A Hyperspectral Remote Sensing image classification Method Based on Semi-supervised Learning

Tian Jie[1], Wu Zongyi[2]

[1] College of Resources, Environment and Materials, Guangxi University;

tj18005005831@outlook.com

[2] School of Arts and Social Sciences, Hong Kong Metropolitan University; jamik1998@qq.com

Abstract. Hyperspectral remote sensing technology is a technology that utilizes spectral imager on flying carriers such as satellites, airplanes, and drones to acquire information about the earth's surface. With the development of deep learning, the application of neural networks in hyperspectral image classification has attracted much attention. However, hyperspectral image classification faces problems such as high labeling sample costs and many redundant samples. Existing methods still have shortcomings in utilizing unlabeled samples and improving classification performance. To solve these problems, this study addresses the critical challenges in hyperspectral image classification, namely high labeling costs and spectral redundancy, by proposing a novel semi-supervised learning framework. The model integrates three key components: (1) a 3D autoencoder for joint spectral-spatial feature extraction, (2) an ECA attention mechanism for channel-wise feature enhancement through adaptive weight learning, and (3) a Siamese network with random sample pairing for supervised feature refinement. Unlike conventional approaches, our framework establishes a synergistic mechanism between unsupervised feature enhancement and supervised correction, connected through innovative skip connections. Experimental validation demonstrates superior performance, achieving 82.3% OA (κ=0.74) on PaviaU and 87.4% OA (κ=0.82) on Salinas datasets, significantly outperforming existing methods while reducing dependence on labeled samples. The results confirm the model's effectiveness in improving feature discriminability and classification accuracy for hyperspectral imagery.

Keywords: Hyperspectral image classification; Semi-supervised Learning; Siamese Neural Network; Attention Mechanism; Autoencoder

1.Introduction

Hyperspectral Remote Sensing Image (Hyper Spectral Image, HSI) typically contains hundreds of spectral bands ranging from the visible to near-infrared spectrum. These spectral bands provide rich spectral and spatial information, effectively reflecting Ground Object Features such as landscape architecture, topography, rivers, and lakes 错误!未定义书签。错误!未找到引用源。. Hyperspectral image classification, as a fundamental task in processing hyperspectral images, plays a critical role in various fields, such as vegetation disease and pest analysis[1]错误!未找到引用源。, soil composition detection 错误!未找到引用源。, and ecosystem monitoring[4]. However, traditional machine learning methods usually require manual feature extraction. With the increasing volume of HSI data and application scenarios, the classification performance of these traditional methods tends to be limited[6]. The success of Deep Learning largely relies on a large number of Labeled Samples, but in HSI, obtaining Labeled Samples is time-consuming and labor-intensive, requiring significant time and computational cost. Many Labeled Samples are redundant, providing duplicate samples with similar or identical information. Therefore, reducing the reliance on Labeled Samples and efficiently mining valuable samples from datasets while eliminating redundant ones has become a research hotspot in current Hyperspectral image classification methods.

Makantasis et al.[7] used Principal Component Analysis (PCA) to project hyperspectral image data into a three-channel tensor and applied a two-dimensional Convolutional Neural Network for classification. Slavkovikj et al.[8]proposed another similar approach, compressing the spatial dimension of hyperspectral image data while retaining the spectral dimension to reduce the three-dimensional data into a two-dimensional image, which is then processed using a Convolutional Neural Network. Subsequently, the Semi-supervised Support Vector Machine and Generative Semi-supervised Classification Method were proposed. In addition to conventional semi-supervised learning, some researchers introduced the concept of few-shot learning and applied it to the field of Hyperspectral image classification [8]. Gao et al. 错误!未找到引用源。proposed a Deep Relation Network for few-shot learning in hyperspectral images, while Li et al. [9]applied the Attention Mechanism for information transfer and

proposed a Deep Cross-domain Few-shot Learning method. These few-shot learning approaches mainly focus on cross-domain information transfer, leveraging known information to assist in learning information from unknown classes.

The model proposed in this paper, based on autoencoders, attention mechanisms, and Siamese network, is a Semi-supervised Learning Model[11]. It can fully utilize a large number of low-cost unlabeled samples to train the autoencoder for feature extraction, and use a small number of labeled samples to train the Siamese network to enhance Feature Extraction, reduce the use of labeled samples, and lower costs.

Few-shot Learning methods often focus on Cross-domain Information Transfer and do not fully utilize unlabeled data for feature correction. To address these issues, this paper proposes an innovative Semi-supervised Learning Model that integrates autoencoders, ECA mechanism, and Siamese networks. The innovations of this model are:

1) Using the 3-D Autoencoder method to train on a large number of unlabeled samples to learn the unsupervised features of hyperspectral image samples. Unlike traditional semi-supervised learning methods, our model can simultaneously extract the Spectral-spatial Features of hyperspectral images through the 3-D Autoencoder without information loss. The encoder part of the autoencoder learns the unsupervised features in the data, providing a basis for subsequent Feature Enhancement and correction.

2) Using the ECA attention mechanism to construct an Attention Layer to help the neural network better focus on the most important spectral channels or features in hyperspectral images during the learning process, for enhancing the hyperspectral image sample features extracted from the encoder in the 3-D Autoencoder. The ECA mechanism specifically addresses the channel redundancy problem in hyperspectral data. By modeling the relationships between channels, it adaptively learns the weights between each channel, allowing the network to focus more on important feature channels, suppress irrelevant channels, and improve classification performance. This mechanism significantly enhances the representation capability of features, providing more accurate input for subsequent feature correction.

3) A Siamese network is trained using limited Labeled Samples and the Random Sample Pair

Generation method to correct the sample features extracted from the Attention Layer. A Residual Module is used to form the final sample features, which are then fed into the classifier for classification. The synergistic mechanism between the Siamese network and the Autoencoder achieves joint optimization of unsupervised features enhancement and Supervised Features correction. Through the training of the Siamese network, the extracted features are corrected using a small number of Labeled Samples, thereby effectively improving the separability and discriminability of the features. The introduction of the Residual Module not only accelerates the training process but also improves the accuracy of the model.

2. Methods for Hyperspectral image classification Based on Semi-supervised Learning

Currently, most Hyperspectral image classification methods[12]based on Semi-supervised Learning effectively combine Unsupervised Learning and Supervised Learning. Common Unsupervised Learning methods include autoencoders, Clustering Method, and Contrastive Learning. Common Supervised Learning methods include Support Vector Machine (SVM), Multilayer Perceptron (MLP)错误!未找到引用源。, Siamese network, and Generative Adversarial Network. This paper combines the autoencoder from Unsupervised Learning and the Siamese network from Supervised Learning to form a Semi-supervised Learning method for Hyperspectral image classification, addressing issues such as the cost of Labeled Samples.

2.1 Unsupervised Feature Extraction Based on 3D Autoencoder

2.1.1 Basic Principles of Autoencoder

Autoencoder is an unsupervised learning model, generally composed of an encoder and a decoder. The encoder encodes the input into a hidden representation h, and the decoder decodes the hidden representation h into an output. The model uses the input data itself as supervision to guide the neural network in learning a latent mapping 错误!未找到引用源。. The training objective of the autoencoder is to make the input and output as similar as possible, which can be expressed as shown in Equation 1.

$$D(X, X^R) = min|X, X^R|^2 \quad (1)$$

2.1.2 Unsupervised Feature Extraction Based on 3D Autoencoder

According to the latest research on autoencoders 错误!未找到引用源。, this paper proposes a Unsupervised Feature Extraction method based on 3D Autoencoder to process 3-D Hyperspectral Image Cube. The 3-D Hyperspectral Image Cube is represented as $X \in R^{H \times W \times D}$, where H, W, and D represent height, width, and number of spectral bands, respectively. At the same time, the three-dimensional neighboring region centered at the spatial location (i, j) is cropped as a Spectral-Spatial Combined Sample. and represent the size of the local region. In this way, a neighboring region dataset is constructed, where i = 1, ..., H and j = 1, ..., W. The autoencoder consists of an encoder and a decoder module, and the hidden representation can carry useful information from the Hyperspectral Image Cube. To simultaneously generate Joint Spectral-Spatial Features from the HSI[15], a 3D Autoencoder approach is introduced, as shown in Figure 1.

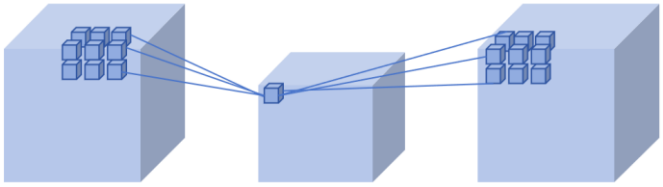

Figure 1. 3D Autoencoder module

2.2 Supervised Feature Extraction Based on Siamese network and Attention Layer

2.2.1 Basic Principle of Siamese network

The Siamese network consists of two highly similar neural networks, hence also known as a twin network. The Siamese network takes two input samples, referred to as a sample pair. Training a Siamese network requires a labeled classification dataset. We need to construct positive samples and negative samples using the training set. Each time, two samples are drawn from the dataset to form a sample pair. If the pair belongs to the same class, it is considered a

positive samples pair and labeled as 1; if they belong to different classes, it is a negative samples pair and labeled as 0, as shown in Figure 2.

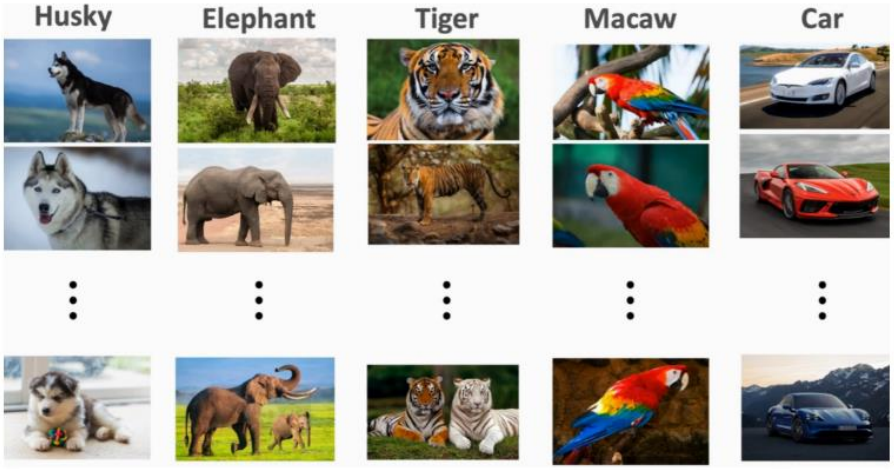

Figure 2. Training Dataset Setup of Siamese network

Assume we use a Convolutional Neural Network to extract features, denoted as $f(x)$, as shown in Figure 2-3. The input to the Siamese network is a sample pair, which is processed by the Convolutional Neural Network to extract the Feature Vector. The Feature Vector extracted from the first sample of the input pair is denoted as$h_1 = f(x_1)$, and similarly, the Feature Vector from the second sample is denoted as$h_2 = f(x_2)$. A vector Z is then used to represent the difference between the Feature Vectors of the two input samples, $Z = |h_1 - h_2|$. This Z vector is further processed by several Fully Connected Layers to produce a scalar output. Then, a sigmoid activation function, ReLU activation function, or other activation function is applied, resulting in a real number between 0 and 1. This output value indicates the similarity between the two images: for a positive samples pair, the output is close to 1; for a negative samples pair, the output is close to 0.We aim for the neural network's output to be close to the label, using the difference between the label and the prediction as the loss function[17]. The loss function can be the cross entropy between the label and the prediction, which measures the discrepancy between them.

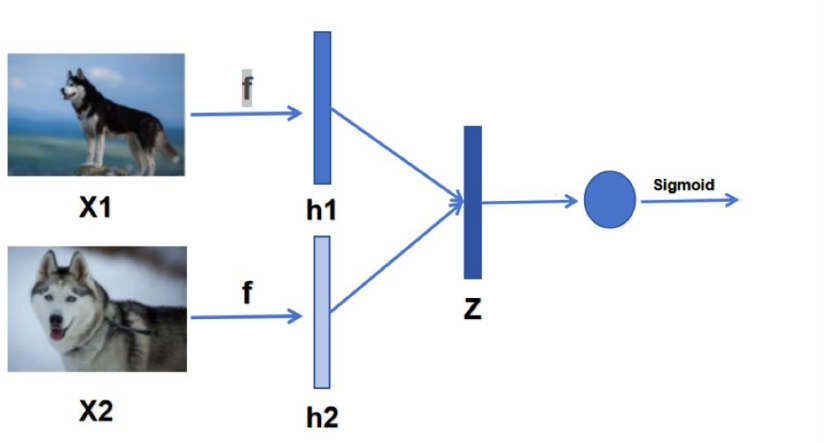

Figure 3. Schematic diagram of the Siamese network training process

2.2.2 Constructing the Attention Layer Using the Efficient Channel Attention Mechanism

The efficient channel attention mechanism is a type of self-attention mechanism that enhances feature representation by modeling the inter-channel relationships. Hyperspectral images contain a large amount of spectral information, with each pixel having data across multiple spectral bands[18], providing richer data features for image classification. The efficient channel attention module can be embedded after the convolutional layer to enhance the discriminative capability of features. This module can adaptively learn the weights among different channels, enabling the network to focus more on important feature channels while suppressing irrelevant ones, thereby improving classification performance. In hyperspectral image classification, where labeled samples are limited, the efficient channel attention module helps the model learn more effectively from the limited labeled data.

In this work, the efficient channel attention mechanism is used to construct an attention layer, which is placed after the encoder of the autoencoder. The features extracted by the autoencoder are passed into the attention layer for feature enhancement, and the enhanced features are then fed into the Siamese network for refinement, further strengthening feature extraction[19].

2.2.3 Random Sample Pair Generation Strategy

The input to the Siamese network in this work adopts the random sample pair generation strategy, which is studied as follows. For hyperspectral image data, we assume there are C

ground object classes and each class has *N* training samples (although in practice the number of samples per class may vary, the final conclusion still holds). The training data for the *c*-th class (where *c* = 1, ..., C) is denoted as $X_C = \{x_{c,1}, \ldots, x_{c,N}\}$, and the corresponding label data is denoted as $Y_C = \{y_{c,1}, \ldots, y_{c,N}\}$. For each $x_{c,n} \in R^{H' \times W' \times D}, n = 1, \ldots, N$, it corresponds to the local region centered at the central pixel, forming the entire training dataset $A = \{(X_1, Y_1), \ldots, (X_C, Y_C)\}$.

A sample pair is constructed $p_{\mathrm{ij}} = \{x_i, x_j\}, i, j = 1, \ldots, C \times N$. The label($p_{\mathrm{ij}}$) for the pair is defined as shown in Equation 2.

$$\mathrm{label}(p_{\mathrm{ij}}) = \begin{cases} 0, y_i \neq y_j \\ 1, y_i = y_j \end{cases} \quad (2)$$

Under the construction method of Equation 2, all possible combinations of sample pairs are obtained, denoted as $P = \{p_{ij}, label(p_{ij})\}, i, j = 1, \ldots, C \times N$. Clearly, the number of elements in *P* can be calculated as shown in Equation 3.

$$E = \frac{1}{2} CN(CN - 1) \quad (3)$$

Represents the total number of distinct combinations of selecting 2 samples from *C*×*N* samples. This can be viewed as a combinatorial problem 错误!未找到引用源。. For the positive samples pair dataset E_{pos}, which contains only sample pairs$\mathrm{label}(p_{\mathrm{ij}}) = 1$ from the same class, the number of elements E_{pos}in is calculated as follows: the first selection has *C*×*N* possibilities, and the second selection, which must be from the same class as the first, has only N−1 possibilities. The calculation is expressed in Equation 4.

$$E_{pos} = \frac{1}{2} CN(N - 1) \quad (4)$$

For Negative Samples pairs, the first selection has C×N possibilities, and the second selection, which must be from a different class than the first, has N×(C−1) possibilities. The calculation is expressed in Equation 5.

$$E_{neg} = \frac{1}{2} CN^2(C - 1) \quad (5)$$

Dividing Equation 5 by Equation 4 yields Equation 6.

$$\frac{E_{neg}}{E_{pos}} = \frac{\frac{1}{2}CN^2(C-1)}{\frac{1}{2}CN(N-1)} = \frac{N(C-1)}{N-1} \approx C-1 \quad (6)$$

From the above calculations, it can be seen that the number of Negative Samples pairs is significantly greater than the number of positive samples pairs[20]. If such sample pairs are used in training the model, the model is more likely to classify pairs as negative, which increases the gradient and slows down the training speed per epoch. To address this issue, a random sample pair generation method is proposed to balance the number of positive samples and negative samples pairs, thereby reducing training time. For any sample in the training set, randomly select one sample from the same class and one from a different class to form a positive pair and a negative pair, respectively. The uniform sampling strategy is used here. This is done in each training iteration $E_{\text{pos}} = E_{\text{neg}}$.

2.2.4 Supervised Feature Extraction Based on Siamese network and Attention Layer

Since unsupervised features may lack separability, making it difficult to effectively classify data, this paper designs a Siamese network and a limited labeled dataset to improve the separability of features. The Siamese network constructed in this paper takes random sample pairs as input. According to the characteristics of the Siamese network, x_i and x_j are processed separately. The following is the processing ofx_i , and the processing of x_j is similar. After any sample is processed by the trained 3-D Autoencoder, i=1,..., C×N, unsupervised features $h_u^i = f_e(x_i)$ are obtained. For the features extracted from the autoencoder, this paper uses the Attention Layer to further enhance the Expressive Power of these features, suppress irrelevant channels, and improve classification performance, to obtain more accurate features: $h_{\text{ue}}^i = f_e(x_i)$. A Siamese network $f_r(h_{ue}^i)$ is connected after the Attention Layer to process the unsupervised features extracted and enhanced by the autoencoder through the Attention Layer: $h_s^i = f_r(h_{\text{ue}}^i)$, as well as the supervised features extracted by the Siamese network.

Therefore, the feature module of the Siamese network f_φ （x_i）used in this paper consists of a strengthened feature module $f_{\text{ue}}(x_i)$ combining Autoencoder and Attention Layer, and a correction module $f_r(h_{ue}^i)$ with skip connections. The parameters of $f_{\text{ue}}(x_i)$ are fixed after

being trained in an unsupervised manner. We assume that each class has an implicit center point, and supervised features h_s^i can make samples closer to their own center point. For the feature fusion method between the correction module and the strengthened feature module, this paper chooses to add them together, which allows the two features to be simply added together, enabling the network to simultaneously consider feature information from both paths. The calculation expression is shown in Equation 7.

$$f\varphi(x_i) = h_{ue}^i + f_r(h_{ue}^i) \quad (7)$$

In this way, the correction module becomes Residual Module, where $f_r(h_{\mathrm{ue}}^i) = f\varphi(x_i) - h_{\mathrm{ue}}^i$. The Residual Module introduces skip connections, which allow gradients to directly skip one or more layers, thereby maintaining the information transfer of gradients and alleviating the problem of vanishing or exploding gradients. At the same time, it can also correct features to improve the training speed and accuracy of the network.

After extracting the features of the input sample pairs, a metric module is used to calculate their similarity. To increase the robustness of the module's hyperparameters, this paper uses a binary Classifier that replaces the metric module to predict the matching probability of sample pairs.

3 Experimental Datasets and Model Methods

3.1 Experimental Datasets

3.1.1 Introduction to hyperspectral image Dataset PaviaU Dataset

The Pavia University dataset is a portion of hyperspectral data captured over the city of Pavia, Italy, in 2003 by the German airborne reflective optics spectrometer imager 错误!未找到引用源。. This hyperspectral imager continuously captures 115 spectral band within the wavelength range of 0.43–0.86μm, with a spatial resolution of 1.3 meters.

Figure 4. Sample Image

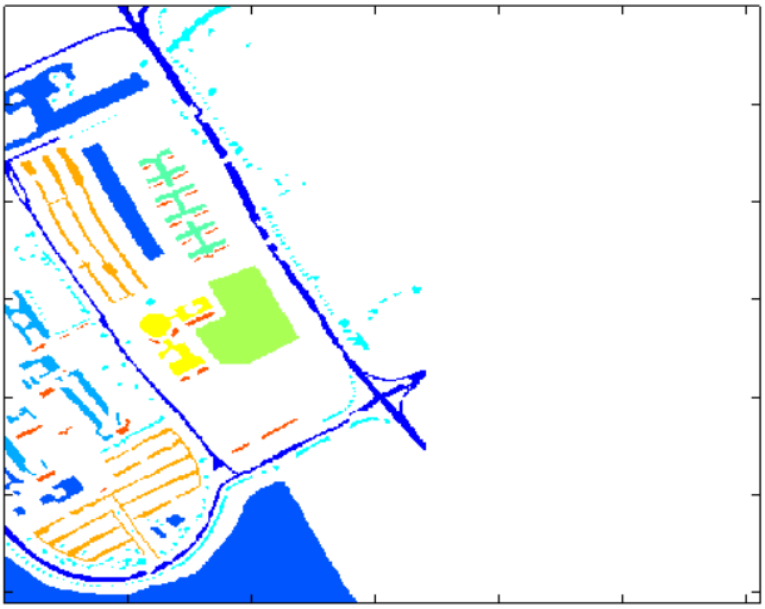

Figure 5. Correctly Classified Image

Table 1. Categories and Corresponding Sample Counts in PaviaU Dataset

No.	Category	Sample Count
1	Asphalt	6631
2	Meadows	18649
3	Gravel	2099
4	Trees	3064
5	Painted metal sheets	1345
6	Bare Soil	5029

No.	Category	Sample Count
7	Bitumen	1330
8	Self-Blocking Bricks	3682
9	Shadows	947

3.1.2 Introduction to hyperspectral image Dataset Salinas Dataset

This scene was captured by the AVIRIS sensor over the Salinas Valley in California, consisting of 224 spectral band and featuring high spatial resolution (3.7 meters per pixel). It represents imagery of the Salinas Valley in California, USA, covering an area of 512 rows by 217 samples.

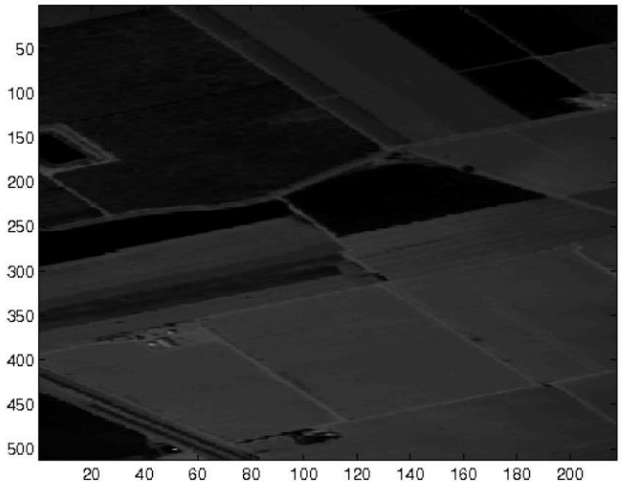

Figure 6. Sample Image

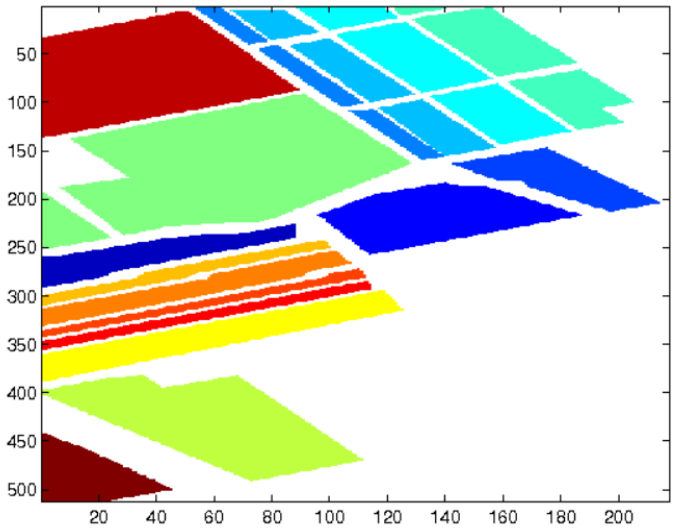

Figure 7. Correctly Classified Image

Table 2 Salinas Dataset Categories and Corresponding Sample Counts

No.	Category	Sample Count
1	Green weeds1	2009
2	Green weeds2	3726
3	Fallow Land	1976
4	Rough fallow	1394
5	Leveled Fallow Land	2678
6	Stubble	3959
7	Celery	3579
8	Untrained grapes	11271
9	Vineyard untrained	6203
10	Corn	3278
11	4-Week-Old lettuce	1068
12	5-Week-Old lettuce	1927
13	6-Week-Old lettuce	916
14	7-Week-Old lettuce	1070
15	Vineyard vertical trellis	7268
16	Vineyard trellis	1807

4 Results and Analysis

This paper constructs models using two Hyperspectral Benchmark Image Dataset datasets: PaviaU and Salinas, to evaluate the model's Overall Accuracy (OA) and Kappa. OA is defined as the proportion of correctly classified samples to the total number of samples. Kappa is a consistency metric based on the Confusion Matrix, $\text{kappa} = (p_o - p_e)/(1 - p_e)$ where p_o accounts for Overall Accuracy and p_e a hypothetical probability representing the likelihood of random agreement. OA directly reflects the proportion of correct classifications; however, it is possible for OA to be high while the classification accuracy for certain classes remains low. Therefore, the Kappa coefficient is used to address this issue. If one class has poor classification results, the Kappa value decreases, making Kappa a more robust evaluation metric than OA. In the experiments, 10 Labeled Samples from each class in the dataset were randomly selected to create the training set. The remaining Labeled Samples were used to form the test set.

4.1 Model Training Parameter Settings

Factors affecting the performance of deep learning models include data quality and quantity, model depth, the number of hidden neurons, the size of the receptive field, batch size, and others. This paper mainly considers the settings of the number of Hidden Neurons and the receptive field size. An excessive number of Hidden Neurons increases model complexity and computational cost, and may lead to Overfitting, where the model performs well on the training set but has poor Generalization Ability on the test set. Too few Hidden Neurons reduce the amount of information stored, which may affect the information transfer process during training. Therefore, keeping other hyperparameters unchanged, experiments were conducted on PaviaU and Salinas Dataset with hidden unit sizes of 64, 128, and 256. Selected OA and Kappa values are presented as references for classification accuracy.

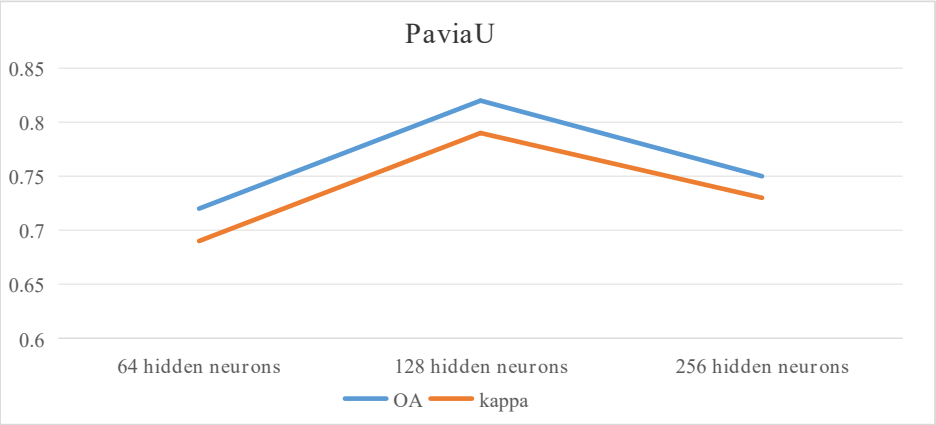

Figure 8. shows the performance on the PaviaU Dataset with different numbers of Hidden Neurons set.

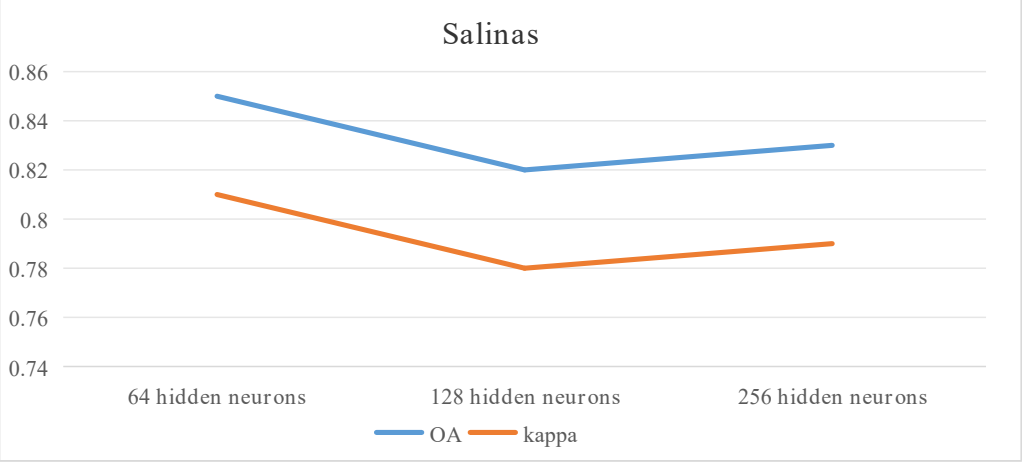

Figure 9. shows the performance on the Salinas Dataset with different numbers of Hidden Neurons set

According to the data tables in Figures 8 and 9, when conducting experiments on the PaviaU Dataset, the optimal number of Hidden Neurons is approximately 128. Similarly, for the Salinas Dataset, the optimal number is approximately 64.

The result region contains spatial information about the central pixel, which is an important parameter for model performance. When the region is too small, the information from neighboring areas is too limited, making it difficult for the model to capture sufficient information, thereby reducing performance. If the region is too large, it may include pixels from other classes, leading to misclassification. This introduces noise and increases the variability among pixels of the same class, especially at the boundaries between different classes.

Therefore, keeping other hyperparameters unchanged, preliminary experiments were conducted with region sizes of 7×7, 13×13, and 19×19 on the PaviaU and Salinas Dataset datasets. The resulting Overall Accuracy and Kappa values were used as references for classification accuracy.

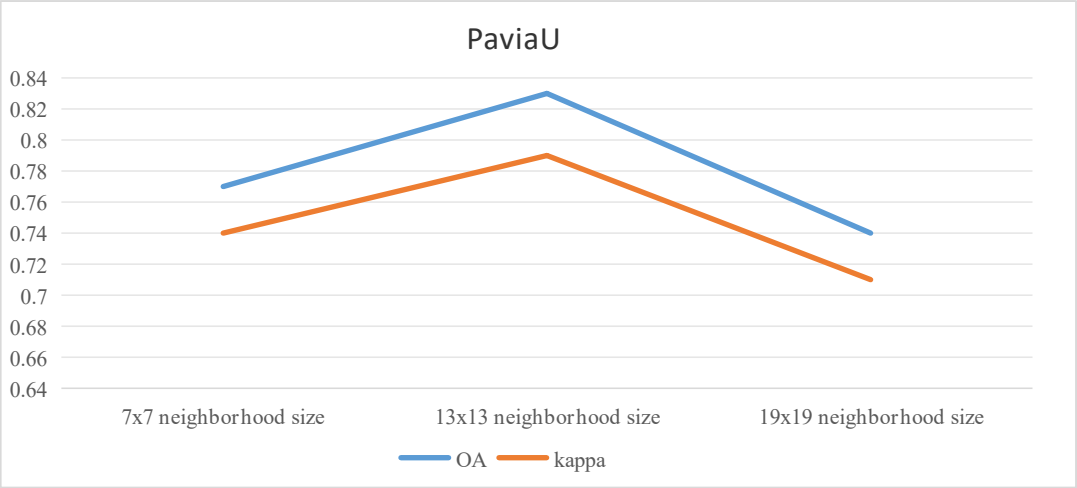

Figure 10. shows the performance of the method with different report region sizes on the PaviaU dataset.

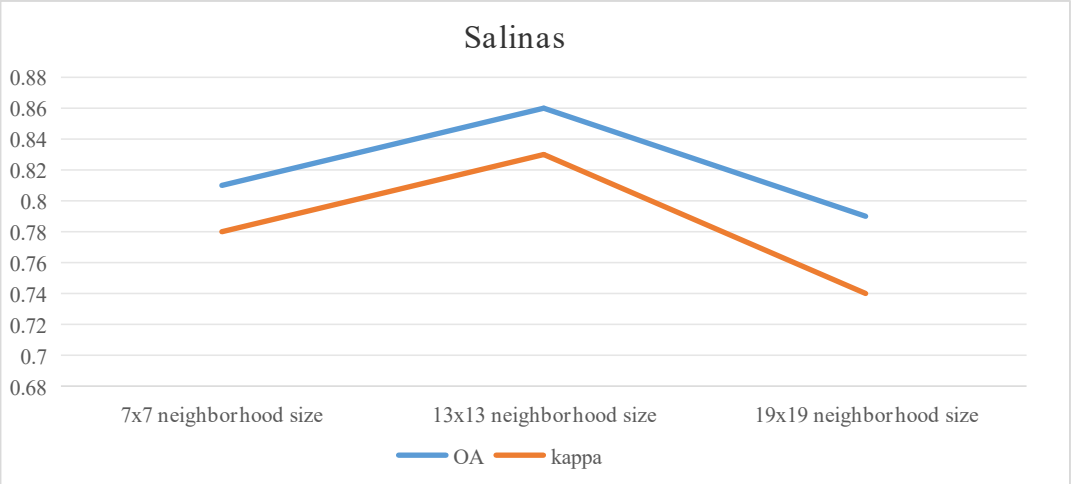

Figure 11. shows the performance of the proposed method with different report region sizes on the Salinas dataset

According to the data tables in Figures 10 and 11, when conducting experiments on the PaviaU and Salinas datasets, the optimal region size is approximately 13×13.Based on the above results, the parameter settings for the table model configuration are summarized in Table 3.

Table 3 shows the parameter settings for the report region size on two datasets PaviaU Dataset Salinas Dataset.

	PaviaU Dataset	**Salinas Dataset**
Hidden Neurons count	128	64
Spatial region size	13*13	13*13

During the training of the 3D Autoencoder, 20% of the data is used for testing and 80% for training. The model parameters are optimized using the Adam Optimizer with a learning rate set to 1e-4. During training, the objective of the model is to reduce the Mean Squared Error (MSE) between the predicted output and the actual output by adjusting the parameters. MSE is a commonly used loss function, particularly suitable for regression problems. It measures the difference between the model's predicted values and the actual values. By minimizing MSE, the model can better fit the training data, thereby improving its performance.

A sparse network is trained using the previously described random sample pair generation method, with a learning rate of 1e-3 using the Adam Optimizer and the Cross Entropy Loss Function. To preserve the unsupervised features extracted by the autoencoder, the parameters

of the encoder module are kept fixed. Finally, a simple logic is used to classify the features.

4.2 Ablation Analysis

3D Autoencoder and the proposed method in the paper: The proposed method enhances the unsupervised features extracted by the autoencoder using the ECA (Efficient Channel Attention) Mechanism, and then combines them with the Supervised Features extracted by the Siamese network. Unlike other approaches that transmit Hyperspectral Image samples to two different networks to separately create unsupervised and Supervised Features and then fuse them, this paper enhances the extracted unsupervised features through ECA attention feature training, and then selects the Supervised Features extracted from the adjacent Siamese network to correct the unsupervised features, in order to improve the modifiability of the features. This part is completed by the module. To validate the improvements of the method proposed in the paper over 3D Autoencoder, both 3D Autoencoder and the proposed method were tested, as shown in Table 4.

Table 4 Performance comparison of the proposed method and 3D Autoencoder on two datasets

Dataset	Method	OA（%）	Kappa（%）
PaviaU Dataset	3D Autoencoder	79	75
	The method	81	77
Salinas Dataset	3D Autoencoder	86	82
	The method	90	86

As shown in the data table, the proposed method in the paper achieves a 3-percentage-point improvement in Overall Accuracy and a 2-percentage-point improvement in Kappa over 3D Autoencoder on the PaviaU Dataset. On the Salinas Dataset, it achieves a 4-percentage-point improvement in Overall Accuracy and a 4-percentage-point improvement in Kappa. This demonstrates that the proposed method, which integrates the Siamese network and Efficient Channel Attention mechanism, effectively addresses the shortcomings in unsupervised features and correction modules.

4.3 Comparative Experiments

This paper selects SAE-LR (Stacked Autoencoder with Logistic Regression)[43], Two-

CNN[44], 3DCAE (3D Convolutional Autoencoder)[45], and the graph convolutional network GCN (Graph Convolutional Network)[46] for comparison.

SAE-LR (Stacked Autoencoder with Logistic Regression) is an autoencoder composed of a Fully Connected Layer network. Before classification, it preprocesses the original spectral and spatial vectors using the PCA (Principal Component Analysis) Technique, and fuses them into a joint spectral-spatial vector. The parameters of the Encoder and decoder modules in this method are bundled together. Two-CNN is a dual-branch model that separately extracts Spectral Features and Spatial Features, and then fuses them. To perform well with limited training samples, this method uses Transfer Learning to pre-train the model, transferring knowledge from the source data to the target dataset. 3DCAE (3D Convolutional Autoencoder) is a 3D CNN autoencoder trained using a large number of unlabeled samples, and then classifies the extracted features using a Support Vector Machine. Since graph data has stronger representation capabilities than grid data, GCN (Graph Convolutional Network) is introduced for Hyperspectral image classification. During training, the graph data includes both labeled and unlabeled nodes. It leverages both labeled and unlabeled information, making it a Semi-supervised method.

The above methods are evaluated on the PaviaU Dataset, with Overall Accuracy and Kappa selected as references for classification accuracy.

Table 5 Performance of different methods on the PaviaU Dataset

	SAE-LR	**Two-CNN**	**3DCAE**	**GCN**	**Proposed Method**
OA(%)	72.4%	76.05%	72.35%	64.36%	82.3%
Kappa	0.54	0.69	0.52	0.65	0.74

The experimental results show that the proposed method outperforms the other four methods in both metrics. SAE-LR (Stacked Autoencoder with Logistic Regression) and 3DCAE (3D Convolutional Autoencoder) are both models based on Encoder, with the difference being that the former is a fully connected network while the latter is a 3-D convolutional network. Their poor performance on the PaviaU Dataset demonstrates that relying solely on Encoder to obtain unsupervised features without further refinement makes it difficult to improve classification

accuracy. This also validates the effectiveness of enhancing Feature Extraction using the Siamese network and ECA (Efficient Channel Attention) Mechanism as proposed in this paper. The performance of GCN (Graph Convolutional Network) depends on the meticulous construction of graphs and node features; poor graph construction and simplistic node feature generation can reduce classification accuracy.

The above methods were evaluated on the Salinas Dataset, using Overall Accuracy and Kappa as classification accuracy metrics.

Table 6 Performance of different methods on the Salinas Dataset

	SAE-LR	**Two-CNN**	**3DCAE**	**GCN**	**Proposed Method**
OA(%)	68.2%	81.25%	75.72%	86.2%	87.4%
Kappa	0.58	0.71	0.70	0.79	0.82

The experimental results show that, unlike the PaviaU Dataset, all models exhibit significant improvements in both Overall Accuracy and Kappa on the Salinas Dataset. This can be attributed to the regular distribution of ground objects in the Salinas Dataset, which facilitates classification. Compared to the SAE-LR (Stacked Autoencoder with Logistic Regression) and 3DCAE (3D Convolutional Autoencoder) methods, the proposed method shows a clear improvement, further proving the effectiveness of using the Siamese network and ECA (Efficient Channel Attention) Mechanism to refine and enhance unsupervised features.

Overall, the proposed model demonstrates certain advantages over the other four methods in terms of Overall Accuracy and Kappa Coefficient on both the PaviaU and Salinas Dataset.

5 Conclusion

The main work of this study is the construction of a Semi-supervised Learning Model for Hyperspectral image classification. The model consists of an autoencoder, an Attention Layer, and a Siamese network. The autoencoder is trained on a large amount of unlabeled data to learn unsupervised features of the samples. These unsupervised features are then enhanced through the Attention Layer to form enhanced features. Subsequently, the Siamese network is trained using a small amount of labeled data and randomly generated sample pairs to learn supervised

features of the samples. The enhanced features are then input into the trained Siamese network, where they are refined using the supervised features to produce corrected features. Finally, the corrected features and enhanced features are fused via skip connections to form the final latent features of the samples. These final latent features, along with the HSI samples, are input into a logistic Classifier for classification to obtain the final results. As a Semi-supervised Learning Model, the model leverages a large number of unlabeled samples and a small number of labeled samples, effectively addressing the high labeling cost issue in Hyperspectral image classification.

1) The unsupervised features extracted by the 3D Autoencoder are input into the Attention Layer for feature enhancement. The enhanced features are then input into the Siamese network, where they are refined using the supervised features learned by the Siamese network. A random sample pair generation method is used to construct input sample pairs for the Siamese network, reducing the reliance on labeled samples.

2) Another major contribution of this study is the validation of the effectiveness of adding the Attention Layer and Siamese network on top of the autoencoder for feature enhancement and correction through ablation experiments. The proposed Semi-supervised Learning Model is applied to two benchmark hyperspectral image datasets, the PaviaU Dataset and the Salinas Dataset, achieving an Overall Accuracy and Kappa Coefficient of 82.3%, 0.74 and 87.4%, 0.82 respectively, demonstrating the effectiveness of the proposed model.

REFERENCES

[1] Deng W H, Zhang W, Zhang L M. Instance-dependent label noise learning via separating style from content [J]. Pattern Recognition Letters, 2025, 196 9-15.

[2] Williams B, Qian L. Semi-Supervised Learning for Intrusion Detection in Large Computer Networks [J]. Applied Sciences, 2025, 15 (11): 5930-5930.

[3] Ying S, Song X, Wang H. Semi-supervised domain generalization with clustering and contrastive learning combined mechanism [J]. Knowledge-Based Systems, 2025, 318 113364-113364.

[4] Chadoulos G C, Tsaopoulos E D, Moustakidis P S, et al. A Multi-View Semi-supervised learning method for knee joint cartilage segmentation combining multiple feature descriptors and image

modalities [J]. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, 2024, 12 (1):

[5] Ming S. A Semi-Supervised Learning-Based Method for Recognizing Volleyball Players' Arm Movement Trajectories [J]. International Journal of High Speed Electronics and Systems, 2024, 34 (01):.

[6] Nie F, Song Y, Chang W, et al. Fast Semi-Supervised Learning on Large Graphs: An Improved Green-Function Method. [J]. IEEE transactions on pattern analysis and machine intelligence, 2024, 64–73.

[7] Tran L, Nguyen H, PhanVo K, et al. Novel directed hypergraph p-Laplacian based semi-supervised learning method: theory and algorithms [J]. International Journal of Information Technology, 2024, 17 (4): 1-7.

[8] Li H, Xu X, Liu Z, et al. Low-Quality Sensor Data-Based Semi-Supervised Learning for Medical Image Segmentation [J]. Sensors, 2024, 24 (23): 7799-7799.

[9] Wang H, Bi J, Hua M, et al. Semi-supervised CWGAN-GP modeling for AHU AFDD with high-quality synthetic data filtering mechanism [J]. Building and Environment, 2025, 267 (PA): 112265-112265

[10] Chen Z, Li B. DyConfidMatch: Dynamic thresholding and re-sampling for 3D semi-supervised learning [J]. Pattern Recognition, 2025, 159 111154-111154.

[11] Guan D, Xing Y, Huang J, et al. S2Match: Self-paced sampling for data-limited semi-supervised learning [J]. Pattern Recognition, 2025, 159 111121-111121..

[12] Chen X, Chen Z, Guo L, et al. Pseudo-label assisted semi-supervised adversarial enhancement learning for fault diagnosis of gearbox degradation with limited data [J]. Mechanical Systems and Signal Processing, 2025, 224 112108-112108.

[13] Yang H, Zhu W, Wang S. Accuracy and generalization improvement for image quality assessment of authentic distortion by semi-supervised learning [J]. Applied Intelligence, 2024, (prepublish): 1-14.

[14] Zhang Y, Lin Q, Li L, et al. Multiobjective band selection approach via an adaptive particle swarm optimizer for remote sensing hyperspectral images [J]. Swarm and Evolutionary Computation, 2024, 101614-101615.

[15] WU Z B, CHEN R. Study on uranium ore sorting method based on semi-supervised learning and support vector machine[J]. Metal Mine, 2024, (3): 229–236.

[16] Du Z, Yang L, Tang M. Saliency-Guided Sparse Low-Rank Tensor Approximation for Unsupervised Anomaly Detection of Hyperspectral Remote Sensing Images [J]. Journal of Circuits, Systems and Computers, 2024, 37(1): 85–87.

[17] WU Z B, CHEN R. Study on uranium ore sorting method based on semi-supervised learning and support vector machine[J]. Metal Mine, 2024, (3): 229–236.

[18] LI Y. Adaptive graph clustering algorithm based on label propagation[J]. Changjiang Information & Communications, 2024, 37(1): 85–87.

[19] LI F, JIA D L, YAO Y M, TU J. Few-shot image classification network combining residual and self-attention-based graph convolution[J]. Computer Science, 2023, 50(S1): 376–380.

[20] AI Mingxi, XU Qing, ZHANG Jin, et al. Improved Semi-supervised Condition Recognition Method for Flotation Process Based on Local Feature Enhancement [J/OL]. Control and Decision, 2025,1-10.

[21] SHI Y X, HE J R, LI Z K, ZENG Z G. 3D convolutional autoencoder model for hyperspectral image classification[J]. Journal of Image and Graphics, 2021, 26(8): 2021–2036.

[22] LI Guohao, LV Yuzeng, DONG Yifan, et al. Semi-supervised Inversion of High-density Resistivity Method Based on Forward Constraint [J/OL]. Geophysical and Geochemical Exploration, 2025,1-9.