

DAE-MLP Based Feature Extraction for Hyperspectral Image Classification of Saint Clair River

Youcef Attallah¹ and Ehlem Zigh¹ and Ali Pacha Adda¹

¹ *Laboratoire LACOSI, University of Science and Technology of Oran – Mohamed Boudiaf, Oran, Algeria*

Received 03th of January, 2024; accepted 6th of June 2025

Abstract

Hyperspectral remote sensing has emerged as a powerful tool for vegetation classification due to its ability to capture detailed spectral information. This study introduces a novel methodology for vegetation classification using exclusively hyperspectral imagery. The proposed approach comprises atmospheric correction using the FLAASH algorithm, followed by dimensionality reduction using PCA and segmentation through the ROI selection and the Spectral Angle Mapper (SAM) module. Subsequently, a deep autoencoder is employed for feature extraction, paving the way for classification using the Multi-Layer Perceptron (MLP) algorithm. The effectiveness of this methodology is evaluated using a hyperspectral image of the Saint Clair River, successfully classifying the image into six main classes: water 1, water 2, grass, tree, reed, corn, and an 'unclassified' category encompassing concrete, roads, bricks, wood, and more. Our findings demonstrate the efficacy of this approach in accurately classifying and mapping vegetation in river ecosystems, offering a promising solution in the face of limited hyperspectral datasets.

Key Words: hyperspectral imaging, mineral identification, Optimized 3D-2D CNN, dimensionality reduction, feature extraction.

1 Introduction

Hyperspectral remote sensing has become an indispensable tool for environmental monitoring and mapping due to its ability to capture information about the spectral reflectance of materials in a given scene [1]. Despite these advantages, accurate classification of hyperspectral data remains difficult, primarily due to the absence of dedicated hyperspectral datasets [2]. Consequently, this gap leads us to rely on unprocessed hyperspectral images captured by satellite or airborne sensors such as AVIRIS or Hyperion EO-1 as data sources [3].

Such imagery introduces several challenges, including notable issues such as noise and other complications related to atmospheric absorption effects. In response to these challenges, there is a strong need to develop and implement advanced classification methods designed to effectively utilize hyperspectral images by employing robust noise reduction and artifact mitigation strategies [4].

The St. Clair River, North America's lifeline, lies at the crossroads of commerce, recreation, and ecological vitality. It serves as a lifeline for commercial shipping and a sanctuary for recreational activities such as fishing and boating. Moreover, the river's unique ecological tapestry is home to remarkable flora and fauna, giving

Correspondence to: youcef.attallah@univ-usto.dz

Recommended for acceptance by Angel D. Sappa

<https://doi.org/10.5565/rev/elcvia.1827>

ELCVIA ISSN:15108-5097

Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

it a central role in ecological exploration. In this dynamic landscape, precision environmental monitoring is becoming increasingly urgent [5].

In recent years, deep learning-based approaches have emerged as a powerful tool for hyperspectral image classification due to their ability to extract complex features and patterns from high-dimensional data [6].

Among these approaches, deep autoencoder-based methods have attracted considerable attention for vegetation mapping and classification in different ecosystems [7]. These studies have demonstrated the efficacy of deep learning-based methods in vegetation mapping and classification, **highlighting** their potential for detecting subtle differences in vegetation types. Furthermore, by leveraging the hierarchical representation learning capabilities of deep autoencoders, these approaches have effectively captured the underlying spatial and spectral characteristics, enhancing classification accuracy and providing valuable insights into ecosystem dynamics [8].

This study uses a feature extraction approach based on a deep autoencoder to classify hyperspectral images of a river landscape area using an MLP classifier. We selected six classes: water 1, water 2, grass, tree, reed, corn, and unclassified. Classification performance was evaluated by applying several machine learning classifiers, including Logistic regression [9], Random forest [10], K-nearest Neighbors [11], Gradient boosting [12], Support Vector Machine [13], and Multilayer Perceptron (MLP) [14].

Our proposed approach for hyperspectral image mapping and classification builds on previous research that has utilized deep learning and auto-encoder-based approaches. For instance, *Xiaorui Ma et al. (2016)* demonstrated the effectiveness of their improved deep network, the Spatial Updated Deep Auto-Encoder (SDAE), in handling small training sets and incorporating spatial information into the learning network [7]. Furthermore, *Zhouhan Lin et al. (2013)* proposed an approach for hyperspectral image classification by integrating deep learning and autoencoders to extract spectro-spatial features, which demonstrated superior classification accuracy compared to traditional methods [15]. Finally, *Wenzhi Zhao et al. (2016)* proposed a spectral-spatial feature-based classification (SSFC) framework that utilizes dimension reduction and deep learning techniques for spectral and spatial feature extraction. Their experimental results on well-known hyperspectral datasets show that the proposed SSFC method outperforms other commonly used methods for hyperspectral image classification [6].

Nevertheless, it is imperative to conduct further research to conduct a rigorous evaluation of the performance of these approaches on larger datasets and under diverse conditions. For example, *Jaime Zabalza (2016)* proposed segmented stacked autoencoders (S-SAE) for hyperspectral feature extraction and classification. S-SAE divides the original features into smaller data segments that are processed by smaller SAEs, resulting in improved efficacy of data abstraction and classification accuracy [16]. The primary contributions of this study are outlined as follows:

- **Atmospheric Correction:** We utilized the FLAASH module to correct hyperspectral data for distortions introduced by atmospheric conditions.
- **Dimensionality Reduction:** We implemented Principal Component Analysis (PCA) to reduce the dimensionality of the hyperspectral data by extracting the most significant components.
- **Segmentation:** The Spectral Angle Mapper (SAM) module was utilized to segment the hyperspectral image into distinct regions.
- **Feature Extraction:** We proposed a novel deep autoencoder model specifically designed to extract meaningful features from the segmented hyperspectral data.
- **Classification:** We classified the extracted features into predefined vegetation categories using the Multilayer Perceptron (MLP) technique.

The rest of this study is organized into Section II, which provides a detailed description of the study area along with specific hyperspectral image information. Section III outlines the proposed method, including the algorithms and modules applied. In Section IV, we offer a concise overview of the autoencoder models employed in

this study, delving into their specific applicability to hyperspectral image classification. Section V presents the experimental results obtained through the proposed approach, offering a detailed analysis of the classification results. Finally, Section VI summarizes the main observations and conclusions drawn from our research.

2 Study area and materials

2.1 Study area

The study area for this research is the St. Clair River, which is a significant waterway located in North America with geolocation coordinates between latitude $42^{\circ} 49' 15.114''$ N and longitude $82^{\circ} 29' 9.6864''$ W. It serves as the primary connection between Lake Huron and Lake St. Clair, separating the Canadian province of Ontario from the U.S. state of Michigan. The river stretches for approximately 64 kilometers (40 miles) and has an average depth of 8 meters (26 feet). It is a critical transportation route for commercial ships and is used for recreational purposes such as fishing and boating. In addition, the river's unique ecosystem supports a diverse range of plant and animal life, making it an important area for ecological research (Figure 1). The St. Clair River serves as a vital transportation corridor and a haven for diverse plant and animal life. As human activities continue to shape the landscape, it is imperative to adopt sustainable land use practices that safeguard the ecological integrity of this precious ecosystem. Figure 2 shows the distribution of different types of land use in the St. Clair River region. Agricultural land is the most widespread, occupying around 40% of the total area. Urban areas occupy around 30% of the total area, while forests, grasslands, and wetlands each occupy around 10% of the total area.



Figure 1: Study Area and Overview of the St. Clair River.

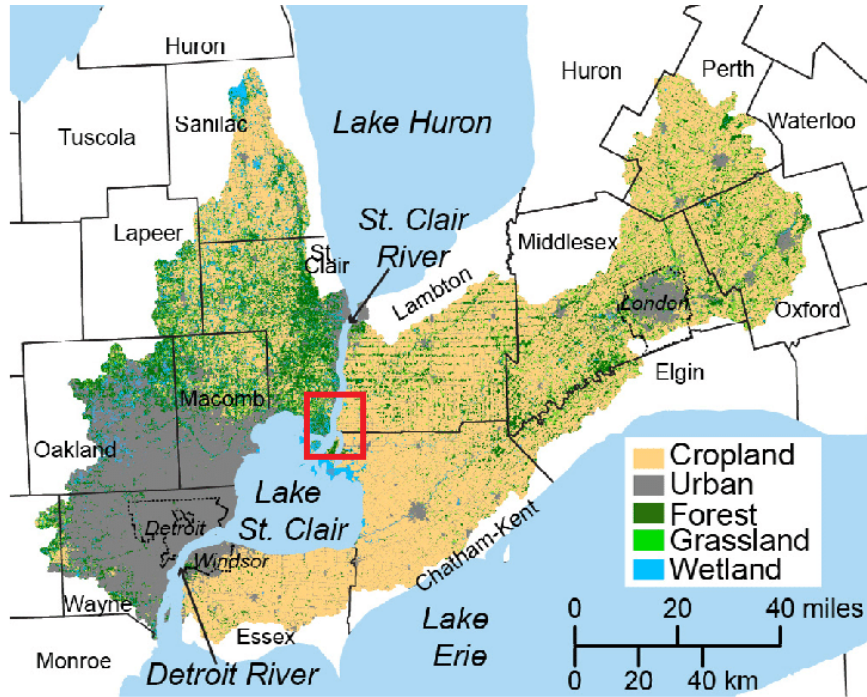


Figure 2: Land use distribution in the St. Clair River.

2.2 AVIRIS data

The Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor aboard a NASA ER-2 aircraft captured the hyperspectral image used in this study. The hyperspectral image is 703 x 738 pixels and contains 224 spectral bands ranging from 0.37 μm to 2.5 μm . This study's region of interest (ROI) has a size of 280 x 280 pixels. The spatial resolution of the image is 17 meters, with a bandwidth of 10 nanometers. There are 79 bands in the visible and near-infrared range (VNIR) and 145 bands in the shortwave infrared range (SWIR). The AVIRIS sensor has been widely used for hyperspectral remote sensing due to its high spectral resolution and accuracy, making it an excellent tool for analyzing the spectral characteristics of the St. Clair River region.

3 Proposed work

Our research focuses on enhancing the classification accuracy of hyperspectral images from the St. Clair River region using a deep autoencoder-based feature extraction approach. Figure 3 illustrates the workflow of our system, which first includes the pre-processing steps, including :

- **BBR and Radiometric Calibration:** This process removes noise-affected bands and converts the image to radiance. It corrects for sensor-related errors, ensuring the quality and consistency of the data.
- **Atmospheric Correction:** This step compensates for atmospheric effects that can alter the spectral characteristics of the images, ensuring accurate reflectance values.
- **Dimensionality Reduction:** Techniques are applied to reduce the data dimensionality, making it more manageable and improving computational efficiency.

Next, the data is processed using a DAE (Deep Autoencoder) architecture that learns a compact and efficient representation of the data in a latent space, capturing the essential features required for accurate classification. Subsequently, the regions of interest selection focus on identifying crucial areas for analysis, while image

segmentation generates the ground truth used as reference data to validate and train the classification model, guaranteeing its accuracy and reliability. The final stage involves classification using machine learning algorithms, followed by performance measurements to assess the model efficiency.

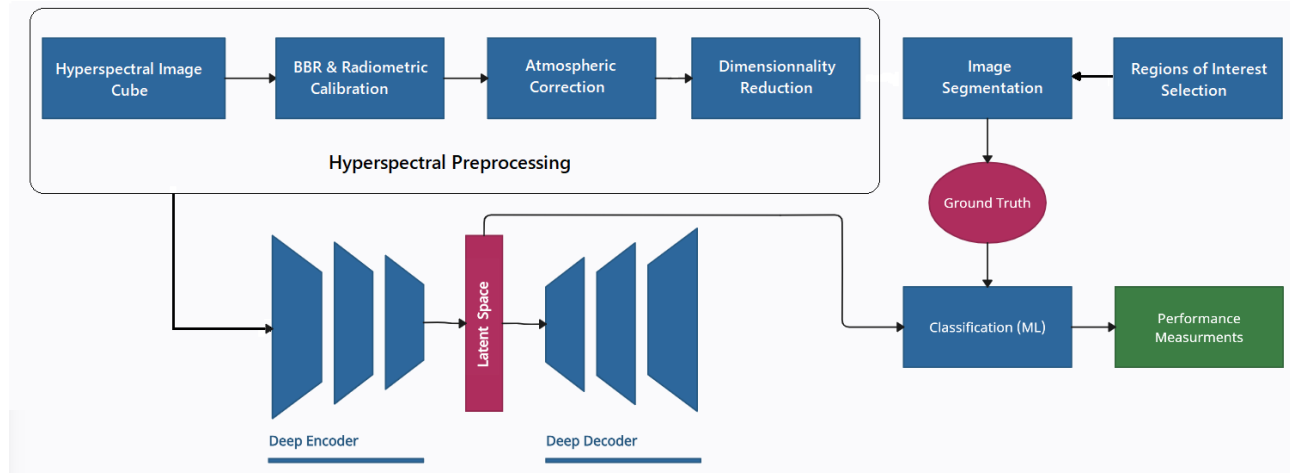


Figure 3: The flowchart of our system.

3.1 Data preprocessing

3.1.1 Bad bands removal (BBR)

Data pre-processing is a crucial step in the analysis of hyperspectral images. One of the main tasks in this stage is to remove bad bands. Our technique is called Bad Band Removal (BBR) [17]. It selects bands with high noise levels or other artifacts. These problematic bands are then excluded from further analysis to prevent them from affecting the quality of feature extraction and classification processes. Table 1 summarizes the specific bands

Removed Bands	Wavelength (μm)
1 to 4	0.37 – 0.40
106 to 116	1.34 – 1.44
152 to 172	1.80 – 1.98
220 to 224	2.46 – 2.50

Table 1: Summary of removed bad bands.

removed during our pre-processing step. The suppressed bands listed in the table cover different wavelength ranges:

- **Bands 1 to 4 (0.37 - 0.40 μm):** These initial bands generally suffer from high noise levels due to sensor limitations and atmospheric scattering effects, which can distort spectral data.
- **Bands 106 to 116 (1.34 - 1.44 μm):** This range includes wavelengths that can suffer from water vapor absorption, resulting in high noise and loss of information. Suppressing these bands minimizes inaccuracies in the data.
- **Bands 152 to 172 (1.80 - 1.98 μm):** In this range, data are sensitive to noise introduced by atmospheric gases and other environmental factors, which can obscure the spectral signals of the materials observed.
- **Bands 220 to 224 (2.46 - 2.50 μm):** At these longer wavelengths, thermal noise, and other sensor-related problems can become pronounced, making these bands unreliable for accurate spectral analysis [18].

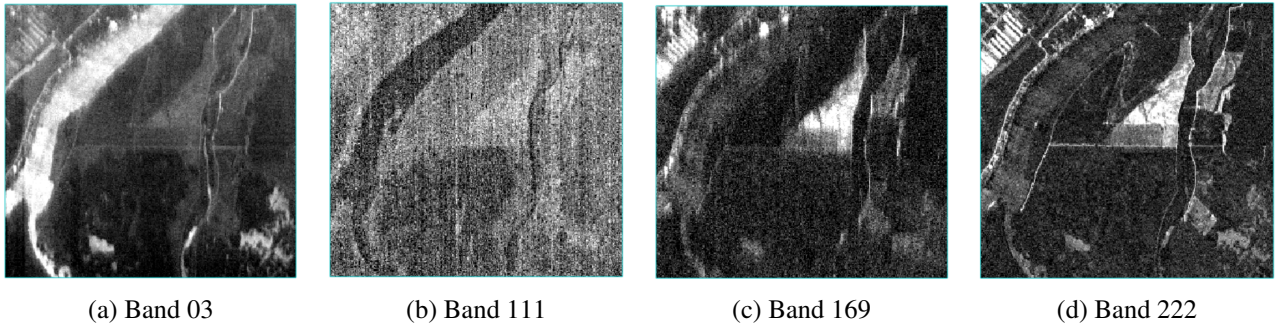


Figure 4: Bad bands in our hyperspectral imaging: visual examples.

Figure 4 provides visual examples of bands affected by noise, such as band 03 (0.38 μm), band 111 (1.39 μm), band 169 (1.95 μm) and band 222 (2.48 μm). These bands are problematic due to the visible noise that can interfere with subsequent data processing and analysis.

3.1.2 Radiometric calibration

The high number of closely spaced and adjacent spectral bands can cause radiometric inaccuracies in hyperspectral data. Therefore, image rectification requires data pre-processing and atmospheric correction [19]. The original, unprocessed image data from the AVIRIS sensor needs correction in radiometry, geometry, and atmospheric conditions. After removing the bad bands, radiometric calibration is essential to convert the raw data into a physical radiance unit. Atmospheric correction is necessary to eliminate atmospheric effects and transform the radiance values into surface reflectance [20]. In this study, we performed radiometric calibration using the following equation:

$$L_{\lambda} = \text{gain} \cdot \text{pixel_value} + \text{offset} \quad (1)$$

L_{λ} Represents the scene's radiance value at the specific wavelength (λ) being measured. The gain factor accounts for the signal amplification during the analog-to-digital conversion (ADC) process. It represents the sensor's sensitivity and how much the raw pixel value is multiplied to obtain the radiance value. Pixel Value is the raw digital value for a specific pixel in the image. It represents the electrical signal generated by the sensor's detector at that location, and the offset factor compensates for any systematic bias in the sensor's response.

3.1.3 Atmospheric correction

The remote sensing sensor measures solar radiation through the atmosphere to obtain the energy reflected by the earth's surface. Atmospheric correction, including coefficients such as altitude, albedo, aerosols, and water vapor, is essential to obtain accurate ground reflectance. In the case of AVIRIS, the correction includes a pixel-by-pixel estimate of water vapor [19]. After radiometric calibration, we applied the FLAASH (Fast Line-of-sight Atmospheric Analysis of Spectral Hypercubes) module, a crucial step in our method of tackling atmospheric interference in hyperspectral imaging. It is designed to correct for atmospheric effects, which can distort the spectral signatures of surface features[21]. By considering these effects, FLAASH improves the accuracy and reliability of subsequent analyses [19]. The following equation can describe the FLAASH correction:

$$L = \frac{A\rho}{1 - \rho e S} + \frac{B\rho}{1 - \rho e S} + La \quad (2)$$

The term (ρ) signifies the surface reflectance of a pixel, representing how much light is reflected by the surface. Meanwhile, (ρe) represents the radiance observed at the sensor, considering the effect of atmospheric spherical albedo. The factor (La) denotes the radiance scattered by the atmosphere in the reverse direction. Notably, the coefficients A and B , contingent upon various atmospheric and geometric conditions, play a pivotal role in

these equations [22]. By understanding and manipulating these elements, we navigate the complex interplay of radiance and atmospheric effects, refining our hyperspectral imagery to represent the underlying surface properties accurately. In our hyperspectral imaging pre-processing, the FLAASH atmospheric correction module assumes essential functions, skilfully compensating for atmospheric scattering and absorption effects [23]. Figure 5 clearly illustrates the significant impact of atmospheric correction on spectral signatures. This figure is divided into two parts, the left showing the area before and after the FLAASH correction, which eliminates atmospheric interference. On the right, the green line represents the original uncorrected spectral data, where fluctuations are visible, indicating the presence of atmospheric interference. In contrast, the red line shows the data after FLAASH correction, revealing distinct and precise spectral characteristics. This transformation highlights the importance of atmospheric correction in obtaining reliable and accurate spectral data. Without this correction, subsequent analyses, such as material or terrain classification, would be biased by atmospheric distortions. Thus, atmospheric correction ensures the accuracy and relevance of the conclusions drawn from spectral data, thereby strengthening the validity of subsequent analyses.

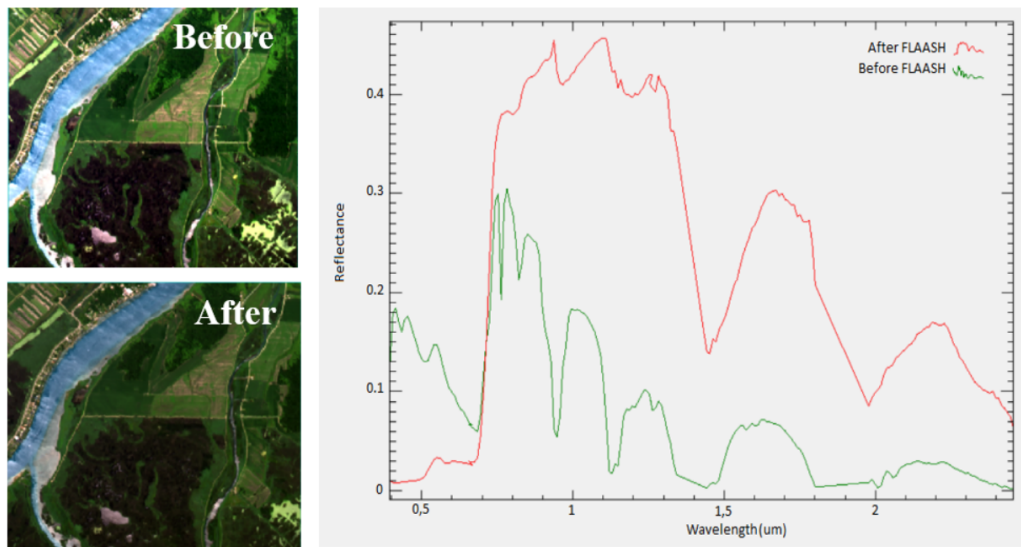


Figure 5: Image and spectra before and after atmospheric correction (FLAASH).

3.1.4 Dimensionality reduction

In the context of hyperspectral image classification for the Saint Clair River, a critical initial step to enhancing feature extraction within the deep autoencoder is spectral dimensionality reduction using PCA. This technique transforms the original data into a new space with reduced dimensions while minimizing information loss. It involves selecting the eigenvalue that contributes most to variance, thereby reducing the dimensionality of the entire dataset without significant loss in classification accuracy and ensuring faster computations [24]. This integration expedites learning convergence and improves the extraction of meaningful and less redundant features in the latent space of the network, capitalizing on the synergies between PCA's dimensionality reduction capabilities and the deep learning strengths of autoencoders [25].

3.2 Data segmentation

Data segmentation was carefully established to provide an in-depth picture of the study area. A preliminary field survey was carried out based on available maps and Google Earth images, supplemented by previous analyses of the area. Regions of interest (ROIs) were precisely selected for each class to identify specific areas for analysis. Figure 6 illustrates the different regions of interest (ROIs) and their locations. Next, the Spectral Angle Mapper (SAM) algorithm was deployed to segment the hyperspectral image and generate a detailed map, accurately characterizing the different regions of the study area. The SAM (Spectral Angle Mapper)



Figure 6: ROI Selections.

algorithm is based on the measurement of the spectral angle between each pixel of the hyperspectral image and the corresponding spectra of the regions of interest (ROI). By calculating this measure, SAM evaluates spectral similarity, enabling precise segmentation based on the spectral properties of different areas [26]. This process promotes the production of a valid map, offering a detailed representation of the distinct features present in the study area.

3.3 Data description

The dataset used in this study comprises 280×280 pixel hyperspectral images, each containing 32 spectral bands (subject to PCA reduction) and originating from the St Clair River region. In Figure 7, the false color representation of the St Clair River region is shown alongside the corresponding segmented image for clarity. This segmented image, measuring 280×280 pixels, comprises distinct classes: Unclassified, Water, Water 2, Corn, Reed, Grass, and Tree. This stratification by class provides a solid basis for assessing the accuracy of the classification. Our labeled data was randomly divided into 70% for training, 10% for validation, and 20% for testing, ensuring that all classes were present in each subset during our experiment. It is crucial to point out that this method of dividing the data improves our model's ability to handle and process unbalanced data.

4 Data modeling

4.1 Feature extraction using DAE

A deep autoencoder is a neural network that aims to produce a lower-dimensional representation of the input data, commonly called latent space [27]. The encoder compresses the input data into a lower-dimensional latent representation while the decoder maps the latent representation back to the original input space [28], [29]. DAEs are applicable in diverse applications, spanning image compression, anomaly detection, and feature extraction [29]. For instance, autoencoders are employed in hyperspectral image classification to acquire a compact representation of spectral bands. It serves as an input for classifiers, thereby enhancing classification performance. Figure 8 represents our deep autoencoder architecture comprising multiple fully connected blocks to extract the features from our hyperspectral image. The architecture encompasses both encoding and

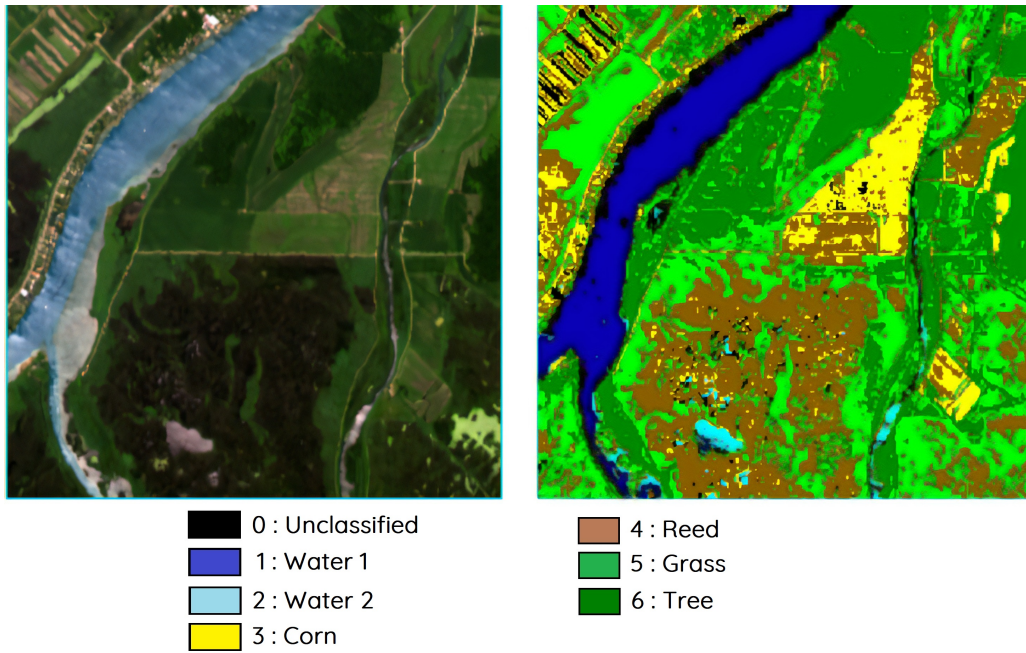


Figure 7: Saint Clair river data. (a) False Color image, (b) Segmented Image.

decoding networks. The coding network includes five fully connected blocks, each housing 1024, 512, 256, 128, and 64 neurons. The decoder network mirrors the encoder's structure and integrates five fully connected blocks with corresponding 64, 128, 256, 512, and 1024 neurons. The autoencoder takes as input a 56448×32 hyperspectral image, where 32 represents the spectral bands after the PCA process, and produces as output a hyperspectral image of identical dimensions; in each block, a dense layer is followed sequentially by an L2 regularization with a penalty coefficient of 0.01. We selected L2 regularization for its ability to effectively mitigate overfitting, thus improving the generalizability of our model. This regularization technique avoids excessive learning from the training data and strikes a balance between model complexity and performance [15]. A subsequent normalization layer is applied to normalize activations, facilitating training convergence by counteracting covariate shift issues. After the encoder, a flattening layer extracts the latent vector pivotal to the classification layer.

The training process of the DAEs is repeated N times based on the designated number of epochs and typically performed by minimizing the reconstruction error between the original input and the reconstructed output, which is commonly measured by the mean squared error (MSE) loss function:

$$L_{MSE} = \frac{1}{N} \times \sum_{i=1}^N \|x_i + \hat{x}_i\|^2 \quad (3)$$

Where N is the number of training samples, x_i is the i^{th} input sample, and $\hat{x}_i$ is the corresponding reconstructed output from the autoencoder. During the training phase, the rectified linear unit (ReLU) activation function is used in all layers of the encoder and decoder networks, except for the last layer of the decoder network, which uses the sigmoid function to ensure the output values are between 0 and 1 [30]. The Adam optimizer is used for effective gradient descent. Following training, the resultant deep autoencoder was harnessed to extract high-level features from hyperspectral data.

4.2 Classification using MLP

In the context of hyperspectral image classification for the Saint Clair River, the extracted features from the Deep Autoencoder (DAE) are subsequently utilized for classification using a Multilayer Perceptron (MLP).

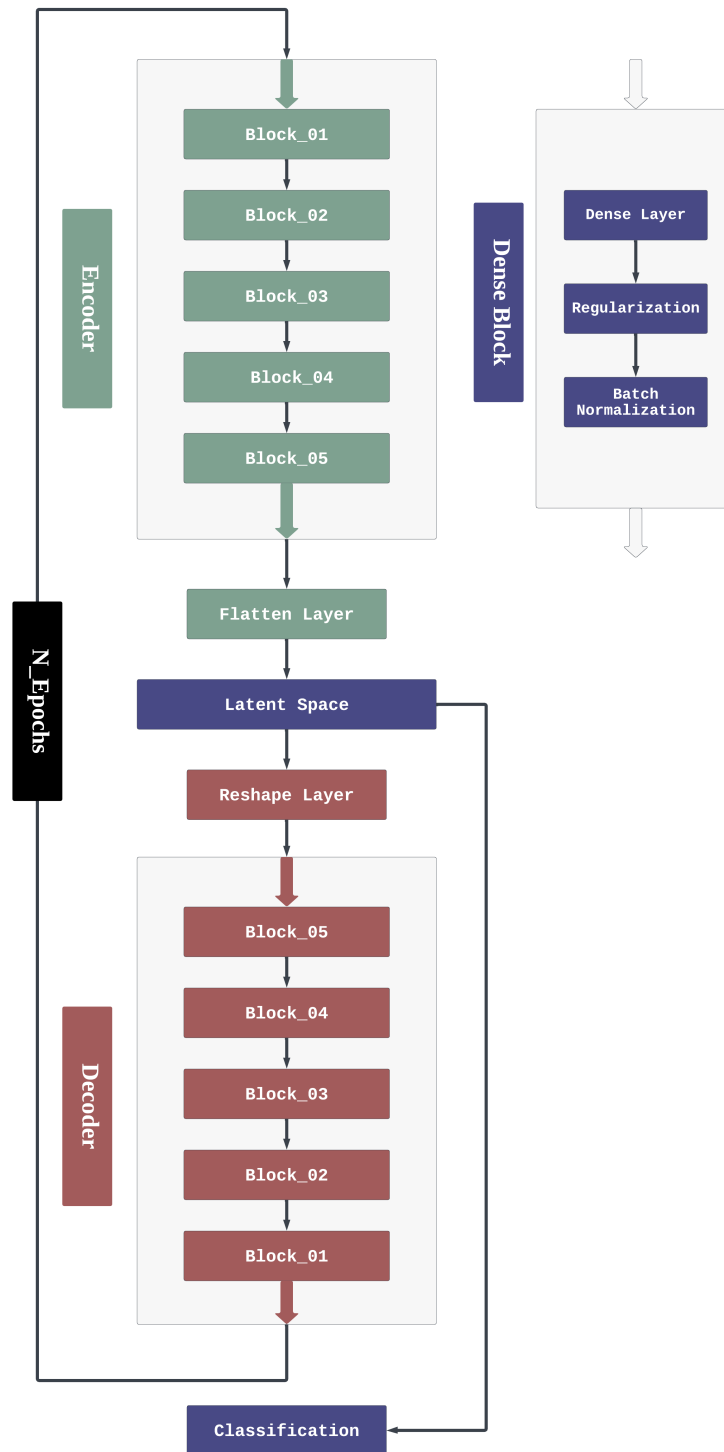


Figure 8: Architectural overview of the proposed DAE model.

The MLP is a type of artificial neural network designed for supervised learning tasks, such as classification [31]. The Multilayer Perceptron consists of an input layer, one or more hidden layers, and an output layer. Each layer is composed of interconnected nodes, or neurons, with each connection characterized by a weight [32]. The latent vector from the Deep Autoencoder serves as the input to the MLP, containing only a single hidden layer with 100 units. The output can be expressed as:

$$a(h) = ReLU(W(h) \times a(h-1)) + b(h) \quad (4)$$

Where $a(h)$ denotes the output activation at the hidden layer; $W(h)$ and $b(h)$ represent the weight and bias parameters, respectively, at the hidden layers. $ReLU$ is the rectified linear unit activation function, defined as follows:

$$ReLU(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \quad (5)$$

The output layer of the DAE-MLP method contains i elements to map to the label of the class designated by y . The formulation is as follows:

$$y = w_1 \times a + b_1 \quad (6)$$

Where w_1 and b_1 represent the weight and bias vector between the hidden layer and the output layer, and y is the output of the DAE-MLP method. To accelerate the convergence of our deep learning model by automatically adjusting the learning rates for each parameter, we used the Adam optimizer.

We conducted our experiments using the Python programming language and harnessed the power of the TensorFlow framework for machine learning and data processing.

Our computing infrastructure had an Intel Core i7-12700F CPU boasting 64 GB of memory. Additionally, we optimized our computations by leveraging the capabilities of a GeForce RTX 3070 Ti GPU, ensuring both efficiency and reproducibility in our results.

5 Results Discussion

5.1 Hyperparameters

In machine learning, hyperparameters are configuration parameters that govern the learning process and impact the performance and convergence of machine learning algorithms [33]. Understanding and appropriately setting hyperparameters are crucial steps in optimizing the performance of machine learning models. In our study, we meticulously selected optimized hyperparameters to enhance the classification performance of our models. Specifically, hyperparameters in the context of machine learning encompass various settings that steer the behavior of the learning algorithm, such as the learning rate, regularization strength, and architectural characteristics of the model. For our Autoencoder (DAE) architecture and each employed classifier—logistic regression, K-Nearest Neighbors, Random Forest, Gradient Boosting, Support Vector Machine, and Multilayer Perceptron—we painstakingly fine-tuned these hyperparameters to strike a delicate balance between model complexity and predictive accuracy. We iteratively experimented with different hyperparameter configurations and used various optimization techniques to adapt our models to the unique characteristics of our dataset. The final selection of hyperparameters was guided by empirical performance measures, ensuring the suitability of our models for the intended task. Table 2 and 3 provide a comprehensive summary of the hyperparameters used in our experiments.

5.2 Evaluation metrics

Various evaluation metrics are employed to assess the performance of our classification approach. Overall Accuracy (OA) assesses the proportion of correctly classified samples.

$$OA = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (7)$$

Feature Extraction	Hyperparameters	Optimized Values
Our Deep Autoen- coder (DAE)	Activation Function	<i>ReLU</i>
	Output Function	<i>Sigmoid</i>
	Kernel Regularizer	L2 Regularizer (0.01)
	Regularization	5
	Batch Normalization	5
	Optimizer	Adam
	Loss Function	Mean Squared Error (<i>MSE</i>)
	Epoch	100
	Batch Size	256

Table 2: Optimized Hyperparameters for Deep Autoencoder.

This metric provides a fundamental assessment of the model's ability to classify objects correctly [34]. The Kappa Coefficient is a widely used measure that considers the agreement expected by random chance, offering a more robust assessment through:

$$Kappa = \frac{P_0 - P_e}{1 - P_e} \quad (8)$$

While P_0 is the observed agreement, and P_e is the expected agreement by chance. Average Accuracy (AA) goes beyond overall accuracy by calculating class-specific accuracies and averaging them:

$$AA = \left(\frac{1}{N}\right) \times \sum_{i=0}^{i=n} \frac{TP_i}{TP_i + FP_i} \quad (9)$$

Where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, FN is the number of false negatives and TP_i and FP_i are the numbers of true positives and false positives, respectively, for class i . This nuanced approach identifies potential disparities in the model's performance across different classes, providing valuable insights.

Additionally, precision, recall, and F1-Score complement these measures.

Precision measures the proportion of authentic positive samples among predicted positives, while recall assesses the proportion of true positive samples among actual positives. The F1-Score is the harmonic mean, offering a balanced evaluation of the classifier's performance.

$$Precision = \frac{TP}{(TP + FP)} \quad (10)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (11)$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (12)$$

Using these additional evaluation measures, we can obtain a more complete understanding of the strengths and weaknesses of the classifier [34].

5.3 Comparative results and analysis

5.3.1 Feature extraction using our DAE

Feature extraction is a pivotal stage in hyperspectral image analysis, primarily owing to the high dimensionality and redundancy of the data [35]. In this study, we harnessed a deep autoencoder-based approach to extract salient features from hyperspectral images captured over the Saint Clair River. This section is dedicated to a

Classification	Hyperparameters	Optimized Values
Logistic Regres- sion	Parameter C	1.0 – 0.1 – 0.001
	Penalty	<i>L2</i> – <i>L1</i> – None
	Solver	<i>Liblinear</i> – <i>Lbfgs</i> – Newton-cg
	Max Iteration	1000
Random Forest	Number of Estimators	500 – 1000 – 1500 – 2000
	Maximum Depth	10 – 20 – 30 – 40 – 50
	Maximum Features	Auto – <i>Sqrt</i> – <i>Log2</i>
	Minimum Number to Split	2 – 5 – 10 – 15 – 20
	Minimum Number for a Leaf	1 – 2 – 5 – 10 – 15
K-Nearest Neighbors	n-neighbors	1 – 2... – 6 – ...10
	Algorithm	Auto – <i>BallTree</i> – <i>Brute</i>
	Parameter P	2
	Weights	<i>Uniforme</i> – Distance
	Leaf Size	30
	Metric	<i>Minkowski</i>
Gradient Boosting	Number of Estimators	100 – 200 – 300 – 400 – 500
	Minimum Samples in Leaf	1
	Learning Rate	0.1 – 0.01 – 0.001
	Max Depth	1 – 2 – 3 – 4 – 5
	Loss	Log-Loss – <i>Exponential</i>
Support Vector Machine	Parameter C	1.0 – 0.1 – 0.01
	Kernel	Rbf – <i>Linear</i> – <i>Poly</i> – <i>Sigmoid</i>
	Degree	1 – 2 – 3 – 4 – 5
	Gamma	<i>Auto</i> – Scale
	Max Iteration	1000
Multi Layer Perceptron	Hidden Layer Sizes	50 – 100 – 150 – 200
	Solver	<i>Lbfgs</i> – <i>Sgd</i> – Adam
	Batch Size	<i>Auto</i>
	Learning Rate	<i>Constant</i> (0.001)
	Activation Function	<i>ReLU</i>
	Max Iteration	1000

Table 3: Optimized Hyperparameters for Classifier Models.

comprehensive examination of our feature extraction results, delving into their impact on the performance of various cutting-edge approaches.

These approaches include convolutional neural networks (CNN) [36], the spectro-spatial feature extraction method (SSFC) [6], the stacked autoencoder (SAE) method [37], and the segmented SAE autoencoder (S-SAE) [16]. These sophisticated approaches were meticulously trained using the mean square error (MSE) loss function and the Adam optimizer for gradient descent, undergoing 100 training epochs with a batch size of 256. Figure 9 compares the training and validation losses across various feature extraction methods, prominently featuring the proposed DAE-MLP approach. This figure is crucial for evaluating the effectiveness and generalizability of each approach, as reflected by the losses observed during the training and validation phases. The green line represents training losses, which indicate the errors the model makes on the training data. In contrast, the red line depicts validation losses, showcasing the model's performance on a separate dataset not used during training. In particular, a lower validation loss indicates a model's ability to generalize well to new, unpublished data. In particular, this figure reveals that our approach has the lowest validation loss, with a value

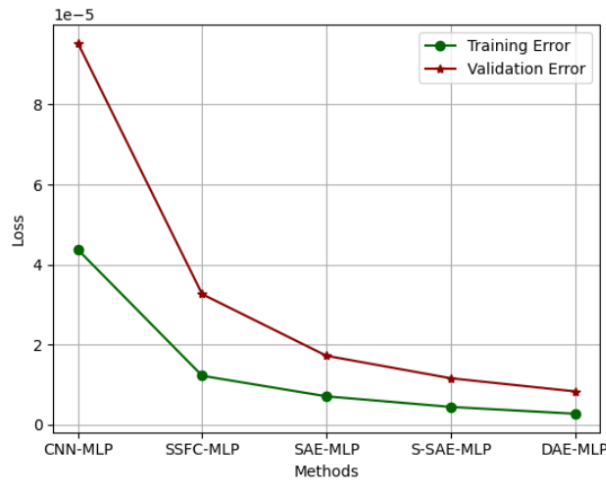


Figure 9: Training and validation loss comparison: our DAE-MLP vs. state-of-the-art methods.

of 9.34×10^{-05} , outperforming all other methods displayed. This finding underscores DAE-MLP's superior generalization capability, suggesting it is exceptionally well-suited for accurately predicting unseen data. Moreover, the DAE-MLP demonstrates a lower training loss than other methods, highlighting its proficiency in capturing and learning the relationships within the training features. These results confirm the robustness of the DAE-MLP approach and highlight its potential for real-world applications. The model's strong performance on training and validation datasets signals its readiness for deployment in practical scenarios where accurate and reliable predictions are crucial.

5.3.2 Classification using DAE with ML classifiers

This section presents an analysis of classification accuracy results for the Saint Clair River dataset. The objective is to evaluate the performance of various machine learning classifiers when combined with our deep autoencoder. Table 4 provides valuable insights into the accuracy performance of the different classifiers. Each classifier's average accuracy (AA), overall accuracy (OA), and Kappa coefficient are presented, along with their standard deviations.

Class	Logistic Regression	Random Forest	KNN	Gradient Boosting	SVM	MLP
Unclassified	48.93±0.81	79.41±0.52	89.53±0.19	88.71±0.36	94.71±0.42	95.03±0.76
Water	92.64±0.20	91.89±0.32	98.66±0.92	98.62±0.10	98.81±0.94	98.46±0.33
Water 2	49.22±2.39	83.40±0.91	83.76±0.36	89.85±0.79	92.83±0.27	94.53±0.30
Corn	83.08±0.46	78.32±1.06	93.52±0.48	90.07±2.38	94.23±0.19	96.46±0.73
Reed	91.12±0.57	92.32±0.72	95.26±0.30	96.36±0.20	98.32±0.53	99.37±0.25
Grass	87.23±1.12	91.33±0.88	94.77±0.21	95.12±0.08	97.69±0.17	98.21±0.18
Tree	92.07±0.27	95.62±0.42	97.03±0.69	97.61±0.42	98.07±0.33	98.31±0.59
OA	88.09±0.31	89.53±0.42	95.09±0.32	95.68±0.37	97.57±0.35	98.51±0.24
Kappa	87.37±0.96	88.21±0.34	93.81±0.72	94.45±0.48	96.87±0.36	98.03±0.23
AA	87.42±0.82	88.73±0.93	95.22±0.45	94.07±0.31	97.01±0.93	97.43±0.12

Table 4: Classification Accuracy for the Saint Clair River Dataset.

The MLP classifier demonstrates a high overall accuracy of 98.51%, eclipsing the performance of all other classifiers by a substantial margin. This remarkable accuracy underscores the exceptional capabilities of MLP

within the classification task. Furthermore, the Kappa coefficient of 98.03% is a testament to the robust agreement between the predicted class labels and the actual segmented image labels obtained through MLP. This coefficient reflects a significant enhancement in classification performance across all classes compared to the other models. This outstanding performance is intrinsically tied to the foundational role played by our proposed Deep Autoencoder (DAE). The synergy between DAE and MLP within our approach has yielded highly proficient results in extracting and classifying hyperspectral images. Comparatively, MLP's accuracy surpasses Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, and the Support Vector Machine (SVM). While delivering commendable performance, SVM lags behind MLP in terms of overall accuracy by approximately 0.94%, highlighting MLP's superiority in classification precision. This remarkable performance by MLP reinforces its status as the preferred classifier within our approach, offering a notable advantage in accurately categorizing hyperspectral data over its counterparts.

In evaluating the performance of various machine learning models on the St. Clair River dataset, we utilized three key metrics: precision, recall, and F1-score. These metrics provide a comprehensive view of each model's ability to correctly classify instances, identify relevant data points, and balance precision and recall. Table 5 summarizes the performance metrics for six models: Logistic Regression, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting, Support Vector Machine (SVM), and Multi-Layer Perceptron (MLP). Figure 10 presents the confusion matrices to give a detailed overview of the accuracy of the models in predicting each class. The MLP model consistently outperforms other models across all metrics, achieving a precision of

-	Logistic Regression	Random Forest	KNN	Gradient Boosting	SVM	MLP
Precision	88.06±0.93	88.73±0.37	95.02±0.23	95.31±0.74	97.22±0.52	98.72±0.13
Recall	87.38±0.57	88.42±0.78	93.88±0.93	95.57±0.47	97.03±0.29	98.43±0.16
F1-Score	87.22±1.01	87.93±0.72	95.06±0.19	93.60±1.03	96.22±0.64	97.27±0.53

Table 5: Comparative analysis of several machine-learning classifiers.

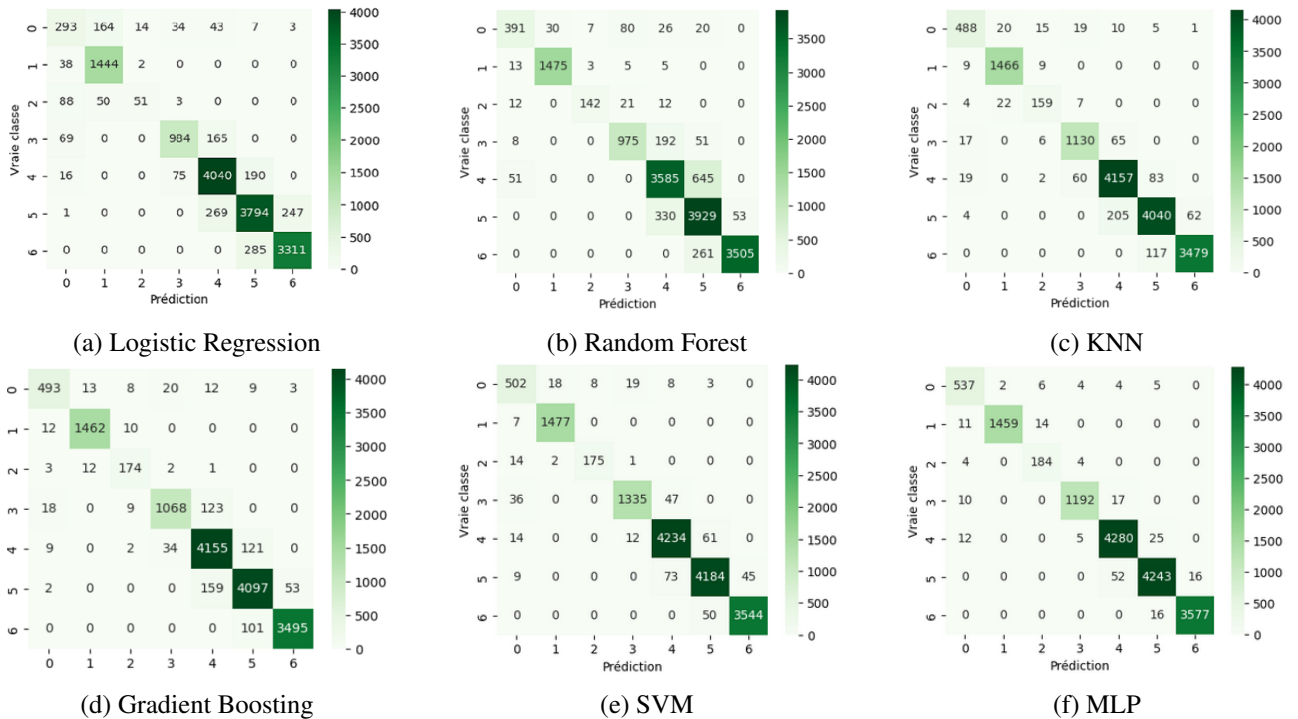


Figure 10: Comparison of a confusion matrix for classification results on Saint Clair river dataset.

98.72%, a recall of 98.43%, and an F1-Score of 97.27%. This performance is attributed to the MLP's ability to capture complex, non-linear relationships in the data through its deep learning architecture, which is well-suited for handling high-dimensional data with intricate patterns, such as the spectral data used in this study.

Logistic regression as a linear model shows the weakest performance, particularly for the unclassified and Reed classes. This limitation is likely due to the model's inability to capture non-linear relationships, which are crucial in distinguishing between closely related spectral signatures.

KNN and Gradient Boosting perform better than Logistic Regression, with Gradient Boosting showing a notable increase in accuracy due to its ensemble learning technique that combines multiple weak learners to form a strong classifier. However, classification errors show the inability to handle overlapping features between classes.

SVM, known for its effectiveness in high-dimensional spaces and when clear margin separation exists between classes, performs better than Logistic Regression and Gradient Boosting but still faces challenges, particularly with the Water 1 and Reed classes. This suggests that the kernel used in SVM might not fully capture the complex boundaries in the data.

The confusion matrices in Figure 10 highlight these differences, showing how MLP's superior ability to minimize misclassification errors across all classes makes it the most robust model for this dataset. The consistent performance of MLP across various evaluation metrics underscores its suitability for tasks requiring high precision and recall, such as environmental monitoring, where accurate classification of land and water features is crucial.

These results justify the comparison by illustrating how different models leverage their inherent strengths and face unique challenges, influencing their effectiveness in this specific application. The MLP model's overall superior performance suggests that deep learning techniques may be more appropriate for datasets with complex, non-linear patterns, providing a clear justification for their use over traditional methods.

5.3.3 Comparison with state of the arts

To demonstrate the superiority of our DAE-MLP model in feature extraction and vegetation classification in the Saint Clair River, we systematically compared it to established state-of-the-art approaches. By combining these approaches with the MLP classifier, our DAE-MLP model generated exceptional results, significantly outperforming other methods.

Class	CNN-MLP	SSFC-MLP	SAE-MLP	S-SAE-MLP	DAE-MLP
Unclassified	91.95±0.07	90.28±0.23	92.74±0.19	93.82±0.21	95.03±0.76
Water	90.64±0.23	90.29±0.43	98.16±0.13	98.82±0.36	98.46±0.33
Water 2	88.12±0.36	89.96±0.73	92.56±0.08	92.85±0.72	94.53±0.30
Corn	88.08±0.62	89.02±0.36	93.59±0.14	94.08±0.38	96.46±0.73
Reed	94.29±0.63	95.42±0.82	97.16±0.32	98.36±0.29	99.37±0.25
Grass	91.45±0.27	90.63±0.85	92.17±0.81	96.12±0.10	98.21±0.18
Tree	92.36±0.71	94.72±0.62	98.07±0.05	98.46±0.52	98.31±0.59
OA	91.48±0.31	91.86±0.69	95.92±0.12	97.58±0.33	98.51±0.24
Kappa	91.77±0.46	92.03±0.34	95.48±0.33	97.28±0.43	98.03±0.23
AA	92.02±0.08	92.93±0.73	96.54±0.08	96.88±0.59	97.43±0.12

Table 6: Comparison of accuracy of our DAE-MLP with state-of-the-art methods.

The classification results, shown in Table 6, highlight the outstanding performance of our DAE-MLP approach. It achieved an average accuracy (AA) of 97.43%, an overall accuracy (OA) of 98.51%, and an impressive Kappa coefficient of 98.03%. In contrast, our model outperforms its closest competitor, S-SAE-MLP, with a difference of 0.55% in AA, about 0.93% in OA, and 0.75% in the Kappa coefficient. This significant dispar-

ity in the three evaluation metrics highlights the notable superiority of our DAE-MLP model compared to its counterparts. Figure 11 presents a comprehensive comparative analysis of the precision, recall, and F1 scores

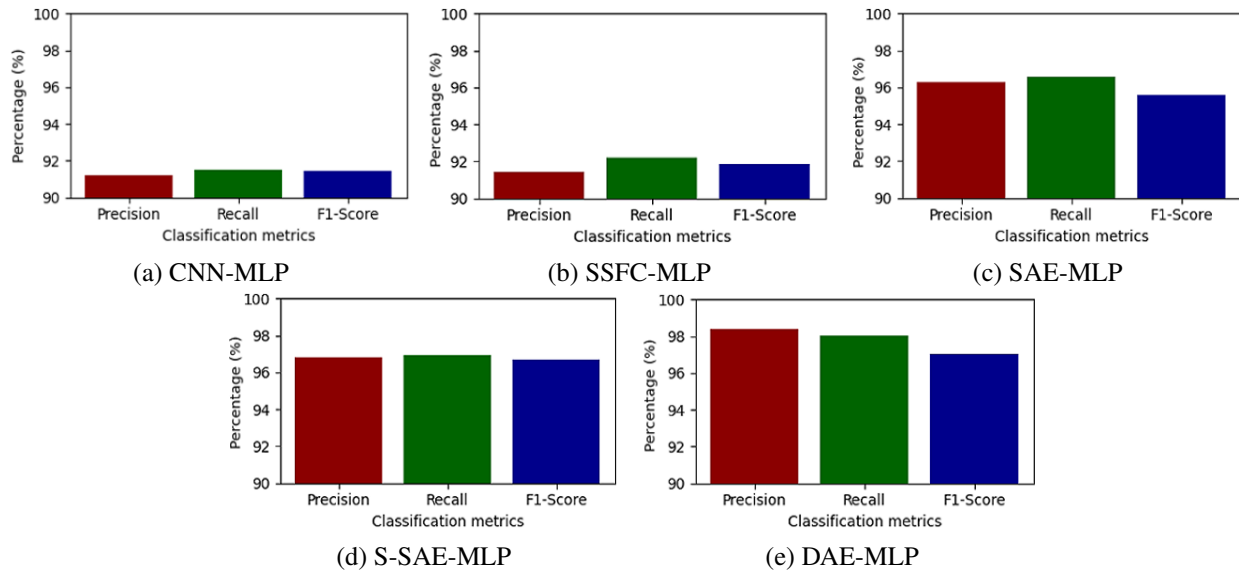


Figure 11: Comparative analysis of our DAE-MLP with state of the art.

across different models. DAE-MLP consistently attains the highest values in all three metrics, demonstrating its exceptional ability to classify vegetation while minimizing misclassification errors. For example, in the "Tree" class, DAE-MLP achieved a precision of 98.46%, recall of 98.31%, and F1-score of 98.38%, outperforming the other models. This suggests that DAE-MLP has a lower rate of false positives and false negatives, making it more reliable in practical applications. The consistency of our DAE-MLP indicates that the model not only identifies vegetation classes accurately but also maintains a high balance between precision and recall, leading to superior F1 scores.

The DAE-MLP model shows remarkable performance across all classes, including those with fewer samples, such as "Water 2" and "Corn." This robustness indicates that the model effectively handles class imbalances, a common issue in hyperspectral image classification. The improved feature extraction and representation capabilities of the DAE-MLP allow it to maintain high accuracy even in less-represented classes. The results for different vegetation types, such as "grass" and "reed," where DAE-MLP achieved accuracies of 98.21% and 99.37%, respectively, highlight its adaptability and effectiveness across diverse categories. This adaptability is crucial for practical applications in environmental monitoring and agricultural management.

The comparison with SAE-MLP and S-SAE-MLP models, which also use autoencoder architectures, underscores the advanced nature of the DAE-MLP's design. The inclusion of denoising autoencoders likely enhances the model's ability to handle noisy data and improves generalization performance, resulting in higher classification accuracy.

The DAE-MLP model demonstrates clear advantages over the state-of-the-art in accuracy, precision, recall, and F1-score. The deep feature extraction capability, effective handling of class imbalances, and adaptability to various classes constitute the main factors contributing to its superior performance. This makes the DAE-MLP model a highly effective tool for hyperspectral image classification tasks in the Saint Clair River and similar environments.

5.4 Comparative evaluation on benchmark datasets

To assess the generalizability and robustness of our proposed method, we applied it to three widely recognized benchmark hyperspectral datasets: Jasper Ridge, Indian Pines, and Salinas Scene. These datasets were selected

due to their diversity in spectral characteristics, spatial resolution, and application domains, offering a comprehensive evaluation of our method's performance across different scenarios.

The Jasper Ridge dataset contains hyperspectral imagery of a natural reserve characterized by a mix of vegetation and bare soil, providing a challenging environment with subtle spectral variations. The Indian Pines dataset, on the other hand, represents an agricultural region with a mixture of crops and forested areas, offering a diverse but structured set of land cover classes. Finally, the Salinas Scene dataset includes agricultural fields with high intra-class spectral similarity, making it an excellent benchmark for evaluating fine-grained classification capabilities [38].

To ensure consistency and replicability, we maintained the same training parameters across all experiments, including the architecture of the autoencoder, the Multi-Layer Perceptron (MLP) configuration, and the preprocessing steps, such as PCA-based dimensionality reduction. This uniform approach allowed us to objectively compare the performance of our method on these datasets while validating its applicability in various ecological and environmental contexts.

Table 7: Comparison of DAE-MLP performance with state-of-the-art methods on Jasper Ridge, Indian Pines, and Salinas Scene datasets.

Dataset	Metric	CNN-MLP	SSFC-MLP	SAE-MLP	S-SAE-MLP	DAE-MLP
Jasper Ridge	OA (%)	96.51 $\pm$ 0.81	96.79 $\pm$ 0.32	98.39 $\pm$ 0.97	98.21 $\pm$ 0.66	99.47 $\pm$ 0.16
	Kappa (%)	96.02 $\pm$ 0.57	95.89 $\pm$ 0.49	98.68 $\pm$ 0.72	99.07 $\pm$ 0.39	99.79 $\pm$ 0.07
	AA (%)	95.67 $\pm$ 0.89	96.30 $\pm$ 0.92	97.94 $\pm$ 0.27	98.62 $\pm$ 0.66	99.78 $\pm$ 0.20
Indian Pines	OA (%)	86.63 $\pm$ 0.68	87.59 $\pm$ 0.27	90.36 $\pm$ 0.53	92.22 $\pm$ 0.33	95.98 $\pm$ 0.54
	Kappa (%)	88.65 $\pm$ 0.72	87.30 $\pm$ 0.53	91.47 $\pm$ 0.79	92.27 $\pm$ 0.82	96.58 $\pm$ 0.09
	AA (%)	87.65 $\pm$ 0.54	88.36 $\pm$ 0.47	90.97 $\pm$ 0.65	92.46 $\pm$ 0.71	97.07 $\pm$ 0.33
Salinas Scene	OA (%)	87.89 $\pm$ 0.28	88.43 $\pm$ 0.34	93.41 $\pm$ 0.54	94.02 $\pm$ 0.23	97.72 $\pm$ 0.71
	Kappa (%)	88.40 $\pm$ 0.41	88.74 $\pm$ 0.93	92.57 $\pm$ 0.39	95.41 $\pm$ 0.55	98.02 $\pm$ 0.22
	AA (%)	88.52 $\pm$ 0.29	87.95 $\pm$ 0.41	93.72 $\pm$ 0.95	95.19 $\pm$ 0.57	98.27 $\pm$ 0.36

The comparative evaluation presented in Table 7 highlights the effectiveness of the proposed DAE-MLP method compared to state-of-the-art approaches across the three benchmark datasets: Jasper Ridge, Indian Pines, and Salinas Scene. The metrics considered include Overall Accuracy (OA), Kappa coefficient, and Average Accuracy (AA), which collectively provide a comprehensive assessment of classification performance. For the Jasper Ridge dataset, DAE-MLP achieved the highest performance across all metrics, with an OA of 99.47%, a Kappa coefficient of 99.79%, and an AA of 99.78%. These results demonstrate the robustness of the proposed method in handling complex vegetation classes with subtle spectral differences, outperforming the second-best approach, S-SAE-MLP, by a notable margin.

In the Indian Pines dataset, characterized by diverse land cover types, DAE-MLP again achieved superior results, with an OA of 96.58%, a Kappa coefficient of 96.50%, and an AA of 97.07%. These metrics indicate the method's ability to accurately classify a mixture of agricultural and forested areas, showcasing its versatility compared to other methods like S-SAE-MLP and SAE-MLP, which performed marginally lower.

For the Salinas Scene dataset, which involves high intra-class similarity, the DAE-MLP method maintained its dominance with an OA of 97.72%, a Kappa coefficient of 98.22%, and an AA of 98.27%. These results underline the method's capacity to distinguish subtle spectral differences between classes, which is crucial for precise mapping in agricultural fields.

Overall, the DAE-MLP method demonstrated consistently superior performance across all datasets, surpassing traditional CNN-MLP and advanced methods like S-SAE-MLP. The incorporation of deep autoencoders for feature extraction, combined with the PCA-based dimensionality reduction and the MLP classifier, significantly contributed to its high accuracy. This consistent performance across diverse datasets underscores the generalizability and robustness of the proposed approach, making it a promising tool for hyperspectral image classification in various ecological and environmental applications.

6 Conclusion

This study highlights the remarkable effectiveness of a combined approach that merges deep autoencoder-based feature extraction with the robust multi-layer perceptron (MLP) classification capabilities for hyperspectral image classification and vegetation mapping within river ecosystems. Our dedicated focus on the Saint Clair River region has unveiled the full potential of this methodology. The results of our study have implications for a range of applications, including forest and river area management and conservation, land use planning, and monitoring environmental change over time. By demonstrating the effectiveness of deep learning-based approaches to hyperspectral image classification, our study contributes to the growing body of research in this area. In addition, it provides insight into the potential of these approaches for environmental monitoring and mapping.

References

- [1] G. Ortac and G. Ozcan, "Comparative study of hyperspectral image classification by multidimensional convolutional neural network approaches to improve accuracy," *Expert Systems with Applications*, vol. 182, p. 115 280, 2021.
- [2] M. Zhao, J. Chen, and Z. He, "A laboratory-created dataset with ground truth for hyperspectral unmixing evaluation," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 7, pp. 2170–2183, 2019.
- [3] F. A. Kruse *et al.*, "Comparison of aviris and hyperion for hyperspectral mineral mapping," in *11th JPL airborne geoscience workshop*, vol. 4, 2002.
- [4] B. Rasti, P. Scheunders, P. Ghamisi, G. Licciardi, and J. Chanussot, "Noise reduction in hyperspectral imagery: Overview and application," *Remote Sensing*, vol. 10, no. 3, p. 482, 2018.
- [5] M. M. Baustian, G. Mavrommati, E. A. Dreelin, *et al.*, "A one hundred year review of the socioeconomic and ecological systems of lake st. clair, north america," *Journal of Great Lakes Research*, vol. 40, no. 1, pp. 15–26, 2014.
- [6] W. Zhao and S. Du, "Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 8, pp. 4544–4554, 2016.
- [7] X. Ma, H. Wang, and J. Geng, "Spectral–spatial classification of hyperspectral image based on deep auto-encoder," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 9, pp. 4073–4085, 2016.
- [8] S. Li, X. Zhu, Y. Liu, and J. Bao, "Adaptive spatial-spectral feature learning for hyperspectral image classification," *IEEE access*, vol. 7, pp. 61 534–61 547, 2019.
- [9] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 11, pp. 4085–4098, 2010.
- [10] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.
- [11] C. Bo, H. Lu, and D. Wang, "Spectral-spatial k-nearest neighbor approach for hyperspectral image classification," *Multimedia Tools and Applications*, vol. 77, pp. 10 419–10 436, 2018.
- [12] Z. He, D. Lin, T. Lau, and M. Wu, "Gradient boosting machine: A survey," *arXiv preprint arXiv:1908.06951*, 2019.
- [13] M. Pal and G. M. Foody, "Feature selection for classification of hyperspectral data by svm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 5, pp. 2297–2307, 2010.

- [14] K. Deng, H. Zhao, N. Li, and W. Wei, "Identification of minerals in hyperspectral imagery based on the attenuation spectral absorption index vector using a multilayer perceptron," *Remote Sensing Letters*, vol. 12, no. 5, pp. 449–458, 2021.
- [15] Z. Lin, Y. Chen, X. Zhao, and G. Wang, "Spectral-spatial classification of hyperspectral image using autoencoders," in *2013 9th international conference on information, Communications & Signal Processing*, IEEE, 2013, pp. 1–5.
- [16] J. Zabalza, J. Ren, J. Zheng, et al., "Novel segmented stacked autoencoder for effective dimensionality reduction and feature extraction in hyperspectral imaging," *Neurocomputing*, vol. 185, pp. 1–10, 2016.
- [17] W. Sun and Q. Du, "Hyperspectral band selection: A review," *IEEE Geoscience and Remote Sensing Magazine*, vol. 7, no. 2, pp. 118–139, 2019.
- [18] L. Ji, X. Geng, Y. Zhao, and F. Wang, "An automatic bad band preremoval algorithm for hyperspectral imagery," *arXiv preprint arXiv:1610.05929*, 2016.
- [19] M. K. Tripathi and H. Govil, "Evaluation of aviris-ng hyperspectral images for mineral identification and mapping," *Heliyon*, vol. 5, no. 11, 2019.
- [20] R. N. Clark, G. A. Swayze, R. Wise, et al., "Usgs digital spectral library splib06a," *US Geological Survey, Open File Report*, vol. 3395, 2003.
- [21] G. W. Felde, G. P. Anderson, T. W. Cooley, M. W. Matthew, A. Berk, J. Lee, et al., "Analysis of hyperion data with the flaash atmospheric correction algorithm," in *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No. 03CH37477)*, IEEE, vol. 1, 2003, pp. 90–92.
- [22] S. Chatteraj, R. Sharma, C. Kumar, P. Champati ray, and V. Sengar, "Identification and characterization of hydrothermally altered minerals using surface and space-based reflectance spectroscopy, in parts of south-eastern rajasthan, india," *SN Applied Sciences*, vol. 2, pp. 1–9, 2020.
- [23] N. Rani, V. R. Mandla, and T. Singh, "Evaluation of atmospheric corrections on hyperspectral data with special reference to mineral mapping," *Geoscience Frontiers*, vol. 8, no. 4, pp. 797–808, 2017.
- [24] H. Li, J. Cui, X. Zhang, Y. Han, and L. Cao, "Dimensionality reduction and classification of hyperspectral remote sensing image feature extraction," *Remote Sensing*, vol. 14, no. 18, p. 4579, 2022.
- [25] C. Rodarmel and J. Shan, "Principal component analysis for hyperspectral image classification," *Surveying and Land Information Science*, vol. 62, no. 2, pp. 115–122, 2002.
- [26] F. A. Kruse, A. Lefkoff, J. Boardman, et al., "The spectral image processing system (sips)—interactive visualization and analysis of imaging spectrometer data," *Remote sensing of environment*, vol. 44, no. 2–3, pp. 145–163, 1993.
- [27] S. Lange and M. Riedmiller, "Deep auto-encoder neural networks in reinforcement learning," in *The 2010 international joint conference on neural networks (IJCNN)*, IEEE, 2010, pp. 1–8.
- [28] B. Liu, Y. Chen, S. Liu, and H.-S. Kim, "Deep learning in latent space for video prediction and compression," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 701–710.
- [29] D. Bank, N. Koenigstein, and R. Giryes, "Autoencoders," *Machine learning for data science handbook: data mining and knowledge discovery handbook*, pp. 353–374, 2023.
- [30] A. Das, K. A. Viji, and L. Sebastian, "A survey on image and video super resolution with deep learning models," *International Journal of Engineering Research & Technology*, vol. 9, no. 13, pp. 30–35, 2021.
- [31] P. M. Atkinson and A. R. Tatnall, "Introduction neural networks in remote sensing," *International Journal of remote sensing*, vol. 18, no. 4, pp. 699–709, 1997.

- [32] F. Pacifici, F. Del Frate, C. Solimini, and W. J. Emery, “Neural networks for land cover applications,” *Computational intelligence for remote sensing*, pp. 267–293, 2008.
- [33] M. Feurer and F. Hutter, “Hyperparameter optimization,” *Automated machine learning: Methods, systems, challenges*, pp. 3–33, 2019.
- [34] A. J. Alberg, J. W. Park, B. W. Hager, M. V. Brock, and M. Diener-West, “Review the use of “overall accuracy” to evaluate the validity of screening or diagnostic tests.,” *JGIM: Journal of General Internal Medicine*, vol. 19, no. 5, 2004.
- [35] L. J. Chmielewski, R. Kozera, A. Orłowski, K. Wojciechowski, A. M. Bruckstein, and N. Petkov, *Computer Vision and Graphics: International Conference, ICCVG 2018, Warsaw, Poland, September 17-19, 2018, Proceedings*. Springer, 2018, vol. 11114.
- [36] C. Zhang, X. Pan, H. Li, *et al.*, “A hybrid mlp-cnn classifier for very fine resolution remotely sensed image classification,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 140, pp. 133–144, 2018.
- [37] C. Xing, L. Ma, X. Yang, *et al.*, “Stacked denoise autoencoder based feature extraction and classification for hyperspectral images,” *Journal of Sensors*, vol. 2016, 2016.
- [38] B. A. M Graña MA Vezanzons, *Hyperspectral remote sensing scenes*, https://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes, Accessed: 2025-01-08.