Anti-Noise Relation Network for Few-shot Learning

Xiaoxu Li*, Jintao Yan*, Jijie Wu*, Yuxin Liu†, Xiaochen Yang‡, Zhanyu Ma§
* Lanzhou University of Technology, Lanzhou, China
† University of Melbourne, Melbourne, Australia
‡ University College London, London, U.K.
E-mail: xiaochen.yang.16@ucl.ac.uk
§ Beijing University of Posts and Telecommunications, Beijing, China
E-mail: mazhanyu@bupt.edu.cn

***Abstract*—Few-shot classification has received great attention in the field of machine learning and computer vision. It aims is to achieve the learning ability close to human recognition by training from a few labelled samples. The existing few-shot classification methods have attempted to alleviate the impact of insufficient samples in a variety of ways, such as meta-learning and metric learning, but they ignore the noise robustness. This work proposes a new *Anti-Noise Relation Network* by embedding an autoencoder network into a classical neural network of few-shot classification, *Relation Network*. Experimental results on the Stanford Car and CUB-200-2011 datasets demonstrate the superiority of the proposed method in both classification accuracy and robustness against different noises.**

I. Introduction

Image classification [1] has been a fundamental and essential task in computer vision [2], [3]. Recently, image classification algorithms trained on large-scale datasets has outperformed human beings [4]. However, challenges still exist for small-sample image classification [5]. Research on few-shot image classification [4], [6], [8] is valuable both theoretically and empirically [9].

The main challenge of few-shot image classification is the deficiency of samples [7], [10]. So far, many methods have been proposed to alleviate this issue founded on different approaches [4], [11], such as data augmentation [12], regularization [11], [13], transfer learning [14], [15], meta-learning [16], and metric learning [17]. In terms of simplicity and effectiveness, metric-based few-shot learning methods obtain state-of-the-art performance on many few-shot classification datasets. Metric-based methods assume that, if it is applicable to learn the metric or similarity measure between images, few-shot classification can be performed by comparing the distance or similarity between a new sample and few samples with known labels.

Depending on whether the distance measure is fixed or learned, the metric-based method can be categorized into two groups. One group aims to learn a feature embedding to adapt to a fixed metric, such as the cosine similarity and the Euclidean distance. Siamese Convolutional Network [18] adopted two networks with the same network parameters to extract features of images and used the L_1 norm to measure the distance of images. Matching Network [8] introduced the attention mechanism and used the cosine similarity for measuring the similarity of images. Prototypical Network [19] introduced the concept of prototype and used the Euclidean distance to measure the distance of an image and prototype of each class. Infinite Mixture Prototypical Network built on Prototypical Network and assumed there are multiple prototypes in each class. The other group of methods focuses on learning both the metric and the feature representation. Relation Network [20] is a representative of this group which employs a relation module to model the distance between images. Built on Relation Network, Nearest Neighbor Neural Network, short for DN4 [21], constructed a module which measures the distance between local features of a query image and those of its nearest neighbor. Compared with other few-shot metric learning methods, the main difference is that DN4 [21] considers local features of images when measuring the distance.

However, these existing few-shot classification methods did not consider the robustness of models to noisy test data. On the other hand, real images usually contain noise because of various reasons, such as inappropriate operation in image storage and image transmission. Therefore, this paper aims to mitigate the sensitivity of few-shot classification methods to noisy data. Specifically, we introduce an anti-noise module into Relation Network and propose a new *Anti-Noise Relation Network* for few-shot classification. The anti-noise module, founded an autoencoder network (AE) [22], [23], synthesizes noise data. By forcing the network to perform well on the synthetic data, robustness to test-time noise is enhanced. Experiments on the Stanford-Cars [24] and the CUB-200-2011 [25] datasets show that the proposed method achieves better classification performance than existing methods on the test data with or without different noise signals.

II. The proposed Anti-Noise Relation Work

This section will first briefly review the problem formulation of few-shot learning and Relation Network, and then introduce the network structure and loss functions of the proposed Anti-Noise Relation Network.

A. Problem Formulation

In a C-way K-shot classification setting, a dataset is divided into three sets, namely a training set, a validation set and a test

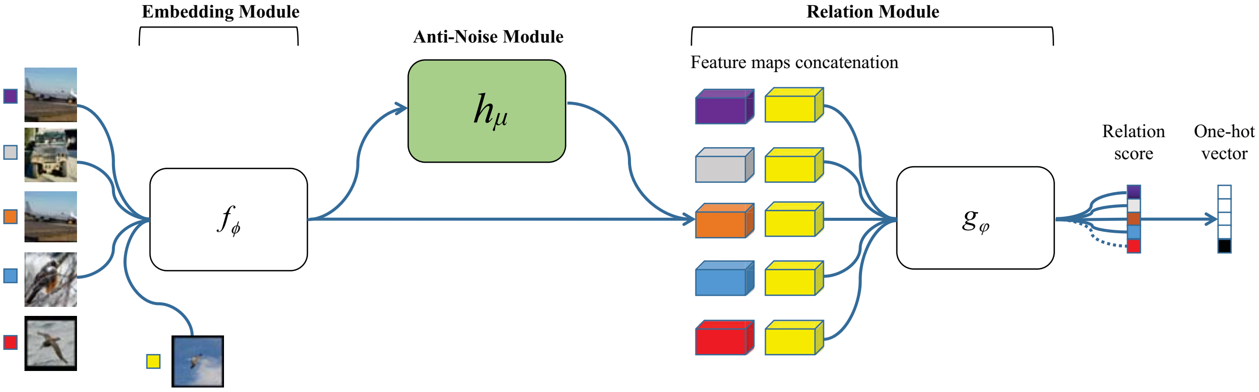

Fig. 1. The Anti-Noise Relation Network. We embed the autoencoder network into Relation Network. The proposed network consists of an embedding module, an anti-noise module and a relation module. The anti-noise module aems to synthesize noise data.

set. The label space of any two sets is disjoint. The training process contains many tasks, and each task consists of a query set Q and a support set S. To form a support set, we randomly select C classes from the training set and K images from each class, i.e. $S = \{(x_i, y_i)\}_{i=1}^{m}$ $(m = C \times K)$. In addition, from the remaining images in each class, we randomly select some samples to form a query set $Q = \{(x_j, y_j)\}_{j=1}^{n}$. We train a model on these tasks continually and select the optimal model based on its performance on the validation set. Finally, the evaluation of C-way K-shot classification is conducted on the test set.

B. Relation Network

Relation Network [20] is a classical method for few-shot classification. This method is elegant and has achieved outstanding empirical performance. It consists of two modules – one is the embedding module, and the other one is the relation module.

The embedding module consists of four convolutional blocks. Each block consists of 64 filters, each of which is 3×3 convolution in size. There is also a batch normalization and a ReLU nonlinearity layer. For the first two convolutions, each convolution is followed by a 2×2 max-pooling layer. The embedding module outputs feature maps, which serve as the inputs to the relation module. After passing through the relation module, the concatenated feature maps are transformed as a relation score.

C. Anti-Noise Relation Network

To alleviate the sensitivity of the model to noise data, we embed the autoencoder network (AE) into Relation Network and propose a new few-shot learning method termed *Anti-Noise Relation Network*. In this section, we will introduce the network structure, followed by the loss function of the proposed network.

As shown in Figure 1, the proposed network consists of an embedding module, an anti-noise module and a relation module. The embedding module and the relation module are the same as the ones in Relation Network. The feature maps of the embedding module are fed into both the anti-noise module and the relation module. The feature maps obtained from the anti-noise module are also fed into the relation module.

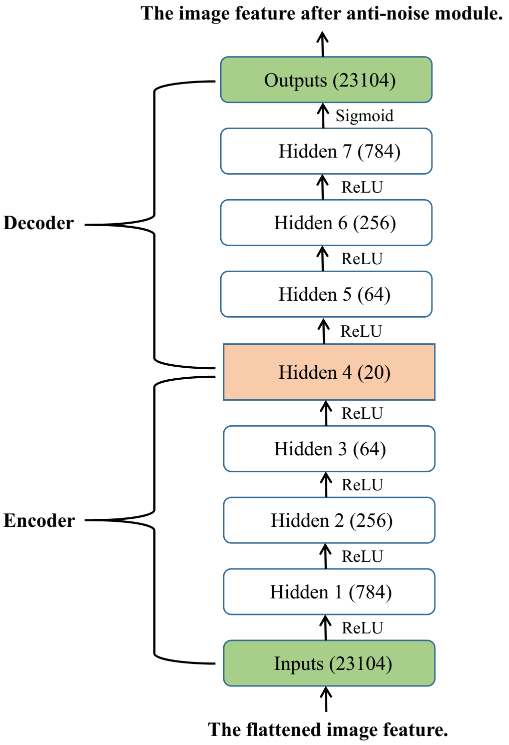

Fig. 2. The anti-noise module is implemented by the autoencoder network.

1) The anti-noise module: The aim of the anti-noise module is to synthesize noise data. The anti-noise module, h_μ, is implemented by the autoencoder network (AE). The network architecture is given in Figure 2.

There are two parts in AE: Encoder and Decoder. The flattened feature maps (23104 dimensions) from the embedding module are fed into encoder first and then fed into decoder.

The output feature of AE has the same size as the input feature. The Encoder contains four layers, and the output dimensions of the four layers are 784, 256, 64 and 20, respectively. The decoder also contains four layers, in which output dimensions are 64, 256, 784 and 23104.

2) Loss function: The loss function of the proposed network consists of three parts, namely a loss function on the original data (Γ), a loss function on the noisy data synthesized via AE ($\hat{\Gamma}$), and a loss function to constrain the distance between the original data and the synthetic noise data (D).

Samples x_i in the support set S and samples x_j in the query set Q are fed into the embedding module, and the output from the embedding module is feature maps $f_\varphi\left(x_i\right)$ and $f_\varphi(x_j)$. We use $F\left(\cdot,\cdot\right)$ to represent a function of concatenating the feature maps. The feature maps $f_\varphi(x_i)$ and $f_\varphi(x_j)$ are concatenated with operator $F(f_\varphi(x_j), f_\varphi(x_j))$. The similarity between the concatenated feature maps is learned through the relation module and the relation score between x_i and x_j, denoted as $r_{i,j}$, is produced:

$$r_{i,j} = g_\phi(F(f_\varphi(x_i), f_\varphi(x_j))),\ i = 1, 2, \cdots, m, \quad j = 1, 2, \cdots, n. \tag{1}$$

Based on the relation score between x_i and x_j, i.e. $r_{i,j}$, and their label y_i and y_j, we obtain the loss function on original data, denoted as Γ:

$$\Gamma = \sum_{i=1}^{m}\sum_{j=1}^{n}(r_{i,j} - \mathbb{1}(y_i == y_j))^2, \tag{2}$$

where $\mathbb{1}(y_i == y_j))$ equals one if y_i is same as y_j and zero otherwise.

Similarly, we can obtain the relation score based on the synthetic noise data $\hat{r}_{i,j}$ and the loss function $\hat{\Gamma}$: let h_μ represents the anti-noise module, then

$$\hat{r}_{i,j} = g_\phi(F(h_\mu(f_\varphi(x_i)), f_\varphi(x_j))) \tag{3}$$

$$\hat{\Gamma} = \sum_{i=1}^{m}\sum_{j=1}^{n}(\hat{r}_{i,j} - \mathbb{1}(y_i == y_j))^2 \tag{4}$$

We introduce the third loss function, denoted as D, to constrain the distance between the original data and the synthetic noise data. $f_\varphi(x)$ and $h_\mu(f_\varphi(x))$ are the input feature maps and output feature maps of the anti-noise module, respectively. The purpose of D is to ensure that the synthetic feature maps and the original feature maps are neither identical nor very distinct from each other. To achieve this goal, we introduce a hyperparameter β into D, which controls the Euclidean distance between feature maps $f_\varphi(x)$ and $h_\mu(f_\varphi(x))$:

$$D = \frac{\left(\beta - \sqrt{\Sigma(h_\mu(f_\varphi(x)) - f_\varphi(x))^2}\right)^2}{n} \tag{5}$$

Finally, integrating Γ, $\hat{\Gamma}$ and D, we obtain the loss function of the proposed method:

$$Loss = \Gamma + \gamma\hat{\Gamma} + \alpha D; \tag{6}$$

α and γ are hyperparameters which adjust the influence of $\hat{\Gamma}$ and D, respectively.

III. Experimental Results

In this section, we evaluate the robustness of the proposed method to noises with increasing intensity levels. After describing datasets and implementation details, we present the experimental results.

A. Datasets and preprocessing

All methods are evaluated on the Stanford Cars [24] and the CUB-200-2011 [25] datasets. The datasets differ in content, the number of classes, and sample size.

CUB-200-2011 (CUB): The CUB dataset contains 11,788 images from 200 bird species, proposed by the California Institute of Technology in 2010. It is also the benchmark dataset for fine-grained image classification and recognition.

Stanford Cars (Cars): The Cars dataset contains 16,185 images of 196 classes of cars. Classes are typically at the level of Year, Make, Model, e.g. 2012 Tesla Model S or 2012 BMW M3 coupe.

For both datasets, classes are divided into the training, validation and test sets as the proportion of 2:1:1. Input images are resized to 84×84 and augmented using standard techniques including random cropping, left and right flipping, and color dithering.

B. Implementation details

Our experiment focuses on the 5-way 5-shot task. For fairness of the experiment, our implementation setting follows the procedure specified in [15]. In the training stage, we train our network for 400 epochs and each epoch contains 100 episodes; that is, 40,000 episodes are trained. The validation set is used to select the training episodes with the highest accuracy. In each episode, we randomly sample 5 classes from the training set; for each class, we sample 5 labeled images to form the support set and 16 images for the query set. The method is trained from scratch by using Adam optimizer. The embedding module of our network uses a 4-layer convolution with max-pooling in only the first two layers. Relation scores are normalized through a softmax layer instead of the L_2 standard to speed up training. The initial learning rate is 10^{-3}.

For comparison, we consider Matching Network (MatchingNet) [8], Relation Network (RelationNet) [20], and Prototypical Network (PrototypeNet) [19], implemented following [15]. All methods are implemented based on PyTorch.

C. Few-shot classification accuracy under different noise conditions

We compare the robustness of our method with the other three methods when test images contain Gaussian noise and Poisson noise. The accuracy of clean images is also reported. We remark that all methods are trained on clean images, and noisy images are used only for testing.

In this experiment, we test Gaussian noise and Poisson noise with different levels of noise intensity. Figure 3 shows an

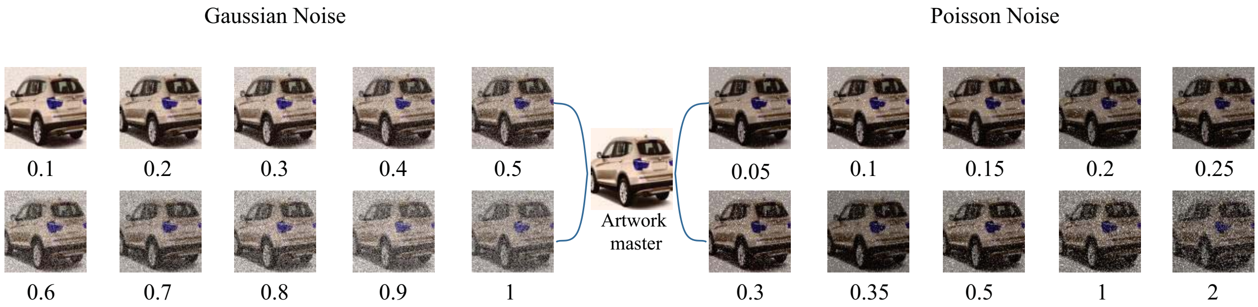

Fig. 3. The change of the image with the gradual increase of Gaussian noise and Poisson noise. The value below the example images represents the noise intensity.

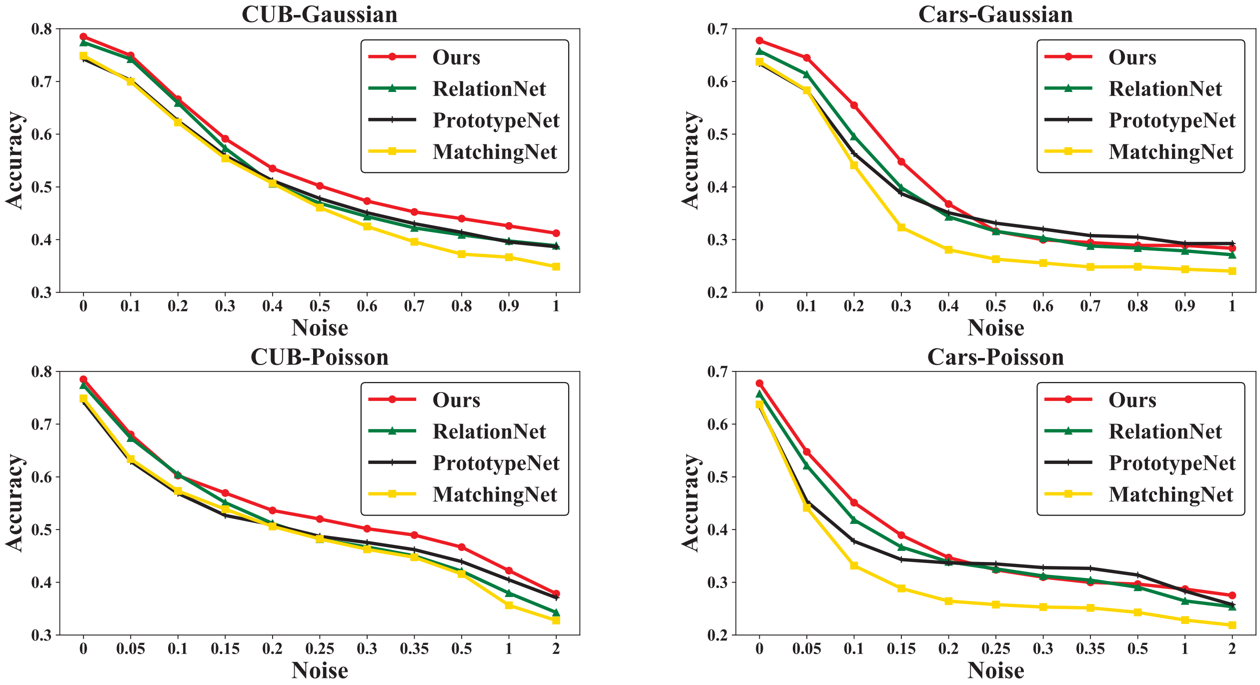

Fig. 4. Comparison of 5-way 5-shot classification accuracy obtained by Matching Network, Prototypical Network, Relation Network and the proposed Anti-Noise Relation Network (Ours) on the CUB-200-2011 (CUB) and Stanford-Cars (Cars) datasets. Gaussian noise and Poisson noise are added to the test data; noise intensities are labeled on the horizontal axis.

example image with gradually increased noise. It is clear that, as the noise intensity increases, the extracted feature picture becomes more and more blurred.

Figure 4 (upper) shows the 5-way 5-shot classification accuracy of all methods in the presence of Gaussian noise; Gaussian noise is randomly synthesized for ten times, and the mean accuracy is reported. First, although our method is proposed to enhance robustness to test-time noise, its classification performance on clean data (i.e. Noise=0) is slightly higher than compared methods. Second, when the intensity of Gaussian noise increases, the performance gain from our method becomes more pronounced on the CUB dataset. On the Cars dataset, our method achieves the highest accuracy in the scenario of low noise, and outperforms the Relation Network in the scenario of high noise. These results verify that the proposed method is more effective in safeguarding against noise.

To further understand the robustness of different methods against Gaussian noise, we present the boxplots of classification accuracy at two noise levels in Figure 5 (upper). In both low and high noise levels, our method has a smaller spread, as indicated by a more narrow interquartile range, which suggests that our method is more reliable in the presence of noises.

In addition to the Gaussian noise, we test the methods under the Poisson noise. Figure 4 (bottom) shows the classification accuracy with gradually increasing noise intensities. Similar patterns to the Gaussian noise can be found. On the CUB dataset, our method outperforms Relation Network in both noise-free and noisy cases, and the advantage is clearer as noise level increases. On the Cars dataset, our method is

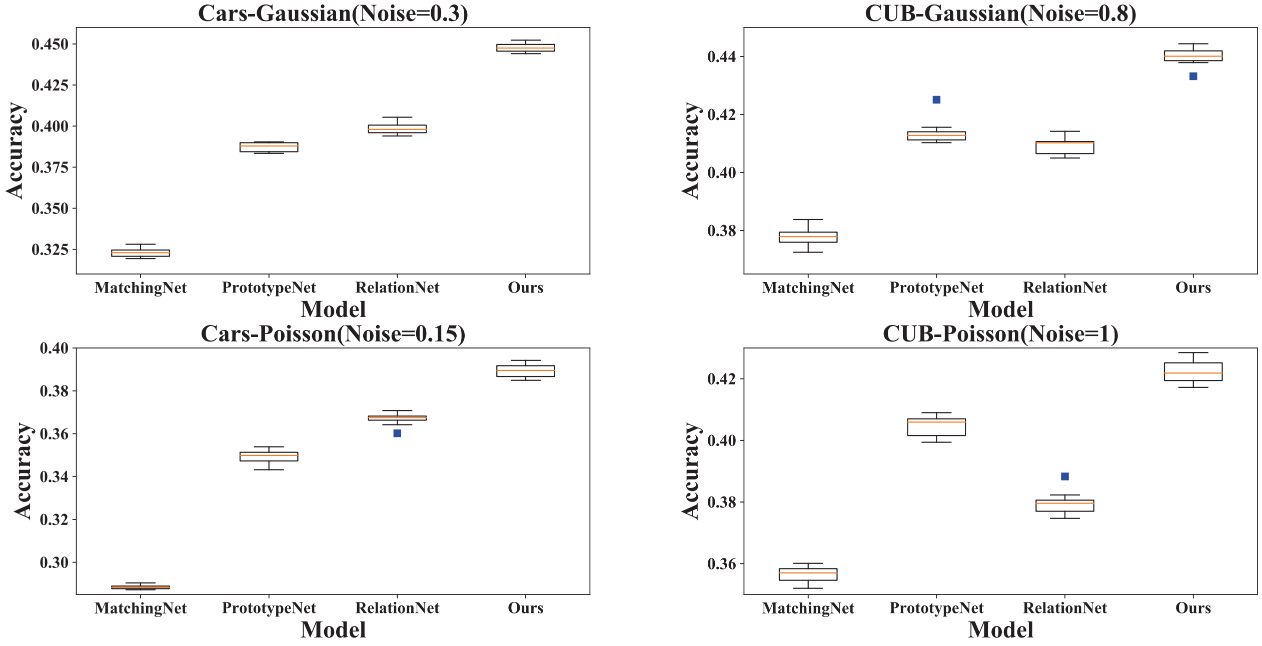

Fig. 5. Comparison of the 5-way 5-shot accuracy via boxplot on the CUB-200-2011 (CUB) and Stanford-Cars (Cars) datasets. The methods include Matching Network, Prototypical Network, Relation Network and the proposed Anti-Noise Relation Network (Ours). Each method runs 10 rounds on each dataset.

optimal in the low-noise case and is competitive in the high-noise case. Figure 5 (bottom) shows the boxplots of accuracy under low and high Poisson noises. The spread of our method is slightly larger than Relation Network. We hypothesize that this is caused by the mismatch between the synthetic noise and the test Poisson noise, as the noise data generated from the anti-noise module is controlled by the Euclidean distance (Eq. 5) and implies a Gaussian noise.

In summary, the experimental results demonstrate that the proposed method achieves better robustness against test-time noises, without sacrificing accuracy on clean data.

IV. Conclusion

In this paper, we have proposed *Anti-Noise Relation Network* to enhance the robustness of few-shot classification methods against potential noise on future unseen data. The proposed network is constructed by embedding an anti-noise module, i.e. an autoencoder network (AE), into Relation Network. Experimental results show that on the Stanford-Cars and the CUB-200-2011 datasets, the proposed method achieves better classification performance than other compared methods on noisy test data. It demonstrates the efficacy of the anti-noise module.

This work can be considered as the first exploration of embedding an anti-noise module in metric-based few-shot classification methods. In the future, we will incorporate it into other metric-based methods to improve their robustness to test-time noise.

Acknowledgement

This work was supported in part by the National Key R&D Program of China under Grant 2019YFF0303300 and Subject II No. 2019YFF0303302, in part by the National Natural Science Foundation of China (NSFC) under Grant 61906080, Grant 61763028, Grant 61773071, Grant 61922015, and Grant U19B2036, in part by the Beijing Natural Science Foundation Project No. Z200002, in part by the Beijing Academy of Artificial Intelligence (BAAI) under Grant BAAI2020ZJ0204, in part by the Beijing Nova Programme Interdisciplinary Cooperation Project under Grant Z191100001119140, in part by Key Research and Development Project of Shandong Province No.2019GGX101036, in part by the Hong-liu Outstanding Youth Talents Foundation of Lanzhou University of Technology.

References

[1] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2019.

[2] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018.

[3] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[4] Jun Shu, Zongben Xu, and Deyu Meng. Small sample learning in big data era. *arXiv preprint arXiv:1808.04572*, 2018.

[5] Dongliang Chang, Yifeng Ding, Jiyang Xie, Ayan Kumar Bhunia, Xiaoxu Li, Zhanyu Ma, Ming Wu, Jun Guo, and Yi Zhe Song. The devil is in the channels: Mutual-channel loss for fine-grained image classification. *IEEE Transactions on Image Processing*, 29:4683–4695, 2020.

[6] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006.

[7] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016.

[8] Xiaoxu Li, Liyun Yu, Xiaochen Yang, Zhanyu Ma, and Jun Guo. Remarnet: Conjoint relation and margin learning for small-sample image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, PP(99):1–1, 2020.

[9] Xiaoxu Li, Zhuo Sun, Jing-Hao Xue, and Zhanyu Ma. A concise review of recent few-shot meta-learning methods. *arXiv preprint arXiv:2005.10953*, 2020.

[10] Xiaoxu Li, Liyun Yu, Dongliang Chang, Zhanyu Ma, and Jie Cao. Dual cross-entropy loss for small-sample fine-grained vehicle classification. *IEEE Transactions on Vehicular Technology*, 68(5):4204–4212, 2019.

[11] Xiaoxu Li, Dongliang Chang, Zhanyu Ma, Zheng-Hua Tan, Jing-Hao Xue, Jie Cao, Jingyi Yu, and Jun Guo. Oslnet: Deep small-sample classification with an orthogonal softmax layer. *IEEE Transactions on Image Processing*, 2020.

[12] Zitian Chen, Yanwei Fu, Yu-Xiong Wang, Lin Ma, Wei Liu, and Martial Hebert. Image deformation meta-networks for one-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8680–8689, 2019.

[13] Xiaoxu Li, Liyun Yu, Dongliang Chang, Zhanyu Ma, and Jie Cao. Dual cross-entropy loss for small-sample fine-grained vehicle classification. *IEEE Transactions on Vehicular Technology*, 68(5):4204–4212, 2019.

[14] Nanqing Dong and Eric P Xing. Domain adaption in one-shot learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 573–588. Springer, 2018.

[15] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *arXiv preprint arXiv:1904.04232*, 2019.

[16] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*, pages 9516–9527, 2018.

[17] Jongmin Kim, Taesup Kim, Sungwoong Kim, and Chang D Yoo. Edge-labeling graph neural network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11–20, 2019.

[18] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.

[19] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087, 2017.

[20] F Sung, Y Yang, L Zhang, T Xiang, PH Torr, and TM Hospedales. Learning to compare: Relation network for few-shot learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1199–1208, 2018.

[21] Tobias Plötz and Stefan Roth. Neural nearest neighbors networks. In *Advances in Neural Information Processing Systems*, pages 1087–1098, 2018.

[22] Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. Adversarially regularized graph autoencoder for graph embedding. *arXiv preprint arXiv:1802.04407*, 2018.

[23] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018.

[24] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3973–3981, 2015.

[25] C Wah, S Branson, P Welinder, P Perona, and S Belongie. The cub-200-2011 dataset. Technical report, Technical report, Cal-Tech, 2011.