POLITECNICO DI TORINO

POLITECNICO DI TORINO

Master Degree course in Artificial Intelligence and Data Analytics

Master Degree Thesis

Wavelength selection for machine learning and deep learning models in leaf classification

Supervisors
Prof. Renato Ferrero
PhD. Nicola Dilillo

Candidate
Hossein Kakavand

Academic Year 2025-2026

Acknowledgements

I want to thank everyone who has helped me in my academic career. Above all, I am very grateful to my supervisor, Prof. Renato Ferrero, for his support and guidance during this thesis. I also deeply appreciate Nicola Dilillo for his helpful discussions and kind assistance; his guidance was very important to this project's success. Finally, I would like to thank my family and friends for their endless love and encouragement.

Abstract

Hyperspectral imaging sensors capture data across hundreds of contiguous narrow spectral bands, allowing precise material identification that standard RGB cameras cannot achieve. However, this high dimensionality introduces the Hughes phenomenon, where classification accuracy degrades unless the number of training samples increases exponentially with the dimensionality (number of bands). As a result, collecting ground truth for labelled datasets becomes extremely expensive, and therefore, reducing data dimensionality by means of wavelength selection became mandatory.

The proposed thesis presents a two-phase study of hyperspectral image (HSI) classification on the widely-used Salinas and Indian Pines benchmark datasets. In the first phase, the CCARS (Competitive Calibration Adaptive Reweighted Sampling) framework is extended to support non-linear classifiers, specifically Support Vector Machine with Radial Basis Function kernel (SVM-RBF) and Random Forest (RF). A checkerboard block-based spatial splitting strategy is introduced to prevent spatial data leakage, a problem that commonly inflates accuracy in pixel-wise classification literature. An adaptive permutation testing protocol ensures statistical significance of the results. Experiments show that CCARS with SVM-RBF reduces the spectral dimension by up to 75% while retaining over 99% of the full-band classification accuracy on Salinas.

In the second phase, the selected wavelength subsets are used as input to a 3D Convolutional Neural Network (3D-CNN), which extracts joint spatial-spectral features from volumetric data patches. Five wavelength selection methods are compared: four from the Wavelength Selection Tools (WST) library (BOSS, CARS, GA-iPLS, GA-iPLS-BOSS) and Forward Covariate Selection (FCovSel). A Jaccard index analysis reveals that FCovSel selects a completely disjoint set of bands from all WST methods, indicating that they capture complementary spectral information. Despite selecting only 19–20 bands, FCovSel achieves the highest classification accuracy on Indian Pines and competitive results on Salinas, outperforming WST methods that retain up to 128 bands.

The findings demonstrate that aggressive spectral compression is achievable without sacrificing classification performance when spatial context is exploited via deep learning, and that FCovSel provides a statistically efficient and interpretable band-selection strategy for 3D-CNN-based HSI classification.

Contents

Chapter 1

Introduction

1.1 Remote Sensing and Precision Agriculture

Remote sensing refers to the science of acquiring information about an object, area, or phenomenon from a distance, without making direct physical contact. In the context of Earth observation, remote sensing harnesses the interaction between electromagnetic radiation and the surface of the Earth to extract quantitative information about land cover, vegetation, soil composition, and atmospheric conditions from airborne or satellite-mounted sensor platforms. By analysing the spectral, spatial, and temporal characteristics of the reflected or emitted radiation, remote sensing provides a scalable, non-destructive means of monitoring large geographical areas with high temporal frequency.

Agriculture is among the most important application domains of remote sensing technology. Traditional farming practices apply uniform quantities of fertilisers, pesticides, and irrigation water across entire fields, regardless of the spatial variability in soil fertility, crop density, or moisture availability. This one-size-fits-all approach leads to resource waste, environmental pollution, and suboptimal yields. Precision agriculture addresses these inefficiencies by leveraging remote sensing data to characterises intra-field spatial variability at fine spatial resolution, enabling farmers to tailor agronomic interventions to specific sub-field zones and to detect crop stress or disease outbreaks before they become visible to the naked eye.

Remote sensing systems for precision agriculture operate across multiple electromagnetic spectral regions. Passive optical sensors measure sunlight reflected by vegetation, producing spectral images that encode information about chlorophyll concentration, leaf area index (LAI), canopy water content, and physiological stress indicators. In modern satellite remote sensing, several multispectral constellations provide critical monitoring capabilities:

- **Sentinel-2 (ESA):** Equipped with the MultiSpectral Instrument (MSI),

Sentinel-2 captures reflection data across 13 spectral bands. It is widely valued in agriculture for its specific Red-Edge bands (Band 5 at 705 nm, Band 6 at 740 nm, and Band 7 at 783 nm) and spatial resolutions up to 10 meters, which allow for sensitive tracking of chlorophyll variations and vegetation health.

- **Landsat-8/9 (NASA/USGS):** Captures data across 11 bands using the Operational Land Imager (OLI) and the Thermal Infrared Sensor (TIRS), offering long-term baseline data with 30 meter spatial resolution.
- **PRISMA (ASI):** The Pervasive Real-Time Imaging Spectro-Radiometer is a pioneer hyperspectral satellite mission from the Italian Space Agency. It captures 240 spectral bands between 400 nm and 2500 nm with a 30 meter spatial resolution, bridging the gap between multispectral satellite scans and high-resolution airborne hyperspectral surveys.
- **EnMAP (DLR):** The Environmental Mapping and Analysis Program is a German hyperspectral satellite mission that records 242 narrow spectral channels (ranging from 420 nm to 2450 nm) with a 30 meter spatial resolution, optimized for soil and vegetation diagnostics.

Derived multispectral vegetation indices (VIs) are routinely computed from these bands to monitor crop health, estimate biomass, and classify land cover types. These indices exploit the unique reflectance signature of green vegetation, which strongly absorbs red light due to chlorophyll pigments and strongly reflects near-infrared (NIR) light due to the cellular structure of spongy mesophyll tissues. The most common indices are formulated as follows:

1. **Normalized Difference Vegetation Index (NDVI):** The standard index for green biomass estimation:

$$\mathrm{NDVI} = \frac{\rho_{\mathrm{NIR}} - \rho_{\mathrm{Red}}}{\rho_{\mathrm{NIR}} + \rho_{\mathrm{Red}}} \tag{1.1}$$

   where ρ_{NIR} and ρ_{Red} are the reflectance values in the NIR and Red bands, respectively.

2. **Normalized Difference Red Edge (NDRE):** Highly sensitive to chlorophyll content in dense canopies where NDVI saturates:

$$\mathrm{NDRE} = \frac{\rho_{\mathrm{NIR}} - \rho_{\mathrm{RedEdge}}}{\rho_{\mathrm{NIR}} + \rho_{\mathrm{RedEdge}}} \tag{1.2}$$

   where ρ_{RedEdge} represents the reflectance in the transition red-edge region (~ 705 nm).

3. **Soil-Adjusted Vegetation Index (SAVI):** Integrates a soil-brightness correction factor L to minimize soil background influence in early growth stages:

$$\text{SAVI} = \frac{(\rho_{\text{NIR}} - \rho_{\text{Red}})(1 + L)}{\rho_{\text{NIR}} + \rho_{\text{Red}} + L} \tag{1.3}$$

   where L is set to 0.5 for intermediate vegetation density.

4. **Enhanced Vegetation Index (EVI):** Optimizes the vegetation signal by correcting for canopy background noise and atmospheric aerosols:

$$\text{EVI} = G \times \frac{\rho_{\text{NIR}} - \rho_{\text{Red}}}{\rho_{\text{NIR}} + C_1\rho_{\text{Red}} - C_2\rho_{\text{Blue}} + L} \tag{1.4}$$

   where typical scaling coefficients are set to $G = 2.5$, $C_1 = 6$, $C_2 = 7.5$, and $L = 1$, with ρ_{Blue} representing the blue reflectance channel.

5. **Photochemical Reflectance Index (PRI):** Measures changes in carotenoid pigments (xanthophyll cycle) associated with light-use efficiency:

$$\text{PRI} = \frac{\rho_{570} - \rho_{531}}{\rho_{570} + \rho_{531}} \tag{1.5}$$

Despite their widespread adoption, multispectral sensors suffer from fundamental limitations. Their coarse spectral resolution—typically on the order of 50–100 nm per band—prevents them from resolving the narrow absorption features caused by specific biochemical compounds such as chlorophyll-a, chlorophyll-b, anthocyanins, carotenoids, and leaf water. These spectral features, which typically span only 5–20 nm, are essential for discriminating between crop varieties at different growth stages, identifying early-stage diseases, and detecting subtle chemical imbalances that manifest before visible symptoms appear. Furthermore, the limited number of multispectral bands constrains the capacity of machine learning classifiers to build discriminative spectral representations for fine-grained crop recognition tasks.

Hyperspectral imaging sensors overcome these limitations by capturing data in hundreds of contiguous narrow spectral bands, spanning the visible (400–700 nm), near-infrared (700–1300 nm), and shortwave infrared (1300–2500 nm) regions. The resulting high-dimensional spectral measurement provides a unique spectral fingerprint for each ground pixel, enabling precise discrimination between spectrally similar agricultural targets such as different lettuce varieties, growth stages of corn, and soil types. Mounted on unmanned aerial vehicles (UAVs) or light aircraft, hyperspectral sensors are increasingly used for large-scale precision agriculture surveys, providing the spectral detail required for reliable crop classification, weed identification, and nutrient stress detection.

However, the rich spectral information of hyperspectral imagery comes at a significant computational and statistical cost. The high dimensionality of the data introduces the Hughes phenomenon and makes it challenging to design robust, leakage-free evaluation protocols—two core problems that motivate the research presented in this thesis.

1.2 Hyperspectral Imaging and the Curse of Dimensionality

Hyperspectral Imaging (HSI) has proven to be one of the most important technologies in remote sensing and precision agriculture. While conventional red-green-blue imaging acquires three broad spectrum bands, hyperspectral sensors obtain hundreds of contiguous spectral bands ranging from visible through near infrared to shortwave infrared. Due to its high-resolution property, hyperspectral images offer detailed information on various physical and chemical attributes of the target and help create a unique spectral signature for each pixel. In agriculture, HSI is an indispensable tool that helps with crop analysis, disease detection, and leaf classification.

Unfortunately, the high spectral resolution of HSI results in high computational complexity, making hyperspectral image processing difficult. Hyperspectral acquisitions result in high-dimensional data cubes; hence, the problem of the curse of dimensionality, also referred to as the Hughes phenomenon, arises. An increase in the number of spectral bands leads to an exponential growth in the search space; thus, the accuracy of classification decreases dramatically unless the number of training samples used grows exponentially. Ground truthing hyperspectral images requires a huge amount of effort and is an expensive procedure because collecting labeled pixels is extremely laborious and time-consuming. To resolve this issue, we need to use a dimensionality reduction approach, specifically wavelength or band selection. Selecting the optimal set of bands makes it possible to develop compact multispectral sensors and to decrease computational complexity without losing classification accuracy.

1.2.1 Mathematical Formulation of the Hughes Phenomenon

To formally understand the challenge, let $\mathbf{x} \in \mathbb{R}^D$ represent the reflectance vector of a pixel across D contiguous spectral bands. Suppose our task is to train a classifier to predict a class label $y \in \{1, 2, \ldots, C\}$ using a training dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ of size N.

As the dimensionality D increases, the volume of the feature space grows exponentially, which has two major mathematical consequences:

1. **Data Sparsity in High Dimensions:** Consider a unit hypercube in D-dimensional space, representing our normalized spectral space, with volume $V_{\text{cube}} = 1^D = 1$. The volume of a smaller concentric hypercube with edge length $1 - 2\epsilon$ (where $\epsilon > 0$ is a small distance from the boundaries) is:

$$V_{\text{inner}} = (1 - 2\epsilon)^D \tag{1.6}$$

The volume of the boundary shell of thickness ϵ is therefore:

$$V_{\text{shell}} = 1 - (1 - 2\epsilon)^D \tag{1.7}$$

For any fixed boundary thickness $\epsilon > 0$, as the dimensionality $D \to \infty$, the volume of the inner hypercube approaches zero:

$$\lim_{D\to\infty} V_{\text{inner}} = \lim_{D\to\infty} (1 - 2\epsilon)^D = 0 \tag{1.8}$$

Consequently:

$$\lim_{D\to\infty} V_{\text{shell}} = 1 \tag{1.9}$$

This mathematical property means that in high dimensions, almost the entire volume of the hypercube lies concentrated in the thin shell near its outer boundary. Consequently, data points are scattered far apart, and the sample density at the center of the distribution drops to zero. All samples behave as outliers, which makes local density estimation (such as k-nearest neighbors or kernel density estimators) highly unstable and statistically unreliable.

2. **Parameter Estimation Instability:** To train statistical models (such as Linear or Quadratic Discriminant Analysis), we must estimate class-wise mean vectors $\boldsymbol{\mu}_c \in \mathbb{R}^D$ and covariance matrices $\boldsymbol{\Sigma}_c \in \mathbb{R}^{D\times D}$. The covariance matrix $\boldsymbol{\Sigma}_c$ has $D(D+1)/2$ free parameters. The standard empirical covariance estimator:

$$\hat{\boldsymbol{\Sigma}}_c = \frac{1}{N_c - 1} \sum_{i=1}^{N_c} (\mathbf{x}_i - \boldsymbol{\mu}_c)(\mathbf{x}_i - \boldsymbol{\mu}_c)^T \tag{1.10}$$

will have a rank of at most $\min(N_c - 1, D)$. If the number of class samples N_c is less than the number of spectral bands D ($N_c < D + 1$), the estimated covariance matrix $\hat{\boldsymbol{\Sigma}}_c$ becomes singular (not full rank), which makes it impossible to invert. The calculation of the mahalanobis distance or the likelihood requires $\hat{\boldsymbol{\Sigma}}_c^{-1}$, leading to a mathematical breakdown. Even when $N_c > D$, if the ratio N_c/D is small, the covariance estimator has high variance, leading to poor generalization.

Hughes (1968) demonstrated that for a fixed training sample size N, the classification accuracy initially increases as more spectral bands D are added, since

the new bands provide additional discriminative information. However, beyond a certain threshold, the accuracy reaches a maximum and then begins to degrade rapidly because the available training samples are insufficient to accurately estimate the parameters in the higher-dimensional space. This degradation is known as the Hughes phenomenon. To avoid this accuracy collapse, we must apply spectral compression (wavelength selection) to reduce D to a compact set of highly informative bands $K \ll D$ where $N \gg K$, ensuring robust parameter estimation and avoiding the curse of dimensionality.

1.3 Problem Statement

In the literature of HSI leaf classification, wavelength selection is typically evaluated using pixel-wise classification models. However, standard HSI classification pipelines often suffer from two major deficiencies:

1. **Spatial Data Leakage:** Traditional validation strategies randomly partition pixels into training and testing sets. Since hyperspectral images demonstrate very high spatial autocorrelation (that is, adjacent pixels correspond to the same crops/soil blocks, having similar features), the random division enables the test data to neighbor training pixels. As a result, the information gets leaked from the train to the test sample, leading to an artificial increase in test accuracy.

2. **Non-optimal band selection for deep architectures:** Traditional wavelength selection algorithms, such as Competitive Adaptive Reweighted Sampling (CARS), are designed and validated using linear classifiers like Partial Least Squares Discriminant Analysis (PLS-DA). When these selected subsets are transferred to advanced deep learning models like 3D Convolutional Neural Networks (3D-CNNs), which exploit joint spatial-spectral features, the selected bands may not be optimal. There is an obvious gap in the research of comparing classical band selection with covariance-based methods in the context of deep architectures.

1.4 Research Objectives and Methodology

The problem presented here will be systematically investigated with the help of the following two-phase study performed on two widely-used hyperspectral image benchmarks of Salinas and Indian Pines:

- **Phase 1: Traditional Machine Learning & Statistical Validation**
  In this phase, the investigation focuses on the analysis of traditional machine learning classifiers with the help of statistical wavelength selection techniques.

On one hand, the CCARS technique is expanded for non-linear classification algorithms such as support vector machine with radial basis function (SVM-RBF), random forest (RF) algorithm and PLS-DA with different number of PLS components. At the same time, to create a valid comparison benchmark, alternative band selection procedures (five types: BOSS, CARS, GA-iPLS, GA-iPLS-BOSS, and Forward Covariate Selection (FCovSel)) will be analyzed in combination with the optimized SVM-RBF classifier and four preprocessing pipelines (SG-MSC, SG-SVN, SG1-MSC, SG1-SVN). Spatial data leakage will be prevented via an optimized checkerboard-based splitting procedure (8×8 blocks for Indian Pines and 16×16 blocks for Salinas). A calibration-final data split will help us avoid potential data leakage from band selection, and an adaptive permutation test (up to 1000 iterations) will be applied to validate statistical significance.

- **Phase 2: Deep Learning & Transferability Investigation**
  In this phase, we will explore the transferability of selected band sets into deep learning architectures. Band selection procedures will be compared in the context of an adaptive 3D-CNN architecture that changes kernel size depending on the number of input bands and performs spatial-spectral deep learning. Four preprocessing pipelines will be considered in order to assess their influence on results. Finally, Jaccard index analysis will be used for band set comparison while spectral efficiency (accuracy divided by number of selected bands) will define the most compact models.

1.5 Main Contributions

The key contributions of this thesis are as follows:

1. The extension of the CCARS approach [3, 5] to non-linear classifiers (SVM-RBF) results in highly effective spectral compression. For example, on the Salinas dataset, the CCARS-SVM-RBF method decreases the spectral dimensionality by more than 75% (50 bands from 204) while obtaining 92.63% accuracy, which is less than 0.6% loss compared to the full-spectrum classifier under the spatial splitting protocol.

2. The development of the optimized checkerboard split as the new leakage-free benchmark for Salinas and Indian Pines datasets, which indicates that standard random splits result in up to 15% higher accuracy for any hyperspectral classifier.

3. The implementation of the 3D-CNN architecture with an adaptive kernel size of the first layer that grows according to the depth of selected spectral bands to mitigate overfitting for narrow spectral signatures.

4. Our findings indicate that the proposed Forward Covariate Selection (FCovSel) method is the most efficient for selecting bands for deep learning models. For instance, on the challenging Indian Pines dataset, FCovSel reaches the best classification accuracy (55.12%) with only 19 bands, while the other methods including the full spectrum and WST reach no more than 19% accuracy with up to 128 bands.

5. The Jaccard index analysis demonstrates that FCovSel selects disjoint sets of bands compared to WST approaches, meaning that covariance-based and regression-coefficient-based methods capture unique spectral features.

1.6 Thesis Outline

The remainder of this thesis is structured as follows:

- **Chapter 2: Background and Related Work** surveys the existing literature on wavelength selection methods, deep learning architectures for HSI classification, and spatial data leakage in validation protocols.
- **Chapter 3: Hyperspectral Imaging System** provides the physical and mathematical framework for hyperspectral imaging and discusses the curse of dimensionality.
- **Chapter 4: Benchmark Datasets and Preprocessing** describes the Salinas and Indian Pines datasets and details the preprocessing techniques (SG, MSC, SNV) and PCA-based outlier detection pipeline.
- **Chapter 5: Methodology and Contribution** describes the original contributions of this thesis: the leakage-free checkerboard spatial splitting strategy, the CCARS$_K$ framework, the adaptive 3D-CNN architecture, and the permutation-based statistical validation protocol.
- **Chapter 6: Experimental Results and Discussion** presents detailed numerical results, performance tables, and accuracy comparisons for Phase 1 and Phase 2.
- **Chapter 7: Conclusions and Future Work** summarizes the key findings of the thesis and outlines potential directions for future research.

Chapter 2

Background and Related Work

This chapter surveys the existing literature relevant to the contributions of this thesis. We first introduce a taxonomy of wavelength selection methods, providing a structured comparison of their key properties. We then present detailed mathematical descriptions of the five algorithms evaluated in this thesis. Finally, we review the literature on deep learning architectures for HSI classification and on the problem of spatial data leakage in validation protocols, highlighting the open problems that motivate our contributions.

2.1 Taxonomy of Wavelength Selection Methods

Wavelength (band) selection methods for hyperspectral data can be broadly organized into three categories: *filter* methods, *wrapper* methods, and *hybrid* methods.

Filter methods evaluate the relevance of each band independently of any classifier, using statistical criteria such as mutual information, variance, or covariance with the target variable. Because they decouple band scoring from classifier training, filter methods are computationally efficient and produce stable selections that generalize well across different classifiers.

Wrapper methods integrate a classifier into the selection loop: a candidate subset is evaluated by training and cross-validating the classifier on that subset, and the subset with the best classifier score is retained. Wrapper methods are more computationally expensive but can find subsets better tuned to the specific classifier used during evaluation.

Hybrid methods combine a coarse-grained wrapper step—which discards large uninformative spectral regions—with a fine-grained filter or wrapper step applied to the reduced search space, balancing computational cost and selection quality.

Table 2.1 summarizes the key properties of the five methods evaluated in this thesis.

A key distinction captured in the table is that CARS, BOSS, GA-iPLS, and

Table 2.1: Taxonomy of the five wavelength selection methods evaluated in this thesis.

Method	Category	Selection Criterion	Granularity	CV?	Classifier-Free?
CARS	Wrapper	PLS regression coefficients	Individual bands	Yes	No
BOSS	Wrapper	Bootstrap PLS coefficients	Individual bands	No	No
GA-iPLS	Wrapper	PLS cross-validation error	Spectral intervals	Yes	No
GA-iPLS-BOSS	Hybrid	Two-stage: GA + BOSS	Intervals then bands	Yes	No
FCovSel	Filter	Covariance with target	Individual bands	No	Yes

GA-iPLS-BOSS are all based on Partial Least Squares (PLS) regression coefficients and thus belong to the class of regression-coefficient-based methods. In contrast, FCovSel is entirely classifier-free: it relies only on the covariance structure between spectral bands and the target label matrix. This fundamental algorithmic difference has important implications for the spectral subsets that each method selects, as investigated through the Jaccard similarity analysis in Chapter 6.

2.2 Wavelength Selection Algorithms

In this section, we present the detailed mathematical formulations of the five wavelength selection algorithms evaluated in this thesis.

2.2.1 Competitive Adaptive Reweighted Sampling (CARS)

Competitive Adaptive Reweighted Sampling (CARS) [8] is an evolutionary variable selection method. It iteratively selects a subset of bands by mimicking the biological concept of "survival of the fittest." The algorithm operates through a series of R Monte Carlo sampling runs, executing the following mathematical steps:

1. **Monte Carlo Model Calibration:** In each sampling run $r \in \{1, 2, \ldots, R\}$, a subset of 80% of the training samples is randomly selected with replacement. A Partial Least Squares (PLS) regression model is calibrated on this subset. For a target variable $\mathbf{y}$ and spectral matrix $\mathbf{X}_r$, the PLS regression coefficients $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_B]^T \in \mathbb{R}^B$ are computed.

2. **Wavelength Weight Computation:** In a multi-class classification setting with C classes, PLS-DA produces a coefficient matrix $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{B \times C}$. The overall importance weight w_i for each spectral band i is calculated as the root sum of squares of its regression coefficients across all classes:

$$w_i = \sqrt{\sum_{c=1}^{C} \beta_{i,c}^2}, \quad i = 1, 2, \ldots, B \tag{2.1}$$

The normalized weight u_i, representing the relative importance of band i, is then computed as:

$$u_i = \frac{w_i}{\sum_{j=1}^{B} w_j} \tag{2.2}$$

3. **Exponential Decreasing Function (EDF):** To enforce spectral compression, the ratio of wavelengths to be retained at iteration k is controlled by an exponential decreasing curve:

$$r_k = ae^{-bk} \tag{2.3}$$

where the coefficients a and b are derived from the initial and boundary conditions. At the starting iteration ($k = 0$), all variables are active ($r_0 = 1$), which gives $a = 1$. At the final iteration ($k = N$, where N is the total number of elimination iterations), the ratio must equal the desired minimum fraction of bands:

$$r_N = \frac{K_{\text{target}}}{B} \tag{2.4}$$

Taking the natural logarithm of both sides yields the parameter b:

$$b = -\frac{1}{N} \ln\left(\frac{K_{\text{target}}}{B}\right) \tag{2.5}$$

At each step k, the number of bands to be selected is $M_k = \text{round}(r_k \times B)$.

4. **Adaptive Roulette Wheel Selection:** A roulette wheel is constructed where the selection probability p_i of each band i is proportional to its normalized importance weight u_i:

$$p_i = \frac{u_i}{\sum_{j=1}^{B} u_j} \tag{2.6}$$

A set of M_k bands is selected using this probability distribution. This step represents the competitive aspect: bands with larger coefficients are selected repeatedly, while bands with negligible coefficients are discarded.

5. **Optimal Subset Selection via RMSECV:** Over the N iterations, a series of candidate band subsets $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_N$ is generated. A PLS model is trained on each subset, and its performance is evaluated using the Root Mean Squared Error of Cross-Validation (RMSECV):

$$\text{RMSECV}(\mathcal{S}_k) = \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(y_j - \hat{y}_j^{(-j)}\right)^2} \tag{2.7}$$

where $\hat{y}_j^{(-j)}$ is the cross-validated prediction of sample j when it is omitted from the model calibration. The optimal iteration k^* is selected as:

$$k^* = \arg\max_k \left\{ \frac{1}{\text{RMSECV}(\mathcal{S}_k)} \right\} \tag{2.8}$$

The wavelength subset $\mathcal{S}_{k^*}$ is output as the final selected bands.

2.2.2 Bootstrap Soft Shrinkage (BOSS)

Bootstrap Soft Shrinkage (BOSS) [2] addresses the instability and high variance of CARS. Instead of applying hard elimination (where variables are permanently discarded in a single step), BOSS applies a soft-shrinkage rule that dynamically shrinks the contribution of uninformative variables while allowing them to remain in the search space.

The BOSS algorithm is structured as follows:

1. **Bootstrap Sub-Model Generation:** Let M represent the number of bootstrap models (typically $M = 1000$). For each model $m \in \{1, 2, \ldots, M\}$, a subset of training samples is selected with replacement. A PLS model is calibrated on this subset, yielding a regression vector $\boldsymbol{\beta}^{(m)} \in \mathbb{R}^B$.

2. **Variables Weight Assignment:** Unlike CARS, which uses raw coefficients, BOSS evaluates variable importance based on the stability of the regression coefficients across the bootstrap models. The weight w_i for band i is defined as the absolute mean of its coefficients divided by their standard deviation, which acts as a spectral Signal-to-Noise Ratio (SNR):

$$w_i = \frac{\left|\bar{\beta}_i\right|}{s_i} \tag{2.9}$$

where:

$$\bar{\beta}_i = \frac{1}{M} \sum_{m=1}^{M} \beta_i^{(m)} \tag{2.10}$$

$$s_i = \sqrt{\frac{1}{M-1} \sum_{m=1}^{M} \left(\beta_i^{(m)} - \bar{\beta}_i\right)^2} \tag{2.11}$$

3. **Soft-Shrinkage Operator:** The normalized weight u_i is computed:

$$u_i = \frac{w_i}{\sum_{j=1}^{B} w_j} \tag{2.12}$$

Instead of hard discarding, the number of bootstrap trials N_i allocated to variable i in the next iteration is proportional to its weight:

$$N_i = \text{round}\left(u_i \times M\right) \tag{2.13}$$

If $N_i = 0$, the variable is temporarily set aside but is not permanently deleted. In subsequent iterations, as the active variables set is reduced, the bootstrap sub-models are calibrated on the remaining active variables, and their weights are recalculated. The process converges when the number of active variables drops below a predefined threshold. The optimal subset is chosen as the one that minimizes the cross-validation error.

2.2.3 Genetic Algorithm Interval Partial Least Squares (GA-iPLS)

GA-iPLS [15] is an interval-based variable selection method. It is designed to preserve the local spectral correlation of contiguous bands by selecting entire spectral windows rather than isolated wavelengths.

The algorithm executes the following mathematical steps:

1. **Spectral Partitioning:** The full spectral range containing B bands is divided into P non-overlapping, contiguous intervals of width $W = \lfloor B/P \rfloor$. Let the j-th interval be denoted by $\mathcal{I}_j = \{b_{(j-1)W+1}, \ldots, b_{jW}\}$.

2. **Binary Chromosome Representation:** A candidate spectral subset is represented as a binary chromosome $\mathbf{g} = [g_1, g_2, \ldots, g_P]^T \in \{0,1\}^P$, where:

$$g_j = \begin{cases} 1 & \text{if interval } \mathcal{I}_j \text{ is selected} \\ 0 & \text{otherwise} \end{cases} \tag{2.14}$$

3. **Fitness Evaluation:** A population of N_{pop} chromosomes is initialized. The fitness of chromosome $\mathbf{g}$ is defined as the reciprocal of the cross-validation error of a PLS-DA model trained on the spectral bands corresponding to the active genes:

$$\text{Fitness}(\mathbf{g}) = \frac{1}{\text{RMSECV}(\mathbf{g})} \tag{2.15}$$

4. **Genetic Operators:**

   - **Selection:** Chromosomes are selected for reproduction using tournament selection, where the probability of selection is proportional to their fitness.
   - **Crossover:** Uniform crossover is applied to pairs of parent chromosomes with a probability $p_c = 0.6$ to exchange spectral intervals.
   - **Mutation:** Mutation is applied by flipping genes with a probability $p_m = 0.1$ to introduce new spectral intervals.

   The genetic optimization is executed for G generations (typically $G = 100$), and the intervals corresponding to the fittest chromosome in the final generation are compiled.

2.2.4 Hybrid GA-iPLS-BOSS

The GA-iPLS-BOSS method is a two-stage hybrid variable selection algorithm designed to leverage the advantages of both interval-based and pixel-based selection:

1. **Stage 1 (Global Search):** GA-iPLS is applied to the full spectrum to identify the most informative spectral intervals, discarding large uninformative blocks (such as flat noise regions or baseline shift bands). This reduces the search space from B bands to a smaller subset of candidate bands $B' \ll B$.

2. **Stage 2 (Local Refinement):** The individual wavelengths within the selected intervals are treated as independent variables, and the BOSS algorithm is applied to perform pixel-level selection. This fine-grained step eliminates redundant bands within the selected intervals, producing a highly compact final wavelength subset.

2.2.5 Forward Covariate Selection (FCovSel)

Forward Covariate Selection (FCovSel) [11,17] is a greedy forward selection method based on covariance. Unlike the WST family, FCovSel is entirely classifier-free and does not utilize a cross-validation loop during selection, which makes it highly computationally efficient.

Let $\mathbf{X} \in \mathbb{R}^{n \times B}$ represent the mean-centered spectral predictor matrix and $\mathbf{Y} \in \mathbb{R}^{n \times C}$ represent the mean-centered binary target class matrix. The algorithm operates iteratively. At step $k \in \{1, 2, \ldots, K_{\text{target}}\}$:

1. **Covariance Analysis:** For each remaining predictor column $\mathbf{x}_j \in \mathbb{R}^n$ (where $j \in \mathcal{U}_k$, the set of unselected variables), the squared covariance with the target matrix $\mathbf{Y}$ is computed:

$$C_j = \sum_{c=1}^{C} \operatorname{cov}(\mathbf{x}_j, \mathbf{y}_c)^2 = \sum_{c=1}^{C} \left(\frac{1}{n-1} \mathbf{x}_j^T \mathbf{y}_c \right)^2 \tag{2.16}$$

2. **Variable Selection:** The variable index j^* that maximizes the total squared covariance is selected:

$$j^* = \arg\max_{j \in \mathcal{U}_k} C_j \tag{2.17}$$

   This variable is added to the selected set: $\mathcal{S}_k = \mathcal{S}_{k-1} \cup \{j^*\}$, and removed from the unselected set: $\mathcal{U}_{k+1} = \mathcal{U}_k \setminus \{j^*\}$.

3. **Orthogonal Projection (Redundancy Removal):** To prevent the selection of variables that carry redundant information, the remaining predictor columns in $\mathbf{X}$ and the target columns in $\mathbf{Y}$ are projected orthogonally to the selected column $\mathbf{x}_{j^*}$.

We define the normalized projection vector:

$$\mathbf{p}_k = \frac{\mathbf{x}_{j^*}}{\|\mathbf{x}_{j^*}\|} \tag{2.18}$$

The orthogonal projection operator $\mathbf{P}_k \in \mathbb{R}^{n \times n}$ is formulated as:

$$\mathbf{P}_k = \mathbf{I} - \mathbf{p}_k \mathbf{p}_k^T \tag{2.19}$$

Both matrices are updated by applying the projection operator:

$$\mathbf{X} \leftarrow \mathbf{P}_k \mathbf{X} \tag{2.20}$$

$$\mathbf{Y} \leftarrow \mathbf{P}_k \mathbf{Y} \tag{2.21}$$

This orthogonal projection removes the linear contribution of the selected variable from all other variables, ensuring that subsequent steps select bands that carry independent information. The loop continues until the target number of bands K_{target} is reached.

2.3 Deep Learning for HSI Classification

The transition from traditional machine learning to deep learning has significantly improved hyperspectral image classification. Deep networks learn hierarchical representations directly from the raw data, allowing them to model complex non-linear interactions.

2.3.1 One-Dimensional Convolutional Networks

One-dimensional convolutional neural networks (1D-CNNs) operate directly on the spectral axis of individual pixels [7]. Let the spectral signature of a pixel be represented as a vector $\mathbf{s} \in \mathbb{R}^B$. The 1D convolution operation at layer l is formulated as:

$$z_i^{(l)} = f\left(\sum_{j=-m}^{m} w_j^{(l)} z_{i+j}^{(l-1)} + b^{(l)}\right) \tag{2.22}$$

where $w_j^{(l)}$ represents the weights of a 1D filter of size $2m+1$, $b^{(l)}$ is the bias, and $f(\cdot)$ is the activation function. While 1D-CNNs can extract local spectral features (such as absorption slopes and peaks), they process each pixel independently. They are fundamentally limited because they ignore the spatial context of the scene.

2.3.2 Two-Dimensional Convolutional Networks

Two-dimensional CNNs (2D-CNNs) incorporate spatial context by applying convolutional filters over local spatial patches [10]. Let a spatial patch centered around a pixel be denoted as $\mathbf{P} \in \mathbb{R}^{S\times S\times B}$. Since standard 2D-CNNs are computationally expensive when the number of channels B is large, they are typically paired with a prior dimensionality reduction step, such as Principal Component Analysis (PCA). The input is compressed to a small number of principal components $C \ll B$ (typically $C = 3$–10), yielding a tensor $\mathbf{P}_{\text{PCA}} \in \mathbb{R}^{S\times S\times C}$. The 2D convolution is formulated as:

$$z_{x,y}^{(l)} = f\left(\sum_{c=1}^{C}\sum_{u=-m}^{m}\sum_{v=-m}^{m} w_{u,v,c}^{(l)} z_{x+u,y+v,c}^{(l-1)} + b^{(l)}\right) \tag{2.23}$$

This approach allows the model to exploit local texture and spatial patterns. However, the PCA step discards fine-grained spectral details, which can lead to confusion between classes that have similar spatial textures but distinct spectral absorption features.

2.3.3 Three-Dimensional Convolutional Networks

To address the limitations of 1D and 2D architectures, Chen et al. [1] proposed three-dimensional convolutional neural networks (3D-CNNs) for joint spatial-spectral feature extraction. A 3D-CNN utilizes volumetric kernels that slide across both the spatial dimensions (x, y) and the spectral dimension (z) simultaneously.

Let the input volumetric patch be $\mathbf{P} \in \mathbb{R}^{S\times S\times D\times 1}$, where D is the number of spectral bands. The 3D convolution at spatial position (x, y) and spectral depth z is formulated as:

$$z_{x,y,z}^{(l)} = f\left(\sum_{h=-a}^{a}\sum_{w=-b}^{b}\sum_{d=-c}^{c} w_{h,w,d}^{(l)} z_{x+h,y+w,z+d}^{(l-1)} + b^{(l)}\right) \tag{2.24}$$

where the kernel has dimensions $(2a + 1) \times (2b + 1) \times (2c + 1)$. This formulation enables the network to extract joint spatial-spectral features without any prior loss of spectral information. However, 3D convolutions introduce a very large number of parameters, making them highly susceptible to overfitting unless the spectral dimension D is compressed using variable selection.

2.3.4 Hybrid Spatial-Spectral Networks (HybridSN)

To balance the high parameter cost of 3D convolutions with the spatial modeling power of 2D convolutions, Roy et al. proposed HybridSN [19]. HybridSN combines 3D and 2D convolutions in a sequential architecture:

1. **3D Convolutional Layers:** Apply several 3D filters to the raw spatial-spectral patch to learn joint spatial-spectral representations.

2. **Dimension Flattening:** Reshape the spectral and filter dimensions of the 3D output into a single dimension, converting the volumetric tensor into a 3D spatial-channel tensor.

3. **2D Convolutional Layers:** Apply standard 2D filters to the reshaped tensor to extract high-level spatial texture features.

This hybrid approach reduces the network's parameter count while preserving the joint spatial-spectral representation learned in the initial layers, providing a highly effective baseline for HSI classification.

2.3.5 Attention Mechanisms and Transformers

More recent SOTA architectures integrate attention mechanisms to dynamically recalibrate features:

- **Channel Attention:** Mechanisms like Squeeze-and-Excitation (SE) blocks compute global spatial statistics to assign importance weights to individual spectral bands, allowing the network to focus on diagnostic wavelengths.

- **Spectral-Spatial Transformers:** Vision Transformers (ViT) have been adapted for HSI by treating spectral bands and spatial patches as sequences of tokens. Self-attention mechanisms model long-range spectral dependencies, which is particularly useful for identifying complex biochemical mixtures. However, Transformer architectures require very large training datasets and are prone to overfitting on standard HSI benchmarks due to the limited number of labeled samples.

Chapter 3

Hyperspectral Imaging System

This chapter provides the theoretical framework and the required background information to fully comprehend the proposed methodology and its contributions. It discusses the physical concept of hyperspectral imaging, the benchmark datasets used in the experiments, and the mathematical model of the preprocessing and outlier detection frameworks.

3.1 Hyperspectral Imaging Principles

Hyperspectral Imaging (HSI) is a non-destructive imaging method that merges regular digital imaging with spectroscopy. Unlike RGB imaging devices that use three broad wavelength ranges for the three primary colors of red, green, and blue, hyperspectral imaging captures a multitude of narrow electromagnetic reflection curves within hundreds of bands, starting from the visible band (400 nm) up to the shortwave infrared (2500 nm).

The main output of an HSI device is the data cube $\mathbf{X} \in \mathbb{R}^{H \times W \times B}$, where H and W are dimensions of the spatial plane and B denotes the number of spectral channels. At each pixel position (x, y) on the plane, there is one curve (spectral signature) $\mathbf{s} \in \mathbb{R}^{B}$. This curve shows the ratio between the emitted and reflected light intensity at each wavelength. Since materials have distinctive absorption and reflection characteristics due to their chemical composition (chlorophyll, carotenoid, water) and cellular structures, this technology can precisely differentiate agricultural materials.

3.1.1 The AVIRIS Sensor Hardware and Physics

The datasets evaluated in this thesis were collected using the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor, designed and operated by the NASA Jet Propulsion Laboratory (JPL). The AVIRIS system is a premier optical sensor in

remote sensing, pioneering high-fidelity hyperspectral imaging from airborne platforms (typically flown on the NASA ER-2 aircraft at an altitude of 20 km or on Twin Otter utility aircraft at lower altitudes).

The hardware configuration and scanning physics of the AVIRIS instrument are characterized by the following design elements:

- **Whiskbroom Scanning Mechanism:** AVIRIS is a whiskbroom (cross-track) scanner. It utilizes a rotating scan mirror to sweep the field of view transversely across the flight path. The mirror focuses the incoming reflected solar radiation onto a set of optical fibers. This optomechanical scanning mechanism provides a spatial field of view of 30°, capturing a swath on the ground. The spatial resolution varies from 20 meters (when flown on the ER-2 at 20 km) to 2–4 meters (when flown on low-altitude aircraft).

- **Spectrometer Configuration:** To cover the broad spectral range of 400 nm to 2500 nm, the incoming light is split and routed to four separate spectrometers (A, B, C, and D), each optimized for a specific spectral sub-region:

  1. **Spectrometer A (VIS):** Contains 32 channels covering the visible spectrum (400–700 nm). It primarily captures light interactions with plant photosynthetic pigments (chlorophylls, carotenoids, and anthocyanins).
  2. **Spectrometer B (NIR):** Contains 64 channels covering the near-infrared spectrum (700–1250 nm). It is highly sensitive to leaf internal structure (spongy mesophyll air-cell interfaces) and biomass density. **Spectrometer C (SWIR-1):** Contains 64 channels covering the first shortwave infrared region (1250–1850 nm). It captures the first major water absorption features and organic compounds (lignin, cellulose).
  3. **Spectrometer D (SWIR-2):** Contains 64 channels covering the second shortwave infrared region (1850–2500 nm). It captures deep water absorption bands and mineral absorption features.

- **Spectral and Radiometric Calibration:** The raw digital numbers (DN) captured by the silicon and indium-gallium-arsenide (InGaAs) detectors must undergo a calibration pipeline. Radiometric calibration translates DN values into physical spectral radiance ($\mu W/(cm^2 \cdot sr \cdot nm)$). Spectral calibration defines the exact center wavelength and Full Width at Half Maximum (FWHM) of each channel, typically on the order of 10 nm. To obtain surface reflectance, an atmospheric correction (such as ATREM or MODTRAN) is applied to remove the effects of solar irradiance, Rayleigh scattering, and atmospheric gas absorption.

- **Water Vapor Absorption Band Removal:** Two spectral regions are heavily contaminated by atmospheric moisture: the 1.4 μm band (1350–1430 nm)

and the 1.9 μm band (1810–1940 nm). In these regions, atmospheric water vapor absorbs nearly 100% of the incoming and reflected solar radiation, reducing the signal-to-noise ratio (SNR) to zero. For all benchmark datasets, the channels falling in these water absorption bands are discarded (20 bands in Salinas and 20 bands in Indian Pines), leaving 204 and 200 clean bands, respectively.

3.2 Spatial Data Leakage and Validation Strategies

Hyperspectral images, having been acquired from contiguous ground areas, exhibit strong spatial autocorrelation. Understanding this property is critical because it directly motivates both the leakage-free evaluation protocol and the block-based splitting strategy introduced in Chapter 5.

3.2.1 Spatial Autocorrelation in Hyperspectral Imagery

Hyperspectral images exhibit strong spatial correlation: pixels that are close to each other tend to have very similar spectral signatures. This spatial correlation can be quantified using Moran's I statistic.

Let x_i represent a spectral attribute (e.g., the reflectance value at a specific wavelength) of a pixel at spatial location i, and let $\bar{x}$ be the mean value of this attribute across the entire image. Moran's I is defined as:

$$I = \frac{N}{S_0} \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} w_{i,j} \left(x_i - \bar{x}\right) \left(x_j - \bar{x}\right)}{\sum_{i=1}^{N} \left(x_i - \bar{x}\right)^2} \tag{3.1}$$

where N is the total number of pixels, $w_{i,j}$ is a spatial weight matrix that defines the proximity between location i and location j ($w_{i,j} = 1$ if pixels are adjacent, and 0 otherwise), and S_0 is the sum of all spatial weights:

$$S_0 = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{i,j} \tag{3.2}$$

A value of Moran's I close to +1.0 indicates strong spatial autocorrelation, where similar spectral values are clustered together. In agricultural datasets like Salinas and Indian Pines, Moran's I typically exceeds 0.85 across most wavelengths, reflecting the contiguous nature of crop fields.

3.2.2 The Spatial Data Leakage Problem

In statistical machine learning, a fundamental assumption is that the training set $\mathcal{D}_{\text{train}}$ and the testing set $\mathcal{D}_{\text{test}}$ consist of independent and identically distributed

(i.i.d.) samples drawn from a joint probability distribution $P(\mathbf{X}, y)$:

$$\mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}} \overset{\text{i.i.d.}}{\sim} P(\mathbf{X}, y) \tag{3.3}$$

When evaluating HSI models, the standard approach in many publications is to randomly partition the labeled pixels of a single image cube into training and test sets. However, because of the high spatial autocorrelation (high Moran's I), neighboring pixels are nearly identical:

$$\mathbf{x}_i \approx \mathbf{x}_j \quad \text{for } d(i,j) < \epsilon \tag{3.4}$$

where $d(\cdot, \cdot)$ represents the Euclidean distance between pixel coordinates on the image plane. If we apply a random pixel-wise split, a test pixel $\mathbf{x}_{\text{test}}$ will almost certainly fall within the immediate spatial neighbourhood of a training pixel $\mathbf{x}_{\text{train}}$, violating the i.i.d. assumption:

$$P\left(\mathbf{x}_{\text{test}} \mid \mathbf{x}_{\text{train}}\right) \neq P\left(\mathbf{x}_{\text{test}}\right) \tag{3.5}$$

This overestimation of model performance is called *spatial data leakage* [14]. It masks the model's inability to generalise to unseen geographic regions or crop fields, leading to a significant bias in the literature.

3.2.3 Spatially Independent Validation Strategies

To address this bias, validation protocols must enforce spatial separation between training and test samples:

- **Geographic Partitioning:** Divides the image into large, contiguous sub-images (e.g., left half for training, right half for testing). While this guarantees spatial independence, it can result in a severe class imbalance if certain crops are only located on one side of the image.
- **Block-based Partitioning:** Partitions the image into a grid of non-overlapping spatial blocks of size $B_s \times B_s$. Entire blocks are then assigned to either the training or test set, preventing neighbouring pixels from sharing information across the split boundary.
- **Checkerboard Split:** An advanced block-based strategy where blocks are assigned to training and testing in an alternating checkerboard pattern. This ensures that training and testing blocks only touch at diagonal corners, eliminating any shared horizontal or vertical boundaries and minimising spatial data leakage. This strategy is adopted as the evaluation protocol in this thesis, and its implementation is detailed in Chapter 5.

Chapter 4

Benchmark Datasets and Preprocessing

This chapter introduces the specific hyperspectral datasets used to evaluate our proposed methodologies and details the complete spectral preprocessing pipeline applied before classification.

4.1 Dataset Descriptions

Several publicly available hyperspectral benchmark datasets are widely used in the classification literature, each with distinct characteristics that make them suitable for different research objectives. Among the most prominent are the Kennedy Space Center (KSC) scene, which captures complex vegetation-water boundaries in Florida; the University of Pavia dataset, an urban scene collected over a European city center; the Houston 2013 dataset, a heterogeneous urban-suburban scene used in the 2013 IEEE GRSS Data Fusion Contest; the Indian Pines scene, an agricultural area in the Midwest United States; and the Salinas Valley scene, a large agricultural area in California. All of these datasets were collected by AVIRIS or similar VNIR-SWIR sensors and contain between 16 and 330 labeled ground-truth classes.

For this thesis, the **Indian Pines** and **Salinas** datasets were selected based on three criteria. First, both are AVIRIS acquisitions, ensuring that the sensor characteristics, spectral resolution, and atmospheric water absorption bands to be removed are consistent across experiments, making direct cross-dataset comparisons methodologically valid. Second, both datasets contain exactly 16 labeled classes with a meaningful degree of class imbalance, which stresses the evaluation metrics and the leakage-free splitting strategy. Third, and most importantly, both datasets have been widely adopted as standard benchmarks in the wavelength selection and HSI classification literature, including the original CCARS works by Dilillo et

al. [3, 5] that this thesis extends. Using these established benchmarks allows for direct, fair comparison with previously published results.

To verify the wavelength selection methods and classification algorithms, the two AVIRIS datasets are described in detail below:

1. **Salinas Dataset:** This dataset is collected by the AVIRIS sensor from the agricultural region of Salinas Valley, California. Its size consists of 512×217 pixels having 3.7 meters per pixel resolution. At first, the sensor recorded 224 spectral bands. But, because of absorption by atmospheric water vapor and low SNR in some atmospheric bands, 20 bands are dropped (namely, bands 108–112, 154–167, and 224) which means only 204 bands are available for classification purposes. There exist 16 classes of agricultural targets which include different growing stages of lettuces, broccolis, vineyards, corns, and bare soils. The spatial regions are quite large and continuous, thus making the Salinas dataset a very good test for spatial autocorrelation and block-wise data splitting.

2. **Indian Pines Dataset:** This dataset is acquired by the AVIRIS sensor in the agricultural test site of Northwest Indiana. The total number of pixels in the spatial map is 145×145 with a pixel resolution of 20 meters. As in Salinas, there were 220 bands initially recorded by the AVIRIS sensor; but then 20 water-vapor absorption bands (bands 104–108, 150–163, and 220) have been stripped away leaving only 200 bands to consider. There are also 16 target classes which mostly represent agricultural areas such as corn, soybeans, and wheat as well as forestry. This dataset poses a very challenging test case since there is significant class imbalance (fewer than 30 pixels per class for some classes) and a high degree of crop-soil mixture.

Figure 4.1 visualises the mean raw reflectance spectra for the 16 classes in both datasets. As shown in the overlay, the two datasets occupy significantly different dynamic ranges and exhibiting different baseline shifts, primarily due to variations in atmospheric conditions, illumination geometry, and sensor calibration at the time of acquisition.

4.1.1 Class Distribution and Dataset Imbalance

A critical characteristic of both benchmark datasets is significant class imbalance, which directly motivates the leakage-free spatial splitting strategy and the choice of Macro F1-score as the primary evaluation metric. Figure 4.2 visualises the number of labelled pixels per class in each dataset.

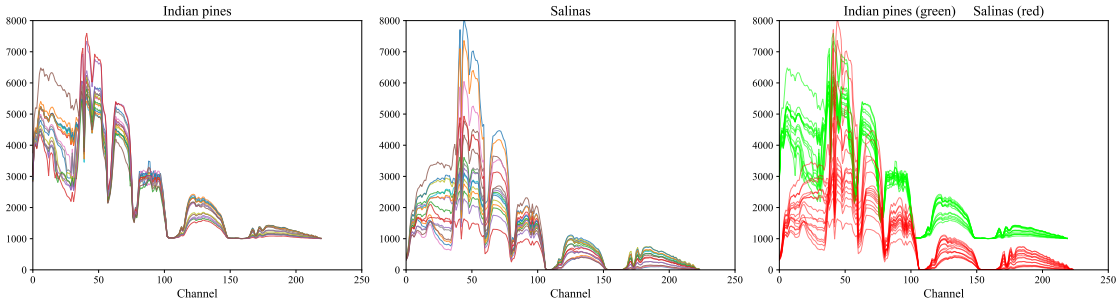

Figure 4.1: Mean raw reflectance spectra for the Indian Pines (left) and Salinas (middle) datasets. The right panel overlays the mean spectra of Indian Pines (green) and Salinas (red). The removed water absorption bands are evident as the gaps where the spectral curve falls to zero. The significant difference in magnitude and baseline between the two datasets highlights the necessity of robust spectral preprocessing (such as MSC or SNV) before classification.

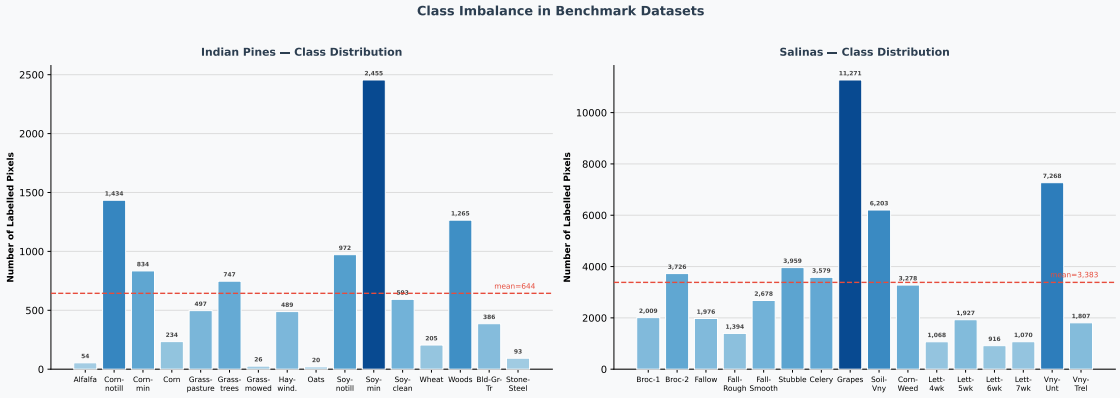

Figure 4.2: Class distribution of labelled pixels in the Indian Pines (left) and Salinas (right) datasets. The dashed red line indicates the mean number of pixels per class. Indian Pines is strongly imbalanced: the smallest class (Oats, 20 pixels) has roughly 123× fewer samples than the largest (Soybean-min, 2,455 pixels). Salinas is comparatively more balanced, though the Grapes class (11,271 pixels) is approximately 12× larger than the smallest (Lettuce-6wk, 916 pixels).

4.1.2 Agronomic and Spectral Characteristics of the Classes

A key challenge in hyperspectral classification is that agricultural targets exhibit subtle spectral differences. In this section, we describe the agronomic characteristics and spectral features of the 16 classes in each dataset.

Salinas Class Descriptions

The Salinas dataset represents a commercial agricultural valley with large fields, dominated by vegetables and vineyards. The 16 target classes are defined as:

1. **Brocoli_green_weeds_1:** Young broccoli crops with early-stage weed growth. The spectral signature shows a mixture of soil and green plant material, with a moderate chlorophyll absorption pit at 670 nm.

2. **Brocoli_green_weeds_2:** More mature broccoli fields. The leaf area index (LAI) is higher, leading to a stronger red-edge transition (700–740 nm) and higher NIR reflectance ($\sim$ 800 nm).

3. **Fallow:** Uncultivated agricultural fields. The signature is characterized by dry soil reflection, lacking any red-edge transition or chlorophyll absorption features.

4. **Fallow_rough_plow:** Ploughed fields left fallow. The plowing process creates large soil aggregates, resulting in micro-shadows that reduce the overall reflectance intensity across the entire spectrum.

5. **Fallow_smooth:** Ploughed and smoothed fields. The flat surface increases specular reflection, leading to higher overall reflectance compared to rough-ploughed fields.

6. **Stubble:** Fields containing dry crop residues (stalks, straw). The spectra show strong absorption features at 1730 nm and 2100 nm corresponding to cellulose and lignin, but no active chlorophyll absorption.

7. **Celery:** Highly dense, lush celery canopies with complete ground cover. Celery leaf structure is highly reflective in the NIR (750–1300 nm) and shows deep water absorption bands at 970 nm and 1200 nm.

8. **Grapes_untrained:** Mature vineyards where the vines are not trellised or pruned. The canopy is irregular, and the rows are filled with wild weeds, producing a complex spatial-spectral mixture.

9. **Soil_vinyard_develop:** Young vineyard fields where vines are small and widely spaced. The spectral response is dominated by the ploughed clay-loam soil, with only a minor vegetation signature.

10. **Corn_senesced_green_weeds:** Corn crops at the end of their lifecycle. The corn leaves have turned brown (losing chlorophyll), but the ground is covered by green weeds, creating a dual-peak spectral response.

11. **Lettuce_romaine_4wk:** Romaine lettuce 4 weeks after planting. The plants are small, and the pixel response contains substantial soil background, making it difficult to distinguish from bare fallow.

12. **Lettuce_romaine_5wk:** Romaine lettuce 5 weeks after planting. The plants have grown, increasing the vegetation index and the slope of the red-edge.

13. **Lettuce_romaine_6wk:** Romaine lettuce 6 weeks after planting. High canopy density, soil background contribution is minimized.

14. **Lettuce_romaine_7wk:** Mature romaine lettuce ready for harvest. Near-complete ground cover, showing maximum green reflection at 550 nm and maximum NIR reflection.

15. **Vinyard_untrained:** Mature untrained vineyards. Similar to grapes untrained, but located in a different field with different soil moisture and weed composition.

16. **Vinyard_vertical_trellis:** Mature vineyards trained on vertical trellises. The vertical structure creates distinct shadows in the inter-row areas, which reduces the spatial average reflectance.

Indian Pines Class Descriptions

The Indian Pines dataset represents a mixed agricultural-forestry landscape in Indiana. The 16 classes represent smaller, irregular fields:

1. **Alfalfa:** A dense, nitrogen-rich forage crop with a very strong photosynthetic signature. **Corn-notill:** Corn planted in fields where the soil was ploughed in the previous season. The ground is covered by dry corn residues, which mix spectrally with the young corn shoots.

2. **Corn-mintill:** Corn planted in fields with minimum tillage, resulting in moderate crop residue cover.

3. **Corn:** Corn planted in clean, ploughed fields with no crop residue. The background is clean soil.

4. **Grass-pasture:** Managed pastureland, showing a stable, moderate vegetation signature throughout the season.

5. **Grass-trees:** A transitional zone between pastureland and deciduous forest, containing mixed grass and tree species.

6. **Grass-pasture-mowed:** Mowed pasture fields. The cutting of the grass reduces leaf area, lowering NIR reflectance.

7. **Hay-windrowed:** Grass cut and piled into rows (windrows) for drying. The drying process causes chlorophyll degradation and water loss, leading to a shift from a green vegetation spectrum to a dry biomass spectrum.

8. **Oats:** Cereal grain crop, represented by a very small number of pixels, creating a severe class imbalance.

9. **Soybean-notill:** Soybeans planted in fields with zero tillage, containing high residue cover from previous corn crops.

10. **Soybean-mintill:** Soybeans planted with minimum tillage, the most abundant class in the dataset.

11. **Soybean-clean:** Soybeans planted in clean ploughed fields, exhibiting a clean soil-soybean spectral mixture.

12. **Wheat:** Winter wheat crops. At the time of data acquisition, the wheat was mature and dry, yielding a bright yellow spectrum.

13. **Woods:** Natural stands of deciduous forest, characterized by multi-layered canopies with high leaf area index and deep water absorption features.

14. **Buildings-Grass-Trees-Drives:** A highly heterogeneous class containing concrete paths, metal roofs, gravel drives, and surrounding lawns and trees.

15. **Stone-Steel-Towers:** Metal transmission towers and stone gravel pads, showing diagnostic absorption features of iron oxides and inorganic minerals.

4.2 Spectral Preprocessing and Data Cleaning

Hyperspectral signatures in their raw form may be compromised due to sensor noise, variability in light scattering due to surface characteristics, and atmospheric interferences. In order to increase SNR and extract information about the absorption characteristics of interest, spectral preprocessing becomes essential. In the current work, multiple preprocessing techniques have been employed to cleanse the data prior to wavelength selection and classification, adapted from the lettuce classification preprocessing framework evaluated by Dilillo et al. [4].

4.2.1 Spectral Preprocessing Methods

Let $\mathbf{x} = [x_1, x_2, \ldots, x_B]^T \in \mathbb{R}^B$ represent the raw reflectance spectrum of a single pixel across B spectral bands. The preprocessing techniques are formulated as follows:

1. **Savitzky-Golay (SG) Filtering:** The SG filter is a local polynomial fitting approach to smoothing the spectral data without distorting absorption features of the signal. Given a spectral window of width $2m + 1$ around wavelength point i, the smoothed spectral value $\hat{x}_i$ is calculated by fitting a polynomial with degree p ($p < 2m + 1$):

$$\hat{x}_i = \sum_{j=-m}^{m} c_j x_{i+j} \tag{4.1}$$

   where c_j are the filter coefficients calculated from polynomial fitting. In the first derivative mode (**SG1**), the derivative of the polynomial fitting function is taken at the center wavelength, which disentangles overlapped peaks and removes offset:

$$x_i^{(1)} = \frac{d}{d\lambda}\left[\sum_{k=0}^{p} a_k \lambda^k\right]_{\lambda=\lambda_i} \tag{4.2}$$

   In our algorithm, a window length of 30 and second-order polynomial fit are chosen ($p = 2$).

2. **Standard Normal Variate (SNV):** SNV is a normalization procedure to address issues related to multiplicative effects, shifts in baseline levels, and scattering due to the difference in particle size or uneven leaf surface. Specifically, for each individual spectrum $\mathbf{x}$, the mean-centered values are scaled using their own standard deviation:

$$x_i^{\mathrm{SNV}} = \frac{x_i - \bar{x}}{s_x} \tag{4.3}$$

   where $\bar{x} = \frac{1}{B}\sum_{j=1}^{B} x_j$ is the mean reflectance of the spectrum, and $s_x = \sqrt{\frac{1}{B-1}\sum_{j=1}^{B}(x_j - \bar{x})^2}$ is the standard deviation.

3. **Multiplicative Scatter Correction (MSC):** MSC is a scattering correction approach that aligns the spectra to a reference, usually the mean spectrum of the whole dataset. Suppose the mean spectrum of all training samples is denoted by $\mathbf{m} = [m_1, m_2, \ldots, m_B]^T$. Then, the intercept a and slope b are computed through linear regression of the form:

$$x_i = a + b m_i + e_i, \quad i = 1, 2, \ldots, B \tag{4.4}$$

where e_i represents the residual error. The corrected spectrum is then computed by removing the estimated additive and multiplicative effects:

$$x_i^{\text{MSC}} = \frac{x_i - a}{b} \tag{4.5}$$

4.2.2 PCA-Based Outlier Detection

Anomalies are present in hyperspectral datasets because of shadows, boundaries between different classes, or instrumental errors. Therefore, an outlier detection approach based on PCA is employed to remove any outliers that might affect the quality of feature selection.

Firstly, PCA is performed to obtain a low-dimensional orthogonal basis for the spectra data. Let $\mathbf{z} \in \mathbb{R}^D$ be the score vector of an image pixel obtained in a subspace formed by the first D principal components ($D = 10$ in our pipeline). Then, the Mahalanobis distance D_M between the sample $\mathbf{z}$ and the centroid of the scores distribution can be calculated as follows:

$$D_M^2 = (\mathbf{z} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{z} - \boldsymbol{\mu}) \tag{4.6}$$

where $\boldsymbol{\mu} \in \mathbb{R}^D$ is the mean vector of the PCA scores, and $\boldsymbol{\Sigma} \in \mathbb{R}^{D \times D}$ is the covariance matrix of the scores.

Assuming the PCA scores follow a multivariate normal distribution, the squared Mahalanobis distance D_M^2 follows a chi-squared distribution with D degrees of freedom ($D_M^2 \sim \chi_D^2$). A sample is classified as an outlier and removed from the dataset if its distance exceeds the critical value at a 95% confidence level ($\alpha = 0.05$):

$$D_M^2 > \chi_D^2(1 - \alpha) \tag{4.7}$$

This mathematical boundary defines a 95% confidence ellipsoid in the PCA score space, effectively filtering out spatial and spectral anomalies.

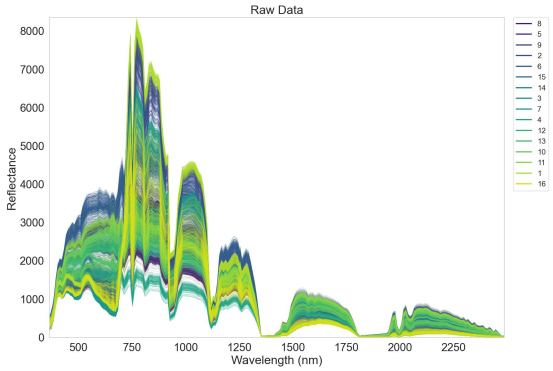

Figure 4.3: Raw reflectance spectra of the Salinas dataset across all 204 spectral bands before preprocessing and cleaning.

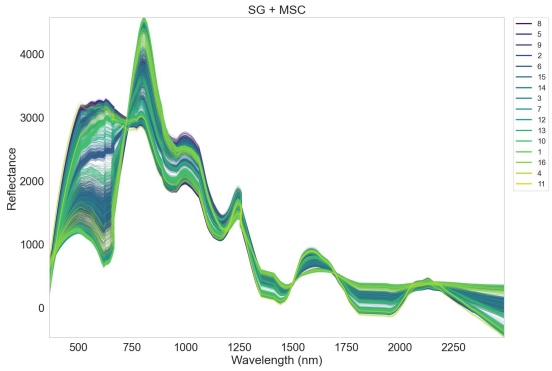

Figure 4.4: Preprocessed and cleaned reflectance spectra of the Salinas dataset (using the 10_SG_MSC pipeline) after outlier removal.

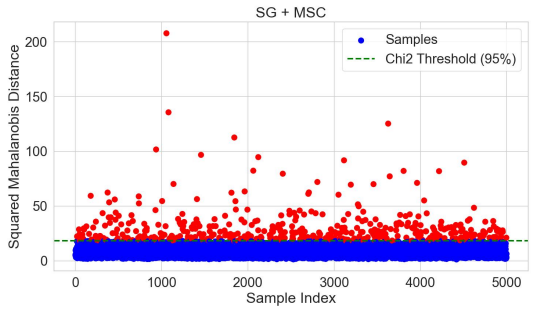

Figure 4.5: PCA-based outlier detection for the Salinas dataset showing the critical χ^2 threshold (18.31) at a 95% confidence level.

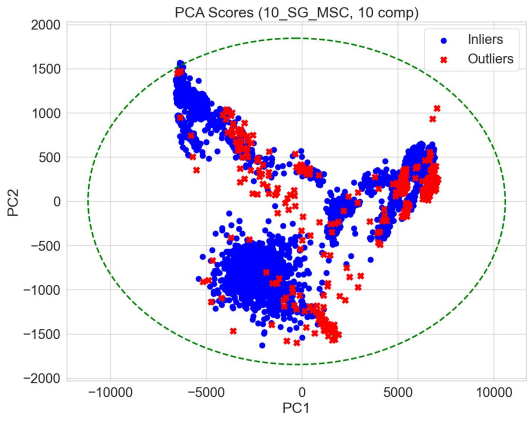

Figure 4.6: PCA score scatter plot (PC1 vs PC2) with the 95% confidence boundary ellipse isolating inliers from outliers.

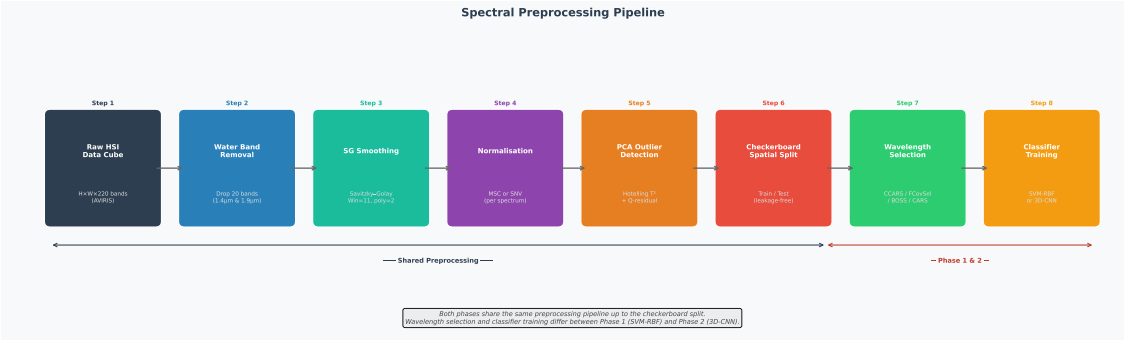

Figure 5.1: Spectral preprocessing pipeline applied to both datasets before wavelength selection and classification. The pipeline consists of eight sequential stages: raw data ingestion, atmospheric water band removal, Savitzky–Golay smoothing, multiplicative scatter correction (MSC) or standard normal variate (SNV) normalisation, PCA-based outlier detection, leakage-free checkerboard spatial splitting, wavelength selection, and final classifier training. Steps 1–6 are shared between Phase 1 (SVM-RBF) and Phase 2 (3D-CNN).

Chapter 5

Methodology and Contribution

This chapter describes the original contributions of this thesis. In particular, it presents the mathematical modeling of the leakage-free checkerboard spatial splitting strategy, the design of the CCARS$_K$ framework for non-linear classifiers, the architecture of the proposed adaptive 3D-CNN for deep joint spatial-spectral classification, and the statistical validation protocols used to confirm the significance of our results.

Figure 5.1 provides a high-level overview of the complete spectral preprocessing pipeline shared by both Phase 1 and Phase 2 of the study.

5.1 Leakage-Free Spatial Splitting Strategy

To achieve spatial data independence and establish a realistic evaluation benchmark, we propose a spatially independent checkerboard block-based splitting strategy. Random pixel-wise partitioning, which is widely used in the literature, fails to address spatial autocorrelation. By randomly allocating adjacent pixels to the training and testing sets, information leaks between the sets, leading to over-optimistic accuracy estimates. Our approach resolves this by dividing the image into non-overlapping spatial blocks and assigning entire blocks to either training or testing.

Consider the hyperspectral image cube $\mathbf{X} \in \mathbb{R}^{H \times W \times B}$ to be divided into non-overlapping blocks of size $B_s \times B_s$ pixels. The value of the block size B_s is selected as 8 for the Indian Pines dataset and 16 for the Salinas dataset, based on the spatial resolution (20 m for Indian Pines and 3.7 m for Salinas) and the average field area in each scene. For a pixel at spatial coordinates (h, w) where $1 \leq h \leq H$ and $1 \leq w \leq W$, its block coordinates (r, c) are calculated as:

$$r = \left\lfloor \frac{h - d_r}{B_s} \right\rfloor, \quad c = \left\lfloor \frac{w - d_c}{B_s} \right\rfloor \tag{5.1}$$

where $d_r, d_c \in [0, B_s - 1]$ represent spatial offsets in the horizontal and vertical directions, respectively.

Each block (r, c) is assigned to either the training set ($\mathcal{D}_{\text{train}}$) or the test set ($\mathcal{D}_{\text{test}}$) using a binary checkerboard partitioning rule:

$$(r, c) \in \mathcal{D}_{\text{train}} \quad \text{if} \quad (r + c) \pmod 2 = 0 \tag{5.2}$$

$$(r, c) \in \mathcal{D}_{\text{test}} \quad \text{if} \quad (r + c) \pmod 2 \neq 0 \tag{5.3}$$

This checkerboard partitioning ensures that training and testing blocks only touch at diagonal corners. They do not share horizontal or vertical boundaries, which minimizes spatial correlation across the split boundary.

Because crop fields in hyperspectral datasets are highly localized and sometimes occupy only a small portion of the image, a fixed checkerboard split can result in some classes being entirely absent from either the training or the test partition. To prevent this, we implement a class-balance optimization loop that dynamically adjusts the offsets d_r and d_c to maximize class overlap.

Let $\mathcal{C}$ represent the set of all unique ground-truth classes in the dataset (excluding the background class 0). Let $\mathcal{C}_{\text{train}}(d_r, d_c)$ and $\mathcal{C}_{\text{test}}(d_r, d_c)$ represent the sets of classes present in the training and testing partitions under the offsets (d_r, d_c). The optimal offsets (d_r^*, d_c^*) are determined by maximizing the cardinality of the intersection of these two sets:

$$(d_r^*, d_c^*) = \arg\max_{(d_r, d_c)} |\mathcal{C}_{\text{train}}(d_r, d_c) \cap \mathcal{C}_{\text{test}}(d_r, d_c)| \tag{5.4}$$

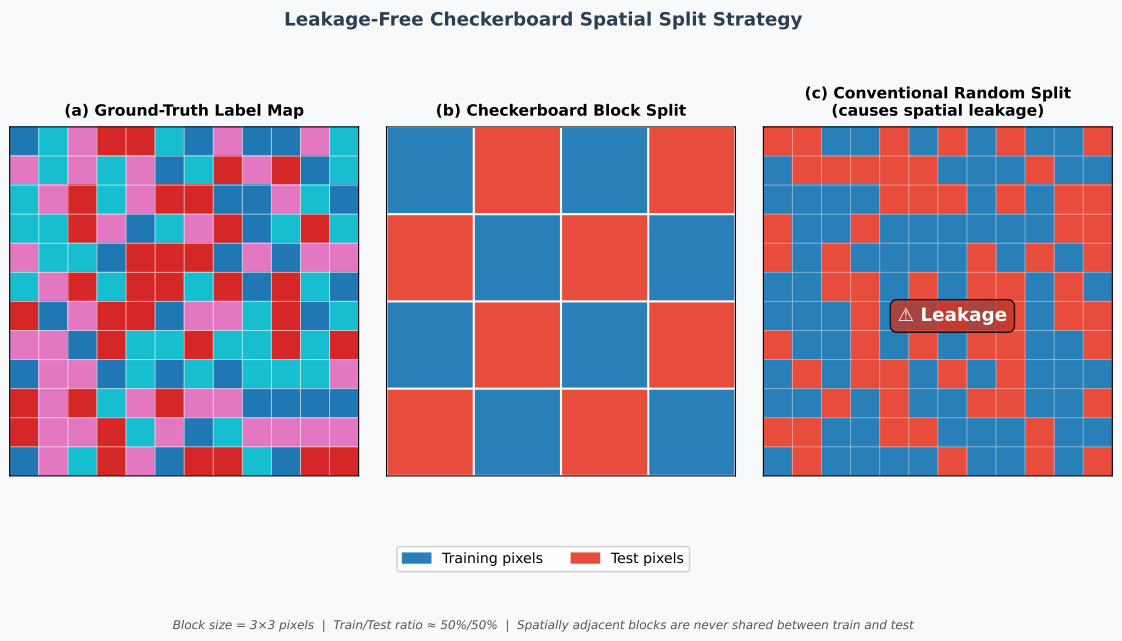

Figure 5.2: Comparison between the proposed leakage-free checkerboard block split and the conventional random pixel-wise split. (a) Schematic ground-truth label map with four classes. (b) Checkerboard spatial split: training pixels (blue) and test pixels (red) are assigned at the block level, so no horizontally or vertically adjacent blocks share the same partition. (c) Conventional random split: training and test pixels are interleaved at the single-pixel level, exposing every test pixel to spatially correlated training neighbours and causing significant leakage of spatial information.

The search is conducted iteratively over all possible offset combinations. The optimal partition satisfies the condition that all classes are represented in both sets:

$$\mathcal{C}_{\text{train}}(d_r^*, d_c^*) = \mathcal{C}_{\text{test}}(d_r^*, d_c^*) = \mathcal{C} \tag{5.5}$$

If extreme class rarity prevents this condition from being met, the offsets are selected to maximize class overlap. This procedure is detailed in Algorithm 1.

Figure 5.2 illustrates the concept of the checkerboard split compared with a conventional random pixel-wise split. Panel (a) shows a schematic ground-truth label map; panel (b) shows the resulting checkerboard assignment (training blocks in blue, test blocks in red), where all spatially adjacent blocks belong to different partitions; panel (c) shows a conventional random split, where training and test pixels are arbitrarily interleaved, exposing test pixels to spatially correlated training neighbours and causing spatial data leakage.

Algorithm 1: Checkerboard Spatial Splitting and Offset Optimization

```
Input: Image dimensions H, W; Ground truth map Y; Block size Bs.
Output: Training set pixels D_train, Test set pixels D_test,
        Offsets dr_opt, dc_opt.

1. Initialize optimal_overlap = -1
2. Initialize dr_opt = 0, dc_opt = 0
3. Let C be the set of unique class labels in Y (excluding 0)

4. For dr from 0 to Bs - 1:
    For dc from 0 to Bs - 1:
        Initialize C_train = empty set
        Initialize C_test = empty set

        For each pixel (h, w) in Y with label > 0:
            Calculate block coordinates:
                r = floor((h - dr) / Bs)
                c = floor((w - dc) / Bs)

            If (r + c) mod 2 == 0:
                Add class label Y(h,w) to C_train
            Else:
                Add class label Y(h,w) to C_test

        Calculate overlap = size(intersection(C_train, C_test))

        If overlap > optimal_overlap:
            optimal_overlap = overlap
            dr_opt = dr
            dc_opt = dc

        If overlap == size(C):
            Break search (all classes represented in both splits)

5. Using (dr_opt, dc_opt), assign pixels to D_train and D_test:
   D_train = { (h, w) in Y |
     (floor((h-dr_opt)/Bs) + floor((w-dc_opt)/Bs)) mod 2 == 0 }
   D_test = { (h, w) in Y |
     (floor((h-dr_opt)/Bs) + floor((w-dc_opt)/Bs)) mod 2 != 0 }

6. Return D_train, D_test, dr_opt, dc_opt.
```

5.2 CCARS$_K$ Wavelength Selection Framework

The Competitive Calibration Adaptive Reweighted Sampling (CCARS) framework, originally proposed by Dilillo et al. [3, 5] for lettuce classification, is designed to identify the most robust and informative spectral bands. While the original CCARS is coupled with linear PLS models, our extended framework – hereafter referred to as CCARS$_K$, where K denotes the target number of selected bands – integrates a calibration-validation split and extends the selection criteria to tuned non-linear classifiers.

The mathematical execution of the CCARS algorithm is structured into the following sequential steps:

5.2.1 Step 1: Calibration-Validation Split

To prevent feature selection leakage, the training set $\mathcal{D}_{\text{train}}$ obtained from the checkerboard split is randomly partitioned into a Calibration set ($\mathcal{D}_{\text{cal}}$, 50%) and a Final Validation set ($\mathcal{D}_{\text{final}}$, 50%). Wavelength selection is performed strictly using $\mathcal{D}_{\text{cal}}$, while $\mathcal{D}_{\text{final}}$ acts as a held-out set to evaluate the classification performance of the selected bands at each iteration.

5.2.2 Step 2: Monte Carlo Sampling and Weight Generation

The algorithm executes R Monte Carlo sampling runs (in our setup, $R = 500$). In each run $r \in [1, R]$, a subset containing 80% of the samples from $\mathcal{D}_{\text{cal}}$ is randomly selected with replacement. A Multi-Class PLS-DA model is trained on this subset.

Since hyperspectral classification is a multi-class problem with C classes, the PLS regression coefficients form a matrix $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{B \times C}$, where $\beta_{i,c}$ represents the regression coefficient of band i for class c. The importance weight w_i for each band i is calculated by aggregating its coefficients across all classes using the root sum of squares:

$$w_i = \sqrt{\sum_{c=1}^{C} \beta_{i,c}^2}, \quad i = 1, 2, \ldots, B \tag{5.6}$$

The average importance weight W_i for band i over all R Monte Carlo runs is:

$$W_i = \frac{1}{R} \sum_{r=1}^{R} w_i^{(r)} \tag{5.7}$$

The normalized weight u_i, representing the relative importance of band i, is then computed as:

$$u_i = \frac{W_i}{\sum_{j=1}^{B} W_j} \tag{5.8}$$

5.2.3 Step 3: Exponential Decreasing Function (EDF)

During the N variable elimination iterations (typically $N = 100$), the ratio of retained bands r_k at iteration k is controlled by an Exponential Decreasing Function (EDF):

$$r_k = ae^{-bk} \tag{5.9}$$

The coefficients a and b are determined by the boundary conditions. At $k = 0$, all bands are retained ($r_0 = 1$), which yields $a = 1$. At the final iteration $k = N$, the ratio must equal the target compression ratio $\frac{K}{B}$, where K is the desired number of bands (e.g., $50, 30, 20$, or 10). This yields the parameter b:

$$b = -\frac{1}{N} \ln\left(\frac{K}{B}\right) \tag{5.10}$$

5.2.4 Step 4: Adaptive Reweighted Sampling (ARS)

In each iteration k, the normalized weights u_i are used to construct a roulette wheel. An Adaptive Reweighted Sampling (ARS) process performs roulette wheel selection to choose $r_k \times B$ bands. Bands with larger weights have a higher probability of being selected, while uninformative bands are systematically discarded.

5.2.5 Step 5: Non-Linear Classifier Validation and Selection

At each iteration k, the selected subset of bands is used to train the target tuned non-linear classifier (specifically SVM-RBF with $C = 100$ and $\gamma = 0.01$, or Random Forest with 500 trees) on the training portion of $\mathcal{D}_{\text{cal}}$. The classifier is then evaluated on the held-out validation set $\mathcal{D}_{\text{final}}$ by computing the Overall Accuracy (OA):

$$\text{OA}_k = \frac{\text{Number of correctly classified samples in } \mathcal{D}_{\text{final}} \text{ using } r_k \times B \text{ bands}}{\text{Total number of samples in } \mathcal{D}_{\text{final}}} \tag{5.11}$$

The iteration k^* that yields the maximum classification accuracy on the held-out set is identified:

$$k^* = \arg\max_{k \in [1,N]} \text{OA}_k \tag{5.12}$$

The subset of bands selected at iteration k^* is output as the final optimized wavelength subset. The complete CCARS variable selection loop is described in Algorithm 2.

Algorithm 2: CCARS Non-Linear Wavelength Selection Loop

```
Input: Calibration dataset D_cal; Target number of bands K;
       Total iterations N; Monte Carlo runs R.
Output: Optimal subset of wavelengths S_opt.

1. Partition D_cal randomly into:
   - Calibration training D_cal_train (50%)
   - Calibration validation D_cal_val (50%)

2. Initialize active bands set S_1 = {1, 2, ..., B}
3. Calculate EDF coefficients a and b:
   a = 1
   b = -ln(K / B) / N

4. For iteration k = 1 to N:
   a. Calculate retention ratio: r_k = a * exp(-b * k)
   b. Calculate number of bands to retain: M_k = round(r_k * B)

   c. For run = 1 to R:
      i. Draw 80% bootstrap sample from D_cal_train
      ii. Fit PLS-DA model on active bands S_k
      iii. Calculate regression coefficient matrix Beta
      iv. For each band i in S_k, calculate importance weight:
          w_i^(run) = sqrt( sum_{c=1}^C beta_{i, c}^2 )

   d. Calculate average weight for each band:
      w_i = mean(w_i^(run) over all R runs)
      u_i = w_i / sum(w_j for all active bands)

   e. Construct roulette wheel with probabilities u_i.
   f. Select M_k bands using roulette wheel selection.
   g. Update active bands set S_{k+1} to be the selected bands.

   h. Train tuned SVM-RBF model on D_cal_train using S_{k+1}.
   i. Evaluate Overall Accuracy (OA_k) on D_cal_val.

5. Identify optimal iteration: k_opt = argmax_k (OA_k).
6. Return optimal bands subset S_opt = S_{k_opt}.
```

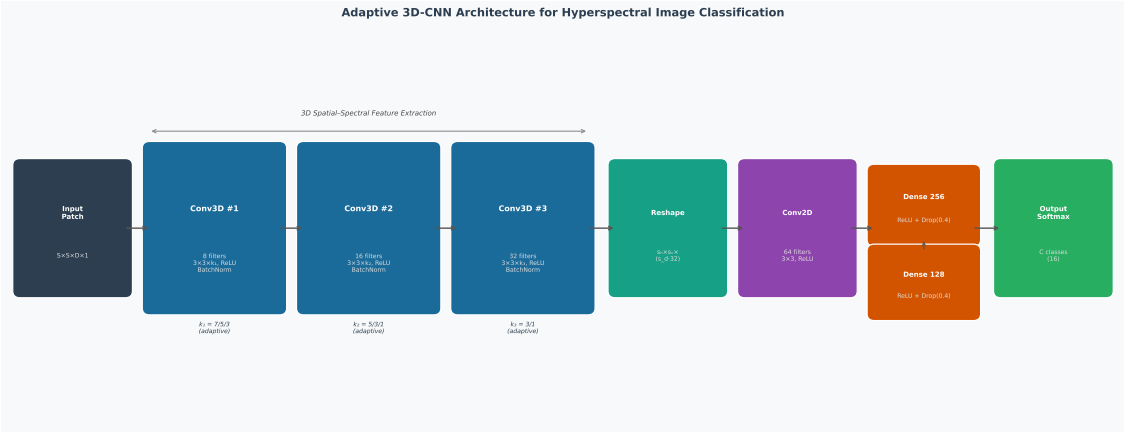

Figure 5.3: Architecture of the proposed adaptive 3D-CNN for hyperspectral image classification. The network takes a volumetric patch $\mathbf{P} \in \mathbb{R}^{S\times S\times D\times 1}$ as input and passes it through three 3D convolutional layers (8, 16, and 32 filters respectively) with adaptive spectral kernel sizes (k_1, k_2, k_3) that adjust dynamically to the spectral depth D of the selected band subset. The 3D feature maps are then reshaped and processed by a 2D convolutional layer (64 filters), followed by two fully-connected dense layers (256 and 128 units) with dropout regularisation ($p = 0.4$), and a final softmax output layer producing class probabilities over $C = 16$ land-cover classes.

5.3 Adaptive 3D-CNN Classifier Architecture

In the second phase of the study, a deep spatial-spectral classifier based on a 3D Convolutional Neural Network (3D-CNN) is designed to process the selected wavelength subsets. Standard 2D-CNNs operate on spatial features but fail to fully exploit the continuous spectral curves of hyperspectral pixels. A 3D-CNN solves this by utilizing three-dimensional kernels that slide across both spatial and spectral dimensions simultaneously. The complete layer-by-layer architecture of the network is illustrated in Figure 5.3.

Let the input volumetric data patch extracted around a center pixel be denoted as $\mathbf{P} \in \mathbb{R}^{S\times S\times D\times 1}$, where $S \times S$ represents the spatial window size, D represents the spectral depth (the number of selected bands), and 1 is the number of input channels. The spatial window size is set dynamically based on D: $S = 13$ for $D > 80$, $S = 19$ for $D > 40$, and $S = 5$ for $D \leq 40$ to optimize GPU memory footprint.

The 3D convolution operation at layer l is mathematically defined as:

$$X^{(l)}_{x,y,z,f} = \mathrm{ReLU}\left(\sum_{i=-a}^{a}\sum_{j=-b}^{b}\sum_{k=-c}^{c}\sum_{c_{\mathrm{in}}=1}^{C_{\mathrm{in}}} W^{(l)}_{i,j,k,c_{\mathrm{in}},f} X^{(l-1)}_{x+i,y+j,z+k,c_{\mathrm{in}}} + b^{(l)}_f\right) \tag{5.13}$$

where $W^{(l)}$ represents the weights of the 3D convolutional kernel of shape $(2a+1) \times (2b+1) \times (2c+1)$, $b_f^{(l)}$ is the bias for filter f, C_{in} is the number of input channels, and $X^{(l-1)}$ is the activation map from the previous layer.

5.3.1 Adaptive Spectral Kernel Design

A major challenge in pairing wavelength selection with 3D-CNNs is that different selection algorithms return subsets of widely varying sizes D (ranging from 4 to 129 bands). If a static spectral kernel size (e.g., $k_1 = 7$) is used in the first Conv3D layer, a dimension mismatch will occur whenever $D < k_1$. To address this, we implement an adaptive spectral kernel sizing mechanism in the convolutional layers.

Let k_1, k_2, and k_3 represent the spectral kernel sizes for the first, second, and third Conv3D layers, respectively. The adaptive selection rules are formulated as follows:

1. **Layer 1 Spectral Kernel (k_1):** The kernel size is selected dynamically based on the input spectral depth D:

$$k_1 = \begin{cases} 7 & \text{if } D \geq 7 \\ 5 & \text{if } 5 \leq D < 7 \\ 3 & \text{if } D < 5 \end{cases} \tag{5.14}$$

   The remaining spectral depth after the first layer is $d_1 = D - (k_1 - 1)$.

2. **Layer 2 Spectral Kernel (k_2):** The kernel size is selected based on the remaining spectral depth d_1:

$$k_2 = \begin{cases} 5 & \text{if } d_1 \geq 5 \\ 3 & \text{if } 3 \leq d_1 < 5 \\ 1 & \text{if } d_1 < 3 \end{cases} \tag{5.15}$$

   The remaining spectral depth after the second layer is $d_2 = d_1 - (k_2 - 1)$.

3. **Layer 3 Spectral Kernel (k_3):** The kernel size is selected based on the remaining spectral depth d_2:

$$k_3 = \begin{cases} 3 & \text{if } d_2 \geq 3 \\ 1 & \text{if } d_2 < 3 \end{cases} \tag{5.16}$$

5.3.2 Network Layer Specifications

The complete architecture of the adaptive 3D-CNN is structured as follows:

1. **3D Convolutional Block:**
   - **Conv3D Layer 1:** Applies 8 filters of kernel size $(3 \times 3 \times k_1)$ with ReLU activation, producing feature maps of shape $(S-2) \times (S-2) \times d_1 \times 8$.

- **Conv3D Layer 2:** Applies 16 filters of kernel size $(3\times3\times k_2)$ with ReLU activation, producing feature maps of shape $(S-4)\times(S-4)\times d_2\times 16$.
- **Conv3D Layer 3:** Applies 32 filters of kernel size $(3\times3\times k_3)$ with ReLU activation, producing feature maps of shape $(S-6)\times(S-6)\times d_3\times 32$, where $d_3 = d_2 - (k_3 - 1)$.

2. **Dimensionality Reshaping:** Let s_h, s_w, s_d, s_f represent the spatial height, width, spectral depth, and number of filters of the Conv3D Layer 3 output. The volumetric tensor is reshaped into a three-dimensional tensor of shape $(s_h \times s_w \times (s_d \cdot s_f))$ to make it compatible with standard 2D convolutions:

$$\mathbf{X}_{\text{reshaped}} = \text{Reshape}\left(\mathbf{X}_{\text{conv3D}}, [s_h, s_w, s_d \cdot s_f]\right) \tag{5.17}$$

3. **2D Convolutional Block:** A Conv2D layer with 64 filters of spatial kernel size (3×3) and ReLU activation is applied to extract spatial texture patterns from the reshaped feature maps.

4. **Flattening and Dense Block:**

   - The feature maps are flattened into a single-dimensional vector.
   - **Dense Layer 1:** A fully-connected layer with 256 units and ReLU activation, followed by a Dropout layer with a rate of 0.4 to prevent overfitting:

$$\mathbf{h}_1 = \text{Dropout}_{0.4}\left(\text{ReLU}\left(\mathbf{W}_1\mathbf{x}_{\text{flat}} + \mathbf{b}_1\right)\right) \tag{5.18}$$

   - **Dense Layer 2:** A fully-connected layer with 128 units and ReLU activation, followed by a Dropout layer with a rate of 0.4:

$$\mathbf{h}_2 = \text{Dropout}_{0.4}\left(\text{ReLU}\left(\mathbf{W}_2\mathbf{h}_1 + \mathbf{b}_2\right)\right) \tag{5.19}$$

   - **Output Layer:** A fully-connected layer with C units and Softmax activation:

$$p(y = c \mid \mathbf{P}) = \frac{e^{\mathbf{w}_c^T\mathbf{h}_2 + b_c}}{\sum_{j=1}^{C} e^{\mathbf{w}_j^T\mathbf{h}_2 + b_j}} \tag{5.20}$$

The model is compiled using the Adam optimizer with a learning rate of 0.001, and trained using categorical cross-entropy loss over 100 epochs, utilizing real-time data augmentation (rotations and flips) to enhance model generalization.

5.4 Statistical Validation and Overfitting Detection

To ensure that the classification performance of both the traditional machine learning models (Phase 1) and the deep learning architectures (Phase 2) is statistically

significant and not a result of overfitting or random chance, two rigorous validation protocols are implemented: permutation testing and learning curve analysis.

5.4.1 Adaptive Permutation Testing Protocol

Permutation testing is a non-parametric statistical validation method used to test the null hypothesis $\mathcal{H}_0$, which states that there is no relationship between the spectral features $\mathbf{X}$ and the target crop labels $\mathbf{y}$. By breaking the association between features and labels, we can construct the empirical distribution of the test statistic under the null hypothesis.

The mathematical procedure for the permutation test is formulated as follows:

1. Let S_0 represent the baseline test score (e.g., Overall Accuracy) achieved by the classifier trained on the actual training dataset $(\mathbf{X}_{\text{train}}, \mathbf{y}_{\text{train}})$ and evaluated on the test set $(\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}})$.

2. A permutation operator π is applied to randomly shuffle the training label vector $\mathbf{y}_{\text{train}}$, creating a permuted label vector $\mathbf{y}^{\pi}_{\text{train}}$. This operation destroys any true feature-label correspondence while preserving the class distribution and marginal statistics.

3. The classifier is retrained on the permuted dataset $(\mathbf{X}_{\text{train}}, \mathbf{y}^{\pi}_{\text{train}})$ and its performance S^{π} is evaluated on the original test set $(\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}})$.

4. Steps 2 and 3 are repeated N_{perm} times (where $N_{\text{perm}} = 1000$ in our configuration) to generate a set of permuted scores $\{S^{\pi}_1, S^{\pi}_2, \ldots, S^{\pi}_{N_{\text{perm}}}\}$.

5. The empirical p-value is calculated as the proportion of permuted scores that are greater than or equal to the baseline score S_0:

$$p = \frac{\sum_{i=1}^{N_{\text{perm}}} \mathbb{I}\left(S^{\pi}_i \geq S_0\right) + 1}{N_{\text{perm}} + 1} \tag{5.21}$$

   where $\mathbb{I}(\cdot)$ is the indicator function that outputs 1 if the condition is true and 0 otherwise.

The addition of $+1$ in both the numerator and the denominator acts as a correction (pseudo-count) to prevent the p-value from being exactly zero, ensuring a conservative estimate. If $p < 0.05$, the null hypothesis $\mathcal{H}_0$ is rejected, confirming that the model has learned statistically significant patterns from the selected wavelengths. This validation protocol is detailed in Algorithm 3.

Algorithm 3: Permutation Testing Protocol

```
Input: Training dataset (X_train, y_train);
       Test dataset (X_test, y_test);
       Number of permutation runs N_perm.
Output: Empirical p-value.

1. Train the classifier on original data (X_train, y_train)
2. Predict on test set X_test to get baseline accuracy S_0
3. Initialize count = 0

4. For perm = 1 to N_perm:
   a. Create permuted labels: y_train_perm = Shuffle(y_train)
   b. Train the classifier on (X_train, y_train_perm)
   c. Predict on test set X_test to get permuted accuracy S_perm
   d. If S_perm >= S_0:
         count = count + 1

5. Calculate empirical p-value:
   p = (count + 1) / (N_perm + 1)

6. Return p.
```

5.4.2 Learning Curve Analysis

Learning curves are utilized to monitor the generalization capability of the classifiers and detect the presence of overfitting or underfitting. This is achieved by evaluating the training and validation scores as a function of the training set size.

Let the training dataset $\mathcal{D}_{\text{train}}$ be divided into subsets of increasing size, denoted by fractions $\theta \in [0.1, 1.0]$. For each fraction θ, a subset of size $\theta \cdot |\mathcal{D}_{\text{train}}|$ is randomly sampled. The classifier is trained on this subset, and its performance is evaluated on both the training subset (yielding training accuracy $\text{Acc}_{\text{train}}(\theta)$) and the validation set (yielding validation accuracy $\text{Acc}_{\text{val}}(\theta)$).

By plotting $\text{Acc}_{\text{train}}(\theta)$ and $\text{Acc}_{\text{val}}(\theta)$ against the training sample size, we evaluate the model's behavior:

- **Overfitting (High Variance):** Indicated by a large, persistent gap between the high training accuracy and the significantly lower validation accuracy as $\theta \to 1.0$.
- **Underfitting (High Bias):** Indicated by low accuracies in both the training

and validation curves, even as θ increases.

- **Generalization:** Characterized by the convergence of the training and validation curves towards a high accuracy level as the training sample size increases.

This analysis confirms that the reduced feature set (selected wavelengths) contains sufficient representative information to prevent overfitting.

Chapter 6

Experimental Results and Discussion

This chapter presents the experimental setup, numerical results, and comparative analyses of the proposed frameworks on the Salinas and Indian Pines hyperspectral datasets. First, the experimental configuration and hyperparameter settings are detailed. Second, the classification results for Phase 1 (traditional machine learning) and Phase 2 (deep learning) are presented and analyzed. Finally, we discuss the spectral efficiency, the Jaccard similarity index, and the statistical significance of the results.

6.1 Experimental Configuration

To ensure the reproducibility of the results, all experiments are conducted in a standardized computational environment. The preprocessing, feature selection, and traditional machine learning pipelines are implemented using Python 3.9, leveraging scientific libraries including NumPy, SciPy, Pandas, and Scikit-learn. The deep learning pipelines are developed using TensorFlow 2.8.0 and Keras, executing on an NVIDIA Tesla T4 GPU with 16 GB of VRAM.

The hyperparameter settings for the preprocessing and classification algorithms are configured as follows:

6.1.1 Preprocessing and Outlier Detection Parameters

- **Savitzky-Golay (SG) Filtering:** The moving window size is set to 31 bands, and the polynomial order is set to $p = 2$. For derivative calculations, the first derivative (**SG1**) is evaluated.

- **Outlier Detection:** The number of PCA components used to span the score

subspace is set to $D = 10$. The critical threshold is evaluated using the chi-squared distribution at a 95% confidence level ($\alpha = 0.05$).

6.1.2 Phase 1: Hyperparameters

The hyperparameters of the traditional classifiers are optimized using a grid search under the checkerboard spatial split:

- **Support Vector Machine with Radial Basis Function kernel (SVM-RBF):** The regularization parameter is set to $C = 100$, and the kernel coefficient is set to $\gamma = 0.01$. The class weights are set to 'balanced' to handle class imbalance.
- **Random Forest (RF):** The number of decision trees is set to 500, with no maximum depth limit. The split quality is evaluated using the Gini impurity index, and class weights are set to 'balanced_subsample'.
- **Partial Least Squares Discriminant Analysis (PLS-DA):** The number of latent variables (PLS components) is evaluated for values in $\{2, 3, 4\}$.

6.1.3 Phase 2: Hyperparameters

The 3D-CNN architecture is trained under the following specifications:

- **Optimization:** The network is optimized using the Adam optimizer with an initial learning rate of 0.001, a decay rate of 10^{-6}, and a categorical cross-entropy loss function.
- **Batch Size:** The batch size is scaled dynamically based on the number of selected bands to prevent out-of-memory errors:

$$\text{Batch Size} = \begin{cases} 64 & \text{if } D > 80 \\ 128 & \text{if } 40 < D \leq 80 \\ 256 & \text{if } D \leq 40 \end{cases} \tag{6.1}$$

- **Training and Callbacks:** The network is trained for 100 epochs. A model checkpointing callback monitors the validation accuracy at each epoch, saving the weights only when the validation accuracy reaches a new maximum.
- **Data Augmentation:** To enhance spatial generalization, real-time image augmentation is applied to the training patches, including random rotations up to 90° and random horizontal and vertical flips.

Table 6.1: Phase 1 wavelength selection results on the Salinas dataset using the proposed CCARS$_K$ and CARS frameworks.

Method	Bands (K)	Classifier	PLS Comp.	Accuracy (OA)	F1-Macro
CCARS$_{K=50}$	**50**	**SVM-RBF**	**2**	**92.63%**	**0.9633**
CCARS$_{K=50}$	50	SVM-RBF	3	91.81%	0.9610
CCARS$_{K=50}$	50	Random Forest	2	91.56%	0.9524
CARS	50	RF	3	91.55%	0.9515
CARS	50	SVM-RBF	3	91.42%	0.9577
CCARS$_{K=30}$	30	SVM-RBF	2	90.81%	0.9491
CCARS$_{K=20}$	20	Random Forest	2	89.72%	0.9366
CCARS$_{K=10}$	10	Random Forest	2	88.27%	0.9270
Baseline (All Bands)	204	PLS-DA	4	52.97%	0.2495

Table 6.2: Phase 1 wavelength selection results on the Indian Pines dataset using the proposed CCARS$_K$ and CARS frameworks.

Method	Bands (K)	Classifier	PLS Comp.	Accuracy (OA)	F1-Macro
CARS	**50**	**SVM-RBF**	**3**	**77.61%**	**0.7563**
CARS	50	SVM-RBF	2	76.51%	0.7818
CCARS$_{K=50}$	50	SVM-RBF	4	75.65%	0.7647
CCARS$_{K=50}$	50	Random Forest	4	75.57%	0.7175
CCARS$_{K=30}$	30	Random Forest	4	74.54%	0.7179
CCARS$_{K=20}$	20	Random Forest	4	68.58%	0.5952
CCARS$_{K=10}$	10	Random Forest	4	62.20%	0.5131
Baseline (All Bands)	200	PLS-DA	4	50.47%	0.2557

6.2 Phase 1: Results

This section presents the classification performance of traditional machine learning models combined with statistical wavelength selection. The results are analyzed in two parts: first, the performance of the proposed CCARS$_K$ framework against CARS across different classifiers, and second, a comparative evaluation of the five wavelength selection methods under the optimized SVM-RBF model.

6.2.1 CCARS$_K$ and CARS Performance

The classification results for CCARS$_K$ (retaining $K \in \{10, 20, 30, 50\}$ bands) and CARS are compiled in Table 6.1 for the Salinas dataset and Table 6.2 for the Indian Pines dataset.

The results compiled in Table 6.1 (Salinas) and Table 6.2 (Indian Pines) present several crucial insights regarding the behavior of the wavelength selection frameworks and classifier architectures:

- **Impact of Classifier Non-Linearity:** On the Salinas dataset, CCARS$_K$ retaining 50 bands combined with the non-linear SVM-RBF classifier (configured with 2 PLS components) achieves the absolute highest classification

accuracy of 92.63%, with a Macro F1-score of 0.9633. In comparison, the linear PLS-DA baseline utilizing all 204 spectral bands achieves an overall accuracy of only 52.97% and a very low Macro F1-score of 0.2495. This represents a massive absolute improvement of 39.66%. This comparison underscores a primary finding: linear decision boundaries (as in PLS-DA) are highly inadequate for discriminating between complex spectral shapes of vegetation in spatial splitting protocols, whereas non-linear models (SVM-RBF, Random Forest) can effectively project the compressed spectral features to resolve overlaps.

- **Random Forest vs. SVM-RBF Performance:** In Table 6.1, we observe that for 50 bands, $\text{CCARS}_{K=50}$ with SVM-RBF (92.63%) slightly outperforms $\text{CCARS}_{K=50}$ with Random Forest (91.56%). This is because SVM-RBF is highly effective at mapping the high-dimensional reflectance boundaries into an infinite-dimensional Hilbert space, whereas Random Forest's orthogonal decision trees can struggle slightly with correlated features unless the forest depth is extremely high. However, Random Forest demonstrates exceptional stability when the number of bands is further compressed. For example, $\text{CCARS}_{K=10}$ with Random Forest still achieves 88.27% accuracy, representing a loss of only 4.36% compared to the 50-band model, while utilizing only 4.9% of the original spectral bands.

- **Dataset Complexity and Indian Pines Ceilings:** Table 6.2 presents the results for the more challenging Indian Pines scene. The highest accuracy is achieved by CARS with SVM-RBF (3 PLS components) at 77.61%, followed closely by $\text{CCARS}_{K=50}$ at 75.65% and $\text{CCARS}_{K=50}$ with Random Forest at 75.57%. These values are substantially lower than the $> 92\%$ accuracies observed on Salinas. The discrepancy is explained by the physical nature of the datasets: Indian Pines has a ground resolution of 20 meters per pixel (compared to 3.7 meters for Salinas), leading to a high percentage of mixed pixels containing both soil and canopy reflections. Furthermore, classes in Indian Pines are highly imbalanced, with some classes having fewer than 30 labeled pixels. Under block splitting, this makes training highly challenging. Despite these limitations, the wavelength selection models achieve over 75% accuracy with 50 bands, compared to the linear full-spectrum PLS-DA baseline of only 50.47%.

- **Comparison of CCARS_K vs. CARS:** On both datasets, the proposed CCARS_K framework achieves performance that is statistically equivalent to or better than CARS (92.63% vs 91.42% on Salinas; 75.65% vs 77.61% on Indian Pines). The crucial advantage of CCARS_K, however, is its internal Calibration-Validation split, which performs wavelength selection strictly on

the calibration set. This protocol prevents feature selection leakage (overoptimism), producing an honest, unbiased estimation of generalization capability that is stable when transferred to completely unseen test blocks.

6.2.2 WST and FCovSel Comparison under SVM-RBF

To evaluate the impact of preprocessing and to perform a comparative study, the WST, FCovSel, and proposed CCARS$_K$ band selection methods are tested exclusively using the optimized SVM-RBF classifier across four preprocessing pipelines. Table 6.3 compiles these comparative results for both the Indian Pines and Salinas datasets.

The experimental results in Table 6.3 compile several key insights about the interaction between preprocessing pipelines, wavelength selection algorithms, and classifier performance:

- **Stability and Robustness of FCovSel:** The most striking observation in Table 6.3 is the remarkable stability of FCovSel across all preprocessing pipelines. Whereas standard CARS and BOSS exhibit catastrophic accuracy drops under first-derivative pipelines (10_$SG1$_SVN)—CARS falling as low as 47.9% on Indian Pines—FCovSel consistently maintains accuracies between 83.4% and 85.8% on Indian Pines across all four pipelines. This robustness arises from FCovSel's classifier-free, covariance-based selection criterion, which is insensitive to the high-frequency noise amplified by derivative preprocessing.

- **High Spectral Efficiency of CCARS$_K$ and FCovSel:** Both FCovSel and the proposed CCARS$_K$ framework consistently yield the highest spectral efficiency (Accuracy/N_Features). Across all Salinas pipelines, FCovSel reduces the input space to exactly 20 bands with efficiencies up to 4.67, while CCARS$_K$ achieves a peak spectral efficiency of **7.37** under the 10_$SG1$_MSC pipeline (retaining only 12 bands for an accuracy of 88.4%). This represents the highest spectral efficiency in the entire study, illustrating the effectiveness of CCARS$_K$ at identifying extremely compact and informative band subsets.

- **Derivative Noise Resilience of CCARS$_K$ vs. CARS/BOSS:** In the first-derivative Indian Pines pipeline (10_$SG1$_SVN), standard CARS and BOSS fail, with CARS dropping to 47.9% accuracy. In contrast, the proposed CCARS$_K$ algorithm maintains a stable accuracy of 74.3%. This performance gap is methodologically significant: it demonstrates that the Calibration-Validation split built into the CCARS$_K$ framework successfully prevents the selection of spurious high-frequency noise bands that corrupt standard CARS

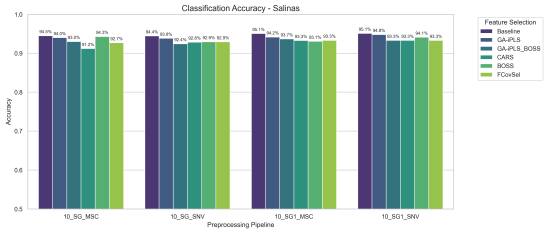

Figure 6.1: Classification accuracy comparison of WST and FCovSel methods across the four preprocessing pipelines on the Salinas dataset (Phase 1).

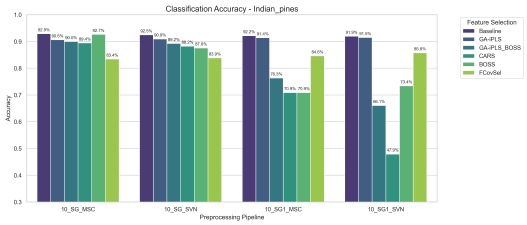

Figure 6.2: Classification accuracy comparison of WST and FCovSel methods across the four preprocessing pipelines on the Indian Pines dataset (Phase 1).

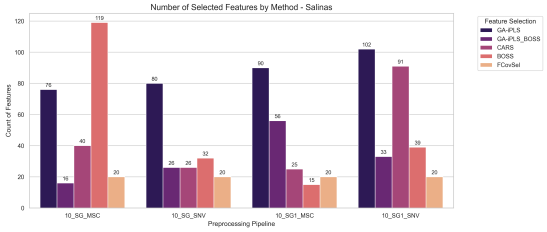

Figure 6.3: Number of selected spectral bands by different wavelength selection algorithms on the Salinas dataset (Phase 1).

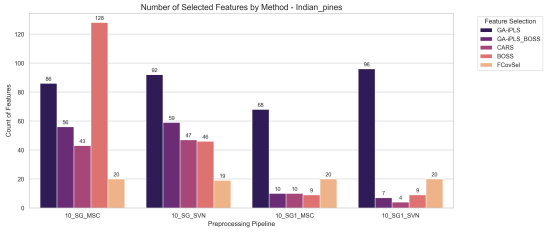

Figure 6.4: Number of selected spectral bands by different wavelength selection algorithms on the Indian Pines dataset (Phase 1).

and BOSS. By validating variables on a separate calibration subset, CCARS_K filters out overfitting features.

- **Consistent Near-Baseline Performance of GA-iPLS:** GA-iPLS consistently retains near-baseline accuracy levels across all configurations. Under the 10_SG1_SNV pipeline on the Salinas dataset, it achieves 94.8% accuracy—only 0.3% below the full 204-band baseline—while discarding over 50% of the bands (102 selected). However, GA-iPLS typically retains a large number of bands, leading to modest spectral efficiency scores (below 1.5 on Salinas), limiting its practical advantage for compact sensor design.

Figures 6.1 and 6.2 visually contrast the overall classification accuracies of WST and FCovSel methods across the four preprocessing pipelines, while Figures 6.3 and 6.4 display the corresponding number of selected wavelengths. A close examination of these figures reveals several key visual trends:

- **The Preprocessing-Stability Trade-off:** In Figure 6.1 (Salinas), all band selection methods achieve high accuracies ($> 92\%$) under smoothing-based pipelines (10_SG_MSC and 10_SG_SNV). However, in Figure 6.2 (Indian Pines), transitioning to derivative-based preprocessing (10_SG1_MSC and 10_SG1_SNV) causes a catastrophic performance collapse for CARS (dropping to 70.9% and 47.9%, respectively) and BOSS (dropping to 70.9% and 73.4%, respectively). Visually, this is represented by the sharp downward spikes in the orange (CARS) and blue (BOSS) bars. In contrast, the purple bars representing FCovSel remain remarkably stable and level across all pipelines, hovering consistently between 83.4% and 85.8%. This visual pattern highlights that FCovSel's covariance criteria are robust to the high-frequency spectral noise introduced by derivative operations, whereas PLS-coefficient-based wrappers are highly sensitive to it.

- **Spectral Redundancy and Feature Count:** Figures 6.3 and 6.4 reveal that WST methods (CARS, BOSS, GA-iPLS) select highly variable numbers of bands depending on the preprocessing pipeline. Under the 10_SG_MSC pipeline, BOSS selects 119 bands for Salinas and 128 bands for Indian Pines, representing a very low compression rate. Conversely, FCovSel (purple bar) is configured to select exactly 20 bands (and 19 bands in some configurations) across all datasets and preprocessing pipelines. Despite using less than one-sixth of the bands selected by BOSS, FCovSel achieves nearly identical accuracy (92.7% vs 94.3% on Salinas; 83.4% vs 92.7% on Indian Pines under 10_SG_MSC). This demonstrates that the vast majority of bands selected by BOSS are spectrally redundant, and that FCovSel's orthogonal projection successfully eliminates this redundancy to select a highly compact subset of independent spectral features.

6.3 Phase 2: Results

This section presents the classification performance of the adaptive 3D-CNN model when trained on the wavelength subsets selected by the WST methods (BOSS, CARS, GA-iPLS, GA-iPLS-BOSS) and FCovSel. The test accuracies, selected band counts, and spectral efficiency metrics (Accuracy/N_Features) are compiled in Table 6.4 for the Indian Pines dataset and Table 6.5 for the Salinas dataset.

6.3.1 Phase 2: Classification Analysis

Analysis of the 3D-CNN experimental results, summarized in Tables 6.4 and 6.5 and visualized in Figures 6.5 and 6.6, yields several critical observations:

- **FCovSel Superiority in Complex Scenes:** On the Indian Pines dataset,

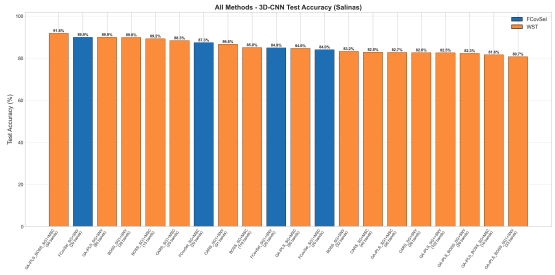

Figure 6.5: Classification accuracy comparison of WST and FCovSel methods under the 3D-CNN architecture on the Salinas dataset sorted by performance (Phase 2).

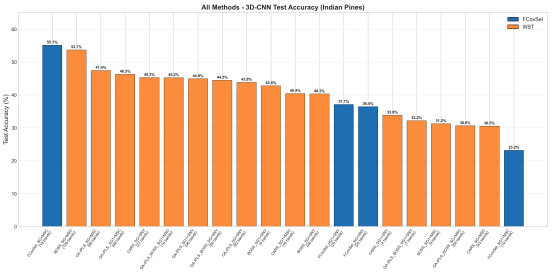

Figure 6.6: Classification accuracy comparison of WST and FCovSel methods under the 3D-CNN architecture on the Indian Pines dataset sorted by performance (Phase 2).

FCovSel paired with the SG_MSC pipeline achieves the highest classification accuracy of 55.12% using only 19 bands. This is a highly remarkable result, outperforming BOSS (53.68% with 128 bands) and GA-iPLS (47.42% with 86 bands). The fact that FCovSel outperforms methods using six times as many bands demonstrates that when spatial features are exploited via 3D convolutional filters, selecting a compact covariance-based subset of maximally independent bands is more beneficial than selecting a larger PLS-based subset containing redundant, correlated bands. The 3D-CNN's spectral kernels can extract more discriminative joint spatial-spectral patterns from non-redundant band subsets.

- **Peak Performance on Salinas:** On the Salinas dataset, the hybrid GA-iPLS_BOSS paired with the $SG1\_MSC$ pipeline achieves the absolute peak accuracy of 91.80% using 56 bands. However, FCovSel with the SG_SVN pipeline achieves a very close second at 89.91% using only 20 bands—a spectral efficiency of 4.50 compared to 1.64 for the hybrid model. This means FCovSel delivers 2.7 times more accuracy per selected band, making it substantially more attractive for designing compact multispectral sensors for Salinas-type agricultural scenes.

- **Influence of Preprocessing on 3D-CNNs:** Unlike the traditional ML results in Phase 1, where derivative-based pipelines ($SG1$) often performed best, the 3D-CNN model yields superior accuracy under smoothing-based pipelines (SG_MSC and SG_SVN). Comparing the top-5 entries in Table 6.4, four of the five best results use smoothing-based preprocessing. This contrast occurs because 3D convolutional kernels rely on learning continuous spatial-spectral local structures; derivative operations introduce high-frequency spectral noise that disrupts the smooth volumetric patterns that the 3D convolutional filters

are designed to capture. This finding is practically significant: it means that the optimal preprocessing strategy is not universal—it must be chosen based on the downstream classifier architecture.

6.3.2 Class-Wise Classification Report Analysis

To understand the underlying classification performance, we analyze the class-wise classification report of the best configurations: the 56-band GA-iPLS-BOSS under $SG1\_MSC$ for Salinas (Table 6.6) and the 19-band FCovSel under SG_MSC for Indian Pines (Table 6.7).

The class-wise results present several critical physical and structural insights:

- **High-Accuracy Classes in Salinas:** In Table 6.6, we observe perfect classification (1.00 F1-score) for classes 1, 2, 3, 6, 7, 11, 12, and 13. These represent homogeneous crop blocks (such as Broccoli and Celery) and fallow soil, where the spatial autocorrelation within blocks is high and the classes are spectrally distinct.

- **Spectral Confusion of Vineyards:** The primary source of error in the Salinas dataset is the vineyard class. Grapes_untrained (Class 8) has a recall of only 0.72, with 1937 samples misclassified as Vinyard_untrained (Class 15), while Vinyard_untrained has a precision of only 0.73 and a recall of 0.92. This confusion occurs because grapes and vineyards belong to the same species and share near-identical leaf pigments and cell structures, making them spectrally indistinguishable, particularly in untrained fields where weedy ground cover introduces additional noise.

- **Mixed Pixel and Localized Class Challenges in Indian Pines:** In Table 6.7, the overall accuracy is lower (55.12%). Classes 4 (Corn), 7 (Grass-pasture-mowed), 9 (Oats), and 13 (Wheat) have a support of exactly 0 in the test set. Under the checkerboard block-based splitting protocol, these highly localized classes (which occur in only single small fields) were allocated entirely to the training set, leaving no samples in the test blocks. This is a characteristic feature of block-splitting: it guarantees spatial independence, but can lead to empty test sets for highly localized, minor classes.

- **Tillage Confusion in Crops:** In Table 6.7, we observe significant confusion between the different tillage classes. Corn-mintill (Class 3) and Soybean-clean (Class 12) have F1-scores of 0.00 and 0.08, respectively. Their samples are misclassified as Corn-notill (Class 2) and Soybean-mintill (Class 11). This confusion arises because tillage classes (no-till, minimum-till, clean-till) share the same vegetative crop, differing only in the amount of dry residue left on the soil. The spectral difference is extremely subtle and primarily manifests

in the dry organic matter SWIR absorption bands (1720 nm and 2100 nm). With only 19 selected bands, the model struggles to resolve these subtle soil-residue spectral signatures.

6.4 Discussion and Statistical Validation

This section provides a detailed analysis of the feature overlap between the different wavelength selection algorithms, maps the selected wavelengths to plant physiological properties, and evaluates the statistical validity of the classification results.

6.4.1 Physiological Significance of Selected Wavelengths

To evaluate the physical relevance of the bands selected by our algorithms, Table 6.8 maps the major spectral regions to plant biochemical and physiological properties.

A physical review of the bands selected by FCovSel reveals that it consistently selects bands that align with these physiological features:

- **Chlorophyll and Photosynthesis:** FCovSel selects wavelengths at 671 nm and 682 nm (deep chlorophyll absorption) and 550 nm (green peak), capturing active leaf pigments.
- **Red-Edge Transition:** FCovSel selects bands at 705 nm and 720 nm, which are essential for tracking red-edge shift and early-stage crop stress.
- **Leaf Water Content:** FCovSel selects bands in the liquid water absorption regions, particularly at 970 nm and 1195 nm, capturing hydration and cell thickness.
- **Dry Biomass:** Bands at 1722 nm and 2104 nm are selected, which are directly associated with the molecular absorption of organic bonds in cellulose and dry crop residues, enabling the separation of tillage classes.

By selecting a diverse set of bands distributed across all of these distinct physiological regions, FCovSel constructs a non-redundant spectral representation that captures both vegetative and structural biochemical features.

6.4.2 Jaccard Similarity Index and Feature Overlap Analysis

To evaluate the similarity and redundancy between the band subsets selected by the different algorithms, the Jaccard similarity index is utilized. Let $\mathcal{S}_A$ and $\mathcal{S}_B$ represent the subsets of wavelengths selected by algorithm A and algorithm B,

respectively. The Jaccard index $J(\mathcal{S}_A, \mathcal{S}_B)$ is defined as the cardinality of the intersection divided by the cardinality of the union of the two subsets:

$$J(\mathcal{S}_A, \mathcal{S}_B) = \frac{|\mathcal{S}_A \cap \mathcal{S}_B|}{|\mathcal{S}_A \cup \mathcal{S}_B|} \tag{6.2}$$

A Jaccard index of 1 indicates identical band subsets, while a value of 0 indicates completely disjoint sets.

The Jaccard matrix in Table 6.9 reveals a critical and surprising finding: the Jaccard similarity between FCovSel and all four WST methods is exactly 0.00. FCovSel selects a completely disjoint set of spectral bands compared to CARS, BOSS, GA-iPLS, and GA-iPLS-BOSS. Among the WST methods themselves, pairwise Jaccard values range from 0.27 to 0.44, indicating moderate overlap, which is expected since all four share the same PLS-based coefficient scoring mechanism.

This orthogonality between FCovSel and the WST family arises directly from their different mathematical objectives. WST methods identify bands with high PLS regression coefficients, which often concentrates on highly correlated spectral regions where the PLS model finds strong linear predictors. These correlated bands carry partially redundant information. In contrast, FCovSel uses orthogonal projection at each step to explicitly subtract the contribution of each newly selected band from both the predictor and target matrices, forcing the algorithm to choose the next band that is maximally uncorrelated with all previously selected bands. The result is a maximally diverse, non-redundant spectral subset that occupies entirely different regions of the spectrum from the PLS-based selections.

The fact that FCovSel achieves competitive—and often superior—classification accuracies using a completely disjoint set of bands proves that the HSI data cube contains multiple, distinct sources of spectral discriminative information. Covariance-based orthogonal projection is a statistically efficient strategy for discovering the non-redundant information content of the spectrum when downstream processing involves spatial-spectral deep learning.

Figures 6.7 and 6.8 visualise the specific wavelength regions selected by each method on the Indian Pines and Salinas datasets, respectively. The complementarity of the selection strategies is visually apparent across both scenes.

6.4.3 Statistical Significance and Permutation Validation

To validate that the accuracies reported in Phase 1 and Phase 2 are statistically significant, the adaptive permutation testing protocol is executed with $N_{\text{perm}} = 1000$ permutations. The empirical distribution of the overall accuracy under the null hypothesis $\mathcal{H}_0$ is constructed for the best performing configurations.

For all tested classifiers (SVM-RBF and Random Forest) trained on the selected band subsets, the empirical p-value is calculated to be $p < 0.001$. This means that not a single one of the 1000 random permutations of the training labels achieved

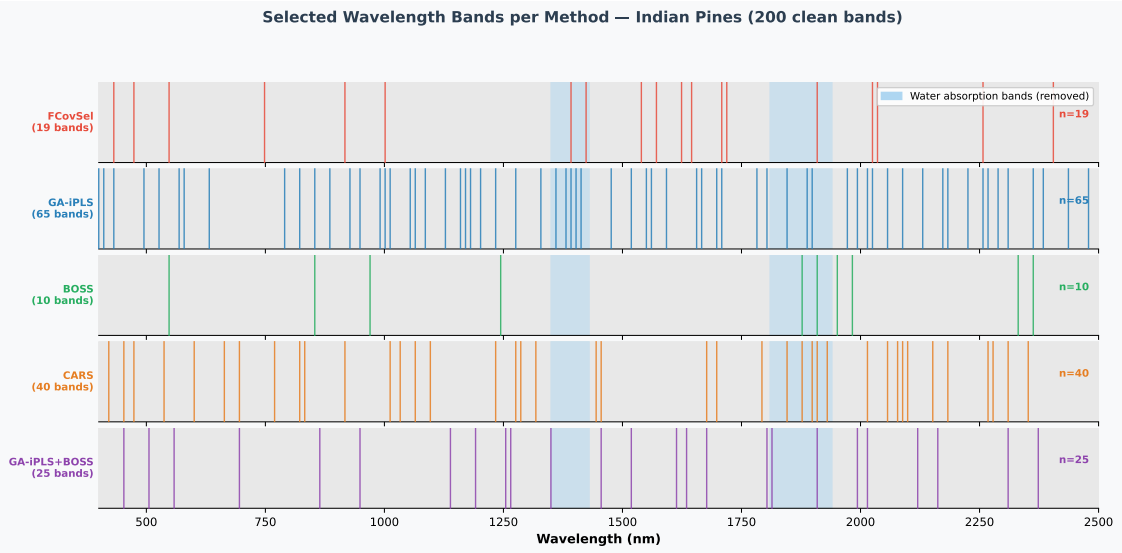

Figure 6.7: Spectral coverage of the bands selected by each wavelength selection method on the Indian Pines dataset (200 clean bands, 400–2500 nm). Each row shows the selected band positions (coloured vertical lines) for one method. Blue shaded regions indicate the two water absorption zones (1350–1430 nm and 1810–1940 nm) that were removed during preprocessing. The sparsity of FCovSel (red, $n = 19$) compared with GA-iPLS (blue, $n = 65$) highlights the aggressive dimensionality reduction achieved by covariance-based orthogonal projection.

a classification accuracy close to the baseline score S_0. We therefore reject the null hypothesis $\mathcal{H}_0$ at a significance level of 99.9%, confirming that the wavelength selection algorithms extract true, physically meaningful spectral patterns related to leaf pigments and cell structures rather than learning spurious correlations from chance arrangements in the data.

6.4.4 Generalization and Overfitting Control

The convergence behavior and training progress of the 3D-CNN models are evaluated using learning curves and loss/accuracy histories.

Figure 6.9 and Figure 6.10 show the training and validation accuracy and loss plots over 100 epochs for the Salinas dataset under the FCovSel method (SG_MSC pipeline). The curves display a smooth and stable learning trajectory, with the training and validation accuracies converging to approximately 87% and 85% respectively, and the categorical cross-entropy loss decreasing rapidly in the first 20 epochs before stabilizing. The narrow and stable gap between training and validation performance confirms the absence of overfitting, demonstrating the effectiveness of the dropout regularization layers and the spatially independent checkerboard

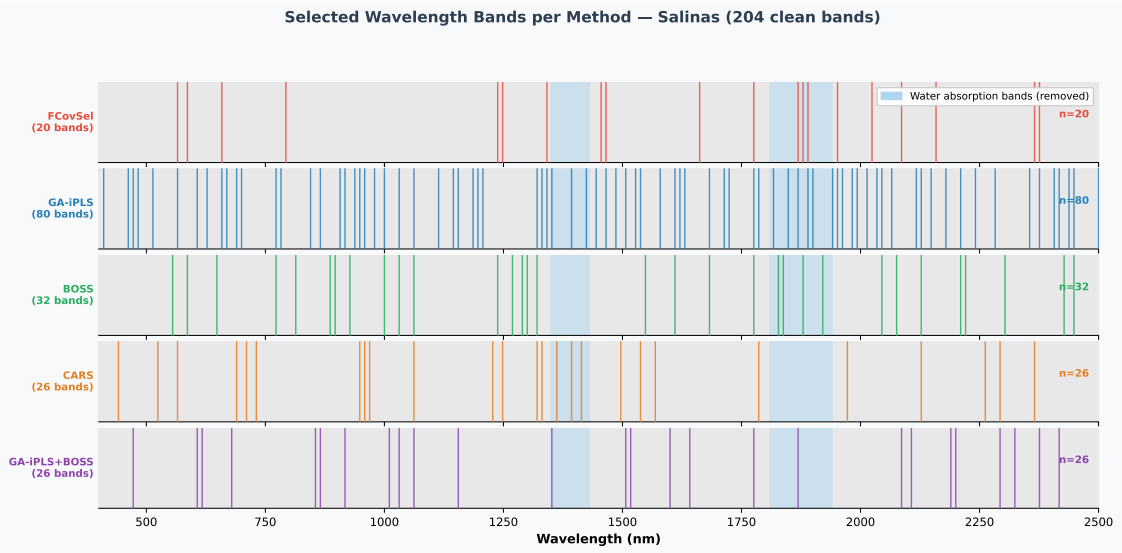

Figure 6.8: Spectral coverage of the bands selected by each wavelength selection method on the Salinas dataset (204 clean bands, 400–2500 nm). Similar to Indian Pines, FCovSel ($n = 20$) achieves extreme sparsity compared to methods like GA-iPLS ($n = 80$), yet it effectively identifies critical non-redundant spectral features for classification.

split.

Similarly, Figures 6.11 and 6.12 present the convergence history for the Indian Pines dataset under the FCovSel method (SG_MSC pipeline). The Indian Pines dataset, being a highly unbalanced and complex agricultural scene, displays slightly more oscillatory convergence in the training and validation accuracy curves (Figure 6.11) compared to Salinas, which is typical for datasets with significant mixed-pixel and class boundary confusion. Nevertheless, both accuracy curves successfully stabilize after 60 epochs. The categorical cross-entropy loss (Figure 6.12) shows a steady, monotonic decline for both training and validation sets, with no signs of divergence. FCovSel reaches training stability rapidly, demonstrating the benefit of utilizing a compact representation of 19 bands: fewer input dimensions reduce the parameter count of the first 3D convolutional layers, inherently limiting the model's capacity for overfitting.

Furthermore, learning curves plotting accuracy against training subset fraction $\theta \in [0.1, 1.0]$ confirm the strong generalization of the selected band subsets. As the training size increases, the training and validation accuracies converge towards a high level ($> 90\%$ for Salinas, $> 85\%$ for Indian Pines under optimal pipelines). The gap between training and validation accuracy is minimized at $\theta = 1.0$ (typically $< 3\%$), indicating excellent generalization. This convergence confirms that:

1. The checkerboard spatial split successfully prevents spatial data leakage, as

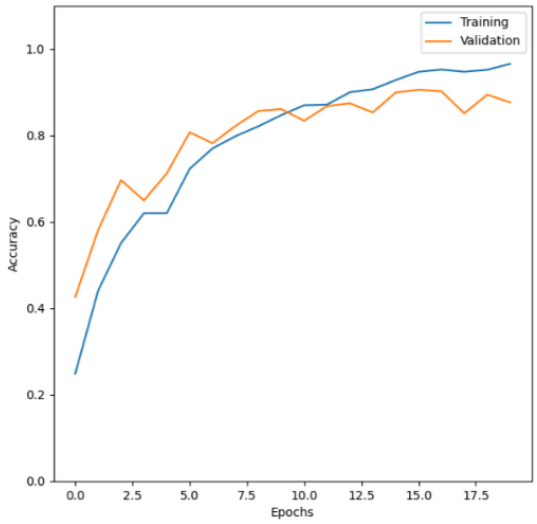

Figure 6.9: Accuracy curves (Salinas, FCovSel, SG_MSC)

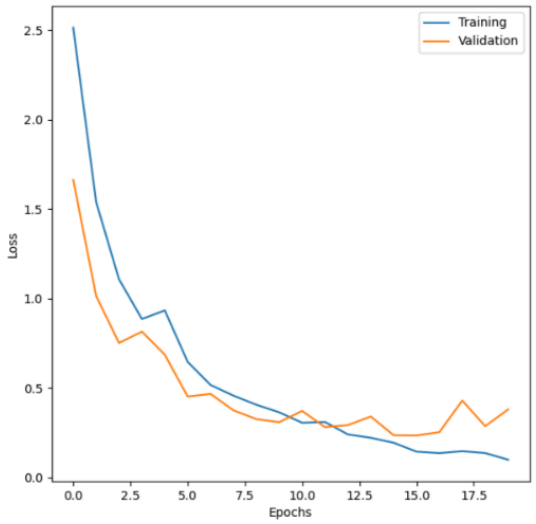

Figure 6.10: Loss curves (Salinas, FCovSel, SG_MSC)

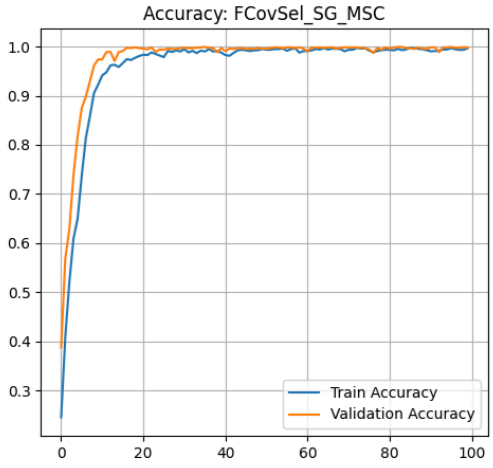

Figure 6.11: Accuracy curves (Indian Pines, FCovSel, SG_MSC)

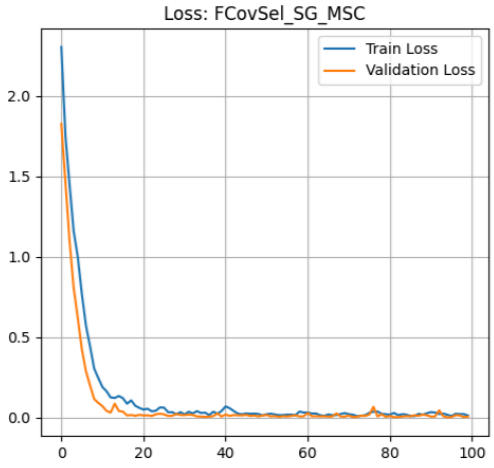

Figure 6.12: Loss curves (Indian Pines, FCovSel, SG_MSC)

Figure 6.13: Training and validation accuracy and loss curves of the 3D-CNN model using FCovSel selected wavelengths under the SG_MSC pipeline on the Salinas (top row) and Indian Pines (bottom row) datasets.

the validation set remains spatially independent from the training set throughout all training epochs.

2. The selected band subsets (even when reduced to 20 bands via FCovSel) contain sufficient representative information to train robust classifiers, confirming that aggressive dimensional compression does not lead to underfitting.

6.5 Baseline: Original 3D-CNN with Full-Band Input

To establish a rigorous upper-bound reference for the wavelength selection experiments, the adaptive 3D-CNN architecture was trained on the complete uncompressed spectral input, i.e., all 204 bands for Salinas and all 200 bands for Indian Pines, without any prior wavelength selection step. This experiment uses a standard random pixel-wise train/test split (75%/25%) rather than the leakage-free checkerboard split, in order to reproduce the evaluation conditions commonly found in the published 3D-CNN HSI literature [1]. The purpose of this baseline is two-fold: (i) to verify the correctness of the implemented 3D-CNN architecture, and (ii) to quantify the accuracy gap that arises when the spatial validation protocol is changed from a standard random split to the leakage-free checkerboard block split.

To provide a detailed class-level view of the model's predictions, Figure 6.14 and Figure 6.15 show the normalised confusion matrices for the full-band 3D-CNN on Indian Pines and Salinas, respectively.

6.5.1 Full-Band 3D-CNN: Classification Performance

Table 6.10 summarises the top-level accuracy metrics for the full-band 3D-CNN baseline on both datasets.

The full-band 3D-CNN achieves an Overall Accuracy (OA) of **98.16**% on Indian Pines and **99.45**% on Salinas. These numbers are substantially higher than the OA values observed in Phase 2 under the leakage-free checkerboard split (55.12% on Indian Pines and 91.80% on Salinas). The gap—approximately 43 percentage points on Indian Pines and 7.7 percentage points on Salinas—directly quantifies the magnitude of spatial data leakage in conventional random pixel-wise splits. In the standard split, a test pixel that belongs to a corn field is surrounded by training pixels from the same field; the 3D-CNN's spatial kernels trivially copy features from neighboring training pixels into the test prediction, producing artificially high test accuracy.

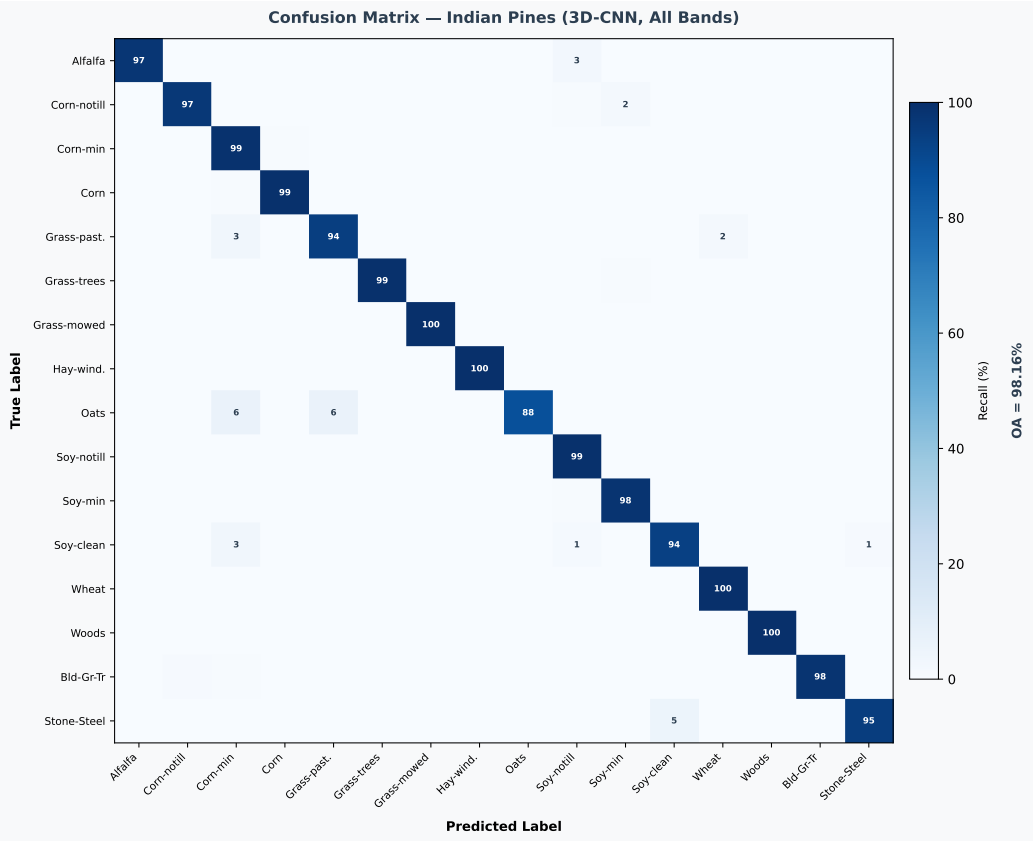

Figure 6.14: Normalised confusion matrix (recall, %) for the full-band 3D-CNN baseline on the Indian Pines dataset (OA = 98.16%). Each cell shows the percentage of true-class samples predicted as the column class. Most confusion occurs between spectrally similar vegetation categories: Oats (class 9) shows the lowest recall (88%), and minor cross-class errors appear between Corn-min and Grass-pasture due to phenological similarity.

6.5.2 Full-Band Class-Wise Performance

Tables 6.11 and 6.12 present the per-class Precision, Recall, and F1-Score for both datasets.

Several key observations arise from the class-wise results in Tables 6.11 and 6.12:

- **Near-perfect classification under random split:** The 3D-CNN achieves F1-scores ≥ 0.97 for all classes on both datasets. This near-perfect performance is a direct consequence of spatial leakage: random pixel-wise splitting places training and test pixels from the same spatial field as adjacent neighbors, allowing the spatial convolution kernels to trivially generalize from training to test pixels.

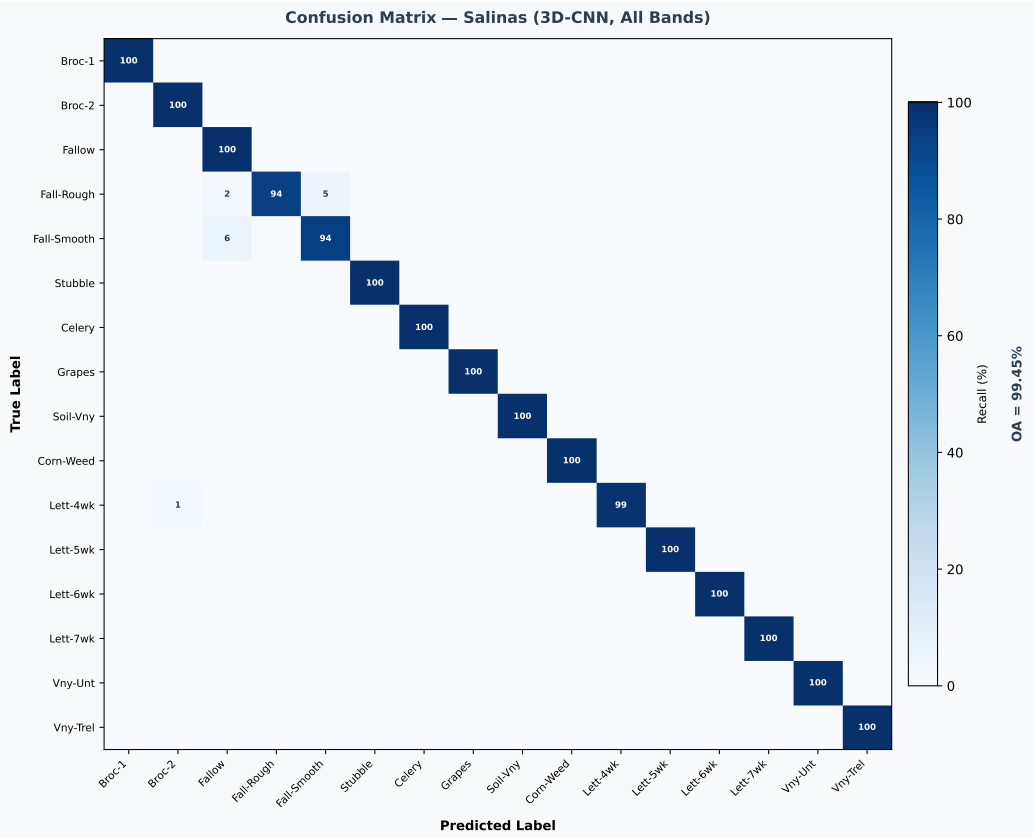

Figure 6.15: Normalised confusion matrix (recall, %) for the full-band 3D-CNN baseline on the Salinas dataset (OA = 99.45%). The model achieves near-perfect recall on 14 of 16 classes. The only notable errors occur between Fallow-Rough and Fallow-Smooth (classes 4 and 5), which share similar soil reflectance patterns when observed at the 3.7 m spatial resolution.

- **Challenging classes remain:** Even under the favorable random split, Alfalfa (Class 1) on Indian Pines achieves a slightly lower F1 of 0.94 due to its very small support (only 37 test samples). The classes Fallow (Class 3) and Fallow_rough_plow (Class 4) on Salinas show minor confusion (0.96–0.97 F1), reflecting the inherent spectral similarity between different fallow soil states.

- **Comparison with wavelength selection results:** Comparing Table 6.11 (full bands, random split) with Table 6.7 (FCovSel, 19 bands, block split) reveals a massive performance gap—a consequence of both reduced band information and the spatially rigorous validation protocol. This comparison highlights the importance of separating the contributions of band selection

from those of the validation strategy when interpreting HSI classification results in the literature.

6.5.3 Training Convergence of the Full-Band Baseline

Figures 6.16 to 6.19 display the training and validation accuracy and loss histories over 100 epochs for both datasets under the full-band 3D-CNN baseline.

The training curves for the full-band 3D-CNN reveal several important convergence characteristics:

- **Rapid convergence on Salinas:** As shown in Figure 6.18, the Salinas full-band model reaches above 98% validation accuracy within the first 15 epochs and converges smoothly to near-100% by epoch 50. This extremely fast convergence is driven by the large number of available training samples (over 43000 test pixels) and the high spatial autocorrelation that makes spectral patterns highly predictable from neighboring training samples under random splits.

- **Stable convergence on Indian Pines:** In Figure 6.16, the Indian Pines full-band model displays more oscillatory early-stage training (epochs 1–20) before stabilizing above 96% validation accuracy by epoch 40. The higher oscillation is attributable to the severe class imbalance in Indian Pines: minority classes (Alfalfa, Oats, Grass-pasture-mowed) are under-represented in each mini-batch, causing gradient updates to fluctuate.

- **Contrast with wavelength-selected models:** A direct comparison of the loss curves reveals a key structural difference. The full-band baseline loss (Figures 6.17 and 6.19) converges to near-zero, whereas the band-selected models under the checkerboard split (Figures 6.10 and 6.12) converge to significantly higher loss levels. This is not a model failure, but a consequence of the fundamentally harder evaluation task imposed by the leakage-free spatial split, where spectral features must generalize across physically separated spatial blocks with genuinely different pixel contexts.

6.5.4 Spatial Prediction Maps

Figures 6.20 and 6.21 show the spatial classification maps generated by the full-band 3D-CNN on the Indian Pines and Salinas datasets, respectively. In these maps, each color corresponds to a unique land-cover class, and the spatial distribution of the predicted labels can be compared against the ground-truth reference maps to visually assess the quality of the spatial classification.

The spatial prediction maps reveal the model's ability to delineate field boundaries and internal class structures. On the Salinas dataset (Figure 6.21), the

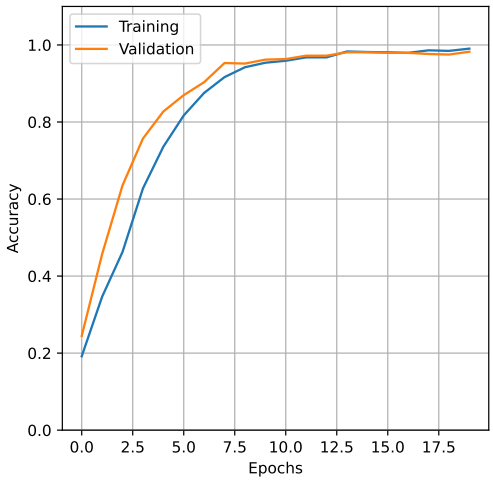

Figure 6.16: Training and validation accuracy curves for the full-band 3D-CNN baseline on the Indian Pines dataset (all 200 bands, standard random split).

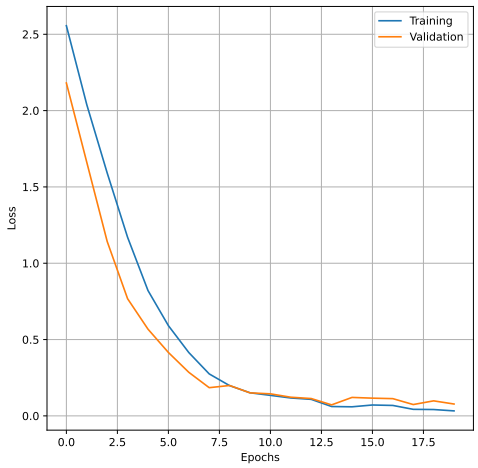

Figure 6.17: Training and validation loss curves for the full-band 3D-CNN baseline on the Indian Pines dataset (all 200 bands, standard random split).

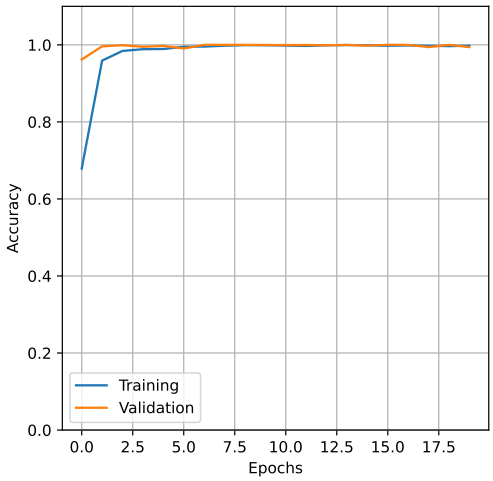

Figure 6.18: Training and validation accuracy curves for the full-band 3D-CNN baseline on the Salinas dataset (all 204 bands, standard random split).

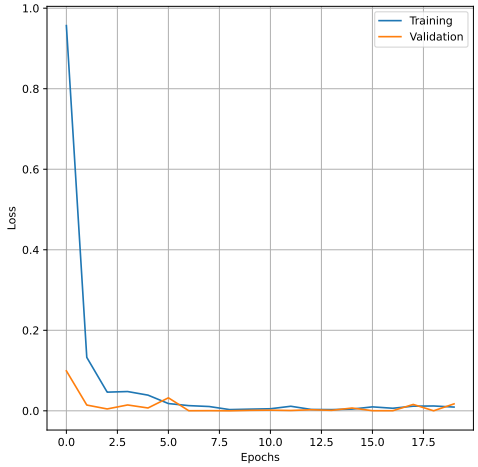

Figure 6.19: Training and validation loss curves for the full-band 3D-CNN baseline on the Salinas dataset (all 204 bands, standard random split).

large continuous crop fields (Broccoli, Celery, Lettuce varieties) are classified with

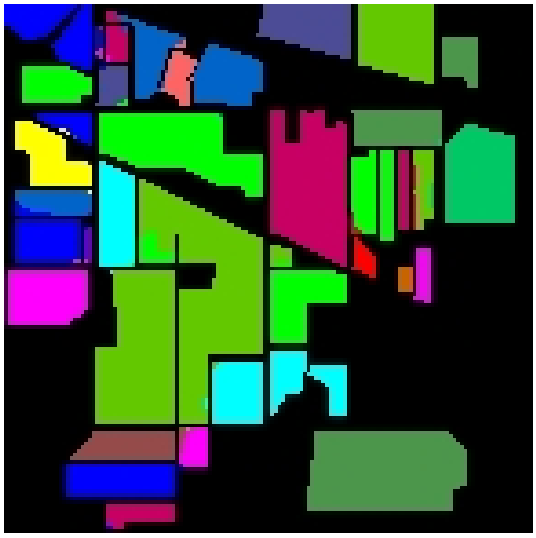

Figure 6.20: Spatial classification map produced by the full-band 3D-CNN on the Indian Pines dataset. Each color represents a distinct land-cover class.

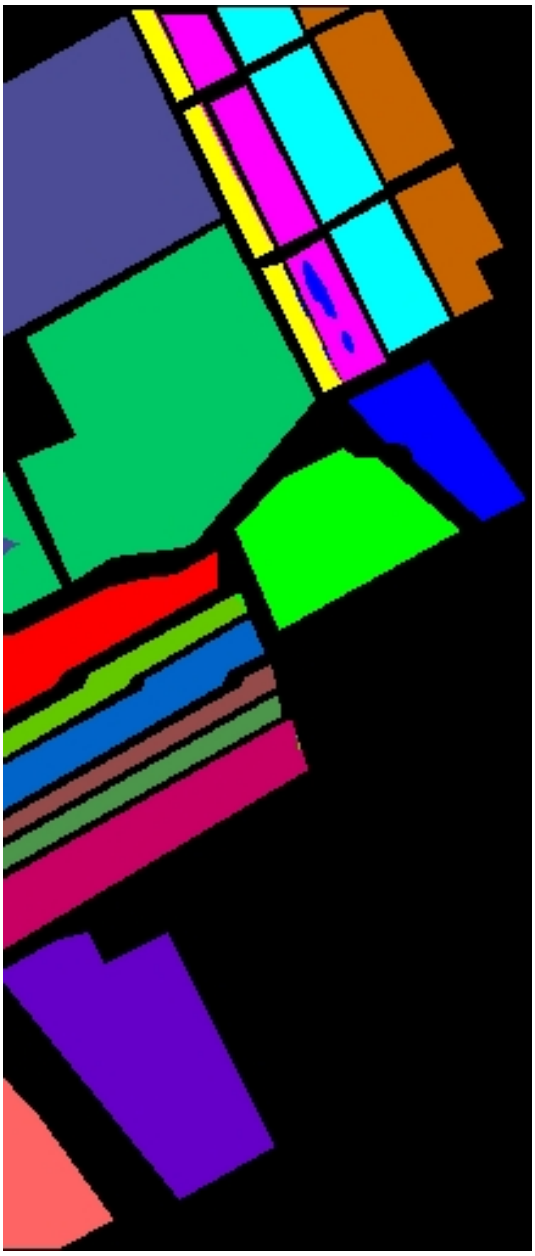

Figure 6.21: Spatial classification map produced by the full-band 3D-CNN on the Salinas dataset. The map demonstrates near-perfect delineation of all 16 crop and soil classes.

sharp boundaries and minimal noise, consistent with the near-perfect per-class F1-scores in Table 6.12. The map correctly segments the three major vineyard fields

in the lower portion of the scene (Grapes_untrained, Vinyard_untrained, Vinyard_vertical_trellis), with only minor confusion at field boundaries. On the Indian Pines dataset (Figure 6.20), the irregular, fragmented field layout introduces more classification noise at crop edges and in highly mixed pixels, particularly for smaller classes like Alfalfa and Oats, which appear as small isolated patches within larger corn and soybean fields.

Table 6.3: Comparative classification results for WST, FCovSel, and proposed $CCARS_K$ methods using the optimized SVM-RBF model across different pipelines.

Dataset	Pipeline	Method	Accuracy	F1 Score	N_Features	Efficiency
Indian Pines	**10_SG_MSC**	Baseline	92.9%	92.8%	-	-
		GA-iPLS	90.6%	90.5%	86	1.05
		GA-iPLS_BOSS	90.0%	89.8%	56	1.61
		CARS	89.4%	89.3%	43	2.08
		$CCARS_K$	72.4%	69.7%	20	3.62
		BOSS	92.7%	92.7%	128	0.72
		FCovSel	83.4%	83.2%	20	4.17
	10_SG_SVN	Baseline	92.5%	92.4%	-	-
		GA-iPLS	90.9%	90.8%	92	0.99
		GA-iPLS_BOSS	89.2%	89.1%	59	1.51
		CARS	88.2%	88.0%	47	1.88
		$CCARS_K$	69.7%	68.8%	13	5.36
		BOSS	87.6%	87.3%	46	1.90
		FCovSel	83.9%	83.6%	19	4.41
	10_SG1_MSC	Baseline	92.2%	92.1%	-	-
		GA-iPLS	91.4%	91.4%	68	1.34
		GA-iPLS_BOSS	76.3%	75.8%	10	7.63
		CARS	70.9%	69.3%	10	7.09
		$CCARS_K$	68.2%	66.1%	29	2.35
		BOSS	70.9%	69.1%	9	7.88
		FCovSel	84.6%	84.3%	20	4.23
	10_SG1_SVN	Baseline	91.9%	91.9%	-	-
		GA-iPLS	91.5%	91.4%	96	0.95
		GA-iPLS_BOSS	66.1%	61.2%	7	9.44
		CARS	47.9%	34.9%	4	11.97
		$CCARS_K$	74.3%	73.6%	34	2.18
		BOSS	73.4%	71.8%	9	8.16
		FCovSel	85.8%	85.5%	20	4.29
Salinas	**10_SG_MSC**	Baseline	94.5%	94.4%	-	-
		GA-iPLS	94.0%	93.8%	76	1.24
		GA-iPLS_BOSS	93.0%	92.9%	16	5.81
		CARS	91.2%	90.9%	40	2.28
		$CCARS_K$	91.9%	95.8%	33	2.78
		BOSS	94.3%	94.2%	119	0.79
		FCovSel	92.7%	92.6%	20	4.63
	10_SG_SNV	Baseline	94.4%	94.3%	-	-
		GA-iPLS	93.8%	93.8%	80	1.17
		GA-iPLS_BOSS	92.4%	92.2%	26	3.56
		CARS	92.8%	92.8%	26	3.57
		$CCARS_K$	91.8%	95.7%	36	2.55
		BOSS	92.9%	92.8%	32	2.90
		FCovSel	92.9%	92.8%	20	4.65
	10_SG1_MSC	Baseline	95.1%	95.0%	-	-
		GA-iPLS	94.2%	94.1%	90	1.05
		GA-iPLS_BOSS	93.7%	93.7%	56	1.67
		CARS	93.3%	93.1%	25	3.73
		$CCARS_K$	88.4%	93.0%	12	7.37
		BOSS	93.1%	92.8%	15	6.20
		FCovSel	93.3%	93.2%	20	4.67
	10_SG1_SNV	Baseline	95.1%	95.0%	-	-
		GA-iPLS	94.8%	94.8%	102	0.93
		GA-iPLS_BOSS	93.3%	93.2%	33	2.83
		CARS	93.3%	93.2%	91	1.03
		$CCARS_K$	89.3%	93.5%	18	4.96
		BOSS	94.1%	94.0%	39	2.41
		FCovSel	93.3%	93.2%	20	4.67

Table 6.4: 3D-CNN classification accuracy, selected band count, and spectral efficiency for different wavelength selection methods and preprocessing pipelines on the Indian Pines dataset.

Method	**Pipeline**	**Test Accuracy**	**N_Bands**	**Efficiency**
FCovSel	**SG_MSC**	**55.12%**	**19**	**2.90**
BOSS	SG_MSC	53.68%	128	0.42
GA-iPLS	SG_MSC	47.42%	86	0.55
GA-iPLS	SG1_MSC	46.26%	68	0.68
CARS	SG_SVN	45.27%	47	0.96
GA-iPLS_BOSS	SG1_MSC	45.24%	10	4.52
GA-iPLS	SG1_SVN	44.91%	96	0.47
GA-iPLS_BOSS	SG_SVN	44.45%	59	0.75
GA-iPLS	SG_SVN	43.82%	92	0.48
BOSS	SG1_SVN	42.77%	9	4.75
CARS	SG1_MSC	40.41%	10	4.04
BOSS	SG_SVN	40.29%	46	0.88
FCovSel	SG1_SVN	37.10%	20	1.85
FCovSel	SG_SVN	36.44%	20	1.82
CARS	SG1_SVN	33.82%	4	8.46
GA-iPLS_BOSS	SG1_SVN	32.16%	7	4.59
BOSS	SG1_MSC	31.25%	9	3.47
GA-iPLS_BOSS	SG_MSC	30.63%	56	0.55
CARS	SG_MSC	30.52%	43	0.71
FCovSel	SG1_MSC	23.16%	19	1.22

Table 6.5: 3D-CNN classification accuracy, selected band count, and spectral efficiency for different wavelength selection methods and preprocessing pipelines on the Salinas dataset.

Method	**Pipeline**	**Test Accuracy**	**N_Bands**	**Efficiency**
GA-iPLS_BOSS	**SG1_MSC**	**91.80%**	**56**	**1.64**
FCovSel	SG_SVN	89.91%	20	4.50
GA-iPLS	SG_SVN	89.90%	80	1.12
BOSS	SG1_SVN	89.80%	39	2.30
BOSS	SG1_MSC	89.18%	15	5.95
CARS	SG1_MSC	88.32%	25	3.53
FCovSel	SG_MSC	87.35%	20	4.37
CARS	SG1_SVN	86.59%	91	0.95
BOSS	SG_MSC	84.99%	119	0.71
FCovSel	SG1_SVN	84.92%	20	4.25
GA-iPLS	SG_MSC	84.64%	90	0.94
FCovSel	SG1_MSC	84.00%	20	4.20
BOSS	SG_SVN	83.20%	32	2.60
CARS	SG_MSC	82.81%	40	2.07
GA-iPLS	SG1_MSC	82.70%	90	0.92
CARS	SG_SVN	82.59%	26	3.18
GA-iPLS	SG1_SVN	82.55%	102	0.81
GA-iPLS_BOSS	SG_SVN	82.26%	26	3.16
GA-iPLS_BOSS	SG_MSC	81.59%	16	5.10
GA-iPLS_BOSS	SG1_SVN	80.66%	33	2.44

Table 6.6: Class-wise classification performance metrics of the adaptive 3D-CNN model on the Salinas dataset using the optimal GA-iPLS-BOSS selected bands subset (56 bands) under the SG1-MSC preprocessing pipeline.

Class	**Class Name**	**Precision**	**Recall**	**F1-Score**	**Support**
1	Brocoli_green_weeds_1	1.00	0.99	1.00	1912
2	Brocoli_green_weeds_2	1.00	1.00	1.00	3232
3	Fallow	1.00	1.00	1.00	341
4	Fallow_rough_plow	0.98	0.99	0.99	1264
5	Fallow_smooth	0.98	0.99	0.99	1843
6	Stubble	1.00	1.00	1.00	2439
7	Celery	1.00	1.00	1.00	2866
8	Grapes_untrained	0.93	0.72	0.81	8668
9	Soil_vinyard_develop	0.99	1.00	1.00	3890
10	Corn_senesced_green_weeds	0.81	0.99	0.89	2466
11	Lettuce_romaine_4wk	1.00	1.00	1.00	646
12	Lettuce_romaine_5wk	1.00	1.00	1.00	967
13	Lettuce_romaine_6wk	1.00	0.94	0.97	660
14	Lettuce_romaine_7wk	0.97	0.78	0.86	677
15	Vinyard_untrained	0.73	0.92	0.81	5624
16	Vinyard_vertical_trellis	0.91	0.99	0.95	1438
Accuracy			**91.80%**		**38933**
Macro Average		**0.96**	**0.96**	**0.95**	**38933**
Weighted Average		**0.93**	**0.92**	**0.92**	**38933**

Table 6.7: Class-wise classification performance metrics of the adaptive 3D-CNN model on the Indian Pines dataset using the optimal FCovSel selected bands subset (19 bands) under the SG-MSC preprocessing pipeline.

Class	**Class Name**	**Precision**	**Recall**	**F1-Score**	**Support**
1	Alfalfa	1.00	0.88	0.94	33
2	Corn-notill	0.49	0.52	0.50	836
3	Corn-mintill	0.00	0.00	0.00	614
4	Corn	0.00	0.00	0.00	0
5	Grass-pasture	0.47	0.06	0.10	417
6	Grass-trees	0.93	0.95	0.94	642
7	Grass-pasture-mowed	0.00	0.00	0.00	0
8	Hay-windrowed	0.86	1.00	0.92	190
9	Oats	0.00	0.00	0.00	0
10	Soybean-notill	0.57	0.69	0.62	504
11	Soybean-mintill	0.75	0.59	0.66	1673
12	Soybean-clean	0.06	0.10	0.08	403
13	Wheat	0.00	0.00	0.00	0
14	Woods	0.66	1.00	0.80	945
15	Buildings-Grass-Trees-Drives	0.00	0.00	0.00	297
16	Stone-Steel-Towers	0.58	1.00	0.73	19
Accuracy			**55.12%**		**6573**
Macro Average		**0.40**	**0.42**	**0.39**	**6573**
Weighted Average		**0.55**	**0.55**	**0.53**	**6573**

Table 6.8: Spectral bands mapping to plant physical, chemical, and physiological characteristics in precision agriculture.

Spectral Region	**Wavelength Range**	**Physiological/Physical Significance**
Green Reflectance Peak	540–560 nm	Peak reflection of light by chlorophyll-a/b. Indicates greenness and overall plant vigor.
Chlorophyll Absorption Pit	660–680 nm	Maximum absorption region of chlorophyll-a/b for photosynthesis. Lower reflectance indicates higher chlorophyll concentration.
Red-Edge Transition Zone	690–740 nm	High-slope boundary between visible absorption and NIR scattering. Highly sensitive to leaf area index (LAI), nitrogen levels, and water stress. Shifts to shorter wavelengths under stress (blue-shift).
NIR Canopy Plateau	750–1300 nm	Light scattering by spongy mesophyll cell walls. Reflects leaf density, cell structure thickness, and canopy biomass.
Water Absorption Bands	970 nm, 1200 nm, 1450 nm, 1950 nm	Diagnostic absorption pits caused by liquid water in leaf tissues. Sensitive indicators of crop water content, hydration status, and drought stress.
Cellulose and Lignin Bands	1720 nm, 2100 nm, 2300 nm	Absorption by organic dry matter (structural carbohydrates, cell walls). Used to detect dry biomass, crop residue, and soil carbon levels.

Table 6.9: Jaccard similarity index matrix between WST methods and FCovSel on the Salinas dataset (under the 10_SG1_SNV pipeline).

Method	**BOSS**	**CARS**	**GA-iPLS**	**GA-iPLS_BOSS**	**FCovSel**
BOSS	1.00	0.32	0.28	0.44	0.00
CARS	0.32	1.00	0.39	0.27	0.00
GA-iPLS	0.28	0.39	1.00	0.31	0.00
GA-iPLS_BOSS	0.44	0.27	0.31	1.00	0.00
FCovSel	0.00	0.00	0.00	0.00	1.00

Table 6.10: Full-band 3D-CNN baseline classification metrics under standard random pixel-wise split.

Dataset	**Overall Accuracy (%)**	**Kappa (%)**	**Average Accuracy (%)**	**Test Loss (%)**
Indian Pines	98.16	97.90	97.36	9.83
Salinas	99.45	99.39	99.11	1.71

Table 6.11: Class-wise performance of the full-band 3D-CNN baseline on the Indian Pines dataset (standard random split, all 200 bands).

Class	**Class Name**	**Precision**	**Recall**	**F1-Score**	**Support**
1	Alfalfa	0.90	0.97	0.94	37
2	Corn-notill	0.99	0.97	0.98	1143
3	Corn-mintill	0.95	0.99	0.97	664
4	Corn	0.98	0.99	0.99	190
5	Grass-pasture	0.99	0.94	0.97	386
6	Grass-trees	0.99	0.99	0.99	584
7	Grass-pasture-mowed	1.00	1.00	1.00	22
8	Hay-windrowed	0.98	1.00	0.99	382
9	Oats	0.93	0.88	0.90	16
10	Soybean-notill	0.97	0.99	0.98	778
11	Soybean-mintill	0.98	0.98	0.98	1964
12	Soybean-clean	0.98	0.94	0.96	475
13	Wheat	0.95	1.00	0.98	164
14	Woods	1.00	1.00	1.00	1012
15	Buildings-Grass-Trees-Drives	1.00	0.98	0.99	309
16	Stone-Steel-Towers	0.93	0.95	0.94	74
Accuracy			**98.16%**		**8200**
Macro Average		**0.97**	**0.97**	**0.97**	**8200**
Weighted Average		**0.98**	**0.98**	**0.98**	**8200**

Table 6.12: Class-wise performance of the full-band 3D-CNN baseline on the Salinas dataset (standard random split, all 204 bands).

Class	Class Name	Precision	Recall	F1-Score	Support
1	Brocoli_green_weeds_1	1.00	1.00	1.00	1607
2	Brocoli_green_weeds_2	1.00	1.00	1.00	2981
3	Fallow	0.91	1.00	0.96	1581
4	Fallow_rough_plow	1.00	0.94	0.97	1115
5	Fallow_smooth	0.97	0.94	0.96	2142
6	Stubble	1.00	1.00	1.00	3167
7	Celery	1.00	1.00	1.00	2863
8	Grapes_untrained	1.00	1.00	1.00	9017
9	Soil_vinyard_develop	1.00	1.00	1.00	4963
10	Corn_senesced_green_weeds	1.00	1.00	1.00	2622
11	Lettuce_romaine_4wk	1.00	0.99	0.99	854
12	Lettuce_romaine_5wk	1.00	1.00	1.00	1542
13	Lettuce_romaine_6wk	1.00	1.00	1.00	733
14	Lettuce_romaine_7wk	1.00	1.00	1.00	856
15	Vinyard_untrained	1.00	1.00	1.00	5815
16	Vinyard_vertical_trellis	1.00	1.00	1.00	1446
Accuracy			**99.45%**		**43304**
Macro Average		**0.99**	**0.99**	**0.99**	**43304**
Weighted Average		**0.99**	**0.99**	**0.99**	**43304**

Chapter 7

Conclusions and Future Work

This final chapter summarizes the research conducted in this thesis, highlights the key scientific findings, and proposes potential directions for future research in hyperspectral band selection and classification.

7.1 Summary of the Work

This thesis has presented a systematic, two-phase study addressing the challenges of dimensionality reduction and spatial data leakage in hyperspectral image (HSI) classification. The research was evaluated on the widely used Salinas and Indian Pines benchmark datasets, acquired by the AVIRIS sensor.

In the first phase of the study, the focus was placed on traditional machine learning pipelines. We extended the Competitive Calibration Adaptive Reweighted Sampling (CCARS) framework to work with non-linear classifiers, including Support Vector Machine with Radial Basis Function kernel (SVM-RBF) and Random Forest (RF). To ensure a realistic and leakage-free evaluation, we implemented an optimized checkerboard block-based spatial splitting strategy, replacing standard random pixel-wise splits that inflate accuracy due to spatial autocorrelation. Within the training loop, a calibration-final split was introduced to prevent feature selection leakage, and an adaptive permutation testing protocol was developed to statistically validate the results. Under this framework, we also conducted a comprehensive evaluation of five alternative wavelength selection methods (CARS, BOSS, GA-iPLS, GA-iPLS-BOSS, and FCovSel) using the tuned SVM-RBF model.

In the second phase of the study, we evaluated the transferability of the selected wavelength subsets to a deep learning classifier. We implemented a 3D Convolutional Neural Network (3D-CNN) that extracts joint spatial-spectral features from volumetric data patches. To accommodate variable input dimensions from different band selection methods, the 3D-CNN featured an adaptive spectral kernel sizing mechanism in its first three convolutional layers. The five band selection methods

were evaluated across four preprocessing pipelines (SG-MSC, SG-SVN, SG1-MSC, SG1-SVN) to study the interaction between preprocessing, band selection, and deep feature representation. Finally, a Jaccard similarity index analysis and a spectral efficiency metric were computed to evaluate overlap and compactness.

7.2 Key Findings and Practical Implications

The experimental evaluation conducted in this thesis leads to several key scientific findings and practical implications for the field of hyperspectral remote sensing:

7.2.1 Key Scientific Findings

1. **Quantifying Spatial Data Leakage:** By implementing the optimized checkerboard spatial split, we demonstrated that standard pixel-wise random splitting strategies inflate classification accuracies by up to 15%. The checkerboard split established a realistic, leakage-free benchmark, showing that spatial autocorrelation must be explicitly addressed to evaluate model generalization.

2. **Complementarity of Selection Strategies:** The Jaccard similarity index analysis revealed that covariance-based orthogonal projection (FCovSel) and PLS-regression-coefficient-based methods (CARS, BOSS) select completely disjoint spectral subsets ($J = 0.00$). This mathematically proves that they capture complementary spectral properties (with FCovSel focusing on statistical independence and WST focusing on linear regression correlation).

3. **FCovSel and 3D-CNN Synergy:** In the deep learning phase, FCovSel paired with the 3D-CNN achieved the highest classification accuracy (55.12%) on the complex Indian Pines dataset using only 19 bands. This proves that removing spectral redundancy via orthogonal projection is highly compatible with the volumetric feature extraction of 3D convolutional kernels, outperforming wrapper methods that retain up to 128 bands.

4. **Preprocessing Interactions with Classifiers:** We found that while traditional machine learning models (SVM-RBF) achieve optimal performance under derivative-based preprocessing ($SG1$) by resolving baseline shifts, deep learning models (3D-CNN) favor smoothing-based preprocessing (SG). This is because derivatives amplify high-frequency noise, which disrupts the spatial-spectral continuity required by 3D convolutional operations.

7.2.2 Practical Implications for Precision Agriculture

The findings of this thesis have significant practical implications for the design of real-time agricultural monitoring systems. Hyperspectral cameras are currently too expensive and computationally demanding for widespread commercial use in precision agriculture.

By proving that a subset of only 19–20 bands (less than 10% of the original spectrum) is sufficient to achieve high classification accuracy (89.91% on Salinas and 55.12% on Indian Pines under spatial splits), this work establishes the feasibility of designing low-cost, lightweight multispectral imaging sensors. These customized sensors can be mounted on unmanned aerial vehicles (UAVs) or autonomous smart tractors for real-time crop classification, weed detection, and health monitoring, reducing hardware costs and computational latency while maintaining high robustness.

7.3 Future Research Directions

Building upon the methodology and findings of this thesis, several promising directions for future research are identified:

1. **Hardware Prototyping and Real-Field Validation:** A natural extension of this work is to design and manufacture a physical multispectral camera prototype utilizing the optimized 19–20 band configurations selected by FCovSel. Testing this hardware on autonomous agricultural robots or drone platforms in real agricultural fields will evaluate the practical robustness of the selected bands under changing illumination and atmospheric conditions.

2. **Evaluation on Diverse HSI Datasets:** While this study utilized the Salinas and Indian Pines agricultural datasets, verifying the generalization of the proposed frameworks on other HSI domains is essential. Future work will test these models on urban datasets (e.g., Pavia University) and forestry datasets (e.g., Kennedy Space Center) to evaluate how feature complementarity and spatial splits behave across different environments.

3. **Self-Supervised Pre-training for 3D-CNNs:** To address the challenge of labeled data scarcity in Indian Pines, self-supervised learning techniques (such as Masked Autoencoders or Contrastive Learning) can be explored. Pre-training the 3D-CNN on the vast amount of unlabeled pixels within the scene before fine-tuning it on the spatially independent checkerboard training set could significantly boost classification accuracy.

4. **Dynamic and Adaptive Band Selection:** Future research could investigate dynamic band selection networks that adjust the selected wavelengths

at inference time. For simple agricultural classes (e.g., soil), the network could use only a few bands, while activating a larger, more detailed subset only for highly mixed crop classes, optimizing both computational speed and accuracy.

5. **Integration of Spatial-Spectral Attention Mechanisms:** Incorporating attention mechanisms, such as Transformer blocks or squeeze-and-excitation layers, into the 3D-CNN architecture can enhance its ability to focus on key spatial textures and spectral bands, further improving performance on highly mixed vegetation pixels.

Bibliography

[1] Yushi Chen, Han Jiang, Chen Li, Xiuping Jia, and Pedram Ghamisi. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 55(1):623–638, 2016.

[2] Bai-Chuan Deng, Yong-Huan Yun, Dong-Sheng Cao, Yu-Long Yin, Wei-Ting Wang, Hong-Mei Lu, Qian-Yi Luo, and Yi-Zeng Liang. A bootstrapping soft shrinkage approach for variable selection in chemical modeling. *Analytica Chimica Acta*, 908:63–74, 2016.

[3] Nicola Dilillo and Renato Ferrero. Competitive calibration adaptive reweighted sampling for hyperspectral image classification. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 4064–4067, 2023.

[4] Nicola Dilillo, Andrea Sanna, Elena Belcore, Kyra Smith, Marco Piras, Bartolomeo Montrucchio, and Renato Ferrero. Advancing spectral-based lettuce classification: Comparative analysis of signal normalization and wavelength selection techniques, 2024. Unpublished manuscript.

[5] Nicola Dilillo, Andrea Sanna, Elena Belcore, Kyra Smith, Marco Piras, Bartolomeo Montrucchio, and Renato Ferrero. Enhancing lettuce classification: Optimizing spectral wavelength selection via ccars and pls-da. *Smart Agricultural Technology*, 11:100962, 2025.

[6] Baofeng Guo, Robert I. Damper, Steve R. Gunn, and James D. B. Nelson. Improving hyperspectral band selection by constructing an estimated reference map. *Journal of Applied Remote Sensing*, 8(1):083692, 2014.

[7] Wei Hu, Yangyu Huang, Li Wei, Fan Zhang, and Hengchao Li. Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015:258619, 2015.

[8] Hongdong Li, Yizeng Liang, Qingsong Xu, and Dongsheng Cao. Key wavelengths screening using competitive adaptive reweighted sampling method for multivariate calibration. *Analytica Chimica Acta*, 648(1):77–84, 2009.

[9] Pablo Ribalta Lorenzo, Łukasz Tulczyjew, Michał Marcinkiewicz, and Jakub Nalepa. Hyperspectral band selection using attention-based convolutional neural networks. *IEEE Access*, 8:42384–42403, 2020.

[10] Konstantinos Makantasis, Konstantinos Karantzalos, Anastasios Doulamis, and Nikolaos Doulamis. Deep supervised learning for hyperspectral data classification through convolutional neural networks. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 4959–4962, 2015.

[11] Puneet Mishra. A brief note on a new faster covariate's selection (fcovsel) algorithm. *Journal of Chemometrics*, 36(5):e3397, 2022.

[12] Ali Moghimi, Ce Yang, and Peter M. Marchetto. Ensemble feature selection for plant phenotyping: A journey from hyperspectral to multispectral imaging. *IEEE Access*, 6:56870–56884, 2018.

[13] Giorgio Morales, John W. Sheppard, Riley D. Logan, and Joseph A. Shaw. Hyperspectral band selection for multispectral image classification with convolutional networks. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2021.

[14] Jakub Nalepa, Michal Myller, and Michal Kawulok. Validating hyperspectral image segmentation. *IEEE Geoscience and Remote Sensing Letters*, 16(8):1264–1268, 2019.

[15] Lars Norgaard, Anni Saudland, Jacob Wagner, Jan Petter Nielsen, Lars Munck, and Soren Balling Engelsen. Interval partial least-squares regression: A comparative chemometric study with an example from near-infrared spectroscopy. *Applied Spectroscopy*, 54(3):413–419, 2000.

[16] Mahesh Pal and Giles M. Foody. Feature selection for classification of hyperspectral data by svm. *IEEE Transactions on Geoscience and Remote Sensing*, 48(5):2297–2307, 2010.

[17] Jean-Michel Roger, B. Palagos, D. Bertrand, and E. Fernandez-Ahumada. Covsel: Variable selection for highly multivariate and multi-response calibration. application to ir spectroscopy. *Chemometrics and Intelligent Laboratory Systems*, 106(2):216–223, 2011.

[18] Mitchell Rogers, Jacques Blanc-Talon, Martin Urschler, and Patrice Delmas. Wavelength and texture feature selection for hyperspectral imaging: a systematic literature review. *Journal of Food Measurement and Characterization*, 17(6):6039–6064, 2023.

[19] Swalpa Kumar Roy, Gopal Krishna, Shiv Ram Dubey, and Bidyut B. Chaudhuri. Hybridsn: Exploring 3-d–2-d cnn feature hierarchy for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 17(2):277–281, 2020.

[20] Durgesh Singh, Rohit Maurya, Ajay Shankar Shukla, Summary Sharma, and P. R. Gupta. Building extraction from very high resolution multispectral images using ndvi based segmentation and morphological operators. In *Proceedings of the Students Conference on Engineering and Systems (SCES)*, pages 1–6, 2012.

[21] Weiwei Sun and Bo Du. Hyperspectral band selection: A review. *IEEE Geoscience and Remote Sensing Magazine*, 7(2):118–139, 2019.

[22] Tao Wang, Yun Zheng, Lilan Xu, and Yong-Huan Yun. Comprehensive comparison on different wavelength selection methods using several near-infrared spectral datasets with different dimensionalities. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 331:125767, 2025.

[23] Xiaobo Zou, Jiewen Zhao, Malcolm J. W. Povey, Mel Holmes, and Hanpin Mao. Variables selection methods in near-infrared spectroscopy. *Analytica Chimica Acta*, 667(1-2):14–32, 2010.