ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

Research on few-shot soil classification algorithm based on feature transformation and dual measurement

Hao Liang[a,*], Jiawei Wang[b], Ling Wei[c]

School of Information Science and Engineering, Chongqing Jiaotong University, Chongqing, China
[a]1422268226@qq.com, [b]154523963@qq.com, [c]2689874779@qq.com
[*]Corresponding author

Abstract: *The soil classification task has become a major challenge in the field of computer vision due to subtle differences between categories, scarce samples, and traditional classification methods that are time-consuming and labor-intensive. To solve this problem, this paper proposes a soil classification algorithm combining adaptive feature transformation (AFT) and dual metric module (CCE) to improve classification accuracy and generalization ability. The AFT module dynamically adjusts image features through affine transformation, enhances the model's adaptability to changes in feature distribution, and captures subtle differences. The CCE module combines the advantages of cosine similarity and Euclidean distance to improve the accuracy of feature similarity evaluation. The classification accuracy on the public data sets CUB-200-2011 and Stanford Dogs reached 87.36% and 81.50% respectively, and the classification accuracy on the soil data set Soil was 72.43%. Experimental results show that this method achieves significant performance improvement in few-shot soil classification tasks, verifying its effectiveness in complex classification problems. Future work will further explore the impact of local feature enhancement strategies on model performance to continuously optimize the classification effect.*

Keywords: *Few-shot Learning, Soil Classification, Feature Conversion*

1. Introduction

Soil classification is a process of systematic classification based on the properties of different soil types. Scientific and reasonable soil classification can help researchers understand soil properties more deeply and formulate more effective management and application strategies. Scientific soil classification not only helps evaluate the production potential of soil, but there are significant differences in nutrient content, drainage and water holding capacity between different types of soil, which directly affects the growth of crops. Therefore, proper soil classification can help farmers select suitable crop types, thereby increasing yields [1]. In addition, soil classification also plays a vital role in urban planning and infrastructure construction. For example, understanding the load-bearing capacity and permeability of soil can help engineers design more stable building foundations. Therefore, scientific and accurate soil classification is crucial.

Traditional soil classification methods mainly rely on laboratory chemical analysis and professional experience. Although these methods have high accuracy, they are usually time-consuming and labor-intensive, and difficult to be efficiently applied in large-scale or continuous soil classification tasks. With the development of machine learning technology, researchers have begun to apply it to the field of soil classification. Although it has alleviated the limitations and efficiency problems of traditional methods to a certain extent, the demand for large-scale data by deep learning models makes it difficult for these methods to fully utilize limited soil data, resulting in easy overfitting of the model and insufficient generalization ability. At the same time, traditional machine learning methods, such as support vector machines and convolutional neural networks, cannot effectively cope with the challenges of large differences between soil categories and small differences within categories, and ignore the problem of inter-domain differences between different soil samples [2].

Therefore, in order to solve the above-mentioned problems of scarce soil samples and inter-domain differences, high sample appearance similarity, and difficulty in classification using traditional technologies, this paper proposes a few-shot soil classification algorithm. In order to solve the problem of few soil samples and inter-domain differences, a metric-based few-shot learning algorithm is proposed,

ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

and adaptive feature transformation is used to improve the model's generalization ability for new data. In order to solve the problem of high soil appearance similarity and difficulty in distinguishing, a dual metric module combining cosine similarity and Euclidean distance is designed to flexibly respond to different similarity evaluation requirements, which helps the model better capture the subtle differences between fine-grained categories under few-shot conditions, thereby improving classification performance.

2. Related Work

2.1. Soil Classification Algorithm

Liu W [3] proposed an end-to-end soil classification method, which directly uses the raw data of the piezoelectric cone penetration test (CPTU) for training, does not require noise reduction, and uses a four-dimensional cluster soil stratification method to accurately locate the soil layer boundary and ensure the accuracy of sample annotation. However, this method relies on limited CPTU data for training, and deep learning models usually require a large amount of data to have strong generalization capabilities, which may limit the performance of the model in practical applications.

Alshahrani H [4] combined computer vision and chaos Jaya optimization algorithm (CJOCV-STC) for soil type classification in smart agriculture. This method generates feature vectors through Squeeze Net and uses the chaos Jaya optimization algorithm for hyperparameter optimization. The classification parameters are adjusted by the chicken swarm algorithm (CSA). Although the optimization algorithm is used for hyperparameter adjustment, the metaheuristic algorithm itself is sensitive to the initial parameters and search space. If it is not set properly, it may affect the performance and stability of the final model.

2.2. Few-shot Classification Algorithm

Wang H [5] proposed to solve the feature difference problem in small few-shot learning by stacking multiple heterogeneous pre-trained feature extractors. This method does not require building a general model, but uses a stack combination of multiple feature extractors to perform target domain tasks. Adaptation, this method will significantly increase the computational overhead when facing a large number of source domains, especially when each source domain has a different extractor, the cost of model fine-tuning and stacked generalization is high. In addition, linear stacking The simplicity of the classifier may limit its ability to handle highly nonlinear features.

Sa L [6] proposed a residual attention module to enhance fine-grained recognition capabilities, and introduced a bilinear measurement function to separately learn and fuse different fine-grained image features. Through the bilinear measurement function structure of non-shared weights, it can Better handle cross-domain fine-grained classification problems.

Zhong J [7] proposed to fine-tune the prototype network to improve its generalization ability by enhancing feature extraction and splitting the target domain data set into a pseudo support set and a pseudo query set. However, this method may not work well when dealing with tasks with large differences in data domains. Facing the bottleneck of generalization ability, more complex feature modeling strategies are needed.

3. Method

Based on the above research and analysis, this paper proposes a few-shot soil classification algorithm that combines feature transformation and dual metrics. The overall framework of the algorithm is shown in Figure 1. First, the model extracts the features of the input image through an adaptive feature transformation layer (AFT) combined with attention. This feature transformation layer not only improves the flexibility and robustness of feature representation, but also can adaptively adjust parameters according to the different feature distributions of the input image, making the model more sensitive to fine-grained category differences; then, the image passes through the positioning module (Foreground Target Positioning, FTP) to remove background interference and reduce the interference of background information on the classification task; then, the extracted target image features and original image features are aligned through the feature alignment module (Feature Alignment, FAM) to ensure that the features between different images can be effectively compared and matched in the same spatial

ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

dimension, reducing the impact of feature scale or perspective differences; finally, the processed image features are input into the metric module (Cross Cosine-Euclidean, CCE), which integrates the advantages of cosine similarity and Euclidean distance, uses cosine similarity to evaluate the angular relationship between features, and measures the absolute difference of features through Euclidean distance. By integrating these two metrics, the model obtains a comprehensive similarity score. This score can classify new categories more accurately, thus significantly improving the performance of fine-grained image classification under few-shot conditions.

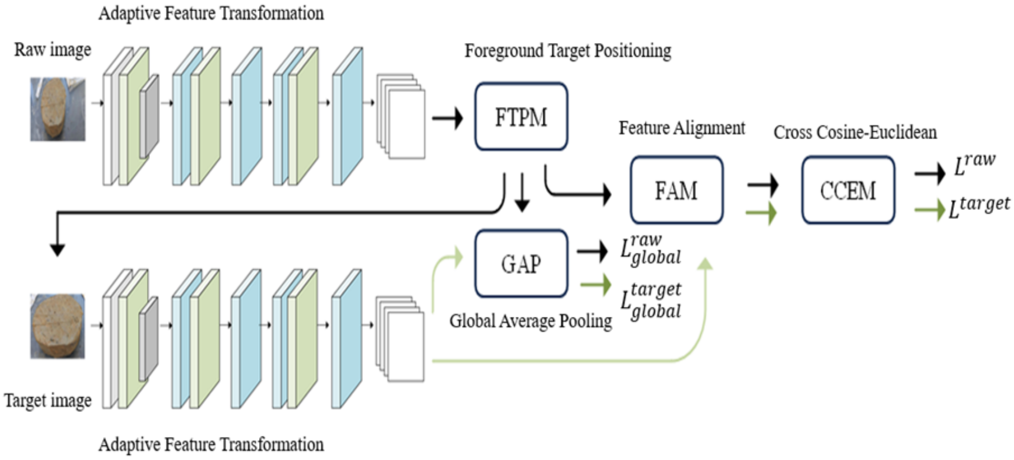

Figure 1: Overall model architecture, the green squares are the feature transition layers, the blue squares are the residual blocks.

4. Adaptive Feature Transformation Module

In convolutional neural networks (CNNs), traditional feature extraction methods are usually static, that is, all input data are processed by the same convolution filter [8]. This unified processing method limits the model's ability to dynamically adjust specific important features in different tasks and scenarios. Therefore, this paper proposes an adaptive feature transformation (AFT) module, which realizes adaptive adjustment of the network's key layer feature map through dynamic scaling and translation operations, and enhances the model's adaptability in specific scenarios and tasks.

The core of the AFT module is to dynamically adjust the feature map based on affine transformation, as shown in Figure 2. The affine transformation linearly scales and translates each channel through the scaling factor γ and the offset β. The specific formula is as follows:

$$\mathrm{AFT}(X) = \gamma \otimes \mathrm{X} + \beta \tag{1}$$

Among them, γ and β are scaling and translation coefficients, and $\otimes$ represents element-wise multiplication. First, use 1x1 convolution to reduce the dimension of the input feature map:

$$\mathrm{Z}_1 = \mathrm{RelU}(Conv(X, W_1)) \tag{2}$$

Then, it is restored to the original dimension through 1x1 convolution. In order to further improve the precision of feature adjustment, the AFT module combines a lightweight attention mechanism and adaptively adjusts γ and β through the generated attention weight matrix Attn, so that the affine transformation of each channel can be adjusted according to the specific features of the input image. The formula is as follows:

$$\mathrm{Attn} = \sigma(\mathrm{Conv}(\mathrm{Z}_1, \mathrm{W}_2)) \tag{3}$$

Among them, $\sigma(\cdot)$ represents the Sigmoid function. According to the attention weight, γ and β are adjusted as follows:

$$\gamma = \gamma_0 \cdot \mathrm{Attn},\ \beta = \beta_0 \cdot \mathrm{Attn} \tag{4}$$

Among them, γ_0 and β_0 are learnable parameters. The final feature transformation is expressed

ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

as:

$$AFT(X) = (\gamma_0 \cdot \mathrm{A}ttn) \otimes X + (\beta_0 \cdot \mathrm{A}ttn) \tag{5}$$

The attention mechanism not only optimizes the values of γ and β, but also dynamically adjusts different parts of the feature map, allowing the model to focus on important features more effectively and suppress redundant information. Through this combination, the AFT module can adaptively adjust the feature distribution based on global and local features, thereby improving the model's generalization ability and its ability to perceive the importance of task-related channels.

The insertion position of the AFT module is carefully designed to maximize its impact on feature learning. Specifically, AFT is deployed after the first convolutional layer of the ResNet12 architecture, and at the end of the 1st and 3rd residual blocks. These positions cover different levels of the network, allowing the AFT module to optimize low-level feature representation and strengthen high-level feature capture, thereby giving the model stronger global adaptability and local feature extraction capabilities.

Compared with traditional channel attention mechanisms (such as SE modules), AFT can not only adaptively adjust the importance of channel features, but also improve the flexibility and adaptability of the model to input data through affine transformation. Therefore, AFT shows stronger feature extraction capabilities in complex tasks such as fine-grained classification. Subsequently, the feature map extracted by the AFT module is input into the introduced FTP positioning module to remove background interference and extract target image features. Finally, the original image features and target image features are input into the feature alignment module for matching.

5. Metrics Module

In few-shot fine-grained classification tasks, accurately evaluating the similarity between image features is crucial to distinguishing small differences between categories. To improve soil classification performance, this paper proposes a Cross Cosine-Euclidean (CCE) module, which enhances the accuracy of feature similarity evaluation by combining cosine similarity and Euclidean distance.

Cosine similarity measures similarity by calculating the angle between feature vectors, focusing on the directionality of features. Compared with absolute eigenvalues, cosine similarity can still work stably when the size of feature vectors changes, especially in few-shot fine-grained classification tasks, it can effectively capture the similarity between images [9]. Euclidean distance focuses on the absolute difference between feature vectors, directly reflecting the distance of features in high-dimensional space, thereby capturing overall differences [10]. When the absolute value of the feature plays an important role in the classification task, Euclidean distance can provide additional information. The formulas for the two are as follows:

$$\mathrm{Cos}Sim(f_{train}, f_{test}) = \frac{f_{train} \cdot f_{test}}{\|f_{train}\| \|f_{test}\|} \tag{6}$$

$$Euclidean(f_{train}, f_{test}) = \|f_{train} - f_{test}\| \tag{7}$$

The core idea of the CCE module is to integrate the advantages of cosine similarity and Euclidean distance, and use their respective advantages to improve the accuracy of similarity calculation. It is particularly suitable for fine-grained classification tasks with few-shot. In order to balance the contribution of these two metrics, the CCE module introduces a learnable parameter α. By weighted combination of cosine similarity and Euclidean distance, the model can adaptively adjust their relative importance in different tasks. The final similarity calculation formula is:

$$S(f_{train}, f_{test}) = 10(\alpha \cdot \mathrm{Cos}Sim(f_{train}, f_{test}) - (1-\alpha) \cdot \mathrm{E}uclidean(f_{train}, f_{test})) \tag{8}$$

Among them, α is a learnable parameter used to dynamically adjust the contribution of cosine similarity and Euclidean distance.

The CCE module makes full use of the complementary advantages of the two measurement methods. Cosine similarity remains stable when the size of the feature vector fluctuates, while Euclidean distance captures the absolute difference of features and provides a more comprehensive similarity assessment. Through the learnable parameter α, the model is able to dynamically adjust the importance of the two measures based on task requirements and data distribution. For some fine-grained categories, the

ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

directionality of features may be more critical, while for other categories, the absolute difference in features may be more important. Through this adaptive metric fusion, the CCE module can flexibly respond to different similarity assessment needs and help the model accurately distinguish categories in the case of few-shot, thereby significantly improving classification performance.

6. Loss Function

This paper uses cross entropy as the classification loss function. The proposed framework contains two independent global classifiers, which are used to classify the basic categories of the features of the original image and the target image respectively. The formula is as follows:

$$\mathrm{L}_{global}^{\mathrm{raw}} = -\sum_{(x_i, y^g)\in t} y^g \log((soft\max(W_1 \cdot GAP(\Theta(x_i))))) \tag{9}$$

$$\mathrm{L}_{global}^{\mathrm{target}} = -\sum_{(\hat{x}_i, y^g)\in t} y^g \log((soft\max(W_2 \cdot GAP(\Theta(\hat{x}_i))))) \tag{10}$$

Where W_1 and W_2 are weight parameters of two global classifiers, GAP(·) represents global average pooling, and y^g is the true basic category label of the original image x^{raw} and the target image x^{target}.The final global classification loss $\mathrm{L}_{\mathrm{global}}$ can be expressed as:

$$\mathrm{L}_{\mathrm{global}} = p \cdot L_{global}^{raw} + q \cdot L_{global}^{target} \tag{11}$$

Among them, p and q are two weighting factors, and the global branch is only a part of the overall framework. Similarly, the local loss can be expressed as:

$$\mathrm{L}^{\mathrm{raw}} = -\sum_{i=1}^{|Q|} \log \frac{\exp(-\mathrm{cce}(A(C_j, F_i), F_i)))}{\sum_{j=1}^{N} \exp(-cce(A(C_j, F_i), F_i))} \tag{12}$$

$$\mathrm{L}^{\mathrm{target}} = -\sum_{i=1}^{|Q|} \log \frac{\exp(-\mathrm{cce}(A(\hat{C}_j, \hat{F}_i), \hat{F}_i)))}{\sum_{j=1}^{N} \exp(-cce(A(\hat{C}_j, \hat{F}_i), \hat{F}_i))} \tag{13}$$

Among them, $\hat{F}_i$ represents the feature $\hat{x}_i$ of the target query image, $\hat{C}_j$ is the corresponding prototype feature calculated by the target support image, and the final local few-shot loss L_{local} is expressed as follows:

$$\mathrm{L}_{local} = a \cdot L^{raw} + b \cdot L^{target} \tag{14}$$

Among them, a and b are weight factors that control the relative contribution of the original image and the target image in the local loss. In all experiments, these two parameters are set to 0.5, and the total loss function of the model is as follows:

$$\mathrm{L}_{total} = L_{global} + \lambda \cdot \mathrm{L}_{local} \tag{15}$$

Among them, λ is the weight factor, which is set to 0.1 in the experiment. It is worth noting that in the inference stage, only the local loss branch L_{local} is used to calculate the similarity of the query sample for classification.

ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

7. Experiment

7.1. Dataset

This paper collects soil image data from China Physical Geological Data Network and soil image data identified by professional inspectors to form a soil image dataset Soil. This dataset has a total of 42 soil categories, each category has 50 soil samples. In addition, this paper also uses the public datasets CUB-200-2011 and Stanford Dogs to verify the classification performance of the model. At the same time, following the previous research settings, the training set and validation set of the public dataset mini-imagenet are used to train the model, and the test set of the public dataset CUB-200-2011 and Stanford Cars is used to evaluate the generalization ability of the model. The detailed information of each dataset is shown in Table 1:

Table 1: Dataset.

Dataset	Class	Train	Val	Test
Soil	42	25	7	10
Mini-Cub	130	64	16	50
Mini- Cars	129	64	16	49
CUB-200-2011	200	100	50	50
Stanford Dogs	120	70	20	30

7.2. Experimental Details

In the experiment, resnet12 was used as the feature extraction network, the size of the input image was 84x84, the stochastic gradient descent method was used to optimize the model, the initial weight was 0.001, the number of training rounds was 90, and it was set to 0.0006 after 60 epochs, and then multiplied by 0.2 for every 10 subsequent rounds. The batch size was 4, and the graphics card V100-32GB (32GB) and the deep learning framework Pytorch1.8.1 were used for training.

7.3. Ablation Experiment

This paper uses the self-made Soil dataset as the basic dataset for ablation experiments, and evaluates the improvement effects of the feature conversion layer module and the metric module on the model. The experimental results are shown in Table 2:

Table 2: Ablation experiment.

Method	Soil	
	1shot	5shot
Base	57.14%	70.52%
Base+AFT	60.19%	72.15%
Base+CCE	59.34%	71.28%

The experimental results in Table 2 show that the use of the AFT module improves 3.05% and 1.63% in 1 shot and 5 shots respectively, and the use of the CCE module improves 2.2% and 0.76% in 1 shot and 5 shots respectively. Therefore, it can be concluded that the AFT module and the CCE module have a certain improvement on the classification ability of the model.

7.4. Comparative Experiment

In order to evaluate the generalization ability of the model, comparative experiments are conducted using the Mini-Cub and Mini-Car datasets. The experimental results are shown in Table 3:

Table 3: Comparative experiment.

Method	Mini-Cub		Mini-Cars	
	1shot	5shot	1shot	5shot
Relationnet	42.44%	57.77%	29.11%	37.33%
Matchingnet	35.89%	51.37%	30.77%	38.99%
GNN	45.69%	62.25%	31.49%	41.28%
Ours	47.62%	62.30%	31.79%	42.07%

From the experimental results in Table 3, it can be seen that the highest accuracy is shown on the

ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

Mini-Cub and Mini-Car datasets, verifying the generalization ability of the model.

In order to further verify the soil classification ability of the algorithm in this paper, the algorithm proposed in this paper is compared with other algorithms in recent years. The experimental results are shown in Table 4:

Table 4: Model classification comparison experiment.

Method	CUB-200-2011		Stanford Dogs		Soil	
	1shot	5shot	1shot	5shot	1shot	5shot
Protonet	63.44%	83.17%	41.61%	76.78%	56.64%	70.33%
Maml	54.73%	75.75%	62.13%	77.91%	58.27%	70.72%
Relationnet	70.92%	84.90%	61.21%	80.27%	48.26%	69.75%
Matchingnet	60.52%	75.29%	55.30%	67.50%	55.96%	69.19%
Baseline	66.52%	83.31%	59.19%	78.21%	57.14%	70.52%
Ours	77.87%	87.36%	63.61%	81.50%	60.87%	72.43%

From the results in Table 4, our model has a classification accuracy of 77.87% and 87.36% on the CUB-200-2011 dataset, 63.61% and 81.50% on the Stanford Dogs data, and 60.87% and 72.43% on the Soil dataset, all of which have reached the highest classification accuracy among the comparison models. The experimental results show that the algorithm proposed in this paper effectively improves the performance of the model in the soil image classification task.

In order to more clearly demonstrate the model's ability to classify soil samples, five categories were selected from the test set of the Soil dataset as new test sets, namely calcareous mudstone, calcareous sandstone, clay, quartz diorite, and silty clay. The test results are shown in Table 5:

Table 5: Soil classification results.

Method	Soil	
	1shot	5shot
Ours	60.65%	72.02%

The classification accuracy of each category is shown in Figure 2:

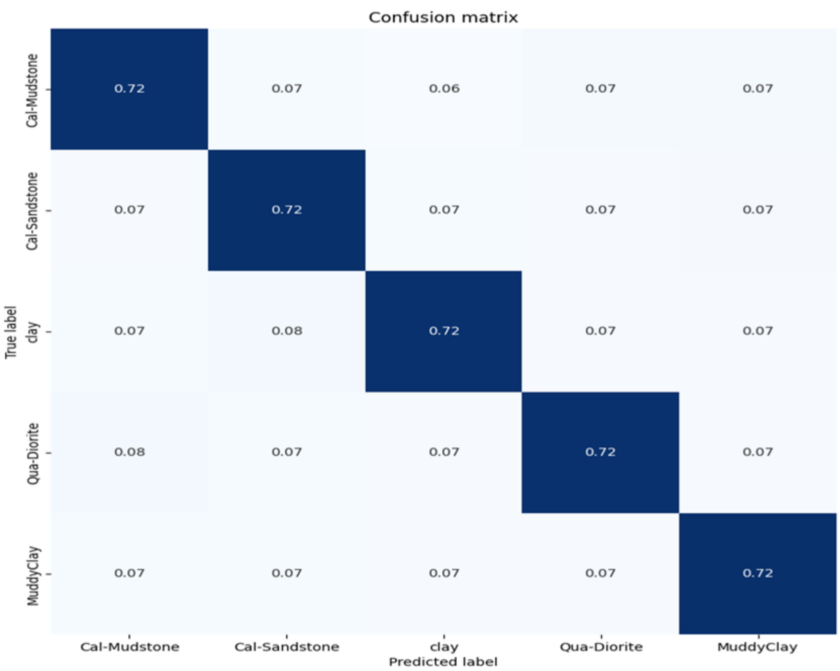

Figure 2: Confusion Matrix.

8. Conclusions

This paper proposes a dual metric module that uses feature transformation and combines cosine similarity and Euclidean distance to enhance fine-grained few-shot soil classification capabilities. It simulates changes in feature distribution through affine transformation and improves the robustness of

ISSN 2616-5775 Vol. 7, Issue 10: 89-96, DOI: 10.25236/AJCIS.2024.071013

the model, especially in the case where there are significant differences between samples, the classification accuracy is significantly improved. At the same time, the CCE module combines the two measurement methods of cosine similarity and Euclidean distance, making full use of their complementary advantages in different feature evaluations to further improve the Classification performance between fine-grained categories.

Experimental results show that the method proposed in this article not only has significant advantages when dealing with fine-grained classification problems, but also enhances the performance of the model under few-shot conditions. Future work can further study the impact of different types of feature transformations on model performance and explore more ways to combine enhancement strategies with existing methods to continue to optimize the performance of classification models.

References

[1] Hou D, Bolan N S, Tsang D C W, et al. Sustainable soil use and management: An interdisciplinary and systematic approach[J]. Science of the Total Environment, 2020, 729: 138961.

[2] Wang D H, Zhou W, Li J, et al. Exploring misclassification information for fine-grained image classification[J]. Sensors, 2021, 21(12): 4176.

[3] Liu W, Tong L, Sun Y, et al. Automatic soil classification method from CPTU data based on convolutional neural networks[J]. Bulletin of Engineering Geology and the Environment, 2024, 83(8): 1-20.

[4] Alshahrani H, Alkahtani H K, Mahmood K, et al. Chaotic Jaya optimization algorithm with computer vision-based soil type classification for smart farming[J]. IEEE Access, 2023, 11: 65849-65857.

[5] Wang H, Frank E, Pfahringer B, et al. Feature extractor stacking for cross-domain few-shot learning[J]. Machine Learning, 2024, 113(1): 121-158.

[6] Sa L, Yu C, Ma X, et al. Attentive fine-grained recognition for cross-domain few-shot classification[J]. Neural Computing and Applications, 2022, 34(6): 4733-4746.

[7] Zhong J, Gu K, Jiang H, et al. A fine-tuning prototypical network for few-shot cross-domain fault diagnosis [J]. Measurement Science and Technology, 2024, 35(11): 116124.

[8] Zangana H M, Mohammed A K, Mustafa F M. Advancements and Applications of Convolutional Neural Networks in Image Analysis: A Comprehensive Review[J]. Jurnal Ilmiah Computer Science, 2024, 3(1): 16-29.

[9] Gao C, Li W, He L, et al. A distance and cosine similarity-based fitness evaluation mechanism for large-scale many-objective optimization[J]. Engineering Applications of Artificial Intelligence, 2024, 133: 108127.

[10] Dharsini S V, Razak M A, Modi S, et al. Captioning based image using Euclidean distance and resNet-50[C]//2022 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI). IEEE, 2022, 1: 1-5.