

A Hybrid Random Forest and CNN Framework for Tile-Wise Oil-Water Classification in Hyperspectral Images

Mehdi Nickzamir

Politecnico di Torino

Turin, Italy

s323959@studenti.polito.it

S. Mohammad Sheikh Ahmadi

Politecnico di Torino

Turin, Italy

s327914@studenti.polito.it

Abstract

This paper presents a novel hybrid Random Forest and Convolutional Neural Network (CNN) framework for oil-water classification in hyperspectral images (HSI). To address the challenge of preserving spatial context, we divided the images into smaller, non-overlapping tiles, which served as the basis for training, validation, and testing. Random Forest demonstrated strong performance in pixel-wise classification, outperforming models such as XGBoost, Attention-Based U-Net, and HybridSN. However, Random Forest loses spatial context, limiting its ability to fully exploit the spatial relationships in hyperspectral data. To improve performance, a CNN was trained on the probability maps generated by the Random Forest, leveraging the CNN's capacity to incorporate spatial context. The hybrid approach achieved 7.6% improvement in recall (to 0.85), 2.4% improvement in F1 score (to 0.84), and 0.54% improvement in AUC (to 0.99) compared to the baseline. These results highlight the effectiveness of combining probabilistic outputs with spatial feature learning for context-aware analysis of hyperspectral images.

Keywords

Hyperspectral Image Classification, Oil Spill Detection, Tile-Wise Classification, Random Forest, Convolutional Neural Network (CNN), Hybrid Learning Framework

1 Introduction

Oil spills have severe ecological, economic, and health consequences. In marine ecosystems, they threaten aquatic species, contaminate seafood, and disrupt the food chain, posing risks to human health. Additionally, oil spills impose financial and reputational damage on companies through fines and legal liabilities[2]. Timely detection is critical to mitigate these effects and prevent further damage.

Hyperspectral imaging (HSI) is a powerful tool for detecting oil spills, capturing detailed spectral information across hundreds of wavelengths. Unlike traditional RGB images, HSI identifies unique spectral “fingerprints” of oil spills, even when they are invisible to the naked eye. However, its effective use faces challenges, including high costs of airborne data collection, noise from environmental factors, and the need for robust models capable of handling high-dimensional data.

Data scarcity is another major hurdle, as obtaining labeled hyperspectral datasets is costly and time-intensive. While unsupervised methods avoid the need for labeled data, they often underperform compared to supervised techniques. Supervised methods like Random Forest excel at pixel-wise classification by leveraging spectral features but fail to incorporate spatial context, which is essential for producing coherent predictions across images.

To address these challenges, we propose a hybrid framework that combines Random Forest and Convolutional Neural Networks (CNNs). Random Forest handles pixel-wise classification, while CNNs capture spatial dependencies by refining the probabilistic outputs of the RF model. This combination balances spectral feature extraction and spatial learning, improving accuracy and consistency. Additionally, we adopt a tile-wise division of images to enhance model generalization. This approach enables faster and more reliable oil spill detection, contributing to environmentally conscious and efficient monitoring systems.

The implementation and dataset used in this study are publicly available on GitHub: [GitHub Repository].

2 Related Works

Oil-water classification using hyperspectral images (HSI) has gained increasing attention due to its environmental significance, particularly in detecting and monitoring oil spills. This section focuses on recent advancements in HSI classification, with an emphasis on works closely aligned with the Hyperspectral Oil Spill Detection (HOSD) dataset, introduced by Puhong Duan.[1]

The HOSD dataset, contains 18 hyperspectral images captured by the ARVIS sensor during the Deepwater Horizon oil spill. These images cover a spectral range of 365 nm to 2500 nm and are accompanied by reference maps that classify each pixel as either oil or non-oil. It is a valuable resource for advancing oil-water classification research.

Early work on the HOSD dataset explored unsupervised detection techniques for pixel level classification, such as the isolation forest (iForest)-based framework proposed by [1]. This method employed a Gaussian statistical model to preprocess noisy spectral bands, followed by dimensionality reduction using kernel principal component analysis (KPCA). Probabilistic outputs from the iForest were refined using clustering algorithms and a support vector machine (SVM), achieving competitive image-wise classification accuracy. While effective, the approach did not integrate spatial relationships, which are critical for context-aware analysis in oil-water classification [1].

To address the limitations of pixel-wise methods, multiscale spectral-spatial learning frameworks have emerged as a promising direction for HSI classification. A notable contribution in this area is the multiscale spectral-spatial CNN (HyMSCN), which introduced an image-based classification framework designed to improve processing efficiency by integrating features from multiple receptive fields [3]. Unlike patch-based approaches, this method minimized redundancy in testing and effectively fused multiscale features to enhance classification accuracy. While the approach achieved strong performance on general-purpose hyperspectral datasets, it was not

tailored for domain-specific datasets like HOSD, highlighting the need for specialized frameworks that address the unique spectral and spatial characteristics of oil spill imagery [3].

Building upon the strengths of spectral-spatial integration, some efforts have emphasized the importance of contextual learning in HSI classification. The contextual CNN(HybridSN) proposed by [4] demonstrated how a multi-scale convolutional filter bank could effectively exploit spatio-spectral relationships, producing a unified feature map for accurate pixel-wise classification. By leveraging deeper and wider architectures, this approach achieved high-ranking performance on standard datasets, such as Indian Pines and Salinas, underscoring the potential of contextual learning in enhancing classification accuracy [4].

While these approaches provide valuable insights, our method simplifies the training process by adopting a CNN on 2D probabilistic images generated by Random Forest rather than directly processing 3D hyperspectral data. This hybrid framework bridges the gap between probabilistic modeling and spatial feature learning, achieving superior oil-water classification performance on the HOSD dataset. By combining the strengths of Random Forest in both pixel-level and tile-wise classification with CNNs' capacity to incorporate spatial context, the proposed approach offers a practical and efficient solution for context-aware analysis of hyperspectral images.

3 Data Preprocessing

3.1 Dataset

The dataset used in this study is the publicly available Hyperspectral Oil Spill Detection (HOSD) dataset [1], which contains hyperspectral images documenting the Deepwater Horizon oil spill. The dataset has 18 images showing spectra from 365 nm to 2500 nm, labeled as oil or non-oil. The spatial resolution of the images varies due to different flight altitudes during data collection, as shown in Table 1. The dataset's class imbalance (95% water, 5% oil) and the complexity of hyperspectral data present unique challenges, as depicted in Figure 1.

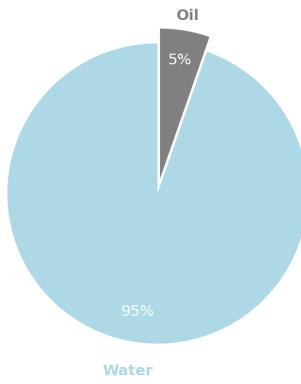


Figure 1: Distribution of Water and Oil Labels through the dataset.

The preprocessing steps applied to address these challenges are summarized in the pipeline shown in Figure 3. These steps include

Table 1: Some features of the HOSD dataset.

Data	Spatial Size	Resolution	Flight Time
GM1	1200×633	7.6m	5/17/2010
GM2	1881×693	7.6m	5/17/2010
GM3	1430×691	7.6m	5/17/2010
GM4	1700×691	7.6m	5/17/2010
GM5	2042×673	7.6m	5/17/2010
GM6	2128×689	8.1m	5/18/2010
GM7	2302×479	3.3m	7/09/2010
GM8	1668×550	3.3m	7/09/2010
GM9	1643×447	3.2m	7/09/2010
GM10	1110×675	7.6m	5/17/2010
GM11	1206×675	7.6m	5/17/2010
GM12	869×649	7.6m	5/06/2010
GM13	1135×527	3.2m	7/09/2010
GM14	1790×527	3.2m	7/09/2010
GM15	1777×510	3.3m	7/09/2010
GM16	1159×388	3.2m	7/09/2010
GM17	1136×660	7.6m	5/17/2010
GM18	1047×550	3.3m	7/09/2010

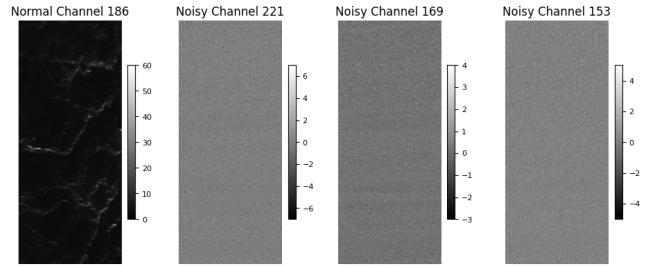


Figure 2: Visualization of three noisy channels and one normal channel in Image 14.

noisy channel removal, normalization, dimensionality reduction using PCA, tiling, and data augmentation. Each of these processes is detailed in the following subsections.

3.2 Noisy Channel Removal and Normalization

To improve dataset quality and facilitate dimensionality reduction, we carefully inspected each hyperspectral channel to detect and remove noisy ones. Noisy channels, characterized by low variance and lack of discernible patterns, fail to provide meaningful information for oil spill detection. Figure 2 illustrates examples of noisy and normal channels in the dataset.

To ensure consistency across the dataset, noisy channels were identified by intersecting channels marked as noisy across all images. This process resulted in the removal of 31 channels, as listed below:

[103, 106, 107, 108, 109, 110, 111, 112, 113, 114, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 221, 222, 223].

After removing noisy channels, normalization was applied to ensure that all channels contributed equally to variance. Figure 4

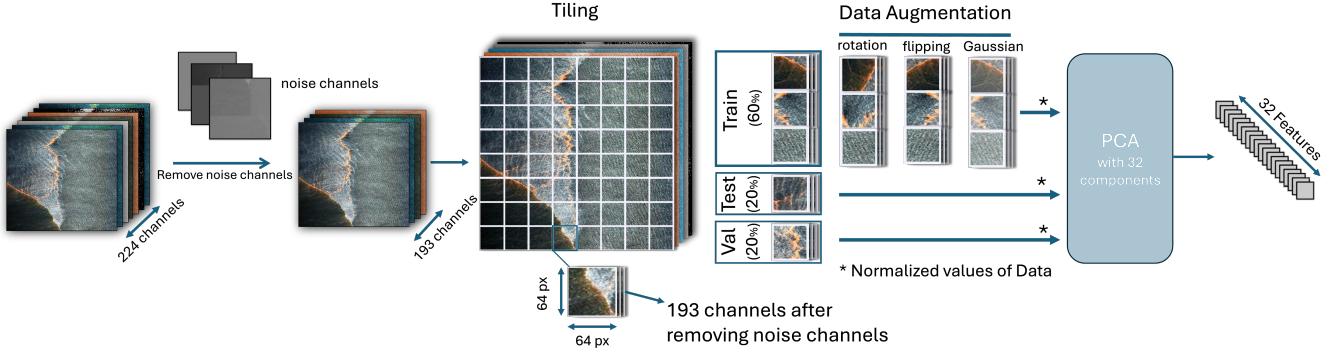


Figure 3: Pipeline of Data Preprocessing

highlights the disparity in value ranges across channels, necessitating normalization.

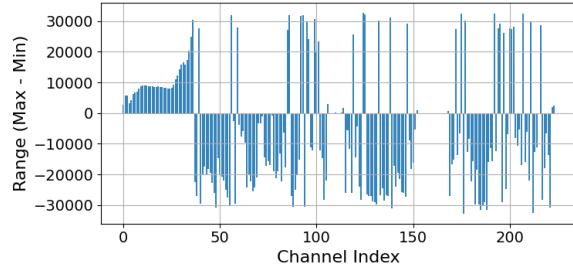


Figure 4: Range of Values in Data

StandardScaler was used to transform each channel to have a mean of 0 and a standard deviation of 1:

$$x_{\text{normalized}} = \frac{x - \mu}{\sigma} \quad (1)$$

Where:

- x is the original feature value,
- μ is the mean of the feature values,
- σ is the standard deviation of the feature values.

Normalization ensured that Principal Component Analysis (PCA) focused on capturing meaningful patterns without being biased by differences in scale.

3.3 Dimensionality Reduction

Although noisy channels were removed during preprocessing, the dataset remained high-dimensional due to the large number of spectral channels, posing challenges for computational efficiency and increasing the risk of overfitting.

To address this, Principal Component Analysis (PCA) was applied, reducing the dataset to 32 principal components while preserving 99% of its variance, as shown in Figure 5. This reduction significantly simplified the dataset, retaining essential spectral information and improving its representation of meaningful features. Notably, the removal of noisy channels reduced the number of components required to explain 99% of the variance from 60 to 32, highlighting the benefits of preprocessing.

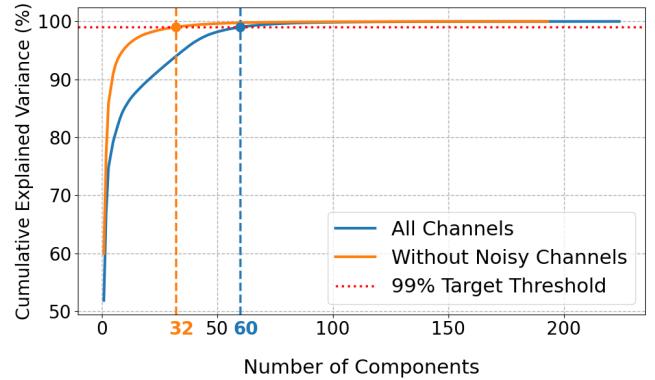


Figure 5: Cumulative explained variance vs. PCA components

3.4 Tiling and Data Splitting into Train, Validation, and Test Sets

To ensure robust training and evaluation, we divided each hyperspectral image into smaller, non-overlapping tiles of size 64*64, instead of assigning entire images to specific splits. This approach avoided overfitting, as the model was not confined to learning the context of some images while leaving others unrepresented.

After tiling, the tiles were shuffled, ensuring a fair distribution of data across the training, validation, and test sets. The shuffled tiles were then allocated as follows: 60% for training, 20% for validation, and 20% for testing. This method distributed spatial contexts from different regions of the dataset into each subset.

For edge tiles, which contained partially empty spaces due to the dimensions of the original images, we applied a padding strategy. These empty spaces were filled with a constant value calculated as the average reflectance of water for the corresponding channels in the affected tile. This ensured data consistency while preserving the integrity of the dataset.

3.5 Data Augmentation

To address the challenge of data scarcity, despite the dataset being relatively larger, in terms of number of images, compared to those used in similar studies, we applied effective data augmentation

techniques: rotation, flipping, and Gaussian noise. The primary objective of these augmentations was to increase the diversity of the dataset while ensuring that no artificial patterns or anomalies were introduced, which could otherwise interfere with the model's learning process.

An interesting and deliberate choice in our augmentation strategy was to exclude water-only tiles from augmentation. This decision was driven by the significant class imbalance in the dataset, where approximately 95% of the labels represented water and only 5% represented oil spills before augmentation. By focusing augmentation efforts solely on tiles containing oil spills, we aimed to improve the balance within the training set and enhance the model's ability to recognize minority-class patterns. As a result, the proportion of oil tiles in the training set increased to approximately 10%, compared to 5% in the validation and test sets, which remained representative of the real-world distribution.

The final dataset distribution of tiles is as follows:

- **Training set:** 6,804 tiles
- **Validation set:** 759 tiles
- **Test set:** 780 tiles

4 Methodology

4.1 Model Choice

In designing our hybrid framework, we carefully selected two complementary models: Random Forest (RF) and Convolutional Neural Networks (CNNs), each addressing specific challenges of hyperspectral image (HSI) analysis.

Random Forest was chosen as the primary model due to its robustness in handling high-dimensional data and its ability to process noisy data effectively. Our dataset, despite being reduced is still high dimensional, and RF's ensemble-based architecture excels in isolating meaningful features from such complex data without requiring extensive parameter tuning. Additionally, RF performs well even with relatively small datasets, which is a critical factor given the scarcity of labeled HSI datasets.

Conversely, CNNs were incorporated to address a key limitation of Random Forest: the lack of spatial awareness. RF operates on individual pixels without considering relationships between neighboring pixels, which can lead to fragmented or inconsistent predictions. CNNs, with their convolutional layers, are adept at capturing spatial context, enabling the model to learn patterns and structures that span multiple pixels. This capability is particularly important for HSIs, where spatial features often complement spectral information in distinguishing oil spills from water.

The hybrid framework leverages the strengths of both models, combining RF's ability to handle high-dimensional noisy data with CNN's capacity to refine predictions by incorporating spatial context. An overview of the hybrid framework is illustrated in Figure 6, which shows how the probabilistic outputs of RF are utilized as inputs for CNN to produce refined predictions.

4.2 Training the Random Forest Model

The dataset was flattened to ensure compatibility with the Random Forest (RF) model. Each pixel, represented by 32 principal components was treated as an independent data point. The RF model was

trained to perform pixel-wise classification, predicting the likelihood of each pixel belonging to the "oil" or "non-oil" class.

Random Forest was configured with the following hyperparameters:

- **Number of Trees:** 100 — ensures a balance between model accuracy and computational efficiency.
- **Random State:** 42 — ensures reproducibility of results.
- **Max Features:** sqrt — uses the square root of the number of features for splitting, a commonly effective default.

The model achieved reliable results with the baseline hyperparameters, demonstrating its robustness. However, due to resource constraints, we did not perform hyperparameter fine-tuning. The dataset's size and memory requirements posed a challenge, as RF's ensemble nature involves maintaining multiple decision trees in memory simultaneously.

4.3 CNN Architecture and Training

4.3.1 Architecture. In the second phase of our hybrid framework, we employ a Convolutional Neural Network (CNN) to refine the probabilistic predictions generated by the Random Forest (RF) model.

The CNN architecture consists of three main components:

1. Downsampling Path (Encoder): The encoder extracts hierarchical features from the input probability maps using convolutional layers with ReLU activations and same padding to retain spatial dimensions. Max-pooling is applied to progressively reduce spatial resolution, allowing the network to capture broader contextual information.

2. Upsampling Path (Decoder): The decoder restores the spatial resolution of feature maps using transposed convolutional layers. Skip connections directly link encoder and decoder features, enabling the integration of low-level spatial details with high-level abstractions. This enhances the refinement of predictions.

3. Output Layer: A final convolutional layer with a sigmoid activation function produces a refined probability map, where each pixel represents the likelihood of belonging to the "oil" class.

4.3.2 Training. The trained RF model generated probabilistic predictions for the validation and test sets, which were then reshaped into 64×64 tiles matching the CNN's input format. To ensure unbiased training, the validation set's prediction maps were split into a new training and validation set, with 80% allocated for training and 20% for validation. This approach prevented the CNN from training on biased probability maps. Since the RF model had already seen the original training data, using its predictions from that set would have created overly optimistic maps too closely aligned with the ground truth. Instead, RF predictions on the validation set provided unbiased and realistic probability maps that were unseen during the RF model's training. By training the CNN on these representative predictions, the model learned to refine its outputs effectively. The best-performing model was selected based on its validation AUC score. To mitigate overfitting, training was limited to 50 epochs due to the small number of tiles, which increased the risk of memorization.

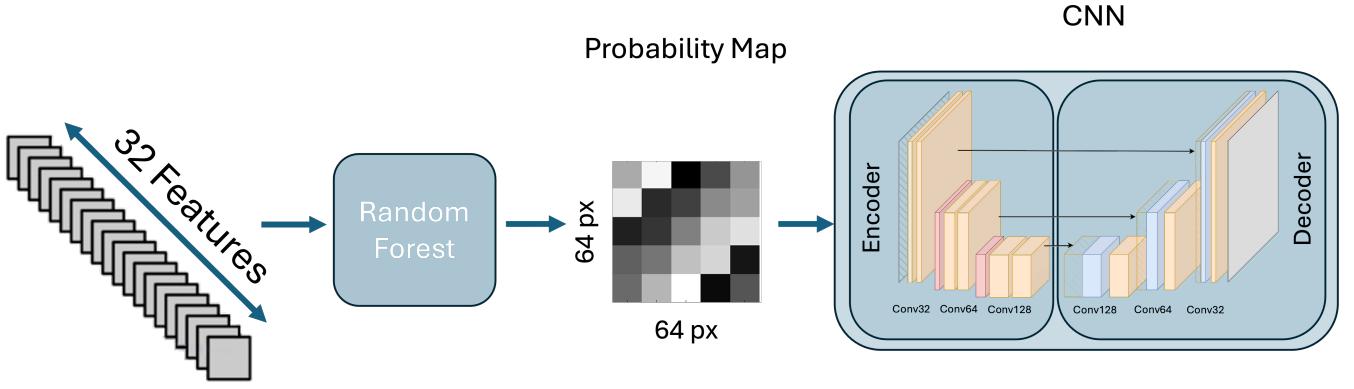


Figure 6: Overview of the Hybrid RF+CNN Framework for Oil Spill Detection

4.4 Metrics

To assess the performance of our hybrid framework, we used five metrics: **Accuracy**, **Recall**, **Precision**, **F1-Score**, and **AUC-ROC**. However, due to the severe imbalance in the dataset (95% water and 5% oil), we prioritized **F1-Score** and **AUC** over accuracy, as the latter can be misleading in such scenarios. For example, a model predicting all labels as "water" would achieve a 95% accuracy but fail entirely to detect oil.

F1-Score, the harmonic mean of precision and recall, was emphasized because it balances the trade-off between detecting oil pixels (recall) and minimizing false positives (precision). The **AUC-ROC**, which measures the model's ability to distinguish between classes across various thresholds, complements F1-Score by providing an overall measure of classification performance.

The formulas for the key metrics are as follows:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \quad (2)$$

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}} \quad (3)$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

$$\text{AUC} = \int_0^1 TPR(FPR) d(FPR) \quad (5)$$

Where:

- TPR is the True Positive Rate, defined as $\frac{\text{TP}}{\text{TP} + \text{FN}}$.
- FPR is the False Positive Rate, defined as $\frac{\text{FP}}{\text{FP} + \text{TN}}$.

By focusing on F1-Score and AUC, we ensured that the evaluation reflected the model's ability to balance precision and recall while effectively detecting the minority class (oil) in the highly imbalanced dataset.

5 Experiments and Results

In this section, we present the results of various models tested on the Hyperspectral Oil Spill Detection (HOSD) dataset. The primary goal of these experiments was to evaluate the performance of different methods, including Random Forest (RF), XGBoost (XGB), HybridSN,

and Attention-based U-Net, and to compare them with our hybrid RF+CNN approach.

5.1 Baseline and Comparison Models

Baseline Model (Random Forest): Random Forest served as our baseline model due to its robustness in handling high-dimensional data and small datasets. It provided a strong starting point for pixel-wise classification, demonstrating high accuracy and acceptable F1 and AUC scores despite its lack of spatial awareness.

XGBoost (XGB): We tested XGBoost as an alternative ensemble learning method, but its performance was slightly lower than RF. XGB demonstrated effective feature handling but did not significantly outperform RF in terms of F1 and AUC.

Attention-Based U-Net: To evaluate whether a complex spatial-spectral model could perform better, we tested an attention-based U-Net. However, its performance was suboptimal due to the small dataset size, leading to overfitting and reduced generalization.

HybridSN: HybridSN, a spectral-spatial deep learning model proposed in prior researches, was also tested. While it outperformed the attention-based U-Net, it underperformed the Random Forest baseline model.

Proposed Hybrid RF+CNN: Our hybrid RF+CNN model was designed to combine the strengths of both RF and CNNs. This approach not only outperformed the baseline but also demonstrated improved consistency across tiles compared to the base model.

5.2 Results

The performance of all models is summarized in Table 2. While RF achieved competitive results, our hybrid RF+CNN approach outperformed it in key metrics such as F1 and AUC, demonstrating the effectiveness of incorporating spatial context.

5.3 Consistency Analysis

In addition to average performance metrics, we evaluated the consistency of predictions across tiles using F1 distributions. Analyzing tile-wise prediction histograms is important because average metrics, such as overall F1 score, can sometimes mask inconsistencies. A model may perform exceptionally well in certain tiles while failing

Model Name	Accuracy	Precision	Recall	F1	AUC
RF (Base Model)	0.9810	0.8409	0.7921	0.8158	0.9886
RF+CNN	0.9822	0.8201	0.8519	0.8357	0.9940
XGBoost (XGB)	0.9796	0.8264	0.7800	0.8025	0.9883
Attention U-Net	0.9769	0.7912	0.7682	0.7795	0.9861
HybridSN	0.9792	0.7814	0.8443	0.8116	0.9919

Table 2: Performance Comparison of Models

in others, resulting in a high average performance but poor reliability in specific contexts. By examining the distribution of F1 scores, we gain deeper insights into how often the model struggles to make accurate predictions, enabling us to identify and address the contexts where the model is less effective. This analysis is particularly critical for hyperspectral oil spill detection, where environmental conditions and data quality vary widely across regions. Figure 7 illustrates the distribution of F1 scores for tiles predicted by the RF baseline and the RF+CNN hybrid approach.

Baseline RF: Approximately 32% of tiles achieved F1 scores below 0.7.

Hybrid RF+CNN: With the hybrid approach, the proportion of tiles with F1 scores below 0.7 dropped to 24%.

The decrease in tiles with low F1 scores in the hybrid model shows how well RF’s ability to handle spectral features and CNN’s spatial context awareness work together. This improvement is vital for real-world applications, where consistent performance in different environmental conditions is key to reliable oil spill detection.

6 Conclusion

This study introduced a hybrid framework combining Random Forest (RF) and a Convolutional Neural Network (CNN) for hyperspectral oil spill detection. By leveraging RF’s robustness in handling high-dimensional, noisy data and CNN’s ability to capture spatial context, the framework effectively addressed the challenges of class imbalance and limited data.

Experimental results demonstrated that the hybrid RF+CNN model consistently outperformed other tested methods, including XGBoost, HybridSN, and attention-based U-Net, achieving higher F1-Score and AUC—key metrics for imbalanced datasets. Additionally, the hybrid model showed improved consistency across tiles, significantly reducing the proportion of low F1 scores compared to the RF baseline.

This framework highlights the potential of combining spectral and spatial features for effective hyperspectral image classification. Future work could explore the integration of advanced spatial-spectral techniques or domain-specific data augmentation strategies to further enhance the model’s robustness and generalization. Moreover, scaling this approach to larger datasets or real-time monitoring scenarios could unlock new possibilities for environmental applications and remote sensing.

References

- [1] Puhong Duan, Xudong Kang, Pedram Ghamisi, and Shutao Li. 2023. Hyperspectral Remote Sensing Benchmark Database for Oil Spill Detection With an Isolation Forest-Guided Unsupervised Detector. *IEEE Transactions on Geoscience and Remote Sensing* 61 (2023), 1–13. <https://doi.org/10.1109/TGRS.2023.3268944>

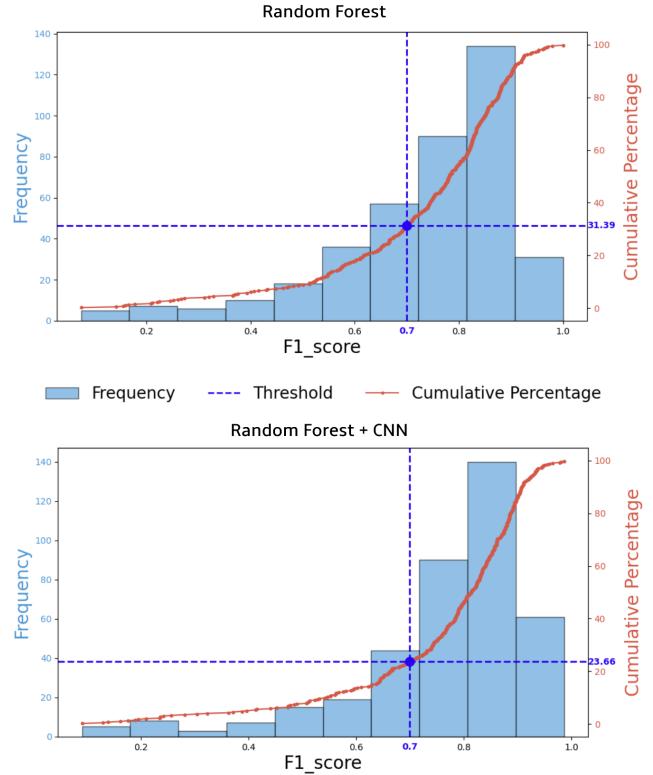


Figure 7: F1 score distributions across tiles for RF and RF+CNN.

- [2] Nada El Moussaoui and Brahim Idelhakkar. 2023. The Impact of Oil Spills on the Economy and the Environment. *European Journal of Economic and Financial Research* 7, 4 (2023), 69–85. <https://oapub.org/soc/index.php/EJEFR/article/view/1570>
- [3] Zhen Xu, Haoyang Yu, Ke Zheng, Lianru Gao, and Meiping Song. 2021. A Novel Classification Framework for Hyperspectral Image Classification Based on Multi-scale Spectral-Spatial Convolutional Network. In *2021 Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*. 1–5. <https://doi.org/10.1109/WHISPERS52293.2021.9483998>
- [4] Liangpei Zhang, Lefei Zhang, and Bo Du. 2020. Going Deeper with Contextual CNN for Hyperspectral Image Classification. *IEEE Transactions on Image Processing* 29 (2020), 4041–4054. <https://doi.org/10.1109/TIP.2020.2965939>