FP_AINET: FUSION PROTOTYPE WITH ADAPTIVE INDUCTION NETWORK FOR FEW-SHOT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

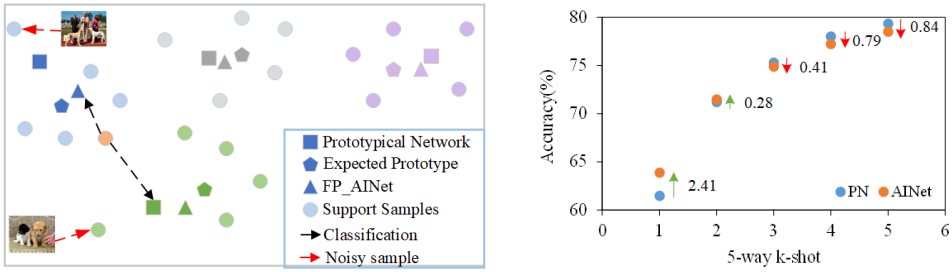
A prototypical network treats all samples equally and does not consider the noisy samples, which leads to a biased class representation. In this paper, we propose a novel fusion prototype with an adaptive induction network (FP_AINet) for few-shot learning that can learn representative prototypes from a few support samples. Specifically, to address the problem of noisy samples, an adaptive induction network is developed, which can learn different class representations for queries and assign adaptive scores for support samples according to their relative significance. Moreover, FP_AINet can generate a more accurate prototype than comparison methods by considering the query-related samples. With an increasing number of samples, the prototypical network is more expressive since the adaptive induction network ignores the relative local features. As a result, a Gaussian fusion algorithm is designed to learn more representative prototypes. Extensive experiments are conducted on three datasets: *mini*ImageNet, *tiered*ImageNet, and CIFAR_FS. The experimental results compared with the state-of-the-art few-shot learning methods demonstrate the superiority of FP_AINet.

1 INTRODUCTION

Few-shot learning aims to learn classifiers for novel classes with limited data. Prototypical network (PN) (Snell et al. (2017)) averages the support features as the prototype. While most of the previous research has achieved promising results, those methods generally assume that the samples used for training were carefully selected to represent their class. The expected prototype should have the smallest distance from all other samples in its class (Liu et al. (2020)), and each sample significantly contributes to the final performance when training from a few labeled samples. Unfortunately, the existing dataset frequently contains mislabeled samples because of weakly automated supervised annotation, ambiguity, or human error (Liang et al. (2022)). In addition, since some images have multiple objects and unrelated background information, the accuracy can be affected by a single noisy example. As illustrated in Figure 1 (a), the PN is easily affected by noisy samples. Meta-learning approaches have become the dominant paradigm for few-shot learning (Chen et al. (2020); Tian et al. (2020); Yao et al. (2021)).

Meta-learning approaches can be roughly summarized into two categories: optimization-based methods (Antoniou et al. (2019); Kao et al. (2022)) and metric-based methods (Vinyals et al. (2016); Sung et al. (2018)). Optimization-based methods readily learn the model's parameters to adapt to each task using gradient descent. However, these methods need to be fine-tuned for the target tasks. Metric-based methods are more efficient and applicable than optimization-based methods. Metric-based methods learn a good metric to calculate the similarity between query and the support samples using a pre-defined distance function, such as cosine similarity (Vinyals et al. (2016)), euclidean distance (Snell et al. (2017); Koch et al. (2015)), earth mover's distance (Zhang et al. (2020)), or a distance parameterized by a neural network (Sung et al. (2018); Zhang et al. (2018)), which has achieved remarkable success due to its fewer parameters.

To obtain more representative prototypes, many methods correct the prototype by using similar samples (Yang et al. (2021); Liu et al. (2020)) or additional knowledge (Zhang et al. (2021)), but since it is easy to introduce sample noise or class differences, a novel method of fusion prototype with an adaptive induction network (FP_AINet) is proposed to solve the issue. The induction network (Geng et al. (2019)) designs a non-linear mapping from sample vector to class vector to diminish the



(a) Prototype with noisy samples.

(b) Test accuracy on *mini*ImageNet.

Figure 1: Different prototype models. (a) shows the sample is misclassified by the PN. Different colors represent different classes. The orange circle denotes the sample to be classified. (b) illustrates the test accuracy of different prototypes on the 5-way k-shot.

prototype bias. But since the model has not seen query samples before extracting support features, some inappropriate features may be extracted, resulting in a significant deviation in prototype estimation. An adaptive induction network (AINet) is proposed to extract more reliable prototypes for each class. The AINet does not take into account the local relative importance of different regions in a sample, while the prototype generated by the PN becomes more discriminative and expressive as the number of support samples increases, as shown in Figure 1 (b). To solve the problem that the calculation of a single prototype is not comprehensive, we assume the estimated prototype follow a multivariate Gaussian distribution (Zhang et al. (2021)). Specifically, the features in the target task are transformed using the Yeo-Johnson transformation, and then two kinds of prototypes are combined, which are generated by AINet and PN, respectively. Finally, the performance of FP_AINet is evaluated on the *mini*ImageNet, *tiered*ImageNet, and CIFAR_FS. Besides, the ablation experiments validate the effectiveness of the FP_AINet. Experimental results show that the FP_AINet can generate a more representative prototype and improve the accuracy of few-shot learning.

The main contributions are summarized as follows:
(1) A novel method of AINet is proposed to assign scores to support samples based on their relevance automatically. (2) A modified Gaussian-based fusion algorithm is employed to aggregates prototypes from PN and AINet by exploring the unlabeled samples. (3) Extensive experiments on three datasets demonstrate the effectiveness of the FP_AINet.

2 Related Work

Unlike conventional machine learning, which provides abundant training examples, few-shot learning requires a classifier that can quickly adapt to novel classes with limited examples. Many efforts have been made to address the issue of data efficiency.

Metric-based methods. To boost the performance of PN, task dependent adaptive metric (TADAM) (Oreshkin et al. (2018)) proposes metric scaling and task conditioning. It is difficult to represent the distribution of a class with limited samples, so many methods have been proposed to correct bias in prototype estimations (Hou & Sato (2021); Yang et al. (2021)). BD-CSPN (Liu et al. (2020)) modifies prototypes by diminishing intra-class and cross-class bias. A pseudo-label is used to reduce intra-class bias, but it is easy to introduce noise. Rather than relying on a pre-defined metric to calculate similarity (Vinyals et al. (2016)), relation network (Sung et al. (2018)) and a deep comparison network (Zhang et al. (2018)) train deep neural networks to compare each query-support image pair. While previous methods adopted the conceptual representation of the first moment (Snell et al. (2017)), CovaMNet (Li et al. (2019)) adopts the second moment rather than the first moment for feature description. Unlike the above methods, multi-level metric learning (Chen et al. (2022)) measures the similarity at three different feature levels. According to the above analysis, most existing methods ignore the noisy samples, resulting in biased class representations. To solve this issue, this paper proposes a more accurate prototype estimate method to improve the few-shot image classification performance.

Transductive few-shot learning. In general, inductive few-shot is employed when data acquisition is expensive, and transductive few-shot is applied when data labeling is expensive (Bendou et al. (2022)). Some studies have tackled the problem by utilizing the additional knowledge from the query dataset or extra unlabeled examples in a transductive setting (Wang et al. (2020); Nichol et al. (2018)). However, they share knowledge between query datasets via batch normalization rather than explicitly modeling the transductive setting as in (Flennerhag et al. (2020)). Task-adaptive feature sub-space learning (TAFSSL) (Lichtenstein et al. (2020)) looks for the discriminative feature sub-spaces for few-shot classification tasks. In contrast to unidirectional label propagation, mutual centralized learning (MCL) considers query and support dataset features as bipartite data and avoids self-reinforcements (Liu et al. (2022)). Inspired by transductive few-shot learning, unlabeled samples are employed to estimate the prototype and enrich the feature representation.

3 METHOD

3.1 PROBLEM DEFINITION

A few-shot classification setting includes two datasets: the base class dataset D^{base} with abundant labeled images and the novel class dataset D^{novel} with few labeled data. Suppose $D^{base} = \{x_t^{base}, y_t^{base}\}_{t=1}^{N^{base}}$, x_t^{base} represents the image sampled from the base class C^{base}, $y_t^{base} \in C^{base}$ is the label of x_t^{base}, there is no intersection between the base class and novel class, that is $C^{base} \cap C^{novel} = \emptyset$, $C^{base} \cup C^{novel} = C$. In each iteration process, one of the episodes means that N classes are selected at random, and each class contains K labeled samples as a support dataset $S = \{(x^s, y^s)\}_{s=1}^{N \times K}$ with a few labeled samples. The query set $Q = \{(x^q, y^q)\}_{q=N \times K+1}^{N \times K'}$ contains examples of the same N classes in S, K' is the quantity of each class in Q. The model needs to predict a class label for a query sample given N support classes, each containing K support samples.

3.2 OVERALL ARCHITECTURE

The FP_AINet consists of three stages, including the pre-training stage, the meta-training stage, and the meta-testing stage. An overview of the FP_AINet is provided in Figure 2.

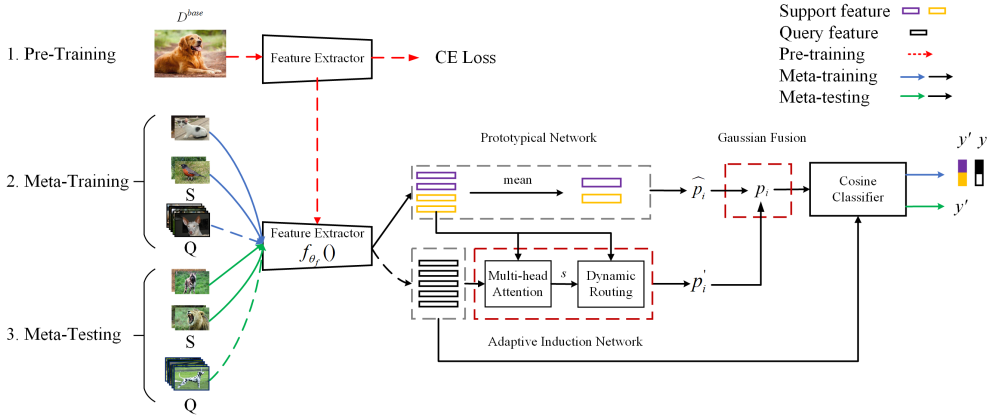


Figure 2: An overview of FP_AINet.

Pre-training stage. During the pre-training stage, an embedding model is trained on the base class dataset D^{base}, the last Softmax layer is removed, and the classifier is transformed into a feature extractor $f_{\theta_f}()$ with parameters θ_f, allowing the model to learn task-agnostic knowledge from base classes and then apply this knowledge to novel classes to produce more reliable prototypes. Then, the feature extractor $f_{\theta_f}()$ is frozen.

Meta-training stage. An N-way K-shot classification task is constructed through episode few-shot learning using the base class dataset D^{base}. In each episode, class C is sampled from D^{base}, K

samples of each class are used as support set S, K' are selected as query dataset Q from the remaining samples in class C. Then the $f_{\theta_f}()$ can be fine-tuned on query dataset. During each episode, estimating the mean-based prototype $\hat{p_i}$ by averaging the labeled support features. Furthermore, AINet is proposed to learn the class prototype p'_i, which is derived from the features $f_{\theta_f}(x)$ of the support and query samples. To obtain more mutual information, the fusion prototype p_i is calculated using the Gaussian-based fusion method. Finally, the cosine similarity of features $f_{\theta_f}(x)$ and p_i is calculated to determine the probability that each sample $x \in Q$ belongs to class i.

Meta-testing stage. The same as the meta-training, and classification task is performed on D^{novel}.

3.2.1 ADAPTIVE INDUCTION NETWORK

The induction module (IM) (Geng et al. (2019)) learns the class-level relationship by considering features and classes to be local-global relationships, but because of the diversity and incompleteness of the support sample, every support sample contributes differently to the class representation when it faces different target query samples. In order to learn a more representative class vector and reduce sample noise, we propose an AINet that pays more attention to effective instances for current query samples. The details of the AINet are shown in Algorithm 1. Using the multi-head self-attention mechanism, the support vector z^s_{ij} and query vector z^q are concatenated to calculate the relationship score; each support vector has its weight attached to the current query vector. Then, we apply dynamic routing to obtain a class vector. The process adjusts the connection's strength dynamically and makes sure that the sum of the coupling coefficients d_i between class i and all of its support samples is 1. The difference is that when adjusting the logits of coupling coefficients in the last step of every iteration, we consider not only the consistency of class candidate vectors and sample prediction vectors but also the relationship between query and support vectors.

Algorithm 1 Adaptive Induction Network

Require: sample vector z^s_{ij} in support dataset S and a vector z^q in query dataset Q, initialize the logits of coupling coefficients: $b_{ij} = 0$

Ensure: Class vector p'_i

for all samples $j = 1, ..., K$ in class i:

$z_{ij} = Concat(z^s_{ij}, z^q)$, z_{ij} is equivalent to concatenate the vector z^s_{ij} and z^q

$s_{ij} = softmax(\frac{z_{ij} z^T_{ij}}{\sqrt{d}}) z_{ij}$, where d is the dimension of z_{ij}

$\hat{z}^s_{ij} = squash(W_s z^s_{ij} + b_s)$, where W_s is transformation weights, b_s denotes bias

for r iterations **do**

$d_i = \textit{softmax}(b_i)$

$p'_i = \sum_j d_{ij} \cdot \hat{z}^s_{ij}$, where $\hat{z}^s_{ij}$ is the prediction vector, p'_i is the class candidate vector

$p'_i = \textit{squash}(p'_i) = \frac{\|p'_i\|^2}{1+\|p'_i\|^2} \frac{p'_i}{\|p'_i\|}$, where $squash$ is a non-linear squashing function

for all sample $j = 1, ..., K$ in class i:

$b_{ij} = b_{ij} + s_{ij} \cdot tanh(\hat{z}^s_{ij} \cdot p'_i)$

end for

return p'_i

3.2.2 PROTOTYPE FUSION

When the number of training samples is limited, p'_i is more accurate because the model needs to focus on more representative features, and $\hat{p_i}$ is more representative as the number of samples increases because the model only considers global features and ignores local features. This means that $\hat{p_i}$ and p'_i can learn mutual affiliations with each other(Zhang et al. (2021)). In order to address the aforementioned issues, a prototype fusion algorithm is proposed to reduce the prototype bias. We assume that the estimated prototype has a Gaussian distribution, and the distributions are independent of each other because samples in the pre-trained space are continuous and clustered. Algorithm 2 describes the Gaussian-based prototype fusion.

To follow a multivariate normal distribution(Yang et al. (2021)), the input features are preprocessed using the Yeo-Johnson transformation(Weisberg (2001)). The Yeo-Johnson transformation can reduce the heteroskedasticity of random variables and increase their normality, resulting in a probabil-

ity density function with a similarity to the normal distribution. At the same time, the Yeo-Johnson transformation can be applied to samples with zero and negative features, making it suitable for statistical analysis of random variables based on the normal assumption, as follows in Equation 1.

$$f_{\theta_f}(x) = \begin{cases} \frac{\left[(f_{\theta_f}(x)+1)^{\lambda}-1\right]}{\lambda}, & \lambda \neq 0, f_{\theta_f}(x) \geq 0 \\ \log(f_{\theta_f}(x))+1), & \lambda = 0, f_{\theta_f}(x) \geq 0 \\ -\frac{\left[(-f_{\theta_f}(x)+1)^{2-\lambda}-1\right]}{2-\lambda}, & \lambda \neq 2, f_{\theta_f}(x) < 0 \\ -\log(-f_{\theta_f}(x)+1), & \lambda = 2, f_{\theta_f}(x) < 0 \end{cases} \tag{1}$$

where $f_{\theta_f}(x)$ is the feature to be transformed and the λ is employed to correct the distribution. Then, the mean-based prototype of $\hat{p_i}$ should be estimated by averaging the features of the support labeled samples, it can be calculated by Equation 2.

$$\hat{p_i} = \frac{1}{|\mathcal{S}_i|} \sum_{x \in \mathcal{S}_i} f_{\theta_f}(x) \tag{2}$$

where S_i represents the support dataset extracted for the class i, and $f_{\theta_f}(x)$ is the feature of support dataset. We assume the $\hat{p_i}$ follows a Gaussian distribution with a mean $\hat{\mu_i}$ and diagonal covariance $diag(\hat{\sigma_i^2})$, and p_i' is a sample from $N(\mu_i', diag(\sigma_i'^2))$. To improve the class representation of the model, learn a Gaussian distribution with mean $\hat{\mu_i} + \mu_i'$ and diagonal covariance $diag(\hat{\sigma_i^2} + \sigma_i'^2)$, then the mean is used to calculate the fusion prototype p_i, as shown in Equation 3.

$$\hat{\theta} \sim N(\hat{\mu_i}, diag(\hat{\sigma_i^2}),\ \theta' \sim N(\mu_i', diag(\sigma_i'^2)),\ \theta \sim N(\hat{\mu_i} + \mu_i', diag(\hat{\sigma_i^2} + \sigma_i'^2) \tag{3}$$

Transductive few-shot learning method is used to calculate the $\hat{\mu_i}$ and μ_i'(Liu et al. (2020)) by leveraging the unlabeled samples. When the class prototype is $\hat{p_i}$ or p_i', the Equations 4 and 5 can be used to calculate the probability of $x \in S \cup Q$, where S is the support dataset with a few labeled samples and Q is the query dataset with unlabeled samples.

$$\hat{P}(y = i \mid x) = \frac{e^{d\left(f_{\theta_f}(x)),\, \hat{p_i}\right)}}{\sum_c e^{d\left(f_{\theta_f}(x),\, p_c\right)}} \tag{4}$$

$$P'(y = i \mid x) = \frac{e^{d\left(f_{\theta_f}(x),\, p_i'\right)}}{\sum_c e^{d\left(f_{\theta_f}(x)),\, p_c\right)}} \tag{5}$$

where $d()$ is the cosine similarity. Then, $\hat{\mu_i}$ and μ_i' can be calculated by regarding $\hat{P}(y = i \mid x)$ and $P'(y = i \mid x)$ as the weights, as shown in Equation 6 and 7.

$$\hat{\mu_i} = \frac{1}{\sum_{x \in S \cup Q} P(i \mid x)} \sum_{x \in S \cup Q} \hat{P}(i \mid x) f_{\theta_f}(x) \tag{6}$$

$$\mu_i' = \frac{1}{\sum_{x \in S \cup Q} P(i \mid x)} \sum_{x \in S \cup Q} P'(i \mid x) f_{\theta_f}(x) \tag{7}$$

Finally, the fusion prototype of p_i can be obtained by $\hat{\mu_i}$ and μ_i, as shown in Equation 8.

$$p_i = \mu_i = \hat{\mu_i} + \mu_i' \tag{8}$$

4 Experimental Setup

4.1 Datasets and Settings

The method of FP_AINet is evaluated on the *mini*ImageNet, *tiered*ImageNet and CIFAR_FS. *mini*ImageNet (Ravi & Larochelle (2017)) contains 100 classes with 600 samples per class. The

Algorithm 2 Prototype Fusion

Require: Support samples $S = \{(x^s, y^s)\}_{s=1}^{N \times K}$, query samples $Q = \{(x^q, y^q)\}_{q=N \times K+1}^{N \times K'}$
Ensure: Fusion prototype p_i
  for each $episode\ iteration$ **do**
    Create the episodic tasks using S and Q, fine-tuned the feature extractor $f_{\theta_f}()$
    Estimate the mean-based prototype $\hat{p_i}$ with Equation 2
    Calculate the class vector p_i' with Algorithm 1
    Use $\hat{p_i}$ and p_i' to calculate the probability of $x \in S \cup Q$ with Equation 4 and 5, respectively
    Calculate $\hat{\mu_i}$ and μ_i' by $\hat{P}(y = i \mid x)$ and $P'(y = i \mid x)$ with Equation 6 and 7, respectively
    Estimate the fusion prototype p_i by $\hat{\mu_i}$ and μ_i'
  end for
  return p_i

dataset is divided into 64, 16, and 20 classes for training, validation, and testing. *tiered*ImageNet (Ren et al. (2018)) consists of a total of 608 classes, which are divided into 34 higher-level classes. The training dataset contains 20 higher-level classes, 351 fine-grained classes; 6 higher-level classes, 97 fine-grained classes as validation sets; 8 higher-level classes, and 160 fine-grained classes as the test datasets. The image size of *mini*ImageNet and *tiered*ImageNet is $84 \times 84 \times 3$. CIFAR_FS (Bertinetto et al. (2019)) contains 100 classes and 600 images in each class, including 64 classes of training datasets, 16 classes of validation datasets, and 20 classes of test datasets. The image size is unified to $32 \times 32 \times 3$.

The classical 5-way $1/5$-shot episodic in few-shot task settings are adopted. The query dataset contains 6 images per class during the meta-training stage, 15 test samples during the meta-testing stage, and 10,000 tasks are randomly constructed. Then test the task and calculate the average classification accuracy of top-1 and the 95% confidence interval as the final result.

4.2 Implementation Details

The experiment is conducted on the feature extractor of ResNet-12 with 640-dimensional for the *tiered*ImageNet. Each residual block contains three 3×3 convolutional layers and a shortcut connection. The WRN-28-10 with a layer number of 28 and a width of 10 is used for tieredImageNet and the extracted features are 512-dimensional. Average pooling is applied at the last block of each architecture to get feature vectors (Mangla et al. (2020)). In the pre-training stage, the base class dataset is trained on 100 epochs with a batch size of 128. SGD with a momentum of 0.9 and weight decay of 0.0005 is adopted as the optimizer to train the feature extractor of ResNet-12, while the Adam optimizer is used for WRN-28-10. In the meta-training stage, data augmentation techniques are used, including random cropping, color jittering, and horizontal flipping. The model is meta-trained for 60 epochs, with each epoch containing 1000 episodes and an initial learning rate of 0.1. When the epochs are set to 20, 40, and 50, the learning rate changes to 0.006, 0.0012, and 0.00024, respectively. λ is set to 0.5 in the Yeo-Johnson transform, and 3 iterations were used for the AINet.

4.3 Experimental Results

4.3.1 Comparison with State-of-the-art Methods

Tables 1 and 2 show the 5-way $1/5$-shot classification results of the FP_AINet and state-of-the-art few-shot learning methods on the *mini*ImageNet and *tiered*ImageNet, respectively. Table 1 shows that the FP_AINet achieves better performance on *mini*ImageNet compared with comparison methods. In the 5-way $1/5$-shot settings, the accuracy of the FP_AINet reaches 72.13% and 84.29%, respectively. Compared to the suboptimal methods Curvature Generation(Gao et al. (2021)) and UniSiam (Lu et al. (2022)), it increased by about 0.34% and 0.89%, respectively. On the *tiered*ImageNet, the accuracy of FP_AINet on 1-shot is higher than 0.49% of the second-best models of BD-CSPN (Liu et al. (2020)), and higher than 0.29% EPNet(Rodríguez et al. (2020)) on a 5-shot setting. The FP_AINet has such an improvement attributed to considering the more important samples. Moreover, the Gaussian-based fusion algorithm alleviates the prototype error and facilitates learning the optimal prototype by exploring the unlabeled samples.

Table 1: 5-way $1/5$-shot accuracy ($\%$) on *mini*ImageNet with 95% confidence intervals.The best two results are highlighted and underlined.

Method	Backbone	Setting	*mini*ImageNet 5-way 1-shot	*mini*ImageNet 5-way 5-shot
Matching Network (Vinyals et al. (2016))	64-64-64-64	Inductive	43.56 ± 0.84	55.31 ± 0.73
Relation Network (Sung et al. (2018))	64-96-128-256	Inductive	50.44 ± 0.82	65.32 ± 0.70
R2D2 (Bertinetto et al. (2019))	96-192-384-512	Inductive	51.20 ± 0.60	68.80 ± 0.10
Baseline++ (Chen et al. (2019))	ResNet-18	Inductive	51.87 ± 0.77	75.68 ± 0.63
TADAM (Oreshkin et al. (2018))	ResNet-12	Transductive	58.50 ± 0.30	76.70 ± 0.30
PN (Snell et al. (2017))	ResNet-12	Inductive	60.37 ± 0.83	78.02 ± 0.57
B+@EST+L2-N (Hou & Sato (2021))	ResNet-18	Inductive	62.44	77.13
MetaOptNet (Lee et al. (2019))	ResNet-12	Inductive	62.64 ± 0.61	78.63 ± 0.46
MetaBaseline (Chen et al. (2021b))	ResNet12	Inductive	63.17 ± 0.23	79.26 ± 0.17
S2M2 (Mangla et al. (2020))	ResNet-18	Inductive	64.06 ± 0.18	80.58 ± 0.12
UniSiam (Lu et al. (2022))	ResNet-34	Inductive	65.55 ± 0.36	<u>83.40 ± 0.24</u>
DeepEMD (Zhang et al. (2020))	ResNet-12	Inductive	65.91 ± 0.82	82.41 ± 0.56
ICI (Wang et al. (2020))	ResNet-12	Transductive	66.8	79.26
DC (Yang et al. (2021))	WRN-28-10	Inductive	68.57 ± 0.55	82.88 ± 0.42
BD-CSPN (Liu et al. (2020))	WRN-28-10	Transductive	70.31 ± 0.93	81.89 ± 0.60
AIM (Lee et al. (2021))	WRN-28-10	Transductive	71.22 ± 0.57	82.25 ± 0.34
Curvature Generation(Gao et al. (2021))	ResNet-12	Transductive	<u>71.79 ± 0.23</u>	83.00 ± 0.17
FP_AINet (OURS)	WRN-28-10	Transductive	**72.13 ± 0.73**	**84.29 ± 0.44**

Table 3 shows the comparison results of the FP_AINet with the main few-shot learning methods on the CIFAR-FS. In the 5-way 1-shot setting, the accuracy of the FP_AINet reaches 81.92%, 0.32% higher than the suboptimal method of SSR (Shen et al. (2021)), which proves that FP_AINet can handle extremely few-shot classification tasks better. In the 5-way 5-shot setting, the accuracy of FP_AINet is 89.38%, which is 0.38% higher than the suboptimal method EASY (Bendou et al. (2022)). The FP_AINet has the highest accuracy with the same backbone, and accurate prototypes are more effective than fully extracted features. Furthermore, accuracy on the 5-shot setting is significantly higher than on the 1-shot setting. The main reason is that fewer annotated samples result in inaccurate prototype estimation, whereas a 5-shot can yield a more representative prototype estimation. It is verified that the FP_AINet can better handle the few-shot learning task with a limited amount of data. The prototype features of the novel class are expressed more abundantly and accurately by fusing the prototypes.

Table 2: 5-way 1-shot$/$5-shot accuracy ($\%$) on *tiered*ImageNet with 95% confidence intervals.

Method	Backbone	Setting	*tiered*ImageNet 5-way 1-shot	*tiered*ImageNet 5-way 5-shot
Relation Network (Sung et al. (2018))	64-96-128-256	Inductive	54.48 ± 0.93	71.32 ± 0.78
B+@EST+L2-N (Hou & Sato (2021))	ResNet-18	Inductive	60.87	81.80
PN (Snell et al. (2017))	ResNet-12	Inductive	65.65 ± 0.92	83.85 ± 0.36
MetaOptNet (Lee et al. (2019))	ResNet-12	Inductive	65.99 ± 0.72	81.56 ± 0.53
MetaBaseline (Chen et al. (2021b))	ResNet-12	Inductive	68.62 ± 0.27	83.74 ± 0.18
DeepEMD (Zhang et al. (2020))	ResNet-12	Inductive	71.16 ± 0.87	86.03 ± 0.58
Meta DeepBDC (Xie et al. (2022))	ResNet-12	Inductive	72.34 ± 0.49	87.31 ± 0.32
SIB (Hu et al. (2020))	WRN-28-10	Transductive	72.9	82.8
ECKPN (Chen et al. (2021a))	ResNet-12	Transductive	73.59 ± 0.45	88.13 ± 0.28
Curvature Generation(Gao et al. (2021))	ResNet-12	Transductive	77.19 ± 0.24	86.18 ± 0.15
ICI v2 (Wang et al. (2021))	ResNet-12	Transductive	77.48 ± 0.62	86.84 ± 0.36
EPNet (Rodríguez et al. (2020))	WRN-28-10	Transductive	78.50 ± 0.91	<u>88.36 ± 0.57</u>
BD-CSPN (Liu et al. (2020))	WRN-28-10	Transductive	<u>78.74 ± 0.95</u>	86.92 ± 0.63
FP_AINet (OURS)	WRN-28-10	Transductive	**79.23 ± 0.70**	**88.65 ± 0.49**

Table 3: 5-way 1-shot/5-shot accuracy (%) on CIFAR_FS with 95% confidence intervals.

Method	Backbone	Setting	CIFAR_FS 5-way 1-shot	CIFAR_FS 5-way 5-shot
Relation Network (Sung et al. (2018))	64-96-128-256	Inductive	55.00 ± 1.00	69.30 ± 0.80
MAML (Finn et al. (2017))	32-32-32-32	Inductive	58.90 ± 1.90	71.50 ± 1.00
B+@EST+L2-N (Hou & Sato (2021))	ResNet-18	Inductive	63.00	77.99
BD-CSPN (Liu et al. (2020))	WRN-28-10	Transductive	72.13 ± 1.01	82.28 ± 0.69
PN (Snell et al. (2017))	ResNet-12	Inductive	72.20 ± 0.70	83.50 ± 0.50
MetaOptNet (Lee et al. (2019))	ResNet-12	Inductive	72.80 ± 0.70	85.00 ± 0.50
S2M2 (Mangla et al. (2020))	ResNet-18	Inductive	74.81 ± 0.19	87.47 ± 0.13
EASY (Bendou et al. (2022))	3xResNet-12	Transductive	76.20 ± 0.20	<u>89.00 ± 0.14</u>
Fine-tuning (Dhillon et al. (2020))	WRN-28-10	Transductive	76.58 ± 0.68	85.79 ± 0.50
ICI v2 (Wang et al. (2021))	ResNet-12	Transductive	79.19 ± 0.63	86.66 ± 0.36
SIB (Hu et al. (2020))	WRN-28-10	Transductive	80.00 ± 0.60	85.30 ± 0.40
SSR (Shen et al. (2021))	WRN-28-10	Transductive	<u>81.60 ± 0.60</u>	86.00 ± 0.40
FP_AINet (OURS)	WRN-28-10	Transductive	**81.92 ± 0.69**	**89.38 ± 0.44**

4.3.2 Ablation Study

Table 4 summarizes the results of FP_AINet and shows that each component is important in few-shot image classification, giving improvements over the state-of-the-art on the *mini*ImageNet. Among them, (i) represents classification using only PN, (ii) is classification result of induction module, (iii) denotes classification using only AINet, and (iv) represents the Gaussian-based fusion algorithm. Obviously, in the 5-way 1-shot setting, if neither module is used, the accuracy drops by more than 10%. The prototype fusion algorithm of FP_AINet achieves better performance than AINet.

Adaptive Induction Network. It can be seen from (iii) in Table 4 that in the 5-way 1-shot, the classification result of AINet is better than the PN, and the main reason is that the module calculates the prototype by using query samples and selection. At the same time, the induction prototype method obtains class-level information and automatically adjusts the coupling coefficient according to the input, which is suitable for few-shot learning and can achieve good results in the presence of noise. In 5-way 5-shot, with the increase of samples, the mean-based prototype obtains better class representation. The results demonstrate that paying more attention to effective support samples is an important factor in the few-shot classification problem.

Prototype fusion. The accuracy of the AINet is improved by about 9% and 6%, respectively, in the 5-way 1/5-shot settings, as shown by the model of (iv) in Table 4. The results indicate that fusion prototypes can improve model performance and alleviate bias in prototype estimates. The primary argument is that prototype fusion utilizes more samples, which can more effectively address the issues of sample noise and incompleteness in few-shot learning. The results show the necessity and effectiveness of learning an optimal class prototype.

Table 4: Ablation studies of 5-way 1/5-shot on *mini*ImageNet.

	IM	AINet	PN	Fusion	5-way 1-shot	5-way 5-shot
(i)			✓		61.47 ± 0.66	79.33 ± 0.48
(ii)	✓				63.85 ± 0.68	78.23 ± 0.48
(iii)		✓			63.88 ± 0.66	78.49 ± 0.49
(iv)		✓	✓	✓	**72.13 ± 0.73**	**84.29 ± 0.44**

The prototypes generated by the FP_AINet are visualized using t-distributed stochastic neighbor embedding (t-SNE). A 5-way 1-shot task of *mini*Imagenet is shown in Figure 3, where circles represent query samples and different colors denote different classes, stars represent PN features, pentagons are AINet features, and squares represent FP_AINet. The prototypes generated by FP_AINet are

much closer to the class center, which can effectively learn the representation of a prototype and improve the capacity of the support dataset.

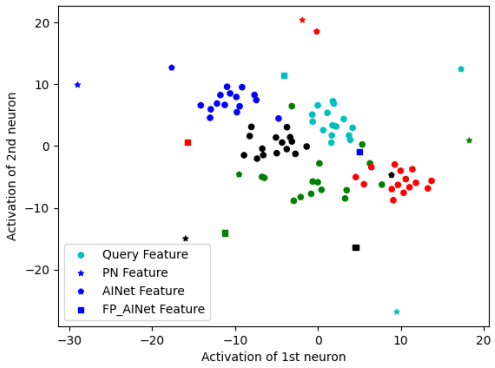
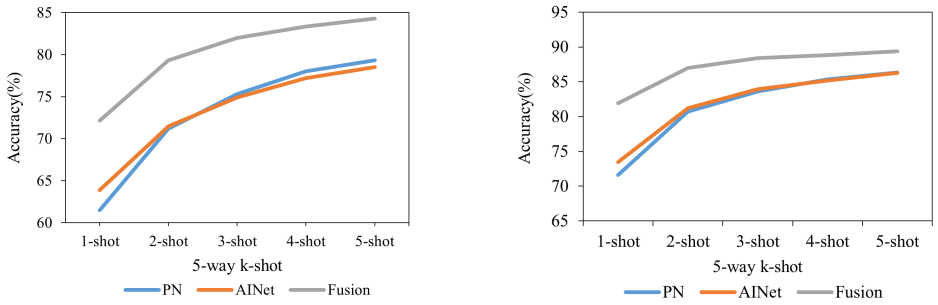


Figure 3: t-SNE visualization of different prototype.

4.3.3 Different Prototype Methods

Figure 4 shows the results of different prototype methods on the 5-way k-shot task. On the *mini*ImageNet, the prototype based on AINet is more accurate in $1/2$-shot tasks. The PN achieves better performance on the $3/4/5$-shot tasks; on the CIFAR_FS, the AINet outperforms the PN on the $1/2/3$-shot classification tasks, while the PN is better at classification on the $4/5$-shot tasks. The main reason is that mean-based prototypes may be far from the expected class center when given very few labeled samples. But the mean-based prototype obtains more training samples and achieves better classification performance as the number of shots increases. The advantages of the two prototypes are fused to create a more representative prototype through Gaussian-based prototype fusion. Meanwhile, when the shot of the support dataset setting on the 5-way k-shot task is increased to 5, the accuracy of the three prototype models on *mini*ImageNet and CIFAR_FS improves.



(a) *mini*ImageNet (b) CIFAR_FS

Figure 4: The accuracy of different prototype methods on different datasets.

5 Conclusion

To address the problem of noisy samples in few-shot learning, we propose a new method based on Gaussian fusion with an adaptive induction network. Firstly, it is significant to exploit different samples for obtaining the class representation, and the AINet can evaluate the significance of different samples adaptively. Secondly, a single prototype method is not comprehensive enough, and a Gaussian-based fusion algorithm is employed to obtain more accurate prototypes. Experiments show that the FP_AINet achieves consistent improvements on three datasets, which verifies the effectiveness of the FP_AINet in few-shot image classification.

REFERENCES

Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your maml. *International Conference on Learning Representations*, 2019.

Yassir Bendou, Yuqing Hu, Raphael Lafargue, Giulia Lioi, Bastien Pasdeloup, Stéphane Pateux, and Vincent Gripon. Easy: Ensemble augmented-shot y-shaped learning: State-of-the-art few-shot classification with simple ingredients. *arXiv preprint arXiv:2201.09699*, 2022.

Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *International Conference on Learning Representations*, 2019.

Chaofan Chen, Xiaoshan Yang, Changsheng Xu, Xuhui Huang, and Zhe Ma. Eckpn: Explicit class knowledge propagation network for transductive few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6596–6605, 2021a.

Haoxing Chen, Huaxiong Li, Yaohui Li, and Chunlin Chen. Multi-level metric learning for few-shot image recognition. In *International Conference on Artificial Neural Networks*, pp. 243–254, 2022.

WeiYu Chen, YenCheng Liu, Zsolt Kira, YuChiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *International Conference on Learning Representations*, 2019.

Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *International Conference on Learning Representations*, 2020.

Yinbo Chen, Zhuang Liu, Huijuan Xu, Trevor Darrell, and Xiaolong Wang. Meta-baseline: Exploring simple meta-learning for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9062–9071, 2021b.

Guneet S Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. *International Conference on Learning Representations*, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135, 2017.

Sebastian Flennerhag, Andrei A Rusu, Razvan Pascanu, Francesco Visin, Hujun Yin, and Raia Hadsell. Meta-learning with warped gradient descent. *International Conference on Learning Representations*, 2020.

Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. Curvature generation in curved spaces for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8691–8700, 2021.

Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. Induction networks for few-shot text classification. *Conference on Empirical Methods in Natural Language Processing-International Joint Conference on Natural Language Processing*, pp. 3895–3904, 2019.

Mingcheng Hou and Issei Sato. A closer look at prototype classifier for few-shot image classification. *International Conference on Learning Representations*, pp. 721–731, 2021.

Shell Xu Hu, Pablo G Moreno, Yang Xiao, Xi Shen, Guillaume Obozinski, Neil D Lawrence, and Andreas Damianou. Empirical bayes transductive meta-learning with synthetic gradients. *International Conference on Learning Representations*, 2020.

Chia Hsiang Kao, WeiChen Chiu, and PinYu Chen. Maml is a noisy contrastive learner in classification. In *International Conference on Learning Representations*, 2022.

Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *International Conference on Machine Learning Deep Learning Workshop*, volume 2, pp. 1–8, 2015.

Eugene Lee, Cheng-Han Huang, and Chen-Yi Lee. Few-shot and continual learning with attentive independent mechanisms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9455–9464, 2021.

Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10657–10665, 2019.

Wenbin Li, Jinglin Xu, Jing Huo, Lei Wang, Yang Gao, and Jiebo Luo. Distribution consistency based covariance metric networks for few-shot learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 8642–8649, 2019.

Kevin J Liang, Samrudhdhi B Rangrej, Vladan Petrovic, and Tal Hassner. Few-shot learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9089–9098, 2022.

Moshe Lichtenstein, Prasanna Sattigeri, Rogerio Feris, Raja Giryes, and Leonid Karlinsky. Tafssl: Task-adaptive feature sub-space learning for few-shot classification. In *European Conference on Computer Vision*, pp. 522–539, 2020.

Jinlu Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning. In *European Conference on Computer Vision*, pp. 741–756, 2020.

Yang Liu, Weifeng Zhang, Chao Xiang, Tu Zheng, Deng Cai, and Xiaofei He. Learning to affiliate: Mutual centralized learning for few-shot classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14411–14420, 2022.

Yuning Lu, Liangjian Wen, Jianzhuang Liu, Yajing Liu, and Xinmei Tian. Self-supervision can be a good few-shot learner. *arXiv preprint arXiv:2207.09176*, 2022.

Puneet Mangla, Nupur Kumari, Abhishek Sinha, Mayank Singh, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Charting the right manifold: Manifold mixup for few-shot learning. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2218–2227, 2020.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. *Advances in Neural Information Processing Systems*, 31: 721–731, 2018.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. *International Conference on Learning Representations*, 2017.

Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *International Conference on Learning Representations*, 2018.

Pau Rodríguez, Issam Laradji, Alexandre Drouin, and Alexandre Lacoste. Embedding propagation: Smoother manifold for few-shot classification. In *European Conference on Computer Vision*, pp. 121–138. Springer, 2020.

Xi Shen, Yang Xiao, Shell Xu Hu, Othman Sbai, and Mathieu Aubry. Re-ranking for image retrieval and transductive few-shot classification. *Advances in Neural Information Processing Systems*, 34: 25932–25943, 2021.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems*, 30:4077–4087, 2017.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2018.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *European Conference on Computer Vision*, pp. 266–282, 2020.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, and Daan Wierstra. Matching networks for one shot learning. *Advances in Neural Information Processing Systems*, 29:3637–3645, 2016.

Yikai Wang, Chengming Xu, Chen Liu, Li Zhang, and Yanwei Fu. Instance credibility inference for few-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12836–12845, 2020.

Yikai Wang, Li Zhang, Yuan Yao, and Yanwei Fu. How to trust unlabeled data instance credibility inference for few-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Sanford Weisberg. Yeo-johnson power transformations. *Department of Applied Statistics, University of Minnesota. Retrieved June*, 1:2003, 2001.

Jiangtao Xie, Fei Long, Jiaming Lv, Qilong Wang, and Peihua 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*, pp. 7972–7981, 2022.

Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. *International Conference on Learning Representations*, 2021.

Huaxiu Yao, Yu Wang, Ying Wei, Peilin Zhao, Mehrdad Mahdavi, Defu Lian, and Chelsea Finn. Meta-learning with an adaptive task scheduler. *Advances in Neural Information Processing Systems*, 34:7497–7509, 2021.

Baoquan Zhang, Xutao Li, Yunming Ye, Zhichao Huang, and Lisai Zhang. Prototype completion with primitive knowledge for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3754–3762, 2021.

Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12203–12213, 2020.

Xueting Zhang, Flood Sung, Yuting Qiang, Yongxin Yang, and Timothy M Hospedales. Deep comparison: Relation columns for few-shot learning. *arXiv preprint arXiv:1811.07100*, 2018.

A EFFECT OF YEO-JOHNSON TRANSFORMATION

Figure 5 shows the 5-way 1-shot accuracy when choosing different λ for the Yeo-Johnson transform in Equation 1. It can be found λ equals 0.5 is the optimum choice, and different values have a significant impact on the classification accuracy. With the Yeo-Johnson transformation, the distribution of features becomes more aligned with the calibrated Gaussian distribution, which favors the classifier that is trained on features from the calibrated distribution.

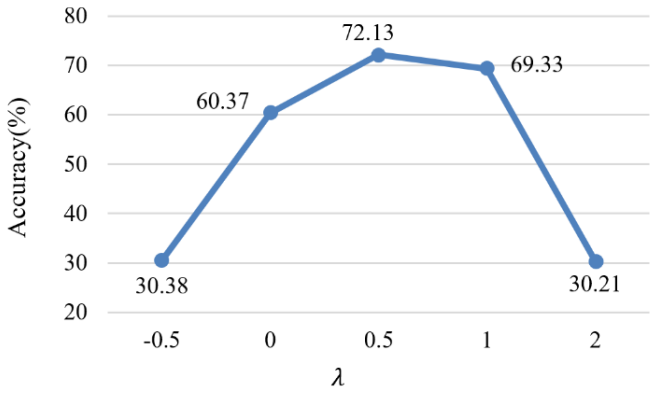
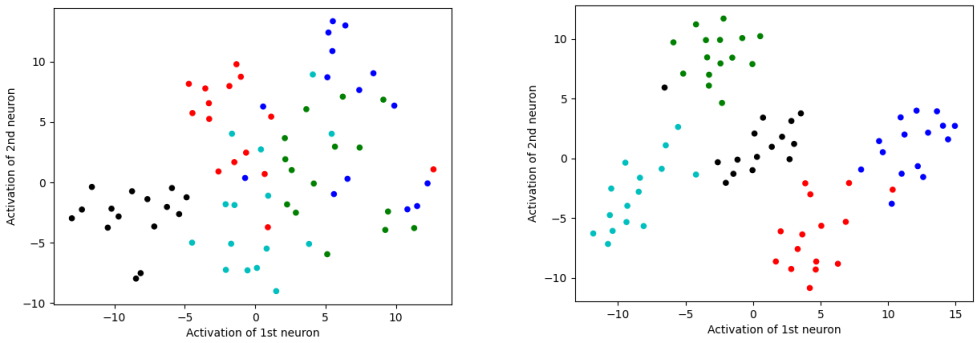


Figure 5: Accuracy of different values of λ on *mini*ImageNet.

The query features before and after the Yeo-Johnson transformation are shown in Figure 6. Different colors represent categories. It is observed that the distribution before transformation is more skewed. The distribution after Yeo-Johnson transformation can very well satisfy the Gaussian assumption. It provides a powerful means of reducing skewness.



(a) Before transformation (b) After transformation

Figure 6: Feature transformation.

B COMPARISON OF COMPUTATION COST

Table 5 shows a detailed analysis of different modules in our method and classification results on *mini*ImageNet. Compared with the baseline of PN, FLOPs increased by 16M. The main reason is that the attention mechanism introduces some attention parameters. The operation of fusion almost without additional calculations. The parameter of ours has increased by 410.9 K.

C COMPARISON OF DIFFERENT BACKBONES

In order to explore the influence of feature embedding vectors, the depth of the backbone network is changed and the same settings were used for the three models. It can be seen from Table 6 that the

Table 5: Comparison of the FLOPs, Params, and Accuracy on *mini*ImageNet.

5-way 1-shot	**FLOPs**	**Params**	**Accuracy**
Backbone (WRN-28-10)	36.19 G	36.47 M	-
PN	+ 0	+ 0	61.47 ± 0.66
AINet	+ 16.0 M	+ 410.9 K	63.88 ± 0.66
FP_AINet(OURS)	+ 16.0 M	+ 410.9 K	72.13 ± 0.73

*mini*ImageNet has achieved the best results on the WRN-28-10 backbone network. The classification results are constantly improving as the number of network layers increases. In few-shot learning, the feature embedding vector is the key factor affecting the classification results. A better backbone network can bring better test performance in few-shot learning.

Table 6: Accuracy (%) on *mini*ImageNet with 95% confidence intervals of different backbone.

	***mini*ImageNet**	
Backbone	**5-way 1-shot**	**5-way 5-shot**
ResNet-10	60.61 ± 0.76	74.80 ± 0.53
ResNet-12	66.63 ± 0.76	79.64 ± 0.51
WRN-28-10	**72.13 ± 0.73**	**84.29 ± 0.44**