Few-Shot Learning with Complex-valued Neural Networks

Zhen Liu[1]
liuzhenbuaa@buaa.edu.cn
Baochang Zhang[1]
bczhang@buaa.edu.cn
Guodong Guo[2]
guoguodong01@baidu.com

[1] Beihang University, Beijing, China
[2] Institute of Deep Learning, Baidu Research. National Engineering Laboratory for Deep Learning Technology and Application.

Abstract

Feature representation is fundamental and attracts much attention in few-shot learning. Convolutional neural networks (CNNs) are among the best feature extractors so far in this field, which are successfully combined with metric learning, leading to the state-of-the-art performance. However, the subtle difference among inter-class samples challenges existing CNN based methods, which only use real-valued CNNs that fail to extract more detailed information. In this paper, we introduce complex metric module (CMM) into metric learning, aiming to better measure the inter- and intra-class relations based on both amplitude and phase information. Specifically, building upon the recent episodic training mechanism, our CMM can enhance the representation capacity by extracting robust complex-valued features to facilitate modeling subtle relationships among samples, which can enhance the performance of the few-shot classification task when only few samples are available. Moreover, we introduce a new transductive method into CMM, by considering not only query and support but also query and query relationships to predict classes of unlabeled samples. Experiments on two benchmark datasets show that the proposed CMM significantly improves the performance over other approaches and achieves the state-of-the-art results.

1 Introduction

Great progress has been made in visiual understanding tasks [7, 11, 19, 22, 25], due to the development of deep learning models and large amounts of labeled data. However, in some scenarios the performance of deep learning are significantly deteriorated due to limited numbers of labeled data. In contrast, human babies can recognize new classes with a few of labeled examples or even some samples of different but similar classes [13]. Few-shot learning aims to solve this problem, which can be combined with deep learning to better recognize unlabeled samples with few labeled instances. A variety of few-shot learning methods have been proposed, which can be roughly divided into three types, i.e., meta-learning, metric learning and data augmentation. The meta-learning methods optimize the (hyper) parameters of neural networks so that the models can easily and efficiently adapt to a new task [5, 6]. Data augmentation is widely used to generate more samples from a small number of available instances [14, 27]. With more training samples, data augmentation can

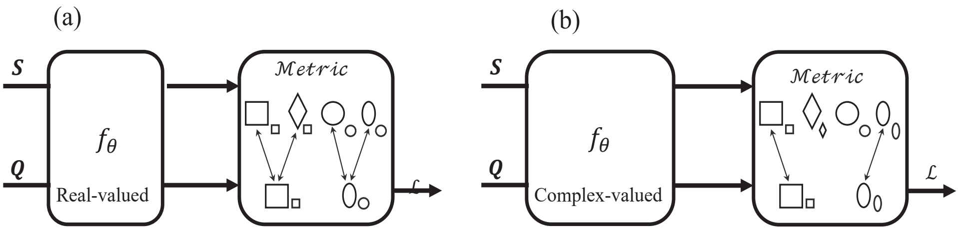

Figure 1: Illustration of the motivation for complex-valued networks. S means support samples and Q means query samples. Large geometry represents real features and small geometry represents extracted features. (a) A real-valued feature extractor extracting only amplitude features cannot find the difference between squares and parallelograms and make a wrong prediction of the unlabeled sample. (b) A complex-valued feature extractor which extracts phase features can recognize the difference between squares and parallelograms and correctly predict the class of the unlabeled sample.

significantly alleviate the overfitting problem. Among the three types, metric learning is more promising for the few-shot classification problem. It learns a mapping from images to an embedding space, where samples from the same class become closer [20, 21, 24]. To make full use of limited data, the line of metic learning methods [20, 21, 24] mainly focuses on learning accurate relationships among samples using different metrics. The drawbacks of existing methods come from the fact that they pay less attention on feature extraction, which is actually a deterministic element for the final performance [22], only based on conventional real-valued convolution neural network (CNNs) for feature representation.

On the contrary, complex-valued neural networks recently receive increasing attention [23, 28], due to their potential to enable easier optimization [16], better generalization [8] and fast learning [1]. They are proved to have a richer representational capacity than real-valued models. These models can extract complex-valued features which consist of both amplitude and phase messages. The phase component is important not only from a biological point of view but also from a signal processing perspective. It has been shown that the amount of information present in the phase of an image is sufficient to recover the majority of the information encoded in its magnitude[26]. In fact, phase provides a detailed description of objects as it encodes shapes, edges, and orientations [23].

In this paper, unlike conventional few-shot learning methods based on real-valued features, we introduce complex-valued CNNs to enhance the capacity of feature representation with richer amplitude and phase information. Also, we propose a unique metric learning method, which can measure embedding distances among samples with amplitude and phase information. By using the distance metric, we utilize the entire query set for transductive inference to deal with the few-shot problem. Specifically, we develop a novel Complex Metric Module (CMM) for few-shot learning by combining deep complex-valued CNNs and complex-value distance metric in the same framework. Firstly, we map the input images to an embedding space using this complex-valued CNNs. Then, we measure sample relations using the complex-valued metric and embedding. With a method of transductive inference, we compute the cross-entropy loss using support-query and query-query scores. Finally, we follow the recent episodic training mechanism, and all parameters can be end-to-end updated during back propagation. We illustrate our idea in Fig. 1, which shows that with the phase information, our complex-valued model can correctly find the difference between ellipses and circles, and also squares and parallelograms, therefore facilitating the prediction of unlabeled samples. The main contributions of our work are as follows:

- We are the first to use complex-valued deep neural networks in few-shot learning, to the best of our knowledge, which allows the model to gain a richer representational capacity.

- A new distance metric in the embedding space is proposed based on the complex-valued features, which significantly improve the performance of real-valued models with the same degree of freedom.
- Experiments on two open datasets miniImageNet and tieredImageNet show that the proposed CMM makes a large improvement in 1-shot and 5-shot accuracy over the state-of-the-art results.

2 Related Work

Few-Shot Learning. The concept of few-shot learning was first introduced by Fei Fei Li and Rob Fergus [13], which can learn much information from just one or a few images. In recent years, there is a growing interest in few-shot learning and a large amount of related work appears. Brenden M Lake *et al.* [12] proposed a hierarchical Bayesian model that can achieve human-level accuracy on alphabet recogition tasks with the setting of few-shot learning. Gregory Koch *et al.* [10] first introduced the Siamese network which computes the pair-wise distance between samples to classify unlabeled samples by the k-nearest neighbors algorithm for few-shot learning. Jake Snell *et al.* [20] built a prototype representation of each class which is the mean of sample embedding features of this class. Flood Sung *et al.* [21] considered that the measurement method is also a very important part of the network, which needs to be modeled, and so trained a relation network (RelationNet) (such as CNN) to learn the measurement method of distance. Lately, meta-learning based approaches rose. Sachin Ravi and Hugo Larochelle [17] designed a model updating the weights of a classifier by an LSTM. Chelsea Finn *et al.* [4] proposed a model agnostic meta-learning (MAML) algorithm to find parameters that are sensitive to changes in the task with a small number of samples. Another line for few-shot learning directly solves the over-fitting problem by data augmentation.

Complex-valued Neural Networks. Using complex parameters has many advantages from computational and biological perspectives[23, 28]. In terms of computation, Ivo Danihelka *et al.* [2] showed that holographic reduced representations store more informance with complex-valued parameters, and so as to efficient and stable retrieval from an associative memory. Unitary-RNN [1] learns a unitary weight matrix, which are a complex generalization of orthogonal weight matrices with eigenvalues of absolute value exactly 1. Compared with other orthogonal counterparts, Unitary-RNN [1] can be easier optimized, provide a richer representation and show the potential in hard tasks involving long-term dependencies. Using complex weights in neural networks is also biologically meaningful [18] where a neural network formulation based on complex-valued neuronal units is introduced. These units are attributed with not only a fire rate but also a phase, which can be used to bulid richer and versatile networks. The complex-valued formula allow one to express the output of neurons according to their firing rate and relative time of activity. The amplitude of a complex neuron represents the former, and its phase represents the latter. Moreover, input neurons with similar phases are view as synchronous since they add constructively, while asynchronous neurons increase destructively. Also, David P Reichert and Thomas Serrec [18] showed that this flexible machanism of neuronal synchrony fulfills multiple functional roles in deep networks.

Metric Learning. Metric learning [3] is one of the most effective categories of few-shot learning approaches which first learns a representation of a sample or class (*it depends on*

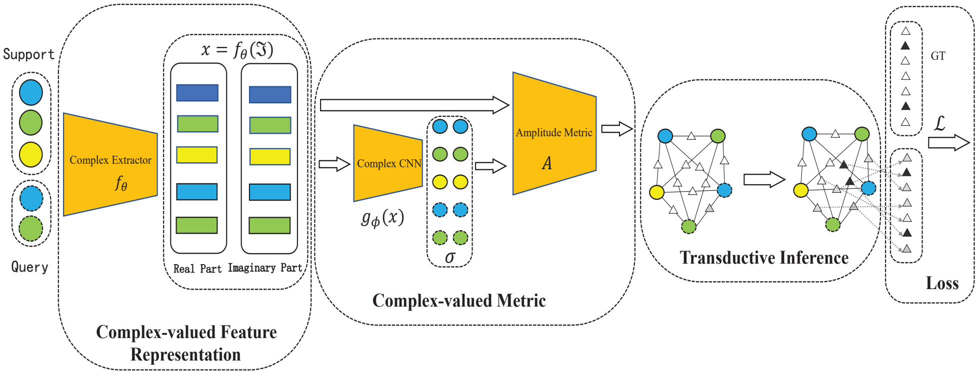

Figure 2: The overall framework of our model in which amplitude and phase information are learned by a sample-wise parameter. It is composed of four components: complex-valued feature repretation, complex-valued metric, transductive inference and loss generation.

whether inter-class information is considered) and then calculates relation scores between query samples and support samples using a metric method. Siamese network [10] trained the network to learn to extract feature embeddings in a supervised way. By calculating the distances of sample pairs, it estimates whether they belong to the same class and generates corresponding probability distributions. Matching network [24] constructed different encoders for the support set and the query set, and the output of the final classifier is a weighted sum of the predicted relation values of the support and query samples. Prototype network [20] was based on the idea that there is a prototype representation for each class, and the prototype of the class is the mean of the support set in the embedding space. Then, the classification problem becomes finding the nearest neighbor in the embedding space. Flood Sung *et al.* [21] believed that the metric is a very important part of the model and a single fixed distance metric may not be optimal, so they trained a network to learn a better distance metric.

3 Our Approach

The proposed method is illustrated in Fig. 2, which utilizes both amplitude and phase information of complex-valued CNNs to improve the performance for the few-shot classification problem.

3.1 Problem Definition

Typically, for few-shot classification tasks, there are two datasets: training set $\mathcal{D}_{train}$ and test set $\mathcal{D}_{test}$, which do not share the same categories. Generally speaking, $\mathcal{D}_{train}$ contains many classes, each of which has multiple samples. In the training stage, we randomly select C categories in $\mathcal{D}_{train}$, K samples of each category as the labeled data which form the support set (S_S), and then select Q samples from the remaining data of these C categories as unlabeled data which form the query set (S_Q). The model is required to learn how to distinguish these $C*Q$ samples in the C categories. Such a task is called the C-way K-shot problem. In each task, the selected data are $(S_S, y_S, S_Q, y_Q) = (\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_{C*(K+Q)}; y_1, y_2, ..., y_{C*(K+Q)})$, where $\mathcal{I}_i$ and y_i denote an image and its label respectively. In the training process, each episode[24] samples different meta-tasks including different combinations of classes. This mechanism enables the model to learn the common knowledge of different meta-tasks, such as how to extract important features and compare samples, and forget the task-related parts of meta-tasks. Through this learning mechanism, samples can be classified well for new meta-tasks.

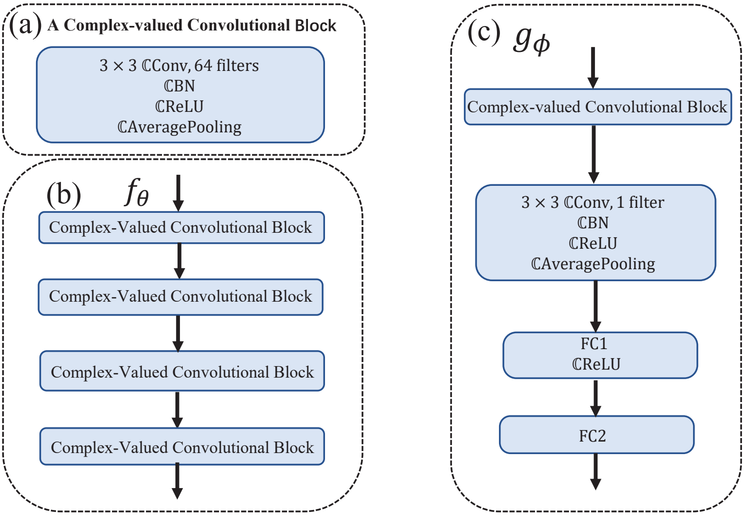

Figure 3: The detailed architecture of CMM. (a) The detailed architecture of a complex-valued convolutional block. (b) The detailed architecture of the feature extractor f_θ. (c) The detailed architecture of the network g_ϕ

3.2 Complex Metric Module

The proposed Complex Metric Moudule (CMM) consists of two components: feature embedding with a complex-valued convolution neural network and a complex metric unit that can measure relationships between samples.

Complex-valued Feature Representation. To make full use of limited samples, we employ complex convolutions and other corresponding components including complex batch-normalization, complex pooling and complex Relu strategies [23] for complex-valued CNNs. The rule is different from traditional CNNs. Assuming there is an input $\mathcal{I} = X + Yi$, and a complex filter matrix $W = A + iB$, where X is an image matrix, Y is initialized to $\mathbf{0}$, and A and B are real matrices since we simulate complex arithmetic using real-valued entities. Specially, the rule of complex convolution is

$$\begin{bmatrix} \Re(\mathcal{I} * \mathrm{W}) \\ \Im(\mathcal{I} * \mathrm{W}) \end{bmatrix} = \begin{bmatrix} \mathrm{A} & -\mathrm{B} \\ \mathrm{B} & \mathrm{A} \end{bmatrix} * \begin{bmatrix} \mathrm{X} \\ \mathrm{Y} \end{bmatrix}. \tag{1}$$

Complex Relu (ℂRelu) and complex Pooling (ℂPooling) both act on the real and the imaginary parts of a neuron separately

$$\mathbb{C}\mathrm{Relu}(\mathrm{z}) = \mathrm{Relu}(\Re(\mathrm{z})) + i\mathrm{Relu}(\Im(\mathrm{z})), \tag{2}$$

$$\mathbb{C}\mathrm{Pooling}(\mathrm{z}) = \mathrm{Pooling}(\Re(\mathrm{z})) + i\mathrm{Pooling}(\Im(\mathrm{z})). \tag{3}$$

We standardize the complex data to the standard normal complex distribution by scaling the data with the square root of their variances. Specially, we multiply the 0-centered data $(x - E[x])$ by the inverse square root of the 2×2 covariance matrix V as

$$\hat{x} = (V)^{-\frac{1}{2}}(x - E[x]), \tag{4}$$

Similar to the real-valued batch normalization algorithm, two parameters, β and γ are used in complex-valued batch normalization. The complex batch normalization is defined as

$$\mathbb{C}\mathrm{BN}(\hat{\mathrm{x}}) = \gamma\hat{\mathrm{x}} + \beta. \tag{5}$$

The chain rule for complex-valued neural networks is also used in the back propagation process. Let L be a real-valued loss function and z be a complex variable such that $z = a + ib$ where $a, b \in \mathbb{R}^D$. Then

$$\nabla_L(z) = \frac{\partial L}{\partial z} = \frac{\partial L}{\partial a} + i\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \Re(z)} + \frac{\partial L}{\partial \Im(z)} = \Re(\nabla_L(z)) + i\Im(\nabla_L(z)). \tag{6}$$

To make a fair comparison in the experiments, the feature extractor f_θ follows the same architecture as in the latest works [4, 20], which consists of four convolutional blocks (see Fig. 3). Each block begins with a 2D complex-valued convolutional layer with a 3 $\times$ 3 kernel and 64 filters, and also includes a complex batch-normalization layer, a complex Relu nonlinearity, and a 2 $\times$ 2 average-pooling layer.

Complex-valued Metric. Different from other methods for few-shot learning, the extracted complex-valued features $x_i = f_\theta(\mathcal{I}_i) = \Re(x_i) + i\Im(x_i)$, $\Re(x_i), \Im(x_i) \in \mathbb{R}^D$, in CMM have both amplitude and phase information, where D is the number of feature dimensions. We design a unique metric learning method to measure relationships between samples. Our complex metric module contains two parts: complex-valued parameter generation and sample relation metric.

To use the amplitude and phase information in the feature embedding, we choose a commonly used Gaussian similarity function based on a learnable complex-valued network g_ϕ to produce a sample-wise length-scale parameter σ_i,

$$\sigma_i = g_\phi(f_\theta(\mathcal{I}_i)) = g_\phi(\Re(x_i) + i\Im(x_i)), \tag{7}$$

where $\sigma_i = \Re(\sigma_i) + i\Im(\sigma_i)$ and σ_i is generated by the amplitude and phase information of feature embedding. The detailed architecture is illustrated in Fig. 3. Then, our relationship matrix is defined below.

$$A_{i,j} = \exp(-\frac{1}{2}d(\mathcal{M}(\frac{x_i}{\sigma_i}), \mathcal{M}(\frac{x_j}{\sigma_j}))), \tag{8}$$

where d denotes the distance function, and $\mathcal{M}$ means the $\mathcal{L}_2$ norm. We can also use the real and imaginary parts of x_i and σ_i to define the operation

$$\frac{x_i}{\sigma_i} = \frac{\Re(x_i) + i\Im(x_i)}{\Re(\sigma_i) + i\Im(\sigma_i)} = \frac{\Re(x_i)\Re(\sigma_i) + \Im(x_i)\Im(\sigma_i) + i(\Re(\sigma_i)\Im(x_i) - \Re(x_i)\Im(\sigma_i))}{\Re^2(\sigma_i) + \Im^2(\sigma_i)}. \tag{9}$$

Then, we normalize A and the overall sample relationship is defined as follows

$$R = D_A^{-1/2} A D_A^{-1/2}, \tag{10}$$

where D_A is the diagonal matrices with the (i,i)-value to be the sum of the i-th row of A. We test our method based on the transductive inference [14]. $A, R \in \mathbb{R}^{(C\times(K+Q))\times(C\times(K+Q))}$ denote all support and query samples. We only keep the k-max values in each row of R to reduce the noise. We empirically set $k = K + Q + C$ that guarantees that the model can learn label information of K support samples and Q query samples of the same class.

3.3 Transductive Method

In the tranductive inference process, we learn query sample labels from support sample labels. Different from the transductive inference of [14], we learn the labels with both support and query samples. In this way, we can learn more accurate inter- and intra-class relationships. Let $R \in \mathbb{R}^{(C\times(K+Q))\times(C\times(K+Q))}$ denote the learned relation matrix whose (i, j)-value is the relationship between the ith sample and the jth sample. Define an initial relation matrix $\mathfrak{I}$ as

$$\mathfrak{I}_{i,j} = \begin{cases} \mathbb{I}(y_i == y_j), & \text{if } x_i, x_j \in S_S, \\ 1/C, & \text{if } x_i, x_j \in S_Q, \\ 0, & \text{otherwise}, \end{cases} \tag{11}$$

where $\mathbb{I}$ is the indicator function. Staring from the initial matrix $\mathfrak{I}$ defined in (11), we iteratively learn the query samples labels from the union set $S_S \cup S_Q$

$$Y_{t+1} = (1-\alpha)RY_t + \alpha\mathfrak{I}, \tag{12}$$

where Y_t denotes the predicted labels at t, R denotes the normalized relation matrix, and α controls the amount of the learned information. Also, it is well known that the sequence $\{Y_t\}$ has a closed-form solution as

$$Y^* = (I-\alpha R)^{-1}\mathfrak{I}, \tag{13}$$

where Y^* denotes the last predicted relationship between samples.

3.4 Label Prediction and Loss Generation

After computing the last learned relation matrix Y^*, we can directly convert the relation matrix Y^* to label scores using softmax as

$$P_{QS}(\tilde{y}_i = j|\mathcal{I}_i) = \frac{\exp(\sum_{z=1}^{K} Y^*_{i,z+K(j-1)})}{\sum_{l=1}^{C}\exp(\sum_{z=1}^{K} Y^*_{i,z+K(l-1)})}, \tag{14}$$

where $\tilde{y}_i$ denotes the final predicted label for the ith sample of the query set. And then we compute classification loss between the predictions of the query set and the ground-truth labels of the union of the support and query sets to end-to-end update all parameters. Firstly, we split the classification loss into two parts as query-support (QS) and query-query (QQ) classification losses. Experiments in Section 4 show that considering the relation among query samples can make the model learn a better relationship and have a better performance. The QS classification loss is defined as

$$\mathcal{L}_{QS} = \sum_{i=CK+1}^{C(K+Q)} \sum_{j=1}^{C} -\mathbb{I}(y_i == j) log(P_{QS}(\tilde{y}_i = j|\mathcal{I}_i)), \tag{15}$$

where y_i is the ground-truth label of $\mathcal{I}_i$. Similar to the QS classification loss, the QQ loss is defined as

$$P_{QQ}(\tilde{y}_i = j|\mathcal{I}_i) = \frac{\exp(\sum_{z=1}^{Q} Y^*_{i,NK+z+Q(j-1)})}{\sum_{l=1}^{C}\exp(\sum_{z=1}^{Q} Y^*_{i,NK+z+Q(l-1)})}, \tag{16}$$

$$\mathcal{L}_{QQ} = \sum_{i=CK+1}^{C(K+Q)} \sum_{j=1}^{C} -\mathbb{I}(y_i == j) log(P_{QQ}(\tilde{y}_i = j|\mathcal{I}_i)). \tag{17}$$

Then, the overall loss is the sum of the QS and the QQ losses:

$$\mathcal{L} = \mathcal{L}_{QS} + \mathcal{L}_{QQ}. \tag{18}$$

Note that P_{QQ} is only used in the training process to make the model learn a more better inter- and intra-class relationship, and P_{QS} is used both in the test and training process.

4 Experiments

We evaluate and compare our proposed CMM with other state-of-the-art approaches on two datasets, miniImageNet and tieredImageNet.

4.1 Datasets

MiniImageNet. The miniImageNet [11], a subset of ImageNet, has 100 classes selected randomly from ImageNet and each class has 600 images. Following the split proposed by [17], the dataset is divided into training, validation, and test sets, with 64, 16, and 20 classes respectively.

TieredImageNet. The tieredImageNet [11] dataset is a larger subset of ImageNet with 608 classes. Different from miniImageNet, it has a hierarchical structure of broader categories of

Table 1: Few-shot classification accuracies on miniImageNet and tieredImageNet. Each result is the average of 600 test episodes.

		mini 5-way		mini 10-way		tiered 5-way		tiered 10-way	
Model	Trans.	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
MAML [4]	B	48.70	63.11	31.27	46.92	51.67	70.30	34.44	53.32
MAML+Trans. [4]	Y	50.83	66.19	31.83	48.23	53.23	70.83	34.78	54.67
Prototypical Net [20]	N	46.14	65.77	32.88	49.29	48.58	69.57	37.35	57.83
Maching Net [24]	N	43.56	55.31	-	-	54.02	70.11	-	-
Relation Net [21]	B	51.38	67.07	34.86	47.94	54.48	71.31	36.32	58.05
Reptile [15]	N	47.07	62.74	31.10	44.66	48.97	66.47	33.67	48.04
Reptile+BN [15]	B	49.97	65.99	32.00	47.60	52.36	71.03	35.32	51.98
TPN [14]	Y	53.75	69.43	36.63	52.32	57.53	72.85	40.93	59.17
Our CMM (QS)	Y	**56.21**	**70.53**	**37.68**	**55.39**	57.12	72.74	**43.30**	**61.71**
Our CMM (QS+QQ)	Y	**56.26**	**70.98**	**38.82**	**55.56**	**58.12**	**73.46**	**43.46**	**61.85**

high-level nodes in ImageNet. This set of nodes is partitioned into 20, 6, and 8 disjoint sets of training, validation, and testing nodes, and the corresponding classes form the respective meta-sets. Therefore, the training classes have distinct semantical samples from the test classes, which makes it a more challenging and realistic task for few-shot learning.

4.2 Experimental Setting

Following the recent work, we use the same episodic training procedure[24] to update our model parameters. To be specific, during the training process, we randomly select C classes in $\mathcal{D}_{train}$ and K samples in each class as the support data, and then select 15 samples from the remaining data of these C classes as the query data. In all experiments, we set α to 0.01 and we take Adam[9] as the optimizer with an initial learning rate of 10^{-3} which is halved for every 25,000 episodes on both miniImageNet and tieredImageNet. All experiments are done without data augmentation.

4.3 Few-Shot Learning Results

We compare our model with several state-of-the-art approaches in various settings. As the proposed CMM belongs to the metric learning type, we mainly compare our model with other state-of-the-art metric learning models including Matching Nets[24], Prototypical Nets[20], Relation Nets[21], and Reptile[15]. Moreover, we also choose TPN[14] and use the simple transductive method named MAML+Transduction designed by [14], both of which explicitly utilize the query set. Experimental results including the combinations of 5 and 10 ways and 1 and 5 shots are shown in Tab. 1, each accuracy is the average of 600 randomly generated episodes from the test set $\mathcal{D}_{test}$ and top results are highlighted. All the methods are divided into three groups with three different inference methods; "N" means inference methods without transduction, "Y" means transductive inference methods where all query samples are simultaneously predicted, and "BN" means query batch statistics are used to share informance among test samples.

The experiments show that the proposed CMM achieves state-of-the-art results and outperforms all the others with a large margin. Especially in the scenario of 5-way 1-shot on miniImageNet, our model can achieve a high accuracy of 56.26% with a significant improvement 2.51% over the best compaired method TPN. Even in a more realistic scenario of 10 ways, the absolute improvements by CMM can also achieve 2.19% and 2.53% for 1-shot and 3.24% and 2.68% for 5-shot on miniImageNet and tieredImageNet respectively.

Another observation is that $\mathcal{L}_{QQ}$ can slightly improve the accuracy of our model. In the process of transductive inference, the loss of CMM consists of two parts, $\mathcal{L}_{QS}$ and $\mathcal{L}_{QQ}$. The first part can make our model learn relationships between support and query samples and predict the labels of query samples in the test. The second part aims to make our model learn better relationships among query samples which can improve the performance of transductive inference. Obviously, with more accuracy relationships among samples, our model can have a better performance.

Table 2: Few-shot classification accuracies on miniImageNet with different metric methods. Each result is the average of 600 test episodes. $\mathcal{RI}$ measures sample distances using the real and imaginary parts of features. $\mathcal{AP}$ measures sample distances using the amplitude part of features.

	5-way Acc		10-way Acc	
Methods	1-shot	5-shot	1-shot	5-shot
TPN-64	53.75	69.43	36.62	52.32
TPN-128	54.75	69.79	37.08	53.53
$\mathcal{RI}$	**54.99**	**70.02**	37.03	**54.43**
$\mathcal{AP}$	**56.26**	**70.98**	**38.82**	**55.56**

4.4 Ablation Experiments

To validate the effects of the proposed complex metric unit named $\mathcal{AP}$, we design a commonly used metric method named $\mathcal{RI}$ which measures the real and imaginary parts of sample features separately. Specially, our complex metric unit firstly generate a sample-wise complex-valued parameter which learn from amplitude and phase information of the feature embedding and measure amplitude distance between samples with the feature embedding and the complex-valued parameter. Different from the complex metric unit $\mathcal{AP}$, $\mathcal{RI}$ measures the Gaussian distance of the real and imaginary parts between samples, and then directly sum them up. This method is commonly used to measure sample distances in [14, 21, 24]. Also, to eliminate the influence of the number of parameters, we design a TPN-128 that has the same number of parameters as CMM and more filters. The experimental results are shown in Tab. 2, which show that the proposed complex metric unit $\mathcal{AP}$ has a better performance than commonly used metric methods. Even with the same number of parameters and fewer filters, CMM still has a higher classification accuracy than TPN.

5 Conclutions

In this work, we have introduced a complex setting for few-shot learning. Our proposed approach, namely Complex Metric Module (CMM), firstly adopts a complex-valued neural network to learn both the amplitude and phase information of samples. Specially, our approach is composed of four parts: complex feature extractor, complex metric unit, transductive inference, and label prediction and loss generation. The complex feature extractor and the complex metric unit are the key components of CMM that extracts and measures the amplitude and phase features, and consequently results in a more accuracy metric. In the transductive inference, we explicitly consider the relationship among query samples, which slightly improves the performance of our model. The proposed CMM can achieve state-of-the-art results on miniImageNet and tieredImageNet. Also, we have shown ablation experiments to verify the effectiveness of our metric method and transductive inference.

6 Acknowledgement

Baochang Zhang is also with Shenzhen Academy of Aerospace Technology, Shenzhen, China, and he is the corresponding author. The work is in part supported by Shenzhen Science and Technology Program (No.KQTD2016112515134654).

References

[1] Martin Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. *ICML*, pages 1120–1128, 2016.

[2] Ivo Danihelka, Greg Wayne, Benigno Uria, Nal Kalchbrenner, and Alex Graves. Associative long short-term memory. *ICML*, 2016.

[3] Istvan Fehervari, Avinash Ravichandran, and Srikar Appalaraju. Unbiased evaluation of deep metric learning algorithms. 2019.

[4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *ICML*, pages 1126–1135, 2017.

[5] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. *CVPR*, pages 4367–4375, 2018.

[6] Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. *CVPR*, pages 21–30, 2019.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, pages 770–778, 2016.

[8] Akira Hirose and Shotaro Yoshida. Generalization characteristics of complex-valued feedforward neural networks in relation to signal coherence. *IEEE Transactions on Neural Networks*, 23(4):541–551, 2012.

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2014.

[10] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. *arXiv: Computer Vision and Pattern Recognition*, 2017.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *NIPS*, pages 1097–1105, 2012.

[12] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[13] Fei Fei Li and Rob Fergus. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):594–611.

[14] 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. *ICLR*, 2019.

[15] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv: Learning*, 2018.

[16] Tohru Nitta. On the critical points of the complex-valued neural network. *ICNIP*, 3: 1099–1103, 2002.

[17] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. *ICLR*, 2017.

[18] David P. Reichert and Thomas Serre. Neuronal synchrony in complex-valued deep networks. *ICLR*, 2014.

[19] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2014.

[20] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. *NIPS*, pages 4077–4087, 2017.

[21] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H S Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. *CVPR*, pages 1199–1208, 2018.

[22] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *CVPR*, pages 1–9, 2015.

[23] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, Joao Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Chris Pal. Deep complex networks. *ICLR*, 2018.

[24] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *NIPS*, 2016.

[25] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. *ECCV*, pages 818–833, 2014.

[26] Baochang Zhang, Shiguang Shan, Xilin Chen, and Wen Gao. Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition. *IEEE Transactions on Image Processing*, 16(1):57–68.

[27] Ruixiang Zhang, Tong Che, Zoubin Ghahramani, Yoshua Bengio, and Yangqiu Song. Metagan: An adversarial approach to few-shot learning. *NIPS*, pages 2371–2380, 2018.

[28] Zhimian Zhang, Haipeng Wang, Feng Xu, and Yaqiu Jin. Complex-valued convolutional neural network and its application in polarimetric sar image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(12):7177–7188, 2017.