Displays 84 (2024) 102795

Contents lists available at ScienceDirect

Displays

journal homepage: www.elsevier.com/locate/displa

FSNet: A dual-domain network for few-shot image classification ☆

Xuewen Yan, Zhangjin Huang *

School of Computer Science and Technology, University of Science and Technology of China, Huangshan Road, Hefei, 230027, China
Anhui Province Key Laboratory of Software in Computing and Communication, Huangshan Road, Hefei, 230027, China
Deqing Alpha Innovation Institute, Huzhou, 313299, China

ARTICLE INFO

Dataset link: https://www.kaggle.com/datasets/arjunashok33/miniimagenet, https://www.kaggle.com/datasets/wenewone/cub2002011

Keywords:
Few-shot learning
Image classification
Frequency domain learning
Channel attention

ABSTRACT

Few-shot learning is a challenging task, that aims to learn and identify novel classes from a limited number of unseen labeled samples. Previous work has focused primarily on extracting features solely in the spatial domain of images. However, the compressed representation in the frequency domain which contains rich pattern information is a powerful tool in the field of signal processing. Combining the frequency and spatial domains to obtain richer information can effectively alleviate the overfitting problem. In this paper, we propose a dual-domain combined model called Frequency Space Net (FSNet), which preprocesses input images simultaneously in both the spatial and frequency domains, extracts spatial and frequency information through two feature extractors, and fuses them to a composite feature for image classification tasks. We start from a different view of frequency analysis, linking conventional average pooling to Discrete Cosine Transformation (DCT). We generalize the compression of the attention mechanism in the frequency domain. Consequently, we propose a novel Frequency Channel Spatial (FCS) attention mechanism. Extensive experiments demonstrate that frequency and spatial information are complementary in few-shot image classification, improving the performance of the model. Our method outperforms state-of-the-art approaches on miniImageNet and CUB.

1. Introduction

The success of deep learning cannot be achieved without the support of a large amount of data. Some deep learning models [1,2] have achieved good results in the field of image classification. However, it is impractical in many scenarios to obtain sufficient high quality labeled samples. Many domains are encumbered by privacy, ethics, and other concerns that hinder the acquisition of high-quality data. Additionally, manual labeling of vast amounts of data is one of the challenges posed by deep learning image classification due to the significant cost in labor. Unlike deep learning models, humans have the remarkable ability to learn quickly with only a few examples when encountering new tasks. To bridge this gap between deep learning models and humans, few-shot learning (FSL) has become an important and widely studied issue. After learning a large number of base classes, few-shot models [3,4] can quickly learn novel classes with only a limited number of samples, allowing the model to adapt to unknown tasks.

An active area in few-shot learning is to propose a model that can extract rich information from a limited number of samples with generalizability. Existing studies on FSL roughly fall into three categories: metric-based methods [5,6], data-augmentation methods [7,8], and meta-learning methods [9–11], respectively. Although their methodologies are different, they all perform feature extraction solely in the spatial domain. Some studies [12,13] have applied frequency learning to feature extraction. Gueguen et al. [12] have devised a variant of traditional CNN architecture that can accommodate DCT coefficients, enabling CNNs to be utilized for extracting frequency features for image classification. Previous approaches [13] have argued that CNNs only accept low-resolution RGB images (e.g., ResNet [14]: 224×224), while many datasets contain higher resolution images (e.g., ImageNet [15]: 482×415). Traditional image classification methods typically downsample high-resolution RGB images before transmitting them to GPUs to reduce computational costs and communication bandwidth requirements. However, image downsizing in the spatial domain results in a loss of information. However, feature extraction from the frequency domain offers potential solutions to this problem by allowing flexible control of the input size. We believe that the benefits of the frequency domain network go beyond this. It provides a global perspective for analyzing images and has the potential to enhance feature extraction when combined with a spatial domain network.

☆ This paper was recommended for publication by Prof. Guangtao Zhai.

* Corresponding author at: School of Computer Science and Technology, University of Science and Technology of China, Huangshan Road, Hefei, 230027, China.
E-mail address: zhuang@ustc.edu.cn (Z. Huang).

https://doi.org/10.1016/j.displa.2024.102795
Received 24 December 2023; Received in revised form 22 June 2024; Accepted 9 July 2024
Available online 14 July 2024

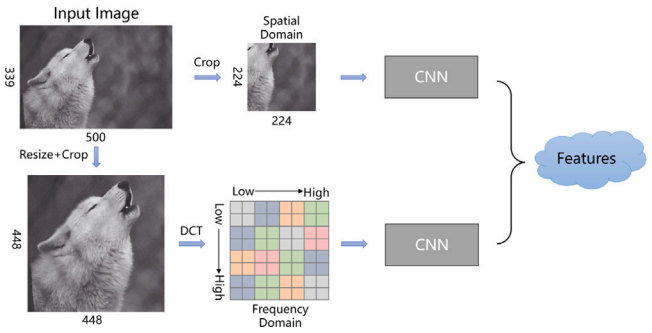

Fig. 1. Combination of spatial and frequency domains.

The dual-branch architecture is commonly used in many fields. Tao et al. [16] used a Siamese network to encode the original view and the masked view with an online branch and a target branch, respectively. Rong et al. [17] also used a dual-branch network with two identical branches to extract the original episode and the transformed episode. Dual-branch networks are superior in terms of feature fusion and comparison. Therefore, we consider using a dual-branch network to extract both spatial and frequency domain features. As shown in Fig. 1, the image is processed in the spatial and frequency domains and then fed into two CNNs to extract the fusion features.

Recently, attention mechanisms have been widely used to improve the feature extraction capability of neural networks. This allows the network to focus on important information while reducing the impact of irrelevant information. Channel attention, represented by SENet [18], learns to attach importance weights to different channels. Due to the constrained computational overhead, a core step of channel attention is to compress each channel into a scalar through global average pooling; however, representing a channel by its mean is too simple to capture the complexity of the input. Several approaches, such as CANet [19], capture spatial long-range dependencies simultaneously by embedding location information in channel attention. However, this approach still relies on 1D average pooling for channel compression, which may lead to information loss.

Based on the above analysis, we propose a novel dual-domain meta-learning model, termed as Frequency Space Network (FSNet). Unlike previous methods that only preprocess images in the spatial domain, FSNet incorporates both spatial and frequency domain preprocessing. In this way, both the spatial location information and the resolution information of the image are preserved. Second, since the sizes of the channels of image representation after frequency domain preprocessing differ from those of RGB images, we use two feature extractors to extract spatial and frequency features. Finally, we fuse these features to feed a prototypical classifier [20] for the classification task. We further propose a novel Frequency Channel Spatial (FCS) attention mechanism to enhance the feature extraction ability of the spatial backbone. We factorize the channel attention into two 1D feature encoding processes in two spatial directions. Additionally, instead of using average pooling for channel compression, we utilize Discrete Cosine Transform (DCT) to retain channel and location information to the greatest extent. The main contributions of this paper can be summarized as follows:

- In this paper, we propose a dual-domain meta-learning model Frequency Space Net (FSNet) that leverages the complementary information from the spatial and frequency domains and ultimately enhances the classification accuracy.
- We propose a novel Frequency Channel Spatial (FCS) attention mechanism that incorporates Discrete Cosine Transform into the attention mechanism to minimize the information loss on the channel.
- We have conducted comprehensive experiments on two real-world datasets, miniImageNet [15] and CUB [21], which demonstrate that our approach can significantly improve the classification accuracy.

The remainder of this paper is organized as follows. In Section 2, we provide an overview of related works on few-shot learning (Section 2.1) and frequency domain learning (Section 2.2). In Section 3, we introduce our proposed FSNet, which includes the problem definition of few-shot learning (Section 3.1), an overview of the model (Section 3.2), and the frequency branch (Section 3.3) and spatial branch (Section 3.4) of our approach. Experimental results are reported in Section 4, with introduction of the datasets in Section 4.1 and details of the experimental setup in Section 4.2. In Section 4.3, we compare our method with state-of-the-art approaches, while Section 4.4 presents the ablation analysis of our model. Additionally, we provide the visualization experiments in Section 4.5. Finally, in Section 5, we conclude the paper and outline potential directions for future research.

2. Related work

2.1. Few-shot learning

Few-shot learning addresses the challenge of recognizing and adapting to novel classes with minimal training data. In recent years, research on few-shot learning has been summarized into the following three categories:

Metric-learning. metric-learning based methods project input data into a metric space and then classify them into corresponding classes based on the similarity between feature vectors. Prototypical network [20] is a classical metric-learning method that represents each class as a prototype. The prototype for each class is simply computed by taking the mean of all support features in the class. Li et al. [22] proposed a task-adaptive subspace mapping to minimize the transfer of task-irrelevant representational information from the source domain. Trosten et al. [23] embeded representations on the hypersphere to eliminate the hubness problem.

Data augmentation. Data augmentation is an intuitive method for enhancing training data by increasing their diversity. Yang et al. [8] calibrated the distribution of novel classes by transferring statistics from base classes and sampled an adequate number of examples from the calibrated distribution. Li et al. [7] erased the discriminative regions of support images and complemented them by image repainting, forcing the network to focus on features in non-critical regions. In addition, some recent works [24,25] have expanded the training samples by utilizing unlabeled datasets in a semi-supervised manner.

Meta-learning. Meta-learning methods leverage prior knowledge to guide learning on unknown tasks. Meta-learning is performed at the task level rather than at the data level, during which generic information is progressively learned across tasks. Finn et al. [9] found a good initialization parameter for the model through a cross-task training strategy. Zhang et al. [26] proposed a prototype completion based meta-learning framework that learns to recover representative prototypes by leveraging primitive knowledge and unlabeled data. Hu et al. [27] proposed a complete three-stage meta-learning framework from pre-training to meta-training to fine-tuning.

All of these few-shot learning methods perform feature extraction in the spatial domain and only accept low resolution RGB images. Unlike these methods, our approach uses a dual-branch network in which one branch extracts features in the spatial domain and the other extracts features in the frequency domain, resulting in richer and complementary features.

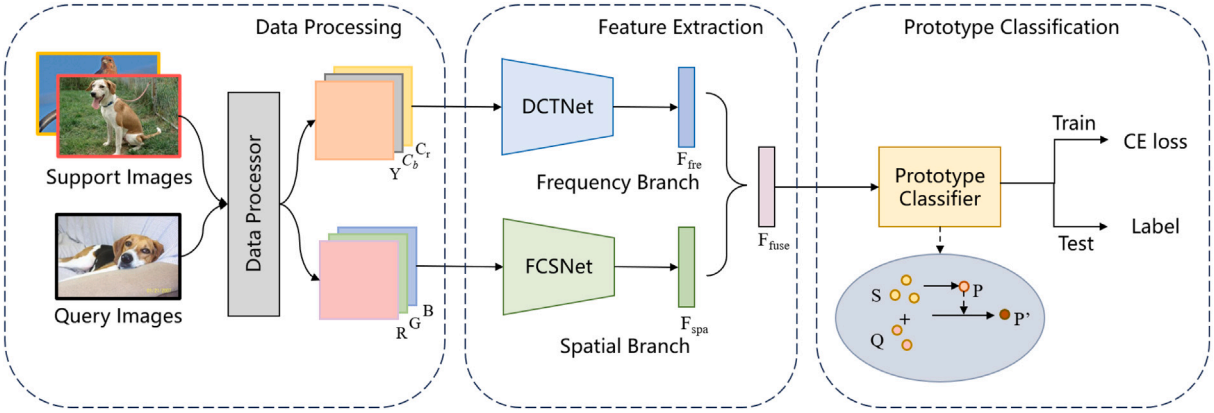

Fig. 2. An overview of our framework.

2.2. Frequency domain learning

Since compressed representations in the frequency domain contain rich patterns, several applications that introduce frequency analysis, such as image compression [28], domain generalization [29], and face forgery detection [30,31], have gradually emerged in the field of deep learning. Frequency analysis also has many applications in image classification [32,33]. Gueguen et al. [12] extracted features from the frequency domain to classify images, and Ehrlich et al. [34] proposed an algorithm to convert CNNs in the spatial domain to the frequency domain. Xu et al. [13] avoided the complex model conversion process from the spatial domain to the frequency domain by slightly modifying the existing CNNs. Moreover, CNNs are more sensitive to low-frequency information; consequently, some high-frequency information can be discarded to reduce computational overhead. Qin et al. [35] demonstrated that traditional Global Average Pooling (GAP) is a special case of DCT. Chen et al. [36] investigated the effect of the DCT filter size on image classification. [37,38] investigated shortcut learning in the frequency domain, and proposed methods to eliminate frequency shortcuts to improve the generalization of networks. Cheng et al. [39] leveraged the task-specific frequency components to adaptively mask the corresponding image information to exploit more discriminative information.

These methods have investigated how to perform deep learning in the frequency domain. Based on this, we combine frequency domain learning with traditional spatial domain learning for feature extraction, and further propose an attention mechanism that combines spatial location and frequency channel information.

3. Proposed approach

In this section, we introduce the few-shot classification problem definition in Section 3.1, and our proposed dual-domain meta-learning model in Section 3.2; moreover, we detail the frequency branch and spatial branch in Sections 3.3 and 3.4, respectively.

3.1. Problem definition

For a few-shot problem, given a labeled dataset $\mathcal{D} = (\mathbf{x}_i, y_i)$, where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in C$ are the feature vector and the class of a sample, respectively. C indicates the set of classes, which is divided into base classes C_b and novel classes C_n, where $C_b \bigcap C_n = \varnothing$ and $C_b \bigcup C_n = C$. There is a large amount of training data in the base dataset, while only a limited number of labeled data is given for each class in the novel dataset. The goal is to train a model on the base dataset that generalizes well to novel tasks. A few-shot meta-task consists of two parts, a training set $S = (\mathbf{x}_i, y_i)_{i=0}^{N\times K}$ containing a small number of labeled samples (called the support set) and a test set $Q = (\mathbf{x}_i, y_i)_{i=0}^{N\times M}$ containing unlabeled samples (called the query set), where N indicates the number of classes in S and Q, and K and M indicate the number of samples in each class in S and Q, respectively.

3.2. Overall framework

Previous approaches usually extract features only in the spatial domain; however, some work [13] has demonstrated that feature extraction in the frequency domain achieves better results. They argued that frequency domain networks have flexible inputs and can handle larger resolution images. We experimentally demonstrate that the frequency domain network is more than that and it can collaborate well with the spatial domain network to extract features.

Inspired by [36], who used two domain features, we propose a dual-domain meta-learning model FSNet which extracts features from both spatial and frequency branches and fuses them for classification. It can be understood that in the frequency branch, we globally perceive important information in a larger resolution image, and in the spatial branch, we understand the detailed information in the core part of the image. In addition, unlike in [36], which uses the trained frequency and spatial CNNs to extract, after concatenating the features, we fuse the above features and send them to the prototype classifier. Moreover, we train frequency and spatial feature extractors simultaneously to promote their fusion effects. As shown in Fig. 2, our proposed model consists of three phases: data processing, feature extraction, and prototype classification.

Data processing. In the data processing stage, we divide the data processing into two branches, processing the images into spatial representation (RGB) and frequency representation (YC_bC_r). Spatial domain data processing involves the use of conventional data augmentation methods, such as cropping, jittering, and flipping. Frequency domain data processing, on the other hand, converts RGB images to frequency representations via Discrete Cosine Transform (DCT).

Feature extraction. In this stage, the preprocessed spatial and frequency representations are fed into the spatial and frequency feature extractors, respectively. The feature maps (F_{fre} and F_{spa}) with the same size extracted by the feature extractors are adaptively weighted and fused to obtain a dual-domain feature F_{fuse}:

$$F_{fuse} = \alpha F_{spa} + (1-\alpha)F_{fre}. \tag{1}$$

where α is an adaptive weighting factor, $\alpha \in [0, 1]$, initially set to 0.5, and its value is adjusted adaptively by the network during training. Specifically, the spatial feature extractor (termed FCSNet) uses ResNet as the backbone and incorporates our FCS attention. The frequency feature extractor (termed DCTNet) uses ResNet with a slight modification of the input layers as the backbone.

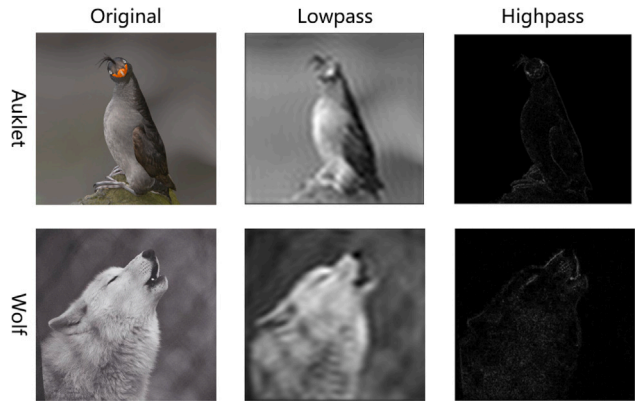

Fig. 3. The result of the original image after filtering.

Prototype classification. In this stage, the query set is added to compute an extended prototype based on the prototypical network [20]. Specifically, we first compute the initial prototype P from the mean of the features of the support set (S). Then P is used to compute the classification probability of each sample in the query set (Q). The probability of each sample $x \in Q$ being class k is estimated based on the proximity between its feature $f(x)$ and P_k. That is:

$$Prob(x,k) = \frac{e^{d(f(x),P_k)\cdot\tau}}{\sum_c e^{d(f(x),P_c)\cdot\tau}} \tag{2}$$

where $d()$ denotes the cosine similarity of two vectors and τ is a learnable scale parameter, we set it to 10 initially. Next, using this probability as the weight, the center of each class is re-estimated as the extended prototype P' by weighting the average of the features of all samples:

$$P'_k = \frac{1}{\sum_{x\in S\bigcup Q} Prob(x,k)} \sum_{x\in S\bigcup Q} Prob(x,k) f(x) \tag{3}$$

where P'_k represents the extended prototype for the kth class. For a sample x within the support set (S) of the kth class, its probability is set to $Prob(x,k) = 1$. The reason for extending the prototype is that each sample contributes the same amount of weight to the mean-based prototype. This approach does not take into account the variance between samples. Moreover, the inclusion of query set information makes the extended prototype more accurate. Finally, we use Cross Entropy (CE) loss as the loss function. The prototype classifier outputs CE loss in training and directly outputs classification results in testing.

3.3. Frequency branch

Previous work [13] has demonstrated the effectiveness of frequency learning in image classification. The frequency branch extracts features in the frequency domain. The image must be converted to the frequency domain before feeding it to the feature extractor. We convert the image into frequency domain by Discrete Cosine Transform (DCT). Then, to control the input size and reduce computational complexity, we crop the features. This is done because low-frequency channels contain the contour information of the image, while high-frequency channels contain noise and detail information.

Fig. 3 shows the result of the original image after filtering. After passing through the lowpass filter, the low-frequency information retained is the outline of the image. The high-frequency information retained after passing through the highpass filter is the edge of the image. The low-frequency features retain most of the information in the image. We crop high-frequency channels appropriately to not only reduce the input size, but also retain important information and remove noise interference. Finally, we group all the components of the same frequency into one channel and concatenate the selected channels into a final input to be fed into the frequency feature extractor. In summary, the frequency branch contains two parts: frequency preprocessing and a frequency feature extractor.

Frequency preprocessing. Like frequency feature extraction process in [13], as shown in Fig. 4, after conventional preprocessing, we transform images into the YC_bC_r color space. Since the Human Visual System (HVS) is more sensitive to luminance (Y) than to chromaticity (C_r and C_b), we use the common sampling format 4:2:0. Next, DCT is performed by dividing each channel by patches of size $f \times f$, where f is the size of the DCT filter. The DCT coefficients at the same frequency are grouped into one channel. Since it was demonstrated in [13] that the CNN is more sensitive to low-frequency channels, we choose the more influential low-frequency channels as the input to the subsequent feature extractor. Here we select 4×4 low-frequency channels from Y and 2×2 low-frequency channels from C_b and C_r proportionally, i.e. the number of channels selected $C_{fre} = 24$. Then, to ensure the consistency of the width W and height H dimensions of each channel, we upsample the frequency channels belonging to C_r and C_b to the same size as the frequency channel of Y. The final input size is $\frac{S_{img}}{f} \times \frac{S_{img}}{f} \times C_{fre}$, where S_{img} is the length of the image (e.g., 448). Therefore, the input size can be controlled by adjusting the DCT filter size, e.g., an input image of size 448×448 can be processed using a filter with f of 8. Larger images can be processed compared to spatial domain processing.

Frequency feature extractor. The image sizes of the frequency domain inputs are different from those of the spatial domain inputs; the height (H) and width (W) dimensions are smaller, but the channel (C) dimension is larger. For the spatial input, W and H are the width and height of the processed image, respectively, and the channel size is 3. For the frequency input, since the frequency features are the result of the image after DCT filtering, they are $1/f$ of the original image size in both the height and width dimensions, and f is the filter size. The channel dimension of the frequency features depends on the number of frequency channels we select, which will generally be larger than 3. Therefore, the same backbone cannot be used for both the spatial and frequency branches exactly.

We follow the design in [13] to minimally modify the backbone network. We skip the input layers of ResNet (a convolutional layer with a stride of 2 and a max-pooling layer), and adjust the size of the input channels of the next layer to the number of frequency input channels. With this simple modification, we obtain a frequency version of the feature extraction network.

3.4. Spatial branch

In the spatial branch, the data processing and feature extraction steps are similar to those of the traditional network, and ResNet is used as the backbone. On this basis, to enhance the feature extraction ability of the backbone, we introduce a novel attention mechanism to focus on the more important regions of the image.

DCT and average pooling. Coordinate Attention [19] (CA) is a kind of channel attention embedded with position information. CA enables the network to avoid losing position information while capturing long-range dependencies. However, we argue that CA still suffers from a large amount of information loss in the channel dimension. The reason for this is that CA still compresses each channel through average pooling, which makes it difficult to capture the complex information of various inputs well. If the average pooling operation is viewed as a compression problem, we can consider extending average pooling to the frequency domain. In the field of signal processing, compared to the Fourier Transform, the Discrete Cosine Transform is more suitable for the processing of discrete signals, especially in the domain of image processing. As a result, the Discrete Cosine Transform is frequently

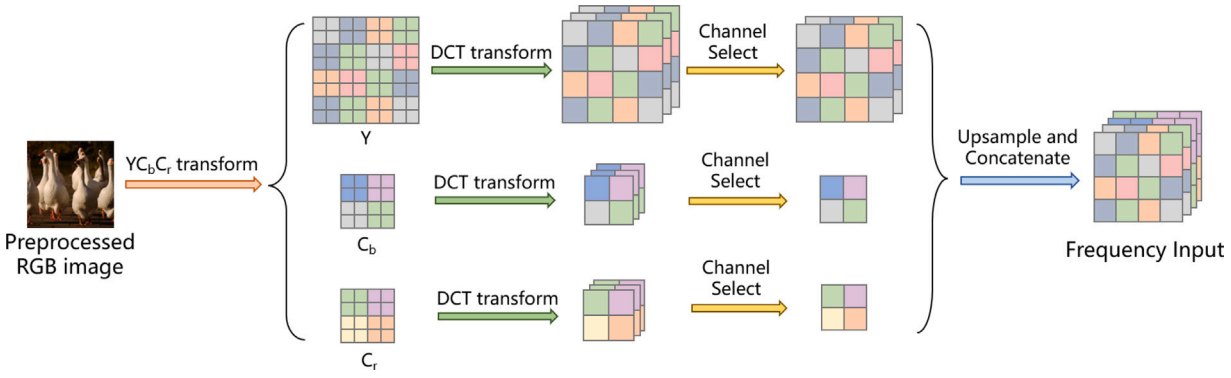

Fig. 4. Frequency preprocess.

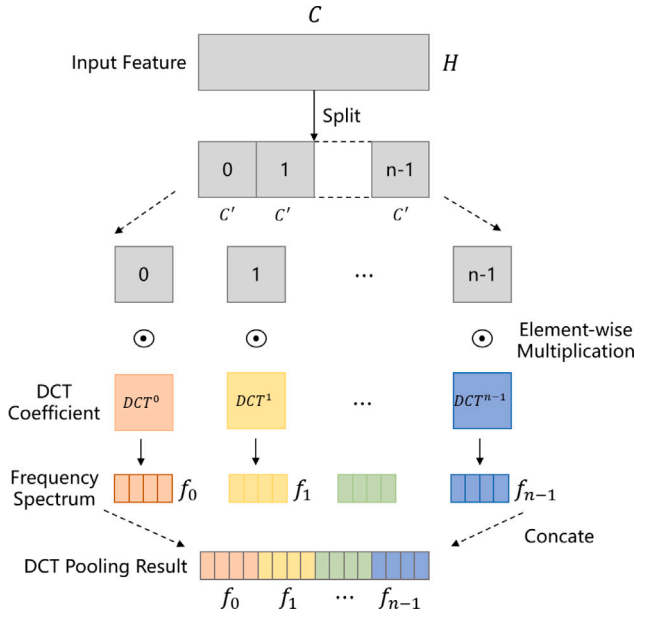

Fig. 5. DCT pooling.

employed for the compression of images. The classical one-dimensional (1D) DCT is:

$$B_l^i = \cos\frac{(i+0.5)\pi}{L} l,$$
$$f_l = \sum_{i=0}^{L-1} x_i B_l^i, \tag{4}$$
$$l \in \{0, 1, \ldots, L-1\}, i \in \{0, 1, \ldots, l-1\}$$

The inverse 1D DCT can be written as:

$$x_i = \sum_{l=0}^{L-1} f_l B_l^i, \tag{5}$$

where x_i is the original signal input, f_l is the frequency spectrum of DCT, L is the 1D length of x, and B_l^i is the DCT coefficient. The constant normalization factor is omitted here, which has no effect on the results. For average pooling along a certain direction, height dimension for example, i.e. $L = H$, is calculated as follows:

$$f(w) = \frac{1}{H}\sum_{i=0}^{H-1} x_i(w) \tag{6}$$

Assuming that l in Eq. (4) is 0:

$$\begin{aligned} f_0(w) &= \sum_{i=0}^{H-1} x_i(w)\cos\frac{(i+0.5)\pi}{L}0 \\ &= \sum_{i=0}^{H-1} x_i(w) \\ &= H f(w) \end{aligned} \tag{7}$$

where w is the coordinate in the W direction. It can be seen that the average pooling function $f(w)$ is a special case of DCT, and its result is equal to that of DCT, which takes only the lowest frequency component. Therefore, we can use DCT pooling instead of average pooling to reduce the loss of information:

$$\begin{aligned} f_{dct}(w) &= f_0(w) + f_1(w) + \cdots + f_{n-1}(w) \\ &= H f(w) + f_1(w) + \cdots + f_{n-1}(w) \end{aligned} \tag{8}$$

where $f(w)$ is the average pooling function in width dimension, $f_{dct}(w)$ is the DCT pooling function in width dimension, and n is the number of channels selected. Utilizing DCT pooling in place of average pooling, the compression result encompasses not only the information of the lowest frequency channel but also includes additional information from n−1 higher frequency channels. Furthermore, the degree of compression can be adjusted by selecting the number of channels. In conclusion, compared to the averaging pooling, DCT pooling contains more frequency information.

Frequency channel spatial attention. We propose Frequency Channel Spatial (FCS) attention, where the channel is compressed using DCT pooling instead of average pooling.

Given an input x, we feed it into two branches to calculate the weights, first, through two separate DCT pooling layers. DCT pooling is performed along the H and W directions to produce a pair of direction-aware feature maps. Here, we select the n lowest frequency components to compute for saving costs, and we set n to 8. As shown in Fig. 5, features are divided into n parts along the channel dimension, and then element-wise multiplied with DCT coefficients of n frequency components.

As shown in Fig. 6, after DCT pooling, we encode each channel along the horizontal coordinate and the vertical coordinate, respectively. The outputs of the cth channel at height h and width w can be formulated as:

$$z_c(h) = \sum_{k=0}^{n-1}\sum_{i=0}^{W-1} x_c(h,i)\cos\frac{(i+0.5)\pi}{W}k \tag{9}$$

$$z_c(w) = \sum_{k=0}^{n-1}\sum_{j=0}^{H-1} x_c(j,w)\cos\frac{(j+0.5)\pi}{H}k \tag{10}$$

Then, we concatenate the results and feed them into a 1 × 1 convolutional layer F_1 and an activating layer δ:

$$\boldsymbol{f} = \delta(F_1([\boldsymbol{z}^H, \boldsymbol{z}^W])) \tag{11}$$

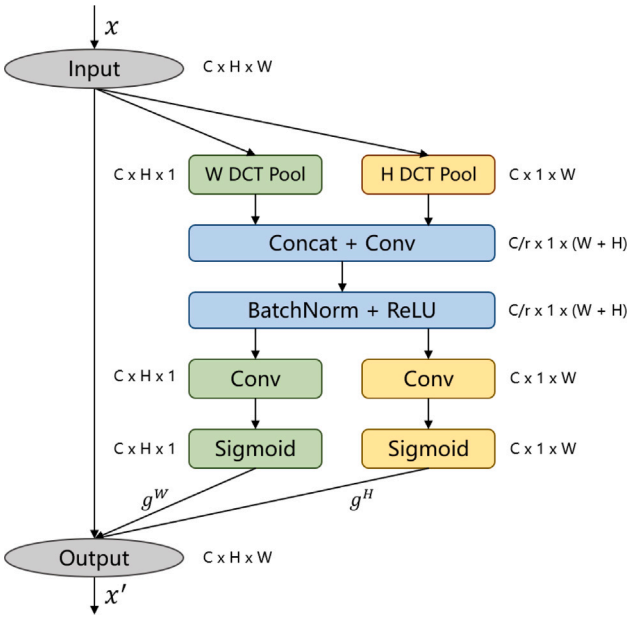

Fig. 6. The structure of frequency channel spatial attention.

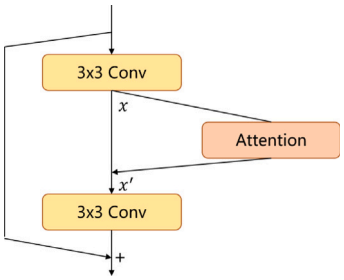

Fig. 7. The residual block structure of FCSNet after inserting FCS attention.

where $\boldsymbol{f} \in \mathbb{R}^{C/r\times(H+W)}$ is the intermediate feature map that encodes spatial information in both the horizontal and vertical direction. r is the coefficient of the reduction ratio used to control the block size. After that, we split $\boldsymbol{f}$ into two tensors $\boldsymbol{f}^H$ and $\boldsymbol{f}^W$ along different dimensions and pass them through a 1×1 convolutional layer and an activating layer again to obtain the outputs $\boldsymbol{g}^H$ and $\boldsymbol{g}^W$, respectively:

$$\boldsymbol{g}^H = \sigma(F_H(\boldsymbol{f}^H)) \tag{12}$$

$$\boldsymbol{g}^W = \sigma(F_W(\boldsymbol{f}^W)) \tag{13}$$

where F_H and F_W are the two convolutional transformations in H and W dimensions, respectively. σ is the sigmoid function. After we have calculated the two weights $\boldsymbol{g}^H$ and $\boldsymbol{g}^W$, we multiply them with the initial input x. Ultimately, the output can be written:

$$x'_c(i,j) = x_c(i,j) \times g_c^H(i) \times g_c^W(j). \tag{14}$$

where $x_c(i,j) \in \mathbb{R}^{C\times H\times W}$ is the input of the cth channel ($c \in C$), $x'_c(i,j) \in \mathbb{R}^{C\times H\times W}$ is the output of the cth channel, i, j are coordinates in the W and H directions, respectively. r is the coefficient of the reduction ratio used to control the block size, and we set it to 32.

FCSNet. Finally, our FCS attention layer is inserted into the residual block of ResNet, which we call FCSNet. The Fig. 7 shows how we plug our attention blocks into the residual block in ResNet (ResNet10 for example).

Table 1
The number of parameters of backbones.

Backbone	ResNet10	ResNet12	ResNet18	ResNet34	WRN28-10	FSNet (Ours)
Params	4.9M	8.0M	11.2M	21.3M	36.5M	9.8M

4. Experiments

4.1. Datasets

Our proposed model has been evaluated on two popular few-shot learning datasets: miniImageNet [15], CUB [21], FC100 [40], CIFAR-FS [41].

miniImageNet. MiniImageNet [5] is sampled from the ILSVRC-12 dataset [15], which includes 100 classes and each class consists of 600 images. Following [26], we split the data set into 64 classes for training, 16 classes for validation, and 20 classes for test, respectively.

CUB. CUB-200-2011 [21] is a fine-grained classification dataset that includes 200 classes and contains approximately 11,788 images. Following [26], we split the data set into 100 classes for training, 50 classes for validation, and 50 classes for test, respectively.

FC100. Fewshot-CIFAR100 (FC100) [40] is a subset of CIFAR-100 [42], which includes 100 classes and each class consists of 600 images. A common split is 60 classes for training, 20 classes for validation, and 20 classes for test, respectively.

CIFAR-FS. CIFAR-FewShot (CIFAR-FS) [41] is also a few-shot classification dataset built on CIFAR-100 [42]. It contains 64, 15 and 20 classes for training, validation and testing, respectively.

4.2. Experiment setup

ResNet is the most commonly used network for few-shot image classification. Therefore, we choose ResNet as the backbone network. Considering that the number of parameters of a dual-branch structure network is doubled compared to that of a single-branch structure, we choose the smaller ResNet10 as the backbone of the two branches. As Table 1 shows, the size of our network is still smaller than that of ResNet18 even if two ResNet10 are used at the same time. Thus, we use ResNet10 which incorporates our FCS attention, as a spatial feature extractor, termed FCSNet10, and crop the input images to 224×224. We use the frequency version of ResNet10 as frequency feature extractor, termed as DCTNet10, and crop the input images to $56 \times f$, where f is the DCT filter size. Note that FCS attention adds almost no additional parameters, so FSNet is exactly twice the size of ResNet10.

For the hyperparameters involved in the experiments, if not specified, we set $f = 8$, $C_{fre} = 24$, $n = 8$, and $r = 32$ for all the experiments, where f is the size of the DCT filter, C_{fre} is the number of channels selected in frequency preprocessing, n is the number of channels selected in FCS attention, and r is the coefficient of the reduction ratio in FCS attention.

We trained our model on base classes via Adam. For the model trained with vanilla prototype classifier, 800 epochs were used; for the model trained with our improved prototype classifier, 1200 epochs were used. Each epoch contains 100 meta-tasks. We conduct few-shot classification on 600 randomly sampled meta-tasks from the test set and report the mean accuracy together with the 95% confidence interval. In each 5-way 1-shot/5-shot task, we randomly sample 15 query images per class for evaluation.

Table 2
Comparison with the state-of-the-art on miniImageNet and CUB. The best accuracy (%) is highlighted.

Method	Backbone	miniImageNet		CUB-200-2011	
		5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
ProtoNet [20]	ResNet10	51.98 ± 0.84	72.64 ± 0.64	73.22 ± 0.92	85.01 ± 0.52
RelationNet [43]	ResNet10	52.19 ± 0.83	70.20 ± 0.66	70.47 ± 0.99	83.70 ± 0.55
MatchingNet [5]	ResNet10	54.49 ± 0.81	68.82 ± 0.65	71.29 ± 0.87	83.47 ± 0.58
MAML [9]	ResNet10	54.69 ± 0.89	66.62 ± 0.83	70.32 ± 0.99	80.93 ± 0.71
ATLNet [44]	Conv4-64	54.30 ± 0.76	73.22 ± 0.63	60.91 ± 0.91	77.05 ± 0.67
UDS [45]	ResNet34	55.04 ± 0.81	75.11 ± 0.63	72.23 ± 0.88	86.57 ± 0.50
MN(s+f) [36]	ResNet10	57.32 ± 0.21	76.27 ± 0.16	–	–
LSTA [46]	ResNet12	59.44 ± 0.67	74.93 ± 0.51	–	–
MDMNet [47]	ResNet12	59.88 ± 0.42	76.60 ± 0.24	–	–
FBM+MTL [48]	ResNet12	61.41 ± 1.87	76.11 ± 0.92	–	–
AFHN [49]	ResNet18	62.38 ± 0.72	78.16 ± 0.56	70.53 ± 1.01	83.95 ± 0.63
PN+VLCL [50]	WRN-28-10	61.75 ± 0.43	76.32 ± 0.49	71.21 ± 0.43	85.08 ± 0.36
Meta-Baseline [51]	ResNet12	63.17 ± 0.23	79.26 ± 0.17	–	–
P-Transfer [52]	ResNet12	64.21 ± 0.77	80.38 ± 0.59	73.88 ± 0.87	87.81 ± 0.48
DAM [53]	ResNet12	60.39 ± 0.21	73.84 ± 0.16	–	–
CxGrad+TSPL [54]	ResNet10	58.55 ± 0.46	72.28 ± 0.41	–	–
FSNet (Ours)	ResNet10	**64.83 ± 0.89**	**80.68 ± 0.60**	**81.80 ± 0.87**	**90.87 ± 0.45**

Table 3
Comparison with the state-of-the-art on FC100. The best accuracy (%) is highlighted.

Method	Backbone	FC100	
		5-way 1-shot	5-way 5-shot
ProtoNet [20]	ResNet10	37.50 ± 0.62	52.52 ± 0.60
MAML [9]	ResNet10	38.07 ± 1.71	50.38 ± 1.22
TADAM [40]	ResNet12	40.10 ± 0.40	56.10 ± 0.40
SimpleShot [55]	ResNet10	40.13 ± 0.18	53.63 ± 0.18
MetaOptNet [56]	ResNet12	41.10 ± 0.60	55.50 ± 0.60
META-RKHS [57]	Conv4-64	41.20 ± 2.21	51.52 ± 0.93
DC [58]	ResNet12	42.04 ± 0.17	57.63 ± 0.23
FSNet (Ours)	ResNet10	**42.74 ± 0.82**	**58.86 ± 0.78**

Table 4
Comparison with the state-of-the-art on CIFAR-FS. The best accuracy (%) is highlighted.

Method	Backbone	CIFAR-FS	
		5-way 1-shot	5-way 5-shot
MatchingNet [5]	Conv4-64	50.53 ± 0.87	60.30 ± 0.82
MAML [9]	Conv4-64	49.28 ± 0.90	58.30 ± 0.80
Dual TriNet [59]	ResNet18	63.41 ± 0.64	78.43 ± 0.64
DSN [60]	ResNet12	72.30 ± 0.80	85.49 ± 0.68
AFHN [49]	ResNet18	68.32 ± 0.93	81.45 ± 0.87
FSNet (Ours)	ResNet10	**74.05 ± 1.00**	**87.02 ± 0.58**

4.3. Comparison to state-of-the-art

In this section, we compare our proposed network FSNet with the state-of-the-art (SOTA) methods on several benchmarks. Table 2 shows the results of our method and the other SOTA methods on miniImageNet and CUB. And Tables 3 and 4 show the results on FC100 and CIFAR-FS. For miniImageNet, our approach increases the best accuracy by 0.62% and 0.30% for 5-way 1-shot and 5-way 5-shot, respectively. For the CUB dataset, the accuracy is increased by 7.92% and 3.06% respectively for the two tasks. Our method outperforms our baseline ProtoNet [20] by 12.85% and 8.04% on miniImageNet and by 8.58% and 5.86% on CUB for the two tasks, respectively. Out FSNet also shows excellent results on FC100 and CIFAR-FS datasets. All the results show that our method is more effective.

Upon comparing the results of the miniImageNet and CUB datasets, it can be found that the enhancement effect of FSNet is more obvious on CUB, a fine-grained dataset, than on miniImageNet. Although some methods [51,52] have similar results to our method on miniImageNet, our method works much better than these methods on CUB. This finding suggested that our method can better recognize subtle differences between images and exclude inter-class noise. There may be two reasons for this: one is that the prototype classifier itself has good fine-grained classification ability, which can be seen from the performance of ProtoNet [20] on CUB. Second, FSNet enhances its fine-grained classification ability through the complementarity of the spatial and frequency domains. The frequency branch discards high-frequency information, which usually contains background noise. As a result, the intra-class variability is decreased. In contrast, the spatial branch captures local subtle differences, better than low-frequency features in the frequency domain, which represent the profile of the image. Hence, the inter-class variability increases. Section 4.5 visualizes this capability of FSNet.

Overall, our method outperforms the methods in the table that use a single-branch structure, although there are several methods that use larger backbone networks, such as ResNet34 and WRN28-10. This proves the superiority of our method. On the other hand, compared with the results of MN(s+f) [36], our method is superior. Compared to simply concatenating the frequency and spatial features only during testing, fusing these features weightedly and adjusting the feature extraction network at training are obviously more reasonable; hence, better results are achieved. Another interesting observation is that, the performance improvement in the 1-shot tasks is more obvious than that in the 5-shot tasks. This is reasonable because the problem of inaccurate extraction of features on 1-shot tasks is more remarkable than that on 5-shot tasks.

Table 5
Component ablation on miniImageNet and CUB. EP: extended prototype. FCS: FCS attention. Fre: frequency branch. The best accuracy (%) is highlighted.

	EP	FCS	Fre	miniImageNet		CUB-200-2011	
				5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
(A)				54.93 ± 0.85	72.79 ± 0.66	70.16 ± 0.91	83.77 ± 0.57
(B)	√			56.97 ± 0.95	74.92 ± 0.64	73.35 ± 0.97	85.12 ± 0.57
(C)	√	√		58.93 ± 0.94	75.49 ± 0.66	74.64 ± 1.03	86.11 ± 0.57
(D)	√		√	63.25 ± 0.95	79.03 ± 0.62	80.47 ± 0.92	89.71 ± 0.50
(E)	√	√	√	**64.83 ± 0.89**	**80.68 ± 0.60**	**81.80 ± 0.87**	**90.87 ± 0.45**

4.4. Ablation study

Component ablation. To demonstrate the effectiveness of our proposed methods, we construct an ablation study on miniImageNet and CUB, as shown in Table 5. Specifically, (A) we remove all the components and classify the samples using only the spatial domain network ProtoNet [20]; (B) we add the extended prototype strategy to (A); (C) we replace the backbone with FCSNet in (B); (D) we add the frequency branch, based on (B); (E) we classify the samples using the complete method that we proposed.

Table 6
Results on miniImageNet with different filter sizes. Spa: Spatial branch. Fre: Frequency branch.

Method	Image size (filter size)	5-way 1-shot	5-way 5-shot	Channels	Size per channel
Spa	224	54.93 ± 0.85	72.79 ± 0.66	3	224 × 224
Fre	112(2)	56.25 ± 0.81	73.13 ± 0.62	12(*all*)	56 × 56
Spa+Fre	224 + 112	58.60 ± 0.84	76.38 ± 0.61	–	–
Fre	224(4)	56.73 ± 0.87	74.22 ± 0.63	24	56 × 56
Spa+Fre	224 + 224	58.73 ± 0.84	76.81 ± 0.58	–	–
Fre	448(8)	57.71 ± 0.84	74.59 ± 0.67	24	56 × 56
Spa+Fre	224 + 448	59.16 ± 0.87	77.03 ± 0.61	–	–
Fre	896(16)	58.10 ± 0.87	75.20 ± 0.62	24	56 × 56
Spa+Fre	224 + 896	59.98 ± 0.81	77.46 ± 0.63	–	–

The experiment results show that each of our proposed components individually improves the model's accuracy, demonstrating the effectiveness of the approach. We can see that ProtoNet in this study works better than the one given in Table 2, this is because we change the distance metric from the Euclidean distance to the cosine distance. Besides, extending the prototype allows the model to utilize information from the query set, which enriches the number of available samples, and therefore enhances the overall effect. It is evident that the enhancement of FCS attention is more prominent in the 5-way 1-shot task than in the 5-way 5-shot task because feature accuracy becomes more crucial when samples are sparse.

Among all the improvements, the addition of the frequency branch has the most obvious positive impact, which shows that learning features from the frequency domain compensates for limitations in spatial domain feature extraction from a different perspective. For example, the noise in the background (high-frequency part) is removed in the frequency branch, thus more attention is given to the target. The frequency branch also perceives a larger area.

Influence of the frequency domain. To further verify that the frequency branch is able to extract features complementary to the spatial branch, we analyze the impacts of frequency domain feature extraction on classification performance with different filter sizes in Table 6.

We test the performances of single spatial branch (without FCS attention), single frequency branch, and dual-branch networks for 5-way 1-shot and 5-way 5-shot tasks on miniImageNet. In all the experiments we use the mean-based prototype instead of the extended prototype. We fix the size of each frequency channel after DCT transformation to 56 × 56, and since the input image size $S_{img} = C_{size} \times f$, where C_{size} is the frequency channel size, we can control the size of the input image S_{img} by changing the filter size f, e.g., $S_{img} = 56 \times 8 = 448$ when $f = 8$. We fixedly selected the 24 lowest frequency channels for feature extraction; for the case of fewer than 24 channels, e.g., $f = 2$, we selected all the channels.

From the table, we can see that the frequency branch outperforms the spatial branch regardless of the DCT filter size. Even when the input image is much smaller than the spatial branch, e.g., when the DCT filter is only 2 × 2 in size, the frequency branch, with an input image half the size of the spatial branch, still extracts richer features at a limited resolution. This finding contradicts the idea presented in the previous paper [13], which suggested that the improvement in classification performance of the frequency domain network was solely due to increased input resolution. Due to the bidirectional integrity of DCT, the spatial-to-frequency domain conversion does not add additional information but rather merely changes the perspective from which the information is viewed. Compared to the spatial representation, each frequency channel contains global information, which is suitable for learning via a CNN.

It is evident that the accuracy of the frequency branch increases as the resolution of the input image increases. This indicates that the ability of the frequency domain network to flexibly change the input size has a positive impact on feature extraction. Furthermore, we find that the combination of spatial and frequency branches always achieves performance beyond that of a single branch regardless of the input size, which again supports the idea that networks operating in the spatial and frequency domains complement each other.

Considering that the average sizes of common datasets are closer to 448 × 448, we choose $f = 8$ in practice to balance the computational cost and accuracy.

Table 7
Results on miniImageNet with different pooling methods on FCSNet. The best accuracy (%) is highlighted.

Method	5-way 1-shot	5-way 5-shot
AvgPool	55.33 ± 0.84	72.40 ± 0.63
MaxPool	55.52 ± 0.84	72.26 ± 0.62
DCTPool	**56.33 ± 0.85**	**73.66 ± 0.67**

Table 8
Results on miniImageNet with different channels and compression ratios on FCSNet. n: number of channels. r: compression ratio. The best accuracy (%) is highlighted.

Backbone	n	r	5-way 1-shot	5-way 5-shot
ResNet	–	–	54.93 ± 0.85	72.79 ± 0.66
CANet	–	32	55.33 ± 0.84	72.40 ± 0.63
FCSNet	1	32	55.02 ± 0.82	73.21 ± 0.63
FCSNet	2	32	56.29 ± 0.85	73.53 ± 0.65
FCSNet	4	32	**57.31 ± 0.83**	72.78 ± 0.65
FCSNet	8	32	56.33 ± 0.85	**73.66 ± 0.67**
FCSNet	16	32	54.93 ± 0.83	73.56 ± 0.67
FCSNet	8	4	55.40 ± 0.88	72.22 ± 0.68
FCSNet	8	8	56.09 ± 0.86	73.02 ± 0.66
FCSNet	8	16	**57.65 ± 0.84**	73.33 ± 0.64
FCSNet	8	32	56.33 ± 0.85	**73.66 ± 0.67**
FCSNet	8	64	56.65 ± 0.80	73.21 ± 0.65

The influence of DCT pooling. To demonstrate the effect of our changes to the pooling operation in FCS attention, we compare DCT pooling with two other different pooling operations on the miniImagenet dataset. In our experiments, we do not use extended prototype and the results are shown in the Table 7. It can be seen that DCT pooling outperforms average pooling and maximum pooling on both the 5-way 1-shot and 5-way 5-shot tasks, which suggests that DCT pooling enriches the valid information contained in the compressed feature maps by retaining more frequency channels.

The influence of channel number and reduction ratio on FCSNet. We examine the effect of the number of channels n and the compression ratio r on FCSNet separately on miniImagenet. We do not use the extended prototype in our experiments, and we add ResNet10 and CANet10 [19] (using ResNet10 with CA attention) for reference. The results are shown in Table 8.

According to the table, when only the lowest frequency channel is selected ($n = 1$), it is equivalent to CANet10 which uses average pooling. The result is also similar to that of CANet, which aligns with our theory. Observing that using more frequencies does not necessarily lead to better performance. When $n = 16$, too many frequencies introduce unwanted noise, resulting in a negative effect on performance.

(a) ProtoNet (b) FSNet

Fig. 8. t-SNE feature visualization results.

The selection of the compression ratio also follows a similar trend. There is an increasing and then decreasing pattern, which indicates that retaining too much or too little information is not favorable for focusing on important information. Based on these experiments, we choose $n = 8$ and $r = 32$ in practice.

4.5. Feature visualization

To visually understand our network's ability to extract features, we use t-SNE [61] to visualize the distribution of the novel class samples of miniImageNet in the feature space as shown in Fig. 8. We randomly selected 5 novel classes and 100 samples for each class. Figs. 8(a) and 8(b) are the features we extracted using the vanilla ProtoNet [20] and our FSNet, respectively. It can be observed that both classes 1, 4 and classes 0, 2 in Fig. 8(a) have an significant overlap, while this phenomenon is absent in Fig. 8(b). This further validates that our method is better for class separation and effectively improves the clustering ability of the model.

5. Conclusion

In this paper, we introduce a dual-domain model that combines spatial and frequency domain networks. It is divided into two branches, the spatial branch and the frequency branch. And it improves the prototype by combining the information from the query set to obtain a more accurate class center. Moreover, it preserves important information by extracting features from both the spatial and frequency domains in a complementary manner. In addition, we propose a novel Frequency Channel Spatial attention mechanism, which is applied to the spatial branch backbone to improve the network's ability to extract critical features by reducing the information loss during the average pooling process. Experiments show that our method achieves superior performance on two common datasets. Moreover, we demonstrate through extensive experiments that the spatial and frequency branches are complementary. In the future, we are interested in exploring additional dual-domain combination methods.

CRediT authorship contribution statement

Xuewen Yan: Conceptualization, Methodology, Software, Writing – original draft. **Zhangjin Huang:** Writing – review & editing, Supervision.

Declaration of competing interest

All authors have no conflict of interest. We confirm that this work is original and has not been published elsewhere, nor is it currently under consideration for publication elsewhere.

Data availability

We used publicly available datasets for our experiments: miniImageNet https://www.kaggle.com/datasets/arjunashok33/miniimagenet and CUB-200-2011 https://www.kaggle.com/datasets/wenewone/cub2002011.

Declaration of Generative AI and AI-assisted technologies in the writing process

Authors declares that there is no AI tool has been used in writing or formatting this manuscript.

Acknowledgments

This work was supported in part by the Anhui Provincial Major Science and Technology Project, China (No. 202203a05020016), the National Key R&D Program of China (Nos. 2022YFB3303400 and 2021YFF0500900), and the National Natural Science Foundation of China (Nos. 71991464 and 61877056).

References

[1] V. Ashwath, O. Sikha, R. Benitez, TS-CNN: a three-tier self-interpretable CNN for multi-region medical image classification, IEEE Access 11 (2023) 78402–78418.
[2] A. Zhao, C. Wang, X. Li, A global+ multiscale hybrid network for hyperspectral image classification, Remote Sens. Lett. 14 (9) (2023) 1002–1010.
[3] J. Quan, B. Ge, L. Chen, Cross attention redistribution with contrastive learning for few shot object detection, Displays 72 (2022) 102162.
[4] K. Liu, S. Lyu, P. Shivakumara, Y. Lu, Few-shot object segmentation with a new feature aggregation module, Displays 78 (2023) 102459.
[5] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al., Matching networks for one shot learning, in: Advances in Neural Information Processing Systems, Vol. 29 (2016) 3630–3638.
[6] J. Xie, F. Long, J. Lv, Q. Wang, P. Li, Joint distribution matters: Deep brownian distance covariance for few-shot classification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7962–7971.
[7] J. Li, Z. Wang, X. Hu, Learning intact features by erasing-inpainting for few-shot classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 9, 2021, pp. 8401–8409.
[8] S. Yang, L. Liu, M. Xu, Free lunch for few-shot learning: Distribution calibration, in: 9th International Conference on Learning Representations, 2021.
[9] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: Proceedings of the 34th International Conference on Machine Learning, Vol. 70, 2017, pp. 1126–1135.
[10] N. Guo, K. Di, H. Liu, Y. Wang, J. Qiao, A metric-based meta-learning approach combined attention mechanism and ensemble learning for few-shot learning, Displays 70 (2021) 102065.
[11] B. Zhang, X. Li, S. Feng, Y. Ye, R. Ye, MetaNODE: Prototype optimization as a neural ODE for few-shot learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2022, pp. 9014–9021.
[12] L. Gueguen, A. Sergeev, B. Kadlec, R. Liu, J. Yosinski, Faster neural networks straight from jpeg, Adv. Neural Inf. Process. Syst. 31 (2018) 3937–3948.
[13] K. Xu, M. Qin, F. Sun, Y. Wang, Y.-K. Chen, F. Ren, Learning in the frequency domain, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1737–1746.
[14] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[15] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (2015) 211–252.
[16] C. Tao, X. Zhu, W. Su, G. Huang, B. Li, J. Zhou, Y. Qiao, X. Wang, J. Dai, Siamese image modeling for self-supervised vision representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 2132–2141.
[17] Y. Rong, X. Lu, Z. Sun, Y. Chen, S. Xiong, ESPT: A self-supervised episodic spatial pretext task for improving few-shot learning, 2023, arXiv:2304.13287.
[18] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.
[19] Q. Hou, D. Zhou, J. Feng, Coordinate attention for efficient mobile network design, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 13713–13722.
[20] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, Adv. Neural Inf. Process. Syst. 30 (2017) 4077–4087.

[21] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, The caltech-ucsd birds-200–2011 dataset, 2011.
[22] P. Li, S. Gong, C. Wang, Y. Fu, Ranking distance calibration for cross-domain few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9099–9108.
[23] D.J. Trosten, R. Chakraborty, S. Løkse, K.K. Wickstrøm, R. Jenssen, M.C. Kampffmeyer, Hubs and hyperspheres: Reducing hubness and improving transductive few-shot learning with hyperspherical embeddings, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7527–7536.
[24] Y. Wang, H. Chen, Q. Heng, W. Hou, Y. Fan, Z. Wu, J. Wang, M. Savvides, T. Shinozaki, B. Raj, B. Schiele, X. Xie, FreeMatch: Self-adaptive thresholding for semi-supervised learning, in: International Conference on Learning Representations, ICLR, 2023.
[25] Y. Jian, L. Torresani, Label hallucination for few-shot classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 7005–7014.
[26] B. Zhang, X. Li, Y. Ye, Z. Huang, L. Zhang, Prototype completion with primitive knowledge for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3754–3762.
[27] S.X. Hu, D. Li, J. Stühmer, M. Kim, T.M. Hospedales, Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9068–9077.
[28] Y. Yu, Y. Wang, W. Yang, S. Lu, Y.-P. Tan, A.C. Kot, Backdoor attacks against deep image compression via adaptive frequency trigger, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 12250–12259.
[29] J. Huang, D. Guan, A. Xiao, S. Lu, Fsdr: Frequency space domain randomization for domain generalization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 6891–6902.
[30] H. Liu, X. Li, W. Zhou, Y. Chen, Y. He, H. Xue, W. Zhang, N. Yu, Spatial-phase shallow learning: rethinking face forgery detection in frequency domain, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 772–781.
[31] Y. Luo, Y. Zhang, J. Yan, W. Liu, Generalizing face forgery detection with high-frequency features, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16317–16326.
[32] H. Wang, X. Wu, Z. Huang, E.P. Xing, High-frequency component helps explain the generalization of convolutional neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 8684–8694.
[33] Y. Rao, W. Zhao, Z. Zhu, J. Lu, J. Zhou, Global filter networks for image classification, Adv. Neural Inf. Process. Syst. 34 (2021) 980–993.
[34] M. Ehrlich, L.S. Davis, Deep residual learning in the jpeg transform domain, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3484–3493.
[35] Z. Qin, P. Zhang, F. Wu, X. Li, Fcanet: Frequency channel attention networks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 783–792.
[36] X. Chen, G. Wang, Few-shot learning by integrating spatial and frequency representation, in: 2021 18th Conference on Robots and Vision, CRV, 2021, pp. 49–56.
[37] S. Wang, R. Veldhuis, C. Brune, N. Strisciuglio, Frequency shortcut learning in neural networks, in: NeurIPS 2022 Workshop on Distribution Shifts: Connecting Methods and Applications, 2022.
[38] S. Wang, R. Veldhuis, C. Brune, N. Strisciuglio, What do neural networks learn in image classification? A frequency shortcut perspective, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 1433–1442.
[39] H. Cheng, S. Yang, J.T. Zhou, L. Guo, B. Wen, Frequency guidance matters in few-shot learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, 2023, pp. 11814–11824.
[40] B. Oreshkin, P. Rodríguez López, A. Lacoste, Tadam: Task dependent adaptive metric for improved few-shot learning, Adv. Neural Inf. Process. Syst. 31 (2018) 721–731.
[41] L. Bertinetto, J.F. Henriques, P.H. Torr, A. Vedaldi, Meta-learning with differentiable closed-form solvers, 2018, arXiv preprint arXiv:1805.08136.
[42] A. Krizhevsky, G. Hinton, et al., Learning Multiple Layers of Features from Tiny Images, Toronto, ON, Canada, 2009.
[43] F. Sung, Y. Yang, L. Zhang, T. Xiang, P.H. Torr, T.M. Hospedales, Learning to compare: Relation network for few-shot learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1199–1208.
[44] C. Dong, W. Li, J. Huo, Z. Gu, Y. Gao, Learning task-aware local representations for few-shot learning, in: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, 2021, pp. 716–722.
[45] Z. Hu, Z. Li, X. Wang, S. Zheng, Unsupervised descriptor selection based meta-learning networks for few-shot classification, Pattern Recognit. 122 (C) (2022) 108304.
[46] F. Gao, X. Luo, Z. Yang, Q. Zhang, Label smoothing and task-adaptive loss function based on prototype network for few-shot learning, Neural Netw. 156 (2022) 39–48.
[47] F. Gao, L. Cai, Z. Yang, S. Song, C. Wu, Multi-distance metric network for few-shot learning, Int. J. Mach. Learn. Cybern. 13 (9) (2022) 2495–2506.
[48] P. Yang, S. Ren, Y. Zhao, P. Li, Calibrating CNNs for few-shot meta learning, in: 2022 IEEE/CVF Winter Conference on Applications of Computer Vision, WACV, 2022, pp. 408–417.
[49] K. Li, Y. Zhang, K. Li, Y. Fu, Adversarial feature hallucination networks for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13470–13479.
[50] X. Luo, Y. Chen, L. Wen, L. Pan, Z. Xu, Boosting few-shot classification with view-learnable contrastive learning, in: 2021 IEEE International Conference on Multimedia and Expo, ICME, 2021, pp. 1–6.
[51] Y. Chen, Z. Liu, H. Xu, T. Darrell, X. Wang, Meta-baseline: Exploring simple meta-learning for few-shot learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9062–9071.
[52] Z. Shen, Z. Liu, J. Qin, M. Savvides, K.-T. Cheng, Partial is better than all: revisiting fine-tuning strategy for few-shot learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 9594–9602.
[53] F. Zhou, L. Zhang, W. Wei, Meta-generating deep attentive metric for few-shot classification, IEEE Trans. Circuits Syst. Video Technol. 32 (10) (2022) 6863–6873.
[54] S. Lee, S. Lee, B.C. Song, Efficient meta-learning through task-specific pseudo labelling, Electronics 12 (13) (2023) 2757.
[55] Y. Wang, W.-L. Chao, K.Q. Weinberger, L. Van Der Maaten, Simpleshot: Revisiting nearest-neighbor classification for few-shot learning, 2019, arXiv preprint arXiv:1911.04623.
[56] K. Lee, S. Maji, A. Ravichandran, S. Soatto, Meta-learning with differentiable convex optimization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 10657–10665.
[57] Y. Zhou, Z. Wang, J. Xian, C. Chen, J. Xu, Meta-learning with neural tangent kernels, in: Proceedings of the 9th International Conference on Learning Representations, ICLR, 2021.
[58] Y. Lifchitz, Y. Avrithis, S. Picard, A. Bursuc, Dense classification and implanting for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9258–9267.
[59] Z. Chen, Y. Fu, Y. Zhang, Y.-G. Jiang, X. Xue, L. Sigal, Multi-level semantic feature augmentation for one-shot learning, IEEE Trans. Image Process. 28 (9) (2019) 4594–4605.
[60] C. Simon, P. Koniusz, R. Nock, M. Harandi, Adaptive subspaces for few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 4136–4145.
[61] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008) 2579–2605.

Xuewen Yan received her Bachelor's degree in Computer Science from University of Science and Technology of China (USTC), Hefei, China, in 2021. She is currently pursuing her Master degree in Computer Science at the University of Science and Technology of China, Hefei, China. Her research interests include deep learning, image classification, few-shot learning, and frequency analysis.

Zhangjin Huang received his B.S. and Ph.D. degrees in computational mathematics from University of Science and Technology of China (USTC), Hefei, China, in 1999 and 2005, respectively. He is currently an associate professor with the School of Computer Science and Technology, and the School of Data Science, USTC, Hefei, China. His current research interests include computer graphics, computer vision, and geometric deep learning.