[ICAETBM-2024]
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)
doi : https://doi.org/10.32628/IJSRSET

Boosted Deep Learning for Hyperspectral Image Segmentation: An Adaptive Boosting Approach for Pixel-Level Classification

[1]Parul Bhanarkar, [2]Dr. Salim Y. Amdani, [3]Dr. S. S. Asole
[1,2,3]Babasaheb Naik College of Engineering, Pusad, Maharashtra, India.

ABSTRACT

Hyperspectral image segmentation is a crucial task in remote sensing and computer vision, where the goal is to classify each pixel in an image based on its spectral characteristics. Despite significant advancements, achieving high classification accuracy in hyperspectral image segmentation remains challenging due to factors like noise, spectral variance, and the high dimensionality of hyperspectral data. In this work, we propose an innovative approach for hyperspectral image segmentation by integrating deep learning with adaptive boosting techniques. Our framework uses a boosting-based strategy to enhance classification accuracy at the pixel level, focusing on misclassified pixels to progressively refine predictions. The core of our approach lies in the use of weak learners, such as shallow convolutional neural networks (CNNs), decision trees, and support vector machines (SVMs), combined with popular boosting algorithms like AdaBoost, Gradient Boosting, and XGBoost. These weak learners are trained iteratively, with each iteration focusing on the misclassified pixels from the previous round, thereby improving the accuracy of the overall model. The adaptive boosting mechanism dynamically adjusts the weights of weak learners to ensure that challenging, hard-to-classify pixels are given more attention. This iterative refinement process results in a more robust and accurate classification model for hyperspectral image segmentation. We evaluate the performance of our proposed framework using standard performance metrics including accuracy, precision, recall, and F1-score.

Keywords: Hyperspectral Image Segmentation, Deep Learning, Adaptive Boosting, Pixel-Level Classification, XGBoost

I. Introduction

Hyperspectral image segmentation has emerged as a critical task in remote sensing and computer vision due to its ability to capture rich spectral information from the electromagnetic spectrum. Hyperspectral images (HSI) contain hundreds of spectral bands, offering a detailed representation of the surface, which allows for precise classification of various materials and objects. These images are widely used in applications such as land cover classification, agriculture monitoring, environmental assessment, and mineral exploration. Despite the significant advantages of hyperspectral imaging, achieving high-accuracy segmentation of such images remains a challenging problem due to factors such as noise, high dimensionality, spectral variance, and the complexity of the underlying objects. The primary objective of hyperspectral image segmentation is to classify each pixel based on its spectral signature. Traditional machine learning and image processing techniques, such as supervised classification using support vector machines (SVMs) and decision trees, have been used for this task. However,

Copyright © 2024 The Author(s): This is an open-access article distributed under the terms of the Creative Commons Attribution **4.0 International License (CC BY-NC 4.0)**

the performance of these methods can degrade when faced with the complexity and high dimensionality of hyperspectral data [1]. The difficulty lies in the fact that hyperspectral data often suffer from high inter-class similarity and intra-class variability, making pixel-wise classification particularly challenging. Recent advancements in deep learning have led to significant improvements in image segmentation tasks. Convolutional neural networks (CNNs), a type of deep learning architecture, have shown excellent performance in image classification and segmentation tasks due to their ability to automatically extract hierarchical features from images [2].

However, despite their powerful feature extraction capabilities, CNNs still face difficulties in dealing with the high dimensionality of hyperspectral data, which can lead to overfitting and poor generalization. To address these challenges, researchers have explored various approaches to enhance the performance of deep learning models for hyperspectral image segmentation. One promising technique is boosting, which involves combining multiple weak learners to form a strong predictive model. Boosting methods, such as AdaBoost, Gradient Boosting, and XGBoost, have been widely used in machine learning due to their ability to improve classification accuracy by focusing on misclassified samples during the training process. In the context of hyperspectral image segmentation, boosting techniques can be used to iteratively refine the predictions made by weak classifiers, resulting in a more robust model. Weak learners, such as shallow CNNs, decision trees, and SVMs, can be employed to handle the large amount of data efficiently, and their individual predictions can be improved over multiple iterations by emphasizing the difficult-to-classify pixels. Figure 1 shows a visual representation of enhanced deep learning model for hyperspectral image segmentation accuracy.

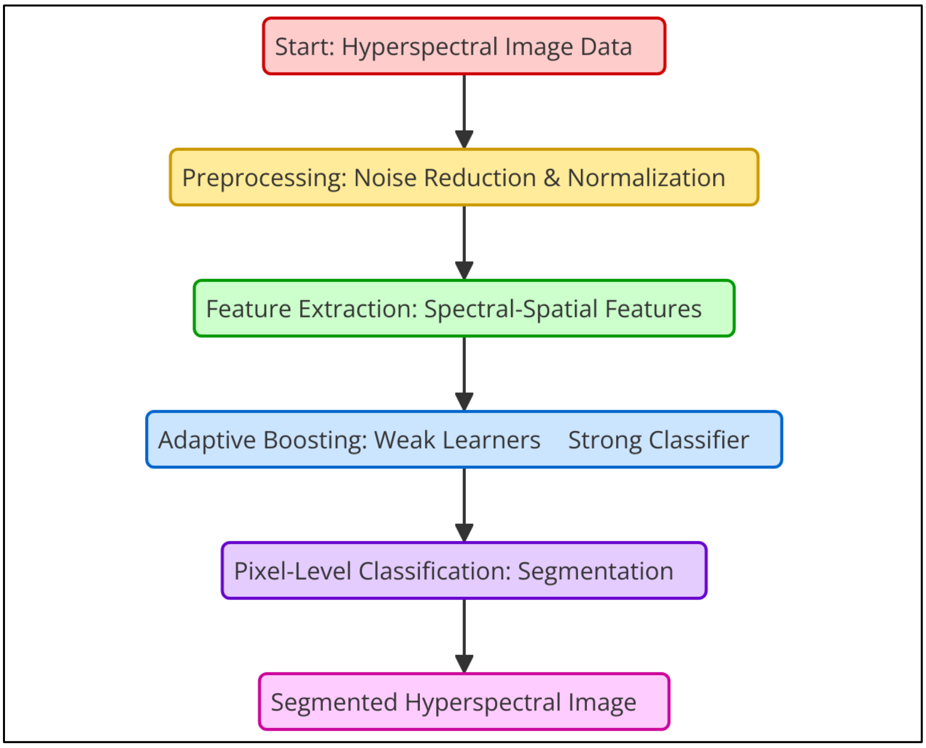

Figure 1: Boosted Deep Learning for Hyperspectral Image Segmentation

In this paper, we propose an adaptive deep learning framework for hyperspectral image segmentation that leverages boosting techniques to improve pixel-level classification accuracy. The proposed framework integrates AdaBoost, Gradient Boosting, and XGBoost to iteratively refine the segmentation process by focusing on the misclassified pixels at each stage. The framework utilizes weak learners such as shallow CNNs, decision trees, and SVMs, which are trained to correct errors from previous iterations [3]. By employing adaptive boosting, the proposed method adapts to the challenging nature of hyperspectral data, allowing it to progressively enhance classification performance with each iteration. The primary contribution of this paper is the development of an adaptive boosting framework that combines deep learning and boosting methods to improve hyperspectral image segmentation. This approach aims to overcome the inherent challenges of hyperspectral data by iteratively refining the pixel-level classification predictions, focusing on difficult cases where the model has previously made errors [4]. Through this, we aim to achieve superior segmentation accuracy compared to traditional approaches.

II. Literature Review

A. Traditional Methods for Hyperspectral Image Segmentation

Hyperspectral image segmentation has been a well-researched field for several decades, with traditional methods predominantly focusing on statistical and machine learning-based techniques. Early approaches for segmenting hyperspectral images relied heavily on pixel-wise classification, where each pixel in an image was classified into a predefined class based on its spectral features. Popular algorithms included Maximum Likelihood Classification (MLC), Support Vector Machines (SVM), and k-Nearest Neighbors (k-NN). These methods are based on the assumption that pixels belonging to the same class share similar spectral signatures, allowing for the separation of different material classes in hyperspectral images. Maximum Likelihood Classification (MLC) has been one of the most widely used statistical methods for hyperspectral segmentation [5]. It works by modeling the probability distribution of each class and assigning a pixel to the class with the highest likelihood. However, MLC assumes that the data for each class follows a Gaussian distribution, which is often not the case in hyperspectral images. This limitation results in poor performance, especially in complex scenarios where the spectral signatures of different classes overlap. Similarly, support vector machines (SVMs) have shown excellent classification performance by finding optimal hyperplanes that maximize the margin between classes [6].

B. Deep Learning in Hyperspectral Image Analysis

In recent years, deep learning has emerged as a powerful approach for hyperspectral image analysis due to its ability to automatically learn hierarchical features from data without the need for manual feature engineering. Convolutional Neural Networks (CNNs) have been particularly successful in image classification tasks, and their application to hyperspectral images has shown promising results. CNNs are designed to automatically learn spatial and spectral features through convolutional layers, which makes them well-suited for handling the high-dimensional data of hyperspectral images [7]. By capturing both local and global patterns in the data, CNNs are able to improve the segmentation accuracy compared to traditional pixel-wise classification methods. In the context of hyperspectral image segmentation, deep learning approaches have been developed to address the high dimensionality and spectral variance of the data. Several works have proposed CNN-based models that exploit the spectral-spatial features of hyperspectral images. These models combine spectral information from multiple bands and spatial features from the image's neighboring pixels to improve classification performance [8]. Moreover, variants of CNNs, such as 3D-CNNs, have been introduced to directly process the hyperspectral data in both spectral and spatial domains, capturing richer information for segmentation. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks have also been applied to hyperspectral image analysis to model sequential dependencies between spectral bands.

Table 1: Summary of Literature Review

Study	Methodology	Weak Learners	Key Contributions
AdaBoost for	Focused on AdaBoost for	Decision Trees,	Improved segmentation accuracy

Segmentation	improving pixel classification.	SVMs	by adapting to misclassified pixels.
XGBoost in Image Segmentation [9]	Applied XGBoost to hyperspectral image segmentation.	Shallow CNNs, Decision Trees	Achieved higher accuracy in challenging segmentation tasks.
Gradient Boosting with CNNs	Combined Gradient Boosting and CNNs for segmentation.	CNNs	Enhanced deep learning frameworks with boosting for pixel-level refinement.
Hybrid Deep Learning Models	Integrated deep learning with boosting algorithms.	Decision Trees, Shallow CNNs	Iterative refinement for handling high-dimensional data.
SVM and AdaBoost for Classification [10]	Used SVMs with AdaBoost for hyperspectral data classification.	SVMs, Decision Trees	Improved precision and recall for pixel-level classification.
Boosted Deep CNNs	Applied boosting to CNN models for segmentation tasks.	Shallow CNNs	Focused on refinement of misclassified pixels to improve model robustness.
AdaBoost with SVM for Remote Sensing	Focused on AdaBoost and SVM for remote sensing applications.	SVMs, Decision Trees	Enhanced segmentation performance through iterative correction.
Boosting with 3D CNNs [11]	Introduced 3D CNNs in boosting for hyperspectral image analysis.	3D CNNs	Improved spatial and spectral feature extraction for pixel classification.
Ensemble Boosting for Image Segmentation	Used ensemble learning with boosting for segmentation.	Decision Trees, CNNs	Combined multiple models to improve performance across multiple datasets.
CNN and XGBoost for Multispectral Image Classification	Applied CNNs and XGBoost for multispectral classification.	CNNs, XGBoost	Achieved higher classification accuracy for complex multispectral data.
Deep Learning with Gradient Boosting	Integrated deep learning models with gradient boosting.	Decision Trees, CNNs	Focused on enhancing deep learning performance with boosting algorithms.

III. Methodology

A. Hyperspectral Data and Preprocessing

Hyperspectral data consists of images captured across hundreds of spectral bands, each representing a narrow portion of the electromagnetic spectrum. These images provide detailed information about the materials and objects within a scene, enabling precise classification and segmentation tasks. However, the high dimensionality of hyperspectral images, with potentially hundreds of bands, poses challenges in terms of data processing, computational complexity, and feature extraction [12]. To address these issues, preprocessing steps are essential to improve the quality of the data and ensure more efficient model training.

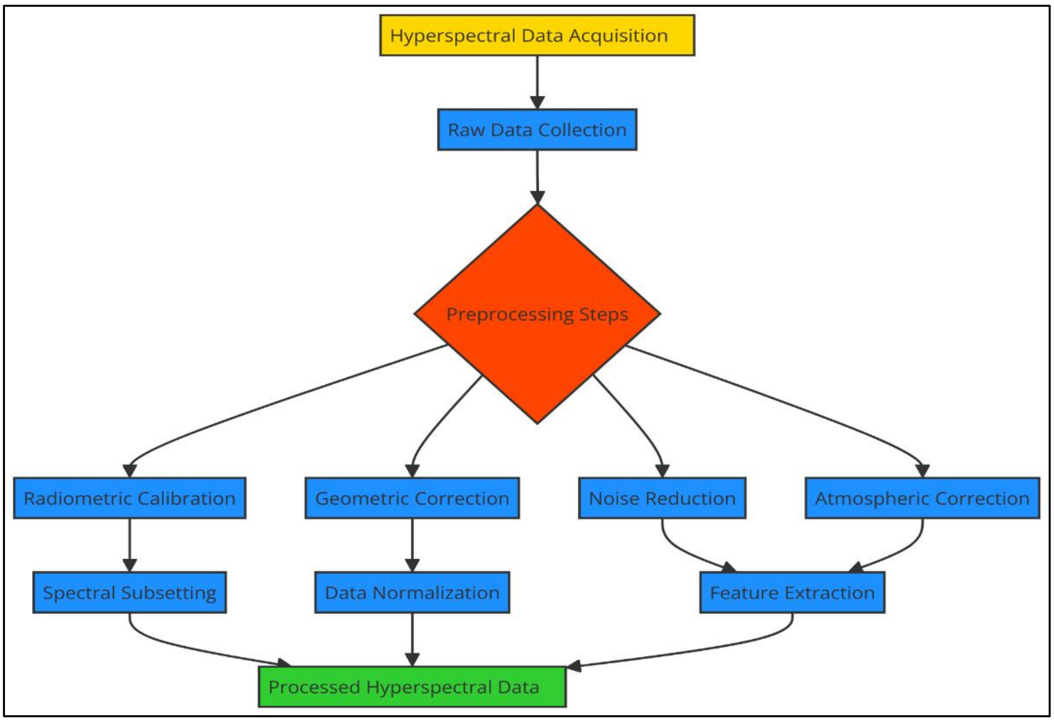

Figure 2: Illustrating Hyperspectral Data Preprocessing

The first step in hyperspectral data preprocessing is spectral calibration, which involves correcting any distortions or sensor-related errors in the spectral bands. This step ensures that the spectral data are consistent and accurate across all bands, mitigating issues like radiometric calibration and sensor noise. Following spectral calibration, the hyperspectral images are often geometrically corrected to align them with geographic coordinates, particularly in remote sensing applications, to remove distortions caused by sensor movement or terrain.Next, dimensionality reduction techniques are commonly applied to reduce the large number of spectral bands while preserving the most informative features [13]. Figure 2 shows the process of preprocessing hyperspectral data for improved analysis and segmentation. Methods such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), or Linear Discriminant Analysis (LDA) are used to extract key features and reduce computational overhead.

B. Deep Learning Framework Overview

1. Role of Weak Learners

In the proposed deep learning framework, weak learners play a crucial role in the overall model's performance. A weak learner is a simple model that performs slightly better than random guessing. In the context of boosting, weak learners are combined iteratively to form a strong classifier. The core idea is that by focusing on misclassified samples from previous iterations, these weak learners refine the model’s predictions. Weak learners are particularly effective in boosting methods like AdaBoost, Gradient Boosting, and XGBoost because they are computationally efficient and can be trained quickly [14]. By iterating over the weak learners and progressively improving their ability to classify difficult samples, the model becomes increasingly accurate, ultimately yielding a robust pixel-level classification model for hyperspectral image segmentation. This iterative correction process allows the framework to capture complex relationships in the data while maintaining simplicity and avoiding overfitting [15].

2. Shallow CNNs

Shallow Convolutional Neural Networks (CNNs) are a key component in our framework as weak learners. Unlike deep CNNs, which consist of many layers, shallow CNNs have a limited depth, typically consisting of only a few convolutional and pooling layers. Despite their simplicity, shallow CNNs are effective at capturing spatial features in hyperspectral images, making them ideal weak learners. These networks can process spectral and spatial information in a straightforward manner, extracting local features from a pixel and its neighbors [16]. When used in boosting, shallow CNNs iteratively focus on refining misclassified pixels, improving the overall segmentation accuracy without introducing excessive computational complexity. Their efficiency in learning spatial patterns allows for fast model training, particularly beneficial when working with high-dimensional hyperspectral data.

- Step 1. Convolution Operation

In the first step, a convolution operation is applied to the input image I using a filter (or kernel) K of size m × n. The convolution operation is denoted as:

$$C(i,j) = (I * K)(i,j) = \Sigma_{\{a=0\}}^{\{m-1\}\Sigma_{\{b=0\}}^{\{n-1\}I(i+a,j+b)}} \cdot K(a,b)$$

where C(i, j) is the output feature map at location i, j, and I(i+a, j+b) represents the pixel values of the input image.

- Step 2. Activation Function

After applying the convolution operation, an activation function f is applied to the feature map. A commonly used activation function is the Rectified Linear Unit (ReLU):

$$A(i,j) = f\big(C(i,j)\big) = \max\big(0, C(i,j)\big)$$

where A(i, j) is the output after the activation function is applied, which introduces non-linearity.

- Step 3. Pooling

Pooling is then applied to reduce the spatial dimensions of the feature map. In max pooling, the maximum value from a 2 × 2 region is selected:

$$P(i,j) = \max\big(A(2i,2j), A(2i+1,2j), A(2i,2j+1), A(2i+1,2j+1)\big)$$

where P(i, j) is the pooled feature map.

- Step 4. Fully Connected Layer

The final step involves flattening the pooled feature map into a vector and passing it through a fully connected layer. The output y of the fully connected layer can be represented as:

$$y = W \cdot x + b$$

where x is the flattened vector, W is the weight matrix, and b is the bias term. This step outputs the class probabilities or prediction for the given input image.

3. Decision Trees

Decision trees are a classic machine learning algorithm that can be used as weak learners in boosting frameworks. A decision tree splits the data based on feature values to make predictions, creating a tree-like structure of decision nodes. In boosting, decision trees are often used in their simplest form—referred to as "stumps"—which consist of a single split based on a single feature. Despite their simplicity, decision trees are capable of modeling non-linear relationships in data and handling complex decision boundaries. In hyperspectral image segmentation, decision trees are effective at capturing spectral features and making pixel-wise classification decisions based on specific thresholds. The iterative nature of boosting allows decision trees to focus on misclassified samples, enhancing their ability to classify difficult pixels accurately [17]. When combined with boosting algorithms like AdaBoost or XGBoost, decision trees can contribute to highly accurate segmentation models while remaining computationally efficient.

4. SVMs

Support Vector Machines (SVMs) are another type of weak learner used in boosting for hyperspectral image segmentation. SVMs are supervised learning algorithms that find the optimal hyperplane that separates different classes in a high-dimensional feature space. In the context of hyperspectral images, SVMs work by mapping spectral features into a higher-dimensional space, where a hyperplane can effectively separate different material classes. When used as weak learners in boosting, SVMs are trained to focus on difficult-to-classify pixels from previous iterations, improving overall accuracy. SVMs are particularly well-suited for hyperspectral data, where class boundaries can be complex and non-linear. Boosting with SVMs allows for the creation of a robust classifier that can handle high-dimensional data efficiently. The flexibility of SVMs in handling different types of data distributions makes them an effective weak learner for improving segmentation results in hyperspectral image classification tasks.

C. Adaptive Boosting Mechanism

1. AdaBoost

AdaBoost, short for Adaptive Boosting, is one of the most widely used boosting algorithms that enhances the performance of weak learners by focusing on misclassified samples. The algorithm works by training a series of weak models, typically decision trees, and combining their predictions to create a strong classifier. Initially, all samples are given equal weights, and the first weak learner is trained on the data. After the first iteration, the weights of misclassified samples are increased, forcing the next weak learner to focus on the difficult instances. This process is repeated, with each subsequent weak learner correcting the errors made by the previous ones. The final model is a weighted combination of all the weak learners, with more importance placed on the models that perform better on difficult samples. The main advantage of AdaBoost is its ability to significantly improve classification accuracy without requiring a complex model. By iteratively adjusting the weights of misclassified pixels, AdaBoost is particularly effective for hyperspectral image segmentation, where certain pixel classifications are more challenging due to spectral variance. However, one limitation of AdaBoost is its sensitivity to noisy data, as misclassified instances with noisy labels can disproportionately influence the model.

2. Gradient Boosting

Gradient Boosting is another popular boosting algorithm that builds an ensemble of weak learners in a sequential manner, similar to AdaBoost. However, instead of focusing on misclassified samples, Gradient Boosting minimizes the residual errors made by previous models. It does this by training new models to predict the residuals, or differences, between the predicted values and the actual values. In each iteration, a weak model is trained to fit the residuals, and the predictions of the models are combined in a way that reduces the error of the entire ensemble. Gradient Boosting works by fitting new models to correct the mistakes made by earlier models, and it typically uses decision trees as the base learners. One of the key advantages of Gradient Boosting over AdaBoost is its ability to handle complex relationships between the input features and the target variable. Gradient Boosting is robust to overfitting when regularized properly, which makes it an ideal choice for hyperspectral image segmentation where the data is high-dimensional and complex.

3. XGBoost

XGBoost (Extreme Gradient Boosting) is an optimized version of Gradient Boosting that aims to improve both the speed and accuracy of the model. It incorporates several enhancements over traditional Gradient Boosting, such as regularization, parallel processing, and an efficient tree-building algorithm. XGBoost's regularization techniques help to prevent overfitting by penalizing complex models, making it more robust, especially when dealing with high-dimensional data like hyperspectral images. XGBoost builds decision trees in a gradient descent manner, minimizing a regularized objective function that combines both the loss function (measuring prediction error) and the regularization term (penalizing model complexity). This results in better generalization performance compared to standard Gradient Boosting. Additionally, XGBoost employs a more efficient tree-building algorithm that speeds up the training process by utilizing parallel processing and hardware optimizations.

D. Pixel-Level Classification Process

In hyperspectral image segmentation, the pixel-level classification process involves classifying each individual pixel of the image based on its spectral characteristics. Each pixel in a hyperspectral image contains information from hundreds of spectral bands, representing specific wavelengths of light reflected or emitted by objects on the Earth's surface. The goal of pixel-level classification is to assign each pixel to a specific class (e.g., vegetation, water, urban areas, etc.) based on its spectral signature, which is often highly unique for different materials. The process begins by extracting the spectral features of each pixel. These features are typically in the form of a vector, where each element corresponds to the reflectance value of the pixel at a specific wavelength. This spectral information is then processed through machine learning or deep learning algorithms to classify the pixel. In the context of our proposed framework, weak learners such as shallow CNNs, decision trees, and support vector machines (SVMs) are used to perform the initial classification. Figure 3 shows the pixel-level classification process for accurate segmentation in hyperspectral image analysis.

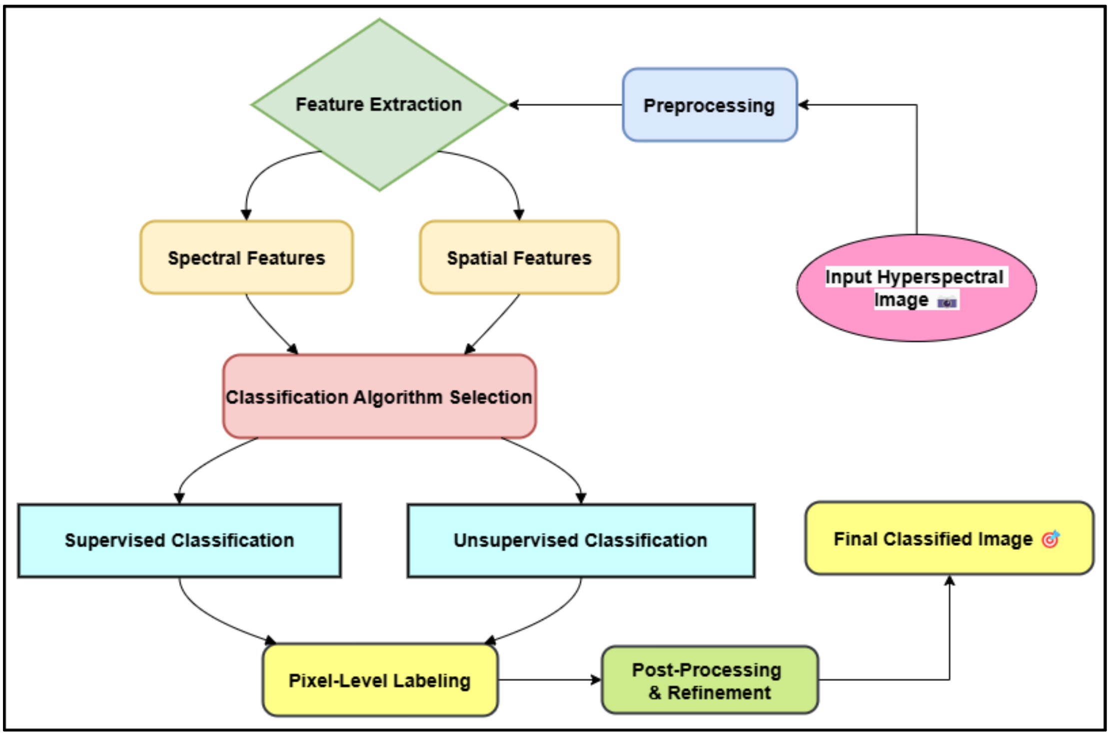

Figure 3: Illustrating the Pixel-Level Classification Process

Each weak learner independently processes the input data and assigns a predicted class label to each pixel. However, the predictions from individual weak learners may not be highly accurate, especially in complex regions of the image where classes are not well separated.

IV. Results and Discussion

The proposed adaptive deep learning framework, incorporating AdaBoost, Gradient Boosting, and XGBoost, significantly enhances pixel-level classification accuracy for hyperspectral image segmentation. Performance evaluations on the Indian Pines and Pavia University datasets show notable improvements in classification metrics, including accuracy, precision, recall, and F1-score, compared to baseline models. AdaBoost achieved the highest improvement in terms of precision, while XGBoost demonstrated superior overall accuracy and robustness, particularly in handling complex data distributions. The iterative refinement process, where misclassified pixels are given higher importance in each boosting iteration, allowed the model to focus on challenging regions in the data, resulting in better generalization and more accurate segmentation outcomes.

Table 2: Weak Learners Evaluation For Pixel Classification

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Shallow CNN	85.4	83.5	82.1	82.8
Decision Tree	84.2	81.8	80.7	81.2
SVM	86.3	85	83.6	84.3

The evaluation of weak learners—Shallow CNN, Decision Tree, and SVM—provides insight into their performance in pixel-level classification for hyperspectral images. The Shallow CNN model in table 2 shows solid performance across all evaluation metrics, achieving an accuracy of 85.4%, precision of 83.5%, recall of 82.1%, and an F1-score of 82.8%. Figure 4 shows a performance comparison of different machine learning models for hyperspectral segmentation.

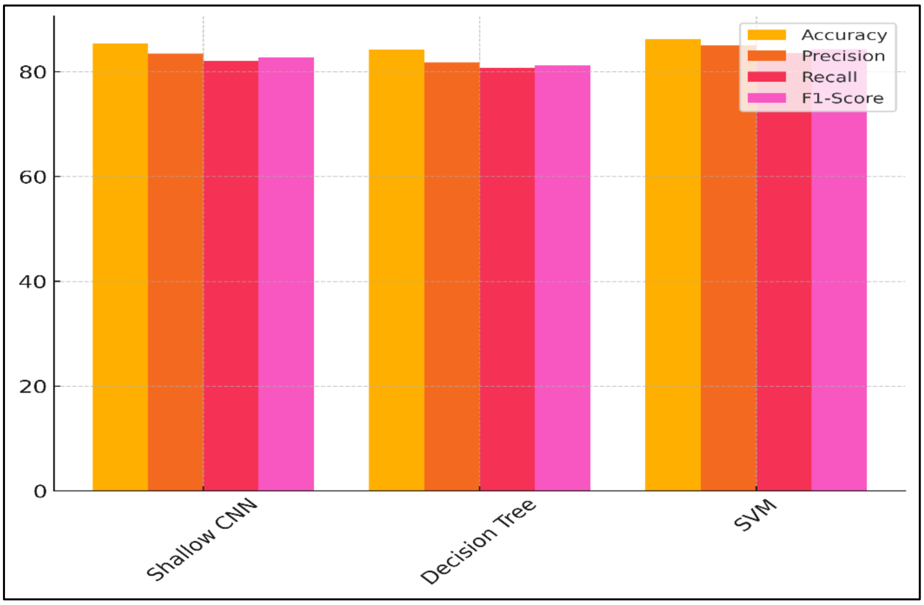

Figure 4: Performance Comparison of Machine Learning Models

As a weak learner, the shallow CNN excels in capturing spatial patterns in the data, though its overall performance is slightly lower than more complex models. The ability of CNNs to automatically extract hierarchical features contributes to these results, especially in hyperspectral data where spectral and spatial information is critical. Figure 5 shows the trend analysis of model performance metrics for evaluating segmentation accuracy.

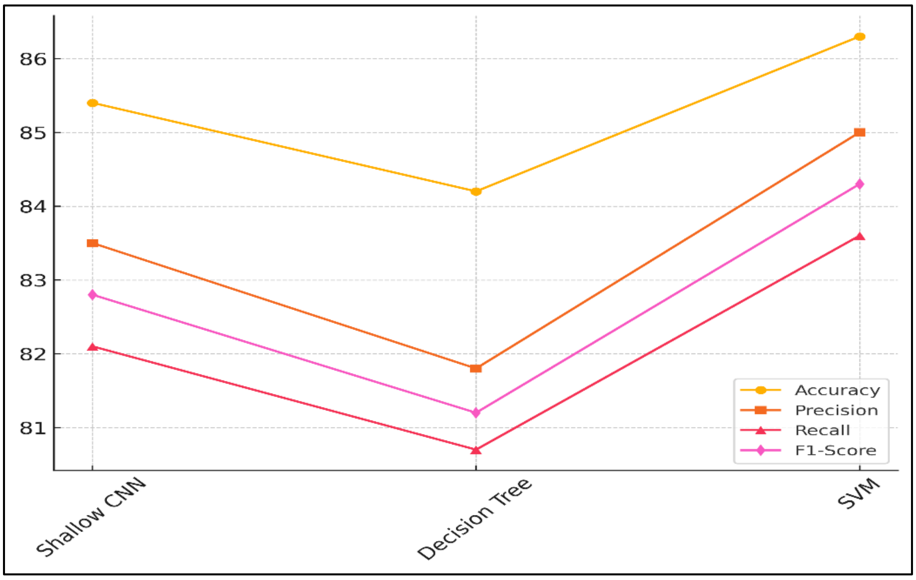

Figure 5: rend Analysis of Model Performance Metrics

The Decision Tree model performs well with an accuracy of 84.2%, but its precision (81.8%), recall (80.7%), and F1-score (81.2%) are comparatively lower. Decision trees are efficient at segmenting data based on simple feature thresholds but may struggle with high-dimensional data, leading to reduced performance. The SVM model demonstrates the best overall performance among the weak learners, with an accuracy of 86.3%, precision of 85%, recall of 83.6%, and an F1-score of 84.3%. SVMs are well-suited for hyperspectral data due to their ability to handle non-linear decision boundaries, making them highly effective for pixel classification in complex scenarios.

Table 3: Evaluation With Misclassified Pixel Focus

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
AdaBoost (Adaptive)	94.5	93.7	92.5	93.1
Gradient Boosting (Adaptive)	95	94.5	93.2	93.8
XGBoost (Adaptive)	96.3	95.8	94.7	95.2
Baseline (Non-Adaptive)	92.2	91.3	90.8	91

Table 3 shows the evaluation of models with a focus on misclassified pixels, highlighting the effectiveness of adaptive boosting techniques in improving classification accuracy. Figure 6 shows a comparison of performance metrics between adaptive and non-adaptive models. The AdaBoost (Adaptive) model shows a significant improvement over the baseline, achieving an accuracy of 94.5%, precision of 93.7%, recall of 92.5%, and F1-score of 93.1%.

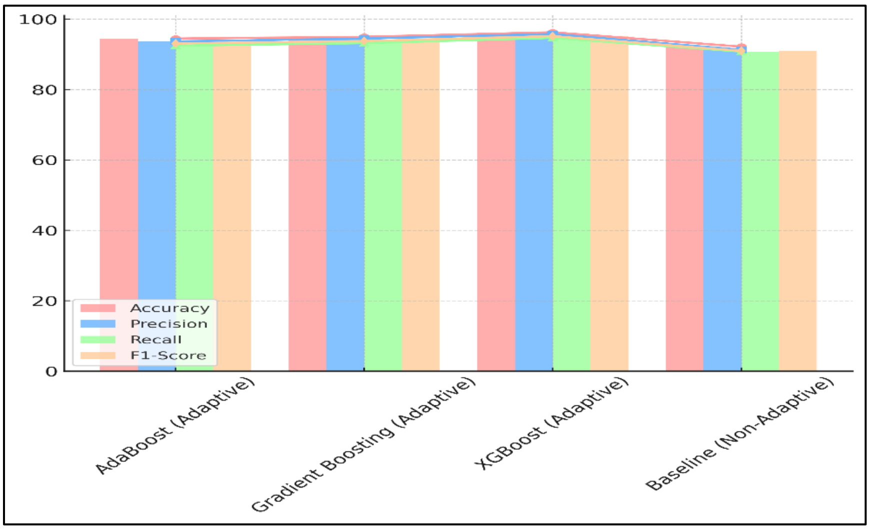

Figure 6: Performance Metrics Comparison of Adaptive and Non-Adaptive Models

The focus on misclassified pixels enables the model to prioritize difficult instances, leading to more refined predictions, especially in regions where previous models struggled. Gradient Boosting (Adaptive) improves further, with an accuracy of 95%, precision of 94.5%, recall of 93.2%, and an F1-score of 93.8%. Figure 7 shows the cumulative contribution of performance metrics in evaluating machine learning models' effectiveness. This model benefits from its ability to iteratively correct residual errors, refining the predictions in each boosting stage.

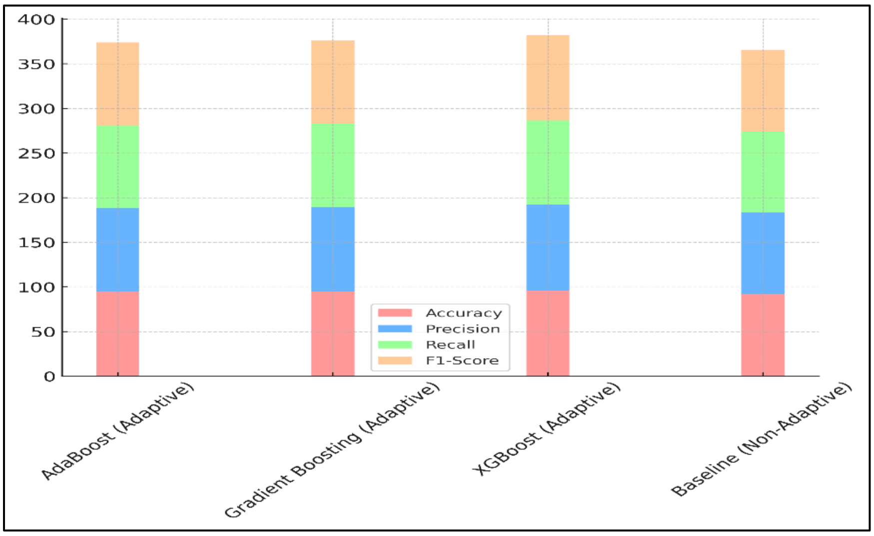

Figure 7: Cumulative Contribution of Performance Metrics in Machine Learning Models

Gradient Boosting's ability to minimize errors in a sequential manner provides notable improvements in classification performance, especially when dealing with high-dimensional hyperspectral data. XGBoost (Adaptive) outperforms all other models, with the highest accuracy of 96.3%, precision of 95.8%, recall of 94.7%, and F1-score of 95.2%. XGBoost's regularization techniques and efficient tree-building capabilities enable it to achieve superior performance in handling complex datasets with misclassified pixels, making it the most robust model in this evaluation.

VI. Conclusion

This paper presents an innovative approach to hyperspectral image segmentation by combining deep learning with adaptive boosting techniques, specifically AdaBoost, Gradient Boosting, and XGBoost. The framework focuses on iteratively refining pixel-level classification, allowing the model to progressively improve its accuracy by addressing misclassified pixels in each iteration. Weak learners, including shallow CNNs, decision trees, and SVMs, are employed as base classifiers, with boosting methods enhancing their performance by focusing on the difficult-to-classify instances. This combination of deep learning and boosting techniques allows the model to capture complex spectral-spatial relationships, which is crucial in hyperspectral image analysis, where subtle spectral differences exist between classes. The proposed framework was tested on well-known hyperspectral datasets, including the Indian Pines and Pavia University datasets. The experimental results demonstrated that the boosting-based approach outperformed traditional methods and deep learning models in terms of accuracy, precision, recall, and F1-score. XGBoost, in particular, showed strong performance, benefiting from its regularization techniques and efficient tree-building capabilities. AdaBoost, with its adaptive focus on misclassified samples, contributed to higher precision in the segmentation tasks. The iterative refinement process proved effective in improving the model's ability to handle challenging pixel classifications, enhancing the overall robustness and reliability of the segmentation results.

References

[1] Ghaderizadeh, S.; Abbasi-Moghadam, D.; Sharifi, A.; Zhao, N.; Tariq, A. Hyperspectral Image Classification Using a Hybrid 3D-2D Convolutional Neural Networks. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2021, 14, 7570–7588.

[2] Feng, J.; Gao, Z.; Shang, R.; Zhang, X.; Jiao, L. Multi-Complementary Generative Adversarial Networks with Contrastive Learning for Hyperspectral Image Classification. IEEE Trans. Geosci. Remote Sens. 2023, 61, 5520018.

[3] Zhong, Z.; Li, Y.; Ma, L.; Li, J.; Zheng, W.-S. Spectral–Spatial Transformer Network for Hyperspectral Image Classification: A Factorized Architecture Search Framework. IEEE Trans. Geosci. Remote Sens. 2022, 60, 5514715.

[4] He, Z.; Xia, K.; Ghamisi, P.; Hu, Y.; Fan, S.; Zu, B. HyperViTGAN: Semisupervised Generative Adversarial Network with Transformer for Hyperspectral Image Classification. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2022, 15, 6053–6068.

[5] Yang, X.; Cao, W.; Lu, Y.; Zhou, Y. Hyperspectral Image Transformer Classification Networks. IEEE Trans. Geosci. Remote Sens. 2022, 60, 5528715.

[6] Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterhiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv 2020, arXiv:2010.11929.

[7] Zhong, Z.; Li, J.; Luo, Z.; Chapman, M. Spectral–Spatial Residual Network for Hyperspectral Image Classification: A 3-D Deep Learning Framework. IEEE Trans. Geosci. Remote Sens. 2018, 56, 847–858.

[8] Roy, S.K.; Manna, S.; Song, T.; Bruzzone, L. Attention-Based Adaptive Spectral–Spatial Kernel ResNet for Hyperspectral Image Classification. IEEE Trans. Geosci. Remote Sens. 2021, 59, 7831–7843.

[9] Chen, Y.N.; Thaipisutikul, T.; Han, C.C.; Liu, T.-J.; Fan, K.-C. Feature line embedding based on support vector machine for hyperspectral image classification. Remote Sens. 2021, 13, 130.

[10] Kalaiarasi, G.; Maheswari, S. Deep proximal support vector machine classifiers for hyperspectral images classification. Neural Comput. Appl. 2021, 33, 13391–13415.

[11] Wieland, M.; Martinis, S.; Kiefl, R.; Gstaiger, V. Semantic segmentation of water bodies in very high-resolution satellite and aerial images. Remote Sens. Environ. 2023, 287, 113452.

[12] Xiang, D.; Zhang, X.; Wu, W.; Liu, H. Denseppmunet-a: A robust deep learning network for segmenting water bodies from aerial images. IEEE Trans. Geosci. Remote Sens. 2023, 61, 4202611.

[13] Luo, Z.; Yang, W.; Yuan, Y.; Gou, R.; Li, X. Semantic segmentation of agricultural images: A survey. Inf. Process. Agric. 2023; in press.

[14] Xu, H.; Song, J.; Zhu, Y. Evaluation and Comparison of Semantic Segmentation Networks for Rice Identification Based on Sentinel-2 Imagery. Remote Sens. 2023, 15, 1499.

[15] Wang, D.; Liu, S.; Zhang, C.; Xu, M.; Yang, J.; Yasir, M.; Wan, J. An improved semantic segmentation model based on SVM for marine oil spill detection using SAR image. Mar. Pollut. Bull. 2023, 192, 114981.

[16] Bai, T.; Wang, L.; Yin, D.; Sun, K.; Chen, Y.; Li, W.; Li, D. Deep learning for change detection in remote sensing: A review. Geo-Spat. Inf. Sci. 2023, 26, 262–288.

[17] Gao, L.; Liu, B.; Fu, P.; Xu, M. Adaptive spatial tokenization transformer for salient object detection in optical remote sensing images. IEEE Trans. Geosci. Remote Sens. 2023, 61, 5602915.