \mathbf{c}_k^{\text{noisy}} &= \frac{1}{|K|} \sum_{\{i:y_i=k\}} \hat{\mathbf{x}}_i^{\text{noisy}}, \end{aligned} \tag{5}$$

where $\mathbf{c}_k^{\text{clean}}$ and $\mathbf{c}_k^{\text{noisy}}$ denotes the prototype of class k constructed from clean support features and noisy support features, respectively.

Next, we compute the Euclidean distance between the query and class prototypes separately for the clean and noisy cases as follows:

$$\begin{aligned} d_{j,k}^1 &= \|\mathbf{x}_j^{\text{clean}} - \mathbf{c}_k^{\text{clean}}\|_2 \\ d_{j,k}^2 &= \|\mathbf{x}_j^{\text{noisy}} - \mathbf{c}_k^{\text{noisy}}\|_2 \end{aligned} \tag{6}$$

For brevity, we will omit the index j hereafter.

Finally, the two distance functions are combined, from which we can obtain the probability of assigning the query image into class k:

$$d_k = \omega_1 d_k^1 + \omega_2 d_k^2 \tag{7}$$

$$P(y=k \mid \mathbf{x}) = \frac{\exp(-d_k)}{\sum_{k'} \exp(-d_{k'})} \ . \tag{8}$$

ω_1 and ω_2 are trainable parameters, which denote the weights of the two distances respectively.

The network is optimized by using the SGD optimizer with the objective of minimizing the negative log probability L of the true class k:

$$L = -\log(P(y=k \mid \mathbf{x})) \tag{9}$$

The training process uses the episodic training approach [7], which selects N classes randomly from the base class data and K samples for each class to mimic the support set, and selects another q samples from each class as the query set.

III. Experiments

A. Datasets

In this paper, we use three fine-grained datasets, CUB-200-2011 [13], Stanford-Dogs [14]. For each dataset we proportionally split into a training set, a validation set, and a test set. CUB-200-2011 is a classical fine-grained image classification dataset containing 11,788 images from 200 bird species. Stanford-Dogs contains 20,580 annotated images from 120 dog species around the world. Stanford-Cars contains a total of 16,185 images of different car models.

B. Implementation Details

We conduct experiments under shallow network architectures, Conv-4 [19, 20]. we preprocess the images with standard data enhancement, including random cropping, random flipping and color dithering. The normalized images allow us to obtain better training stability. We trained all Conv-4 models for 1,200 epochs. The initial learning rate is set to 0.1 and the weight decay is set to 5e-4. After every 400 epochs, the learning rate decreases by a factor of 10. We train the model using the 30-way 5-shot setting in the three fine-grained datasets mentioned above. In addition, we selected the best-performing model based on the validation set and validated it every 20 epochs. For all experiments, we report the average accuracy of 10,000 randomly generated tasks on datasets with 95% confidence intervals on the standard 5-way 1-shot and 5-way 5-shot settings.

C. Comparison with State-of-the-arts

The proposed method is compared with 10 few-shot image classification methods and results are reported in Table I. We see that our method is consistently better than the baseline prototypical network on all datasets and in all settings. Moreover, our model achieves superior performance on 5-way 5-shot tasks, outperforming the state-of-the-art methods.

D. Evaluation of Robustness

In order to demonstrate the robustness of our experiments, we use the trained model f_φ as the embedding module and add Gaussian noise to query images of the novel dataset. The noise level is set as $\mu = 0.1, 0.5$. For comparison, we add the same noise to prototypical network.

Before presenting the results, we visualize the effect of Gaussian noise. Figure 2 show some randomly selected images

TABLE I

TABLE 1: 5-WAY FEW-SHOT CLASSIFICATION PERFORMANCE ON THE CUB, DOGS AND CARS DATASETS.

Model	CUB		Dogs		Cars	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
MAML[15]	55.92 ± 0.95	72.09 ± 0.76	46.67 ± 0.87	62.56 ± 0.80	48.37 ± 0.81	65.41 ± 0.77
MatchingNet[7]	60.06 ± 0.88	74.57 ± 0.73	46.10 ± 0.86	59.79 ± 0.72	44.73 ± 0.77	64.74 ± 0.72
ProtoNet[8]	63.64 ± 0.23	84.23 ± 0.15	45.12 ± 0.21	69.16 ± 0.16	48.42 ± 0.22	71.38 ± 0.18
RelationNet[12]	63.94 ± 0.92	77.87 ± 0.64	**47.35 ± 0.88**	66.20 ± 0.74	46.04 ± 0.91	68.52 ± 0.78
Baseline++[16]	62.36 ± 0.84	79.08 ± 0.61	44.49 ± 0.70	64.48 ± 0.66	46.82 ± 0.76	68.20 ± 0.72
DeepEMD[17]	64.08 ± 0.50	80.55 ± 0.71	46.73 ± 0.49	65.74 ± 0.63	**61.63 ± 0.27**	72.95 ± 0.38
DSN[9]	**71.57 ± 0.92**	83.51 ± 0.60	44.33 ± 0.81	60.04 ± 0.68	48.16 ± 0.86	60.77 ± 0.75
LRPABN[5]	63.63 ± 0.77	76.06 ± 0.58	45.72 ± 0.75	60.94 ± 0.66	60.28 ± 0.76	73.29 ± 0.58
MixFSL[18]	53.61 ± 0.88	73.24 ± 0.75	43.96 ± 0.77	64.43 ± 0.68	44.56 ± 0.80	59.63 ± 0.79
Ours	63.68 ± 0.23	**85.18 ± 0.15**	45.95 ± 0.21	**70.14 ± 0.16**	49.02 ± 0.22	**73.31 ± 0.18**

TABLE II

COMPARISON OF PROTOTYPICAL NETWORK AND THE PROPOSED METHOD ON 5-WAY FEW-SHOT CLASSIFICATION WITH GAUSSIAN NOISE ADDED TO QUERY IMAGES; NOISE MAGNITUDE IS CONTROLLED BY μ ($\mu = 0.1$ – LOW NOISE, $\mu = 0.5$ – HIGH NOISE).

Model	CUB		Dogs		Cars	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ProtoNet[8] (μ=0.1)	60.72 ± 0.23	81.58 ± 0.16	40.82 ± 0.20	62.49 ± 0.17	44.41 ± 0.22	67.57 ± 0.18
Ours (μ=0.1)	**61.39 ± 0.23**	**82.19 ± 0.16**	**41.46 ± 0.20**	**63.40 ± 0.17**	**44.84 ± 0.21**	**67.74 ± 0.18**
ProtoNet[8] (μ=0.5)	28.48 ± 0.15	34.93 ± 0.19	21.19 ± 0.06	23.26 ± 0.08	22.82 ± 0.80	25.93 ± 0.10
Ours (μ=0.5)	**30.78 ± 0.16**	**38.02 ± 0.19**	**21.76 ± 0.06**	**23.70 ± 0.08**	**24.00 ± 0.10**	**28.22 ± 0.12**

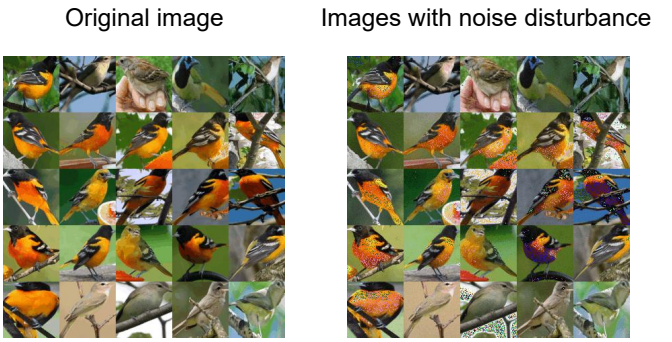

Fig. 2. The left image is the original unprocessed CUB image, and the right image is the image with Gaussian noise added.

TABLE III

CLASSIFICATION ACCURACY UNDER DIFFERENT TYPES OF NOISES ON THE CUB DATASET.

Model	Backbone	1-shot	5-shot
ProtoNet[8] (Gaussian)	**Conv-4**	60.72 ± 0.23	81.58 ± 0.18
Ours (Gaussian)	**Conv-4**	**61.39 ± 0.23**	**82.19 ± 0.15**
ProtoNet[8] (Possion)	**Conv-4**	62.20 ± 0.23	83.86 ± 0.16
Ours (Possion)	**Conv-4**	**63.89 ± 0.23**	**84.24 ± 0.16**

from the CUB dataset, with the left panel displaying the original images and the right panel displaying noisy images. As we can see, images with Gaussian noise will show some more obvious pixel dots, which appear randomly at various locations of the image.

Table II lists the classification accuracy of the baseline prototypical network and our method in the presence of Gaussian noise. Both methods deteriorate when facing noisy queries, particularly in the case of a high level of noise (i.e., $\mu = 0.5$). The proposed method achieves higher accuracy than the baseline in all scenarios, indicating its effectiveness to withstand test-time noises.

In addition to Gaussian noise, we also explore the robustness of our method to other types of noises, aiming to understand if it has the capacity to defend noises which are unseen before. Table III lists the performance of ProtoNet and our method under Gaussian and Poisson noise on CUB; $\mu = 0.1$ is used for both types of noises. In turns out the influence of Poisson noise is weaker than the Gaussian noise at the same noise magnitude. Still, our method is more robust than ProtoNet.

IV. CONCLUSION

In this paper, we propose a Dual Prototypical Network (DPN) to enhance the robustness of the prototypical network when facing different levels of noise perturbation. By adding noise perturbation to images and adopting a dual network to generate two distance functions, the class prototypes in the metric space become more representative and thus improving the overall performance of the model. Experiments show that on CUB, Dogs and Cars datasets, the proposed method achieves better classification accuracy and robustness than prototypical network in all settings, and it is superior to the state-of-the-art methods in the 5-way 5-shot setting, demonstrating its effectiveness. This work is a metric-based

approach for few-shot classification, and in the future, we will incorporate different metric methods or feature alignment methods to further improve the generalization ability and robustness of few-shot classification methods.

ACKNOWLEDGEMENT

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62176110, 62111530146, 61906080, Young Doctoral Fund of Education Department of Gansu Province under Grant 2021QB-038, Hong-Liu Distinguished Youth Talents Foundation of Lanzhou University of Technology.

REFERENCES

[1] X. Li, Z. Sun, J.-H. Xue, and Z. Ma, "A concise review of recent few-shot meta-learning methods," *Neurocomputing*, vol. 456, pp. 463–468, 2021.

[2] J. Shu, Z. Xu, and D. Meng, "Small sample learning in big data era," *ArXiv*, vol. abs/1808.04572, 2018.

[3] Y.-X. Wang, R. Girshick, M. Hebert, and B. Hariharan, "Low-shot learning from imaginary data," in *IEEE conference on computer vision and pattern recognition*, 2018, pp. 7278–7286.

[4] H. Zhang, J. Zhang, and P. Koniusz, "Few-shot learning via saliency-guided hallucination of samples," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2770–2779.

[5] H. Huang, J. Zhang, J. Zhang, J. Xu, and Q. Wu, "Low-rank pairwise alignment bilinear network for few-shot fine-grained image classification," *IEEE Transactions on Multimedia*, vol. 23, pp. 1666–1680, 2021.

[6] D. Wertheimer, L. Tang, and B. Hariharan, "Few-shot classification with feature map reconstruction networks," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8008–8017, 2021.

[7] O. Vinyals, C. Blundell, T. P. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *NIPS*, 2016.

[8] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," *ArXiv*, vol. abs/1703.05175, 2017.

[9] C. Simon, P. Koniusz, R. Nock, and M. T. Harandi, "Adaptive subspaces for few-shot learning," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4135–4144, 2020.

[10] H. Wang and D. Chen, "Few-shot image classification based on ensemble metric learning," *Journal of Physics: Conference Series*, vol. 2171, 2022.

[11] T. Asakawa and M. Aono, "Visual sentiment analysis for few-shot image classification based on metric learning," *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 1081–1086, 2020.

[12] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2018.

[13] C. Wah, S. Branson, P. Welinder, P. Perona, and S. J. Belongie, "The caltech-ucsd birds-200-2011 dataset," 2011.

[14] A. Khosla, N. Jayadevaprakash, B. Yao, and L. Fei-Fei, "Novel dataset for fine-grained image categorization : Stanford dogs," 2012.

[15] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *ICML*, 2017.

[16] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y. Wang, and J.-B. Huang, "A closer look at few-shot classification," *ArXiv*, vol. abs/1904.04232, 2019.

[17] C. Zhang, Y. Cai, G. Lin, and C. Shen, "Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12 200–12 210, 2020.

[18] A. Afrasiyabi, J.-F. Lalonde, and C. Gagn'e, "Mixture-based feature space learning for few-shot image classification," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9021–9031, 2021.

[19] M. Tong, S. Wang, B. Xu, Y. Cao, M. Liu, L. Hou, and J.-Z. Li, "Learning from miscellaneous other-class words for few-shot named entity recognition," in *ACL*, 2021.

[20] K. Lee, S. Maji, A. Ravichandran, and S. Soatto, "Meta-learning with differentiable convex optimization," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10 649–10 657, 2019.