Metric-Based Learning for Nearest-Neighbor Few-Shot Image Classification

Min Jun Lee, Jungmin So
Department of Computer Science and Engineering
Sogang University
Seoul, Korea
roblee100@sogang.ac.kr, jso1@sogang.ac.kr

2021 International Conference on Information Networking (ICOIN) | 978-1-7281-9101-0/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICOIN50884.2021.9333850

***Abstract*—Few-shot learning task, which aims to recognize a new class with insufficient data, is an inevitable issue to be solved in image classification. Among recent work, Meta-learning is commonly used to figure out few-shot learning task. Here we tackle a recent method that uses the nearest-neighbor algorithm when recognizing few-shot images and to this end, propose a metric-based approach for nearest-neighbor few-shot classification. We train a convolutional neural network with miniImageNet applying three types of loss, triplet loss, cross-entropy loss, and combination of triplet loss and cross-entropy loss. In evaluation, three configurations exist according to feature transformation technique which are unnormalized features, L2-normalized features, and centered L2-normalized features. For 1-shot 5-way task, the triplet loss model attains the uppermost accuracy among all three configurations and for 5-shot 5-way task, the identical model reaches the foremost accuracy in unnormalized features configuration.**

***Index Terms*—Few-shot learning, Metric-learning, Triplet loss, Nearest-neighbor, Embedding network, Image classification**

I. Introduction

As deep learning technology evolves, remarkable achievements are shown in various types of fields. In computer vision, the image classification for certain dataset obtain notable accuracy with high-level neural networks [1]–[3]. However, classifying images of a specific category has limitations for being implemented in reality. There are infinite types of classes with insufficient data to be classified [4]. Hence, research on classifying novel classes with small number of images which is called few-shot learning is actively in progress [5]–[7].

Meta-learning, which enables classifier learn how to learn, is vital for reaching the objective of few-shot learning task. Normally, meta-learning is attempted in the form of episodic training [8]–[10]. Episodic training is a way to enhance generalization performance by inducing the classifier to derive learning rules on its own through a training task similar to a few-shot learning task. Meta-learning can roughly be divided into two, metric-based and optimization-based [5].

In this study, we stress on metric-based learning which is a method based on distance measure [11]. Specifically, few-shot classification using nearest-neighbor [12] is being tackled. This method classifies images with nearest-neighbor classification, but distance metric is not utilized for training. Thus, we propose a scheme to raise the accuracy of nearest-neighbor classification for miniImageNet [13] through metric-based learning.

Particularly, we analyze the performance of the convolutional neural network trained with three losses, triplet loss [14], cross-entropy loss, and mixed loss through nearest-neighbor classification. Mixed loss indicates the mixture of triplet loss and cross-entropy loss. Evaluation consists of three configurations, each named unnormalized (UN) features, L2-normalized (L2N) features, and centered L2-normalized (CL2N) features. UN simply compares feature vectors from the embedding network, L2N measures L2-normalized feature vectors, and CL2N performs L2-normalization after subtracting the mean vector of training classes from the feature vectors [12]. We evaluate on two few-shot settings, 1-shot 5-way and 5-shot 5-way for each configuration. The classifier trained with triplet loss display the leading accuracy of 51.37% in 1-shot 5-way setting. For 5-shot 5-way setting, triplet loss classifier only has the leading accuracy of 67.14% in UN configuration, but all configurations indicate similar accuracy.

II. Related Work

A. Metric-Based Few-Shot Learning

Metric-based learning is a meta-learning method that learns how to measure similarity between data through distance metric [11]. There are several metric-based few-shot learning models that are noteworthy. Siamese Network [8] trains a model to distinguish whether two input images are in the same category or not. Matching Network [13] avoids fine-tuning to adapt to novel class by learning how to connect unlabelled sample with small amount of labelled support set. Prototypical Network [10] first defines a prototype feature vector by averaging image feature vectors of the equivalent class in the support set. Then, it trains to induce the feature vector of unlabelled image close to the prototype feature vector of the identical class. A self-supervised model AMDIM [15] applies Prototypical Network by fine-tuning a self-supervised model with prototype feature vector. SimpleShot [12] trains a convolutional neural network linked to a fully connected layer utilizing cross-entropy loss and extracts image feature vectors from convolutional neural network without fully connected layer. Then it classifies feature vectors through nearest-neighbor rule. Among many other metric-based models, we focused on metric learning using triplet loss [14]. Triplet loss is applied not only to achieve face recognition [16] and image classification [17], but also to achieve the goal of few-shot learning [18], [19].

B. Optimization-Based Few-Shot Learning

This approach is to transform the optimization algorithm so that the classifier can sufficiently learn even with a small number of samples [20]. The optimization-based method for few-shot learning is implemented with Long Short-Term Memory (LSTM) [21] based meta learner model [22] which efficiently updates learner parameters with small support set so that learner can rapidly adjust to novel tasks. The reason for employing LSTM is because the gradient-based update of backpropagation is similar to the cell-update of LSTM. Another optimization-based model compatible with all models that are trained through gradient descent is Model-Agnostic Meta-Learning (MAML) [23]. This allows the parameters to be located in optimal position of several tasks so that even with small steps of gradient descent, it can derive proper parameters for certain task. Recent work has shown a classifier that combines Graph Neural Network with MAML [7]. Graph Neural Network is employed in this work because it generates more flexible description of data besides Euclidean space.

C. Transductive Learning

Transductive learning is an efficient way to overcome the lack of labeled data [24]. It learns to utilize the unlabeled test data for classification. A common way of implementing transductive approach into few-shot learning is to update the prototype of each class using the confidence of query samples [25], [26]. Depending on each technique, confidence of query samples are measured in a different way leading it to update prototypes diversely. Meta-Confidence Transduction (MCT) [27] meta-learns the confidence of training query samples, and allocates optimal weights to testing queries. This is done by meta-learning an input-adaptive distance metric. Other than approach that updates prototype, [5] demonstrates a method that attaches a new fully connected layer to pre-trained model, and retrains it with support set. Then transductive fine-tuning is done which regularizes classifier parameters with query samples.

III. Preliminaries

A. Nearest-Neighbor classification for Few-Shot learning

We start by introducing the problem of few-shot learning. Denoting image as x and class by y, the training set $\mathcal{D}$ consisting of k images can be expressed as $\mathcal{D}_{base} = \{(x_1, y_1), ..., (x_k, y_k)\}$, which has class $y_i \in \{1, ..., A\}, 1 \leq i \leq k$. Suppose we have a support set $\mathcal{D}_{support}$ which has N images for each of C novel classes, and $\mathcal{D}_{query}$ as a query set consisting of C novel class images. The goal of the few-shot learning is to train a model that correctly classifies $\mathcal{D}_{query}$ using $\mathcal{D}_{support}$. This is generally referred to as N-shot C-way setting. Fig. 1 represents an episodic training for few-shot learning. In each episode, a support set is given to learn how to match the query set. Test phase proceeds image classification

Fig. 1: Example of few-shot learning. In order to achieve the objective of few-shot learning, meta-learning is carried out by episodic training.

in the same way, and episodic training is performed to obtain satisfactory results in this test.

There is a intuitive approach called SimpleShot [12] consisting of a feature extraction step and a nearest-neighbor classification step to fulfill the purpose of few-shot learning. In the feature extraction step, embedding network with cross-entropy loss can be defined as f_θ^{ce} which has parameter θ_{ce}. The cross-entropy loss is minimized by training θ_{ce} with sample $\mathcal{D}_{base}$, which can be expressed as:

$$\text{argmin}_{\theta_{ce},W} \sum_{(x,y)\in\mathcal{D}_{base}} \ell_{ce}(W^\top f_\theta^{ce}(x), y), \tag{1}$$

where W is a fully connected layer attached at the end of f_θ^{ce}. After the training, feature extraction starts with defining $\mathcal{D}_{support} = \{(x_1^{'}, 1), ..., (x_C^{'}, C)\}$ and each $\mathcal{D}_{query}$ image as $x^{''}$. All the $\mathcal{D}_{support}$ and $\mathcal{D}_{query}$ images are extracted as a feature with embedding network, $f_\theta^{ce}(x'), f_\theta^{ce}(x'') \in \mathbb{R}^D$.

In the nearest-neighbor classification stage, distance is measured between $f_\theta^{ce}(x')$ and $f_\theta^{ce}(x'')$, using distance metric d as $d(f_\theta^{ce}(x'), f_\theta^{ce}(x''))$. The classification rule for $x^{''}$ is then expressed as follows:

$$y(x'') = \text{argmin}_{c\in\{1,...,C\}} d(f_\theta^{ce}(x_c^{'}), f_\theta^{ce}(x'')). \tag{2}$$

If $N > 1$ in N-shot C-way, images that fall under the same category are averaged and represented as a single feature vector. Assuming l_2 distance as base, there are two feature transformation method available at feature space. One is L2-normalization, which can be denoted as $f_\theta^{ce}(x') = \frac{f_\theta^{ce}(x')}{\left\|f_\theta^{ce}(x')\right\|_2}$. The other is centralization, which is $f_\theta^{ce}(x') = f_\theta^{ce}(x') - f_\theta^{ce}(\bar{x})$ when $f_\theta^{ce}(\bar{x}) = \frac{1}{\mathcal{D}_{base}} \sum_{x\in\mathcal{D}_{base}} f_\theta^{ce}(x)$.

B. Triplet Loss

The goal of the triplet loss is to minimize the distance between images of the same category and maximize the distance between images of the different categories [14].

Fig. 2 exemplifies the general training procedure of triplet loss. Suppose that L2-normalized image feature vector is denoted as $f(x) \in \mathbb{R}^d$. In order to derive the triplet loss, there must be an anchor image x, positive image x^+, and negative image x^-. Each image in $(x, x^+, x^-) \in \mathcal{D}_{base}$ possesses a single label in label set $Y = (y_1, y_2, ..., y_k), 1 \leq i, j \leq k$, which anchor and positive have the same y_i of $x \neq x^+$ and negative image x^- has the y_j of $y_j \neq y_i$. At this point, the positive distance d_+ and the negative distance d_- can be measured as:

$$d_+ = \left\| f(x) - f(x^+) \right\|_2^2 \tag{3}$$

$$d_- = \left\| f(x) - f(x^-) \right\|_2^2, \tag{4}$$

and in order to induce learning in the desired direction, the following equation has to satisfy:

$$d_+ + \alpha < d_-. \tag{5}$$

Here, a is a margin which enlarges dissimilarity between d_+ and d_-. Triplet loss then can be demonstrated as:

$$\ell_{trp} = [d_+ - d_- + \alpha]_+, \tag{6}$$

which has to be minimized to 0.

IV. Approach

The method we propose is to train a model appropriate for nearest-neighbor few-shot classification. SimpleShot [12] is a very facile and effective method, but intuitively, there is a finer way to train for nearest-neighbor classification. With the purpose of nearest-neighbor classification for novel class, it is more reasonable to perform metric learning about which classes are the same or different, rather than conducting supervised learning for a specific class. This is because nearest-neighbor rule is based on a distance metric, so training with distance metric will induce the embedding network to extract features that benefit when classifying through nearest-neighbor algorithm. Therefore, we train the embedding network with triplet loss and accomplish the objective of few-shot learning through nearest-neighbor classification. We examine two types of loss, triplet loss and mixed loss in addition to cross-entropy loss for the embedding network.

A. Triplet Loss Training for Nearest-Neighbor Few-Shot Classification

Let us denote θ_{trp} as the parameter of embedding network f_θ^{trp} that minimizes the triplet loss. In the training step, there are I iterations per epoch and θ_{trp} is updated by learning B batches for each iteration. Each batch b_{trp} consists of $\mathrm{b}_{trp} = (x, x^+, x^-) \in \mathcal{D}_{base}$ and they have the same condition as introduced above which is $x \neq x^+$ with y_i and x^- having y_j of $y_j \neq y_i$. The equation of minimizing the triplet loss to 0 then can be expressed as:

$$\mathrm{argmin}_{\theta_{trp}} \sum_{(x,x^+,x^-)\in\mathcal{D}_{base}} \ell_{trp}(f_\theta^{trp}(x), f_\theta^{trp}(x^+), f_\theta^{trp}(x^-)), \tag{7}$$

which updates θ_{trp} for I times per epoch. In the training equation, ℓ_{trp} represents (6) and the distance metric is based

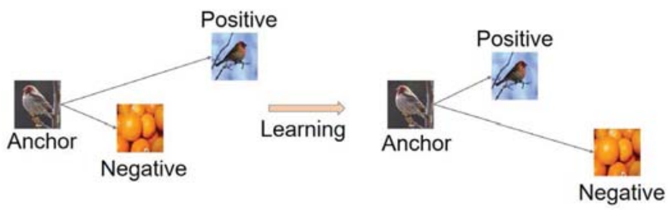


Fig. 2: Illustration of triplet loss learning process. It reduces the distance between anchor and positive samples, and increases the distance between anchor and negative samples.

on l_2 distance. Following the training step, nearest-neighbor classification (2) is done with the N-shot C-way setting. For $\mathcal{D}_{support}$ and $\mathcal{D}_{query}$, instead of extracting the embedding vector $f_\theta^{ce}(x')$ and $f_\theta^{ce}(x'')$, we use $f_\theta^{trp}(x')$ and $f_\theta^{trp}(x'')$ with l_2 distance to calculate the similarity.

B. Mixed Loss Training for Nearest-Neighbor Few-Shot Classification

We employed both cross-entropy loss and triplet loss, which is the mixed loss, for the embedding network, allowing it to better represent the features of each class while training for similiarity. Denoting parameter of the embedding network f_θ^{mix} that minimizes mixed loss as θ_{mix}, we compute ℓ_{ce} and ℓ_{trp} first with b_{trp} and coalesce them to obtain ℓ_{mix}:

$$\ell_{mix} = \ell_{ce}(W^\top f_\theta^{mix}(\mathrm{b}_{trp}), \mathrm{b}_y) + \ell_{trp}(f_\theta^{mix}(\mathrm{b}_{trp})). \tag{8}$$

Here, $f_\theta^{mix}(\mathrm{b}_{trp})$ expresses feature vector of each (x, x^+, x^-) from embedding network f_θ^{mix}, and b_y indicates label set of b_{trp}. The overall equation of minimizing ℓ_{mix} is then shown as follows:

$$\mathrm{argmin}_{\theta_{mix},W} \sum_{(\mathrm{b}_{trp}\mathrm{b}_y)\in\mathcal{D}_{base}} \ell_{mix}(f_\theta^{mix}(\mathrm{b}_{trp}), \mathrm{b}_y). \tag{9}$$

Equation (9) iterates for I times per epoch, equivalent to (7). Classification also operates identical as demonstrated in triplet loss training approach.

V. Experiment

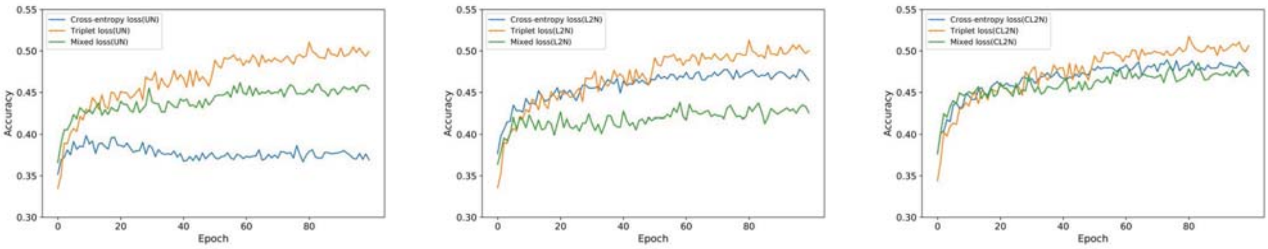
A. Experimental Setting

a) Dataset: We use miniImageNet [13] for the experimental dataset. This dataset is a sample of 100 classes from ImageNet [28], and 600 images exist for each class. Since the images have different size each, they are resized to a shape of $84 \times 84 \times 3$ [13], and 64 out of 100 classes are used as a training set, 16 as a validation set, and 20 as a test set [22].

b) Model and Training Details: The backbone network employed in the experiment is a basic 4-layer convolutional neural network (Conv-4) with 64 channels each. Adam is chosen as the optimizer, which has a learning rate 0.001. Training is progressed for 100 epochs. SimpleShot [12] which applies cross-entropy loss utilizes a batch size of 256 per

TABLE I: Nearest-neighbor few-shot classification accuracy results on miniImageNet on 1-shot 5-way and 5-shot 5-way. A basic 4-layer convolutional neural network is employed as a backbone. All outcomes are stated with 95% confidence intervals.

Method	Shot	Test Accuracy		
		UN	L2N	CL2N
SimpleShot	1	38.38 $\pm$ 0.18	49.42 $\pm$ 0.19	51.14 $\pm$ 0.19
Proposed$_{trp}$	1	**50.97 $\pm$ 0.20**	**51.37 $\pm$ 0.20**	**51.34 $\pm$ 0.19**
Proposed$_{mix}$	1	47.51 $\pm$ 0.19	44.76 $\pm$ 0.19	49.23 $\pm$ 0.19
SimpleShot	5	66.31 $\pm$ 0.16	**68.22 $\pm$ 0.16**	**68.51 $\pm$ 0.16**
Proposed$_{trp}$	5	**67.14 $\pm$ 0.16**	67.21 $\pm$ 0.17	67.14 $\pm$ 0.16
Proposed$_{mix}$	5	65.84 $\pm$ 0.16	65.34 $\pm$ 0.17	65.91 $\pm$ 0.16



(a) UN (b) L2N (c) CL2N

Fig. 3: Validation accuracy during training in 1-shot 5-way environment, for models trained using three different loss functions. Validation is done by nearest-neighbor few-shot classification. Accuracy for each epoch is derived from the average of 500 tasks using 16 classes of validation set in miniImageNet. Distance is measured as Euclidean distance with three configurations, unnormalized (UN) features, L2-normalized (L2N) features, and centered L2-normalized (CL2N) features.

epoch. For triplet loss and mixed loss training, we use batch size B=300, which has total of 900 images and iterates for I=150 times in each epoch. Only the 64 training set classes of miniImageNet are applied for training.

c) Evaluation protocol: For evaluation, images of 20 classes from miniImageNet is used and $\mathcal{D}_{support}$ is formed under two condition: 1-shot 5-way and 5-shot 5-way. There are 10000 tasks for each N shot C way evaluation method, and N support images for each C novel class per task. Furthermore, 15 query images exist per C novel class in $\mathcal{D}_{query}$. When evaluating models, we also use three types of feature transformation that SimpleShot [12] implemented, which are unnormalized (UN), L2-normalization (L2N) and centered L2-normalization (CL2N). Accuracy is expressed as the average of entire classification cases, and a 95% confidence interval is provided as well.

B. Experimental Result

We show the results of triplet loss and mixed loss training for nearest-neighbor classification on miniImageNet in few-shot learning. We designate model trained with triplet loss as Proposed$_{trp}$, and model trained with mixed loss as Proposed$_{mix}$. In Table I, we can see that 1-shot nearest-neighbor classification on Proposed$_{trp}$ displays the highest accuracy for all 3 configurations, unnormalized (UN), L2-normalization (L2N), and centered L2-normalization (CL2N). In particular, UN shows significant improvement from SimpleShot [12] with the accuracy of 50.97%. For L2N, Proposed$_{trp}$ exhibits the foremost accuracy among all the cases in 1-shot environment which is 51.37%. Accuracy for CL2N is 51.34% which has slightly increased from SimpleShot [12].

As for 5-shot 5-way setting, the UN configuration of Proposed$_{trp}$ displays the leading accuracy of 67.14, and SimpleShot [12] maintains the uppermost accuracy in the rest of configurations. Even if the accuracy of SimpleShot [12] comes out a little higher with feature transformation, all three accuracy of Proposed$_{trp}$ are comparable. This can be seen that the Proposed$_{trp}$ extracts feature in a somewhat normalized state for various images.

The mixed loss model Proposed$_{mix}$ fails to present satisfactory accuracy overall as it does not appear to have a harmonious impact on the two losses.

Fig. 3 demonstrates validation accuracy of nearest-neighbor classification on 1-shot 5-way task while training the three models. Each epoch tests 500 tasks from validation set, and Euclidean distance with UN features, L2N features, CL2N features is measured for distance. As observed in Fig. 3, Proposed$_{trp}$ sustains the best accuracy in all three configurations as the epoch increases. This clarifies that triplet loss benefits metric-based classification than cross-entropy loss in 1-shot 5-way setting. The Proposed$_{mix}$ model leads only at the beginning in two configurations, UN and CL2N. Afterwards, The accuracy of the Proposed$_{mix}$ rises slowly, and from this it seems that the mixed loss does not show significant impact for nearest-neighbor classification.

VI. Conclusion

We proposed a metric-based training approach to enhance the existing nearest-neighbor classification method for few-shot learning [12]. We trained the embedding network using triplet loss, cross-entropy loss, and mixed loss, then analyzed the result for nearest-neighbor classification. In 1-shot setting, utilizing the triplet loss performed a significant effect. For 5-shot setting, the same loss had the finest accuracy in unnormalized configuration, and analogous accuracy were shown for entire configurations. However, the suggested triplet loss training model consumes excessive GPU memory and due to this, it is arduous to implement high-level backbones such as ResNet [1] or DenseNet [2]. Therefore, we plan to improve our current model to be able to train on those backbones with limited GPU memory. Mixed loss will also be developed to better learn similarity, and an ensemble model that combines model trained with triplet loss and cross-entropy loss will be discussed in future studies.

Acknowledgment

This work was supported in part by the National Research Foundation (NRF) of Korea under grant No. 2019R1A2C1005881.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 770–778.

[2] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jul. 2017, pp. 4700–4708.

[3] S. Zagoruyko and N. Komodakis, “Wide residual networks,” in Proc. British Mach. Vision Conf., 2016.

[4] G. V. Horn and P. Perona, “The devil is in the tails: Fine-grained classification in the wild,” ArXiv, vol. abs/1709.01450, 2017.

[5] G. S. Dhillon, P. Chaudhari, A. Ravichandran, and S. Soatto, ”A baseline for few-shot image classification,” in Proc. Int. Conf. Learn. Representations, 2020.

[6] L. Song, J. Liu, and Y. Qin, “Fast and generalized adaptation for fewshot learning,” arXiv e-prints, p. arXiv:1911.10807, Nov 2019.

[7] J. Cai and S. M. Shen, “Cross-domain few-shot learning with meta finetuning,” arXiv preprint arXiv:2005.10544, 2020.

[8] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese neural networks for oneshot image recognition,” in Proc. Int. Conf. Machine Learning Deep Learning Workshop, 2015.

[9] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, “Learning to compare: Relation network for few-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1199–1208.

[10] J. Snell, K. Swersky, and R. S. Zemel, “Prototypical networks for few-shot learning,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2017, pp. 4077–4087.

[11] E. Xing, A. Ng, M. Jordan, and S. Russell, ”Distance metric learning with application to clustering with side-information,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2002.

[12] Y. Wang, W. L. Chao, K. Q. Weinberger, and L. S. van der Maaten, ”Simpleshot: Revisiting nearest-neighbor classification for few-shot learning,” arXiv preprint arXiv:1911.04623, 2019.

[13] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, “Matching networks for one shot learning,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2016, pp. 3630–3638.

[14] M. Schultz and T. Joachims, ”Learning a distance metric from relative comparisons.” in Proc. Int. Conf. Neural Inf. Process. Syst., 2004.

[15] D. Chen, Y. Chen, Y. Li, F. Mao, Y. He, and H. Xue, ”Self-supervised learning for few-shot image classification,” arXiv preprint arXiv:1911.06045, 2019.

[16] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 815–823.

[17] E. Hoffer and N. Ailon, “Deep metric learning using triplet network,” in Proc. Int. Workshop Similarity-Based Pattern Recognit., 2015, pp. 84–92.

[18] S. Puch, I. Sánchez, M. Rowe, ”Few-shot learning with deep triplet networks for brain imaging modality recognition,” in Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data, 2019, pp. 181-189.

[19] M. Ye and Y. Guo, “Deep triplet ranking networks for one-shot recognition,” arXiv preprint arXiv:1804.07275, 2018.

[20] J. Schmidhuber, “Learning to control fast-weight memories: An alternative to dynamic recurrent networks,” Neural Computation, vol. 4, no. 1, pp. 131–139, 1992.

[21] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[22] S. Ravi and H. Larochelle, “Optimization as a model for few-shot learning,” in Proc. Int. Conf. Learn. Representations, 2017, pp. 1–11.

[23] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in Proc. Int. Conf. Machine Learning, 2017, pp. 1126–1135.

[24] V. N. Vapnik, “An overview of statistical learning theory,” IEEE Trans. Neural Networks, vol. 10, pp. 988–999, Sept. 1999.

[25] R. Hou, H. Chang, M. Bingpeng, S. Shan, and X. Chen, “Cross attention network for few-shot classification,” in Proc. Int. Conf. Neural Inf. Process. Syst., 2019, pp. 4005–4016.

[26] M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J. B. Tenenbaum, H. Larochelle, and R. S. Zemel, ”Meta-learning for semi-supervised fewshot classification,” in Proc. Int. Conf. Learn. Representations, 2018, pp. 1–5.

[27] S. M. Kye, H. B. Lee, H. Kim, and S. J. Hwang, “Meta-learned confidence for few-shot learning,” arXiv preprint arXiv:2002.12017, 2020.

[28] O. Russakovsky et al., “ImageNet large scale visual recognition challenge,” Int. J. Comput. Vis., vol. 115, pp. 211–252, 2015.