# NEW DATA ANALYSIS AND DIMENSIONALITY REDUCTION METHODS FOR HYPERSPECTRAL IMAGERY

**Samson Damilola Fabiyi**

In the fulfilment of the requirement

for the degree of Doctor of Philosophy

Centre for excellence in Signal & Image Processing

Department of Electronic & Electrical Engineering

University of Strathclyde

Supervised by

**Dr Paul Murray**

**Dr Jaime Zabalza**

## Declaration and Copyright

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.



Signed:

Date:

## Acknowledgement

My special gratitude goes to my supervisor, Dr Paul Murray, for his support, advice, and encouragement. I would like to thank him for believing in me from the beginning and for the opportunity to be his PhD student. He has been an inspiring supervisor and mentor for me.

My thanks goes to my secondary supervisor, Dr Jaime Zabalza. His advice and guidance have been very helpful. I would like to thank him for his patience and for always showing interest in my work. I would also like to say a big thank you to Professor Jinchang Ren (my former secondary supervisor) for his support and advice. I would like to thank him for his effort and time spent on reviewing my work. His feedback have been very helpful. My thanks also goes to all the collaborators in Vietnam and United Kingdom for their advice.

I very much appreciate the financial support I received from the Faculty of Engineering and Department of Electronic and Electrical Engineering at the University of Strathclyde, Glasgow. I am using this opportunity to thank them for awarding me the scholarship that covered my tuition fees and monthly stipends.

Many thanks to my family, fellow church members, members of the Nigerian community at the University of Strathclyde, and my colleagues at school (especially those in room RC 349) for their support, patience, encouragement, and advice.

## Abstract

Hyperspectral data contains rich spectral information and so have become very useful in data classification. However, hyperspectral data contains several spectral bands (usually in hundreds) which bring about curse of dimensionality and limits its potential in classification applications. With a focus on addressing this problem, this thesis applies Linear Discriminant Analysis (LDA) for hyperspectral data dimensionality reduction and proposes novel extensions of LDA.

LDA is a supervised technique which can reduce the number of dimensions in data. One problem with LDA is that the number of features it can produce is limited to $c - 1$ where $c$ is the number of classes in the data. Also, LDA gives sub-optimal performance when applied on small training samples, which limits its use on hyperspectral data since such data does not always contain enough samples for training.

Firstly, this thesis applies LDA on spectral features extracted from hyperspectral data to reduce its dimensionality and combines the LDA outputs with spatial features from RGB images. Comparative performance analysis of LDA and PCA is also performed. Results show that LDA can perform better than PCA and that combining spectral features with spatial features from RGB images can improve performance of classification models. Secondly, Folded LDA (F-LDA), a novel extension of LDA, is proposed for hyperspectral data dimensionality reduction. F-LDA is based on a mathematical 'trick' (folding the pixels) which was inspired by previous work to extend PCA using a similar innovative step. Results show that F-LDA achieves higher accuracy than LDA and other state-of-the-art methods when applied on small training samples. When compared with LDA, F-LDA achieves reduction in computational

complexity and memory requirement, and can extract many more features. Finally, F-LDA is applied on optimal spectral features selected by Genetic Algorithms. Results show that a novel combination of GA and F-LDA can achieve further reduction in computational complexity and memory requirement in certain applications.

# Table of Contents

## List of Symbols

| | |
|---|---|
| I | Two dimensional image |
| K | Number of rows in a two dimensional image |
| L | Number of columns in a two dimensional image |
| B | Number of spectral bands in a hyperspectral data cube |
| $\mathbf{X}$ | Hyperspectral data matrix |
| s | Number of samples in a data |
| F | Number of features in a data |
| $\boldsymbol{x_n}$ | Spectral vector in a hyperspectral data at position n |
| N | Position of a spectral vector in a hyperspectral data |
| $c$ | Number of classes in a data |
| $c_j$ | Class $j$ in a data |
| $J$ | Position of a class among all the classes in a data |
| $N_j$ | Number of samples in a class j |
| $I$ | Position of a sample among other samples in a class |
| $x_{ij}$ | Position of a sample in a particular class $j$ |
| $\boldsymbol{m_j}$ | Mean of the samples in each class |
| $\boldsymbol{m}$ | Mean of all the samples in the data matrix $\boldsymbol{X}$ |
| $\boldsymbol{V_W}$ | Within-class variance in LDA |
| $\boldsymbol{V_B}$ | Between-class variance in LDA |
| $\epsilon$ | Belonging to (is a member of) |
| $\Re$ | Real numbers |
| $\boldsymbol{T}$ | Transformation matrix |

| | |
|---|---|
| $\lambda$ | Eigenvalues |
| $\boldsymbol{v}$ | Eigenvectors |
| $d$ | Number of columns selected in eigenvectors |
| $\boldsymbol{V_d}$ | Reduced eigenvectors after the selection of $d$ columns |
| $Y$ | Data in a lower dimensional space in LDA |
| $x_e$ | Each feature column in a data matrix |
| $e$ | Position of each feature column in a data matrix |
| $c_e$ | Mean of each feature column in a data matrix |
| $\bar{x}_e$ | Each centralised feature column in a data matrix |
| $\mathbf{W}$ | Covariance matrix of a data matrix |
| $A_j$ | Accuracy for each class in a data |
| $P$ | Precision |
| $R$ | Recall |
| $t_p$ | Number of true positives |
| $f_p$ | Number of false positives |
| $t_n$ | Number of true negatives |
| $f_n$ | Number of false negatives |
| $kc$ | Kappa coefficient |
| $P_o$ | Observed agreement |
| $P_e$ | Probability of chance agreement |
| $Y$ | Hyperspectral data cube |
| $w$ | Wavelength, belonging to a set of wavelengths ($\Lambda$) |

| | |
|---|---|
| $\Lambda$ | Set of wavelengths |
| $X$ | Pixel in a 2D image |
| $G$ | Number of groups in a converted matrix (folded vector) |
| $B$ | Number of bands in a converted matrix (folded vector) |
| $\boldsymbol{P_n}$ | Matrix formed from folding of a spectral vector, $\boldsymbol{x_n}$ |
| $p_n$ | Each element in the converted matrix, $\boldsymbol{P_n}$ |
| $\boldsymbol{M_j}$ | Mean of the folded samples in each class |
| $\boldsymbol{M}$ | Mean of all the folded samples in a data |
| $\boldsymbol{P_{ij}}$ | Position of a converted matrix among other converted matrices in class $c_j$ |
| $\boldsymbol{V_{PW}}$ | Within-class variance in F-LDA |
| $\boldsymbol{V_{PB}}$ | Between-class variance in F-LDA |
| $\boldsymbol{V_{Pd}}$ | Reduced eigenvectors in F-LDA |
| $\boldsymbol{T_P}$ | Transformation matrix in F-LDA |
| $\boldsymbol{Y_n}$ | Projected matrix of each sample in F-LDA |
| $\boldsymbol{M_{mJ}}$ | Folded version of $\boldsymbol{m_j}$ **(**overall mean of each class in LDA**)** |
| $Pc$ | Parameter, penalty of the RBF kernel |
| $G$ | Parameter, gamma of the RBF kernel |
| $r$ | Rank of the between-class variance matrix |
| $\boldsymbol{v_{Pd}}$ | Single projection vector in 2D LDA |
| $y_{Pd}$ | Projected sample in 2D LDA |

## List of Acronyms

| | |
|---|---|
| AA | Average Accuracy |
| AVIRIS | Airborne Visible/InfraRed Imaging Spectrometer |
| BPNN | Back Propagation Neural Network |
| CCD | Charged-Couple Device |
| CNN | Convolutional Neural Network |
| CONFIG | Configuration |
| DA | Discriminant Analysis |
| DRT | Dimensionality Reduction Technique |
| EVD | EigenValue Decomposition |
| F-LDA | Folded Linear Discriminant Analysis |
| FSB | Full Spectral Band |
| GA | Genetic Algorithm |
| GDA | Generalized Discriminant Analysis |
| His | Hyperspectral Imaging |
| ICA | Independent Components Analysis |
| IRRI | International Rice Research Institute |
| KNN | k-Nearest Neighbours |
| KPCA | Kernel Principal Component Analysis |
| LDA | Linear Discriminant Analysis |
| LR | Logistic Regression |
| LS | Least Squares |
| NASA | National Aeronautics and Space Administration |

| | |
|---|---|
| NCPNVGGP | National Center of Protection of New Varieties and Goods of Plants |
| NIR | Near Infrared |
| NWFE | Nonparametric Weighted Feature Extraction |
| OA | Overall Accuracy |
| OCT | Optical Coherence Tomography |
| PCA | Principal Component Analysis |
| PC | Principal Component |
| PLS | Partial Least Squares |
| PNN | Probabilistic Neural Network |
| RBF | Radial Basis Function |
| RF | Random Forest |
| RGB | Red, Green, and Blue |
| RNN | Residual Neural Network |
| ROSIS | Reflective Optics System Imaging Spectrometer |
| SAR | Synthetic Aperture Radar |
| SSA | Singular Spectral Analysis |
| SSS | Small Sample Size |
| SVD | Singular Value Decomposition |
| SVM | Support Vector Machines |
| VIS | Visible |
| VNIR | Visible and Near-infrared |
| XML | Extensible Markup Language |

## 1. Introduction

Hyperspectral imaging (HSI) systems have become very useful in data classification applications due to the rich spectral information which are present in hyperspectral images (produced by such systems) [1],[2],[3] and the major role that hyperspectral images play in precision agriculture [4], environment monitoring [5], national security [6], etc. Hyperspectral imaging systems offer simultaneous acquisition of spectral and appearance-based information [7],[8],[9]. Consequently, chemical and physical profiles of samples are captured in the acquired hyperspectral images and are utilised for enhanced classification [8],[10]. This is different from the conventional RGB imaging and spectroscopy which have been applied separately for classification [8],[9]. RGB imaging can only provide appearance-based information of samples in the acquired images and not their constituents [8]. Similarly, spectroscopy can only provide information on the molecular composition [8],[11] and no information on the appearance of samples is captured.

Hyperspectral imaging systems have produced very promising results in classification [12],[13],[14],[15],[8],[10]. However, hyperspectral imaging still faces some challenges which limit its potential for classification. Firstly, traditional classification models suffer from curse of dimensionality due to the presence of very high number of spectral features (usually in hundreds) in the acquired hyperspectral images, and this degrades performance of such models [16],[17],[18]. Hence, the development of innovative data dimensionality reduction techniques for redundant data removal while retaining important information becomes imperative for enhanced classification. The research community working on providing optimal classification systems still faces the challenges of getting the best dimensionality reduction tools to deal with the

problem of high number of spectral features, which are present in the acquired hyperspectral imaging data. Secondly, while RGB imaging captures reflectance in 3 channels (Red, Green and Blue), hyperspectral imaging captures reflectance in several spectral bands (usually in hundreds) with increased spectral resolution. However, hyperspectral imaging, when compared to RGB imaging, often provides a decreased pixel density due to spatial binning and to enhance the data (acquired) robustness [19],[20]. Hence, fidelity in the appearance-based features is reduced especially when discriminating small objects such as rice seeds which are studied as one application in this thesis.

This thesis is therefore aimed at conducting investigations and proposing solutions to address these challenges. Firstly, this thesis evaluates the effectiveness of combining spectral information from high spectral resolution hyperspectral images and appearance based information from high spatial resolution RGB images for hyperspectral data classification while also applying Linear Discriminant Analysis (LDA) for dimensionality reduction of the spectral data before combining it with the spatial data for enhanced classification. It is necessary to justify the selection of LDA for this study since it is a less commonly applied technique than Principal Component Analysis (PCA) for dimensionality reduction of spectral data [15],[21],[10],[13]. Performance of LDA is therefore compared with that of the commonly applied PCA. The results from the comparative study of LDA and PCA show that LDA can perform better than PCA when applied to reduce the dimensionality of hyperspectral data. Performance evaluation of the proposed approach is performed on hyperspectral data of rice seeds. The rice seed data is selected for this study due to the role that classification plays as an important step in rice seed screening exercises, the promising

results achieved in recent papers by hyperspectral imaging data in rice seed classification and the continued need for rice seed screening exercises to be automated and enhanced [15],[22],[23]. Experimental results show that the proposed combination of spectral features from the hyperspectral images and spatial features from the RGB images can improve performance of classification models.

Secondly, based on the superior performance achieved by LDA when applied on hyperspectral data, this thesis also proposes an extended and improved version of LDA, named Folded LDA (F-LDA). Performance of the proposed F-LDA is evaluated on five hyperspectral datasets from different sensors namely Hyperion sensor [5], ROSIS (Reflective Optics System Imaging Spectrometer) sensor [24], and AVIRIS (Airborne Visible/InfraRed Imaging Spectrometer) sensor [25]. These five datasets are selected for this study because their use will broaden the scope of the proposed technique since the selected datasets are widely used in related papers [24],[26],[27]. Results obtained demonstrate the ability of the proposed F-LDA to outperform the conventional LDA in terms of classification accuracy, computational complexity, and memory requirement.

Finally, this thesis proposes the hybridization of GA and F-LDA (i.e., GA+F-LDA) for dimensionality reduction of spectral data. This is based on the promising results achieved in other publications where a Genetic Algorithm (GA) was applied to select optimal features in the data prior to feature extraction using PCA [28],[29] and LDA [30],[31]. The proposed GA+F-LDA is achieved by applying F-LDA on reduced hyperspectral datasets (datasets containing optimal spectral features selected by the GA). Performance of the proposed approach (GA+F-LDA) is evaluated on hyperspectral datasets of rice seeds and sugar. The continued need for rice seed

screening exercises to be automated and enhanced [15],[22],[23] makes it a good application domain for F-LDA and the proposed GA+F-LDA where they would be deployed to enhance rice seed classification (an important step in rice seed screening exercises). The sugar dataset is selected to demonstrate the potential of the proposed approach in enhancing the classification of other Agri-tech products. Results obtained show that the proposed GA+F-LDA can further reduce the computational complexity and memory requirement at the different stages of F-LDA. These benefits, however, come with a slight reduction in classification performance (in terms of accuracy and $F_1$ score) when compared with those attained by the standard F-LDA.

### 1.1. Original Contribution

The novel contributions of this thesis are summarised as follows:

1) The effectiveness of an innovative framework for combining spatial and spectral features is evaluated. (Chapter 4).
2) Comparative study of LDA and PCA as dimensionality reduction techniques for hyperspectral data is conducted (Chapter 4).
3) Performance of the innovative framework in 1) above is carried out on a large, diverse dataset of 90 rice seed species and similarity assessment of species (classes) is recommended in rice seed classification (Chapter 4).
4) The large hyperspectral dataset of rice seeds used in this work is made publicly available to the community (Chapter 4).
5) F-LDA, an extended and improved version of LDA, is proposed for hyperspectral data dimensionality reduction (Chapter 5).
6) Hybridization of GA and F-LDA (GA + F-LDA) is proposed for dimensionality reduction of hyperspectral data (Chapter 6).

Detailed explanation of these contributions is provided below:

In Chapter 4, effectiveness of fusing spatial features from RGB images and spectral features for hyperspectral data classification is evaluated. It is necessary to reduce the dimensionality of the spectral features prior to fusing them with the spatial features to enhance the classifier's performance. Also, since PCA is a more commonly applied dimensionality reduction techniques for hyperspectral data, it is necessary to compare its performance with LDA in preliminary analysis and use the superior technique in the final analysis where the performance of the framework for fusing spatial and spectral features will be evaluated. Hence, LDA and PCA are applied separately on the spectral data to reduce its dimensionality. It is observed that LDA can outperform PCA when used to reduce dimensions of spectral data. LDA is therefore utilised to reduce the dimensionality of spectral features, the outputs of which are then combined with the spatial features. It is also observed that the combination of spatial and spectral features improves discrimination ability and classifier's performance.

State-of-the-art techniques (as will be shown in Chapter 3) which are related to the approach evaluated in Chapter 4 were mainly evaluated on datasets with small variety of species; usually, 5-6 with the exception of [32] (30 species) and [33] (754 species). It is therefore difficult to confirm if the varying results achieved in those papers were due to superiority of algorithms employed, effectiveness of feature descriptors used for model training, or, differences in the inter-class or intra-class variation of species used in each paper since the datasets used in those papers are not publicly available. Consequently, performance evaluation of the proposed approach is evaluated on a large, diverse hyperspectral dataset of 90 rice seed species with 96 seeds per species.

It is then observed that performance of classification models can be impacted by varying the number of species in the datasets. Hence, similarity assessment of species is recommended. The large dataset of rice seeds is made publicly available [34],[35] to the community to assist in the benchmarking of the framework for feature combination. Specifically, the rice seed dataset was made available at an online research repository named Zenodo [36] in January 2020 and has been downloaded 6,205 times as at 27th June 2022. However, the dataset is yet to be cited based on the information available at Zenodo [36].

In Chapter 5, an extended and improved version of LDA, named Folded-LDA (F-LDA) is proposed. The proposed F-LDA is based on a simple but effective mathematical 'trick' (folding the pixels) which was inspired by previous work where an extension of PCA was proposed [37]. The proposed F-LDA shares the concepts of 'folding the pixels' with the two dimensional (2D) LDA [38] but also improves it by treating resulting eigenvectors individually and unfolding the projected samples. These allow more discriminant features to be extracted by the proposed F-LDA. These differentiate the proposed F-LDA from the 2D LDA where the eigenvectors are combined into a single projection vector and consequently, projection samples are also not folded. Performance of the proposed technique is evaluated on five publicly available hyperspectral datasets. Experimental results show that the proposed F-LDA produces more informative features and achieves higher classification accuracy than the original feature space, conventional LDA, 2D LDA and other state-of-the-art methods namely Generalized Discriminant Analysis (GDA) [39], Nonparametric Weighted Feature Extraction (NWFE) [40], Kernel PCA (KPCA) [41] and Folded PCA (F-PCA) [37] when applied in small sample size scenarios. When compared with

the conventional LDA, the proposed F-LDA can also achieve reduction in contiguous memory requirement and computational complexity.

In Chapter 6, hybridization of GA and F-LDA (GA + F-LDA) for dimensionality reduction of hyperspectral data is proposed. The use of GA to select optimal spectral features in the data is followed by the application of F-LDA on the selected features for feature extraction and further dimensionality reduction. Performance of the proposed approach (GA + F-LDA) is evaluated on two hyperspectral datasets of 10 and 20 rice seed species (with both containing 256 spectral features), and another hyperspectral dataset of sugar containing 9 species and 160 spectral features. Experimental results show that, while the standard F-LDA gives slightly higher classification results than the proposed GA+F-LDA, applying F-LDA on datasets containing GA-selected optimal spectral features can further reduce the computational complexity and memory requirement of F-LDA.

## 1.2. Thesis Organisation

The remainder of this thesis is arranged as follows:

Chapter 2 provides relevant technical background information on hyperspectral imaging and various dimensionality reduction approaches for hyperspectral imaging data. The techniques presented are relevant to the methods that are used, developed and extended in the later chapters of this work.

Chapter 3 presents a review of approaches used by various authors in the literature for dimensionality reduction and classification of hyperspectral imaging data.

Chapter 4 proposes a novel combination of spatial features (extracted from high resolution RGB) and spectral features (extracted from hyperspectral images) for

hyperspectral data classification. It also applies LDA and compares its performance with that of the widely used PCA for dimensionality reduction of hyperspectral data.

Chapter 5 presents a novel dimensionality reduction method named F-LDA. This proposed technique is an extended and improved version of LDA for dimensionality reduction of hyperspectral imaging data.

Chapter 6 explores the effectiveness of hybridizing GA and F-LDA (GA+ F-LDA) for dimensionality reduction of hyperspectral imaging data.

Chapter 7 presents a summary of the thesis contributions and suggestions for future research directions.

## 2. Theoretical Background

Based on the research motivation presented in Chapter 1, this chapter provides theoretical background on hyperspectral imaging data classification with a focus on dimensionality reduction techniques for hyperspectral data in classification applications. Specifically, this chapter explains the workings of various data dimensionality reduction approaches for hyperspectral imaging data as well as related classification steps which are relevant to the contributions made in this thesis which will be described in the next chapters.

### 2.1. Hyperspectral Imaging

Hyperspectral images, as illustrated in Figure 2.1, are data cubes which consist of a set of 2D images $I$ (of rows $k$ and columns $l$) captured at different wavelengths, $w$ (third dimension) of the acquiring HSI sensors. Hence, each pixel in the data cube represents a set of spectral bands of reflected light in the wavelength range of the sensors [42],[34],[43]. Dimensions of the data cube can be denoted as $k \times l \times b$, where b is the number of spectral bands in the data [7].

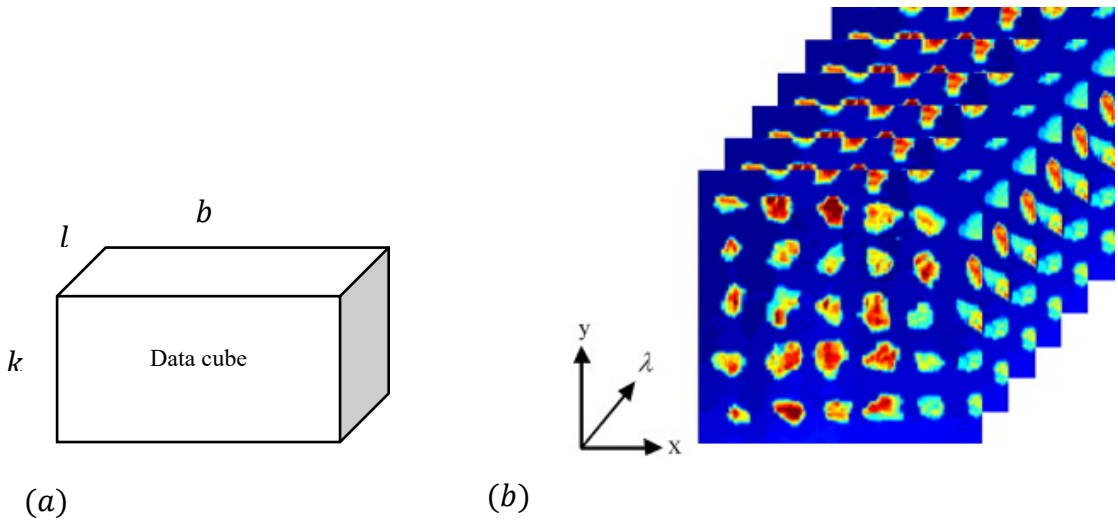



Figure 2.1 (a) Schematic diagram of the hyperspectral data cube, where $k \times l$ and $b$ denote the number of samples (pixels) and features (bands) in the hyperspectral data respectively (b) Pictorial diagram of the hyperspectral data cube where its spatial dimension is denoted as x × y and the spectral dimension as λ [43].

Hyperspectral imaging provides an alternative way of acquiring and processing images to the conventional RGB imaging and spectroscopy which have been applied separately for classification and process automation [8],[9]. RGB imaging can only provide morphological, colour and textural information of acquired images and not the constituents [8]. Similarly, traditional point spectroscopy can provide information on the molecular composition [8],[11] and no information on the spatial context. Hyperspectral imaging systems produce images with very high spectral resolution and offer simultaneous acquisition of spatial, textural and spectra information [8],[9],[7]. Hence, both the chemical and physical profiles of acquired hyperspectral images are utilised for improved model performance [8],[10].

## 2.2. Hyperspectral Imaging Data Classification

Hyperspectral imaging data classification is performed using the following 4 key steps [2],[39],[23]: data acquisition, data pre-processing, dimensionality reduction and classification. The classification steps are illustrated in Figure 2.2 and explained fully in the following subsections.

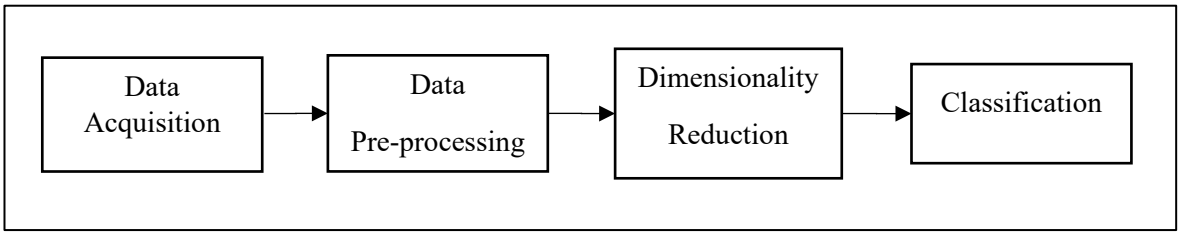



Figure 2.2 A block diagram showing the steps for hyperspectral imaging data classification

### 2.2.1. Data Acquisition

Hyperspectral images of samples are usually acquired at this stage using hyperspectral imaging systems. Such images can be acquired either by positioning the acquiring systems over a scene containing the samples for remote data acquisition or by installing the acquiring systems in a laboratory where samples whose images are to be captured are positioned below the cameras attached to the acquiring systems (As illustrated in Figure 2.3).

The hyperspectral imaging system can acquire data using different approaches namely whiskbroom, push-broom, and staring [44]. Whiskbroom is a technique used in

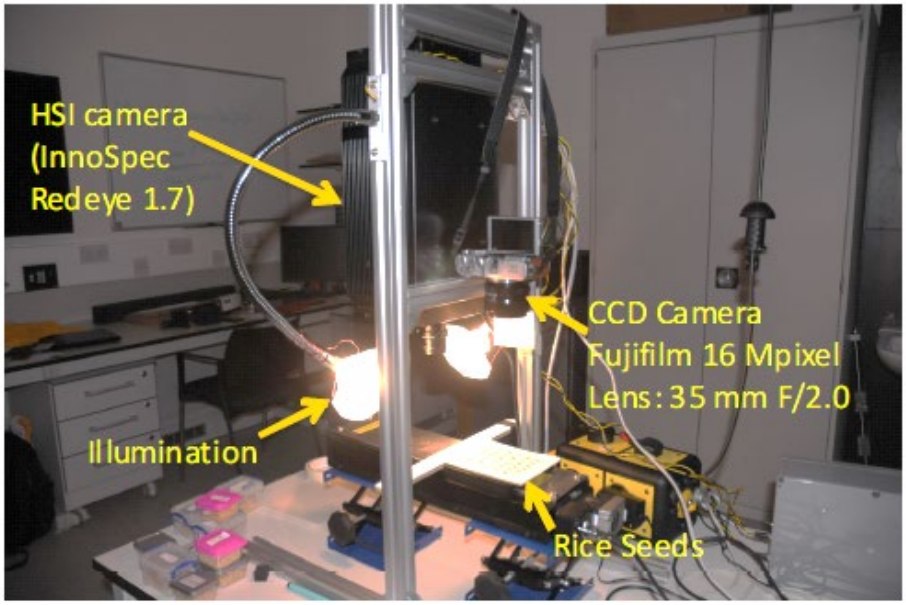



Figure 2.3 A pictorial diagram of the hyperspectral imaging camera set up in the laboratory. Samples whose image are to be acquired are positioned on a stage and below the hyperspectral imaging camera

hyperspectral imaging system to capture a point (pixel) in the image at a time. A dispersive element such as prism can then be used to disperse the spectral information in the captured point on a line detector. The hyperspectral data cube can therefore be formed by scanning the entire image point by point. Hence, whiskbroom scanning is also referred to as point or pixel scanning [1],[45]. When operating in the push-broom mode, the hyperspectral imaging system can capture a line (a set of points or pixels) in the image at a time. The spectra information of the scanned image line is then dispersed on a matrix detector. One axis of the matrix detector always shows the spatial information on the scanned image line while the other axis shows the corresponding spectral information. The hyperspectral data cube can therefore be formed by scanning the entire image in the spatial domain line by line. Hence, push-broom scanning is also

referred to as line scanning [2],[45],[46]. When using the staring approach, the hyperspectral imaging system can capture the entire image at a time. The captured image is passed through a tunable filter and acquired at a narrow frequency range. This is possible by tuning the filter to the corresponding wavelength (as a function of time). The process above is repeated at different wavelengths of the filter. The resulting hyperspectral data cube is therefore an accumulation of images acquired at different spectral bands [45],[47]. Another approach which can be used is snapshot [44]. In snapshot mode, the acquiring system does not employ the scanning approach. Instead, it captures both the spectral and spatial information in one shot. Hence, snapshot is also known as single-shot [45]. In Chapter 4 of this thesis, where a hyperspectral imaging system is used to acquire images of rice seeds of different species, the push-broom approach is adopted. This is due to the high spectral resolution it offers [45] and its successful deployment in recently published related papers [13],[15],[48].

It is also worth noting that there are existing publicly available hyperspectral imaging data which were captured remotely using popular sensors such as Hyperion sensor [5], ROSIS (Reflective Optics System Imaging Spectrometer) sensor [24], and AVIRIS (Airborne Visible/InfraRed Imaging Spectrometer) sensor [25]. For instance, Hyperion sensor on the National Aeronautics and Space Administration (NASA) EO-1 satellite was deployed at the Okavango Delta to capture Botswana hyperspectral data at a spectral range of 400-2500 nm. The acquired Botswana data contains 14 different classes and originally has 242 spectral bands and a spatial dimension of 1476 × 256 pixels. ROSIS sensor was deployed over Pavia in northern Italy to capture Pavia Centre hyperspectral data. There are 9 different classes in the acquired hyperspectral data which originally has 114 spectral bands and a spatial dimension of 1096 × 1096

pixels. ROSIS sensor was also used to acquire another hyperspectral data named Pavia University Scene when deployed over Pavia, in northern Italy. There are 9 different classes in the acquired Pavia University hyperspectral data which has 115 spectra bands and a spatial dimension of 610 × 340 pixels. AVIRIS sensor was deployed over Salinas Valley in California to capture Salinas-A hyperspectral data. 6 different classes are present in the acquired Salinas-A hyperspectral data which has a spatial dimension of 86×83 pixels and originally contains 224 spectral bands. AVIRIS sensor was also deployed at Indian Pine test site in North-western Indiana to capture Indian Pine hyperspectral data at a range of 400-2500 nm. The acquired Indian Pine data contains 16 different classes and originally has 224 spectral bands and a spatial dimension of 145 × 145 pixels. The Botswana, Pavia Centre, Pavia University, Salinas-A and Indian Pine data are all used in Chapter 5 of this thesis.

Also, hyperspectral images of rice seed samples were acquired in the laboratory using a hyperspectral imaging system for use in this thesis. The hyperspectral imaging system, which consisted of a Specim V10E Imaging Spectrograph and Hamamatsu ORCA-05G CCD camera, was used to acquire the hyperspectral images at a Visible - Near Infrared (VIS/NIR) range of ~ (385 – 1000) nm. This is a part of a bigger data acquisition system, based on which the performance of a new data fusion technique is evaluated in Chapter 4. Hence, full description of the system will be presented in that chapter.

### 2.2.2. Data Pre-processing

At this stage, the acquired hyperspectral imaging data is pre-processed to reduce or remove noises which are present in the acquired hyperspectral data. Discarding noisy

bands and subtraction of dark current from the acquired hyperspectral data [49] are two well-known approaches for achieving this and are described below:

A. Removal of noisy bands

The spectral range of hyperspectral imaging cameras are often characterised by the presence of noisy bands at both ends of such range [49]. It is necessary to remove these noisy bands to prevent them from limiting the performance of models when presented with hyperspectral data for classification.

For instance, the Botswana, Pavia Centre, Pavia University, Salinas-A and Indian Pine data were all pre-processed using this approach before they were made publicly available as illustrated in Table 2.1 below [5],[24],[25],[50].

Table 2.1 Number of bands in the Botswana, Pavia Centre, Pavia University, Salinas-A and Indian Pine data before and after the removal of noisy bands

| Datasets | Number of bands (originally) | Number of discarded bands | Number of retained bands |
|---|---|---|---|
| Botswana | 242 | 97 | 145 |
| Pavia Centre | 114 | 12 | 102 |
| Pavia University | 115 | 12 | 103 |
| Salinas-A | 224 | 20 | 204 |
| Indian Pine | 224 | 24 | 200 |

B. Subtraction of dark currents

Dark currents are electronic currents which flow through detectors in hyperspectral imaging systems and constitute noise in the images acquired by the systems [49],[51]. Dark currents are usually generated due to the difference in temperature between the environment and the sensor [51]. Charged-Couple Device (CCD) detector is an example of detectors in where thermally induced current can be generated [49]. Subtraction of a dark current image (which is usually captured by covering the camera lens using its cap while the light source is turned off [8],[13]) from the acquired hyperspectral image is a common approach for removing the influence of dark current [49],[14].

### 2.2.3. Dimensionality Reduction of HSI Data

Dimensionality reduction of HSI data is the process of reducing the number of spectral features which are present in the data. Dimensionality reduction of HSI data can be classified into feature extraction and feature selection.

Feature extraction is the process of extracting useful features from the data by transforming it from the original feature space to a lower dimensional space. The lower dimensional space contains new features which provide important information for improved model performance. After applying feature extraction techniques on HSI data, the features extracted are usually fed to models for further action such as classification. Feature selection, on the other hand, is the process of selecting an optimal feature subset from several subsets of the original feature set. Since the new space is a subset of the original set, it usually contains features which are also present in the original set. It retains important information in the data while discarding the others.

Explained below are the reasons why dimensionality reduction of HSI data is necessary for improved model performance:

1. For a data with a fixed number of samples, classification accuracy will continue to increase as more features are added to the data until the classification accuracy reaches a peak. A decline in the classification accuracy begins after the peak is reached following the initial rise. This is called Hughes phenomenon. HSI data is characterised by the presence of many spectral bands, usually in the order of hundreds. Also, the applications of HSI data are faced with the problem of small sample size (limited number of labelled samples for training). Lack of enough samples for training and high dimensionality of data are responsible for the Hughes phenomenon which limits the performance of traditional machine learning classifiers when applied on HSI data. Hence, dimensions of HSI data are often reduced before they are presented to classification models to avoid the Hughes phenomenon.
2. Many of the spectral bands which are present in HSI data are highly correlated. High correlation among the bands results in data redundancy and noise which in turn limit the performance of models when applied on HSI data. Hence, dimensionality reduction tools are often applied on HSI data to produce or retain features which contain key information while discarding features which constitute noise and redundancy in the data.
3. Models perform various mathematical operations such as multiplication on hyperspectral data matrices. The computational complexity of these operations are often dependent on the dimension of the data matrices. Hence, reducing the

dimensionality of the hyperspectral data matrices will reduce the computational complexity as well as the space required to store such matrices.

While pre-processed hyperspectral imaging data can be fed directly to models for classification, dimensionality reduction techniques have proven to be effective in improving performance of such models when applied on the data prior to classification. A description of selected dimensionality reduction techniques are presented below - with a focus on those which will be applied and extended in the contribution chapters of this thesis (Chapter 4 - 6):

#### *2.2.3.1. Principal Component Analysis*

Principal Component Analysis (PCA) performs data dimensionality reduction by transforming the data into a linearly uncorrelated (orthogonal) components by looking for directions with the most variations in the data. PCA then ranks these Principal Components (PCs) based on the percentage of variance they explain in the data. This is illustrated in Figure 2.4 where the first principal component (PC1) accounts for the most variation in the data followed by the second principal component (PC2) which is selected to be orthogonal and ranked (based on the amount of variation it accounts for) next to PC1. PCA achieves this without considering the classes in the data. Hence, it is classified as an unsupervised dimensionality reduction technique.

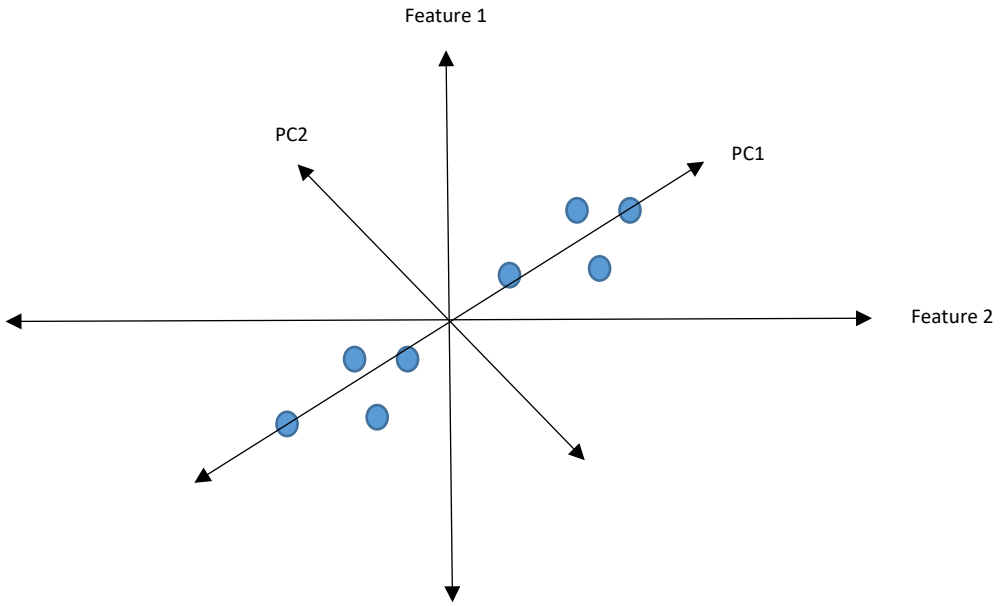
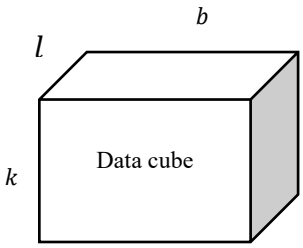


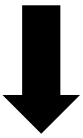

Figure 2.4 Transformation of a 2 dimensional data into two orthogonal PCs by PCA.

PCA consists of the following three main steps: (1) Covariance matrix computation (2) Eigenvalues and eigenvectors computation and (3) Data projection. Implementation steps of the PCA and its applications for feature extraction and dimensionality reduction of hyperspectral data are fully described below:

1. Covariance matrix computation

Firstly, as illustrated in Figure 2.5, the hyperspectral data cube is converted into a data matrix, $\boldsymbol{X}$ of size $s * f$ where $s = k \times l$ and $f = b$ represent the number of samples (pixels) and features (bands) in the hyperspectral data respectively. The elements of $\boldsymbol{X}$ are as shown in (2.1).



$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1f} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2f} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{s1} & x_{s2} & x_{s3} & \cdots & x_{sf} \end{bmatrix}$$

Figure 2.5 Conversion of the hyperspectral data cube into a 2D matrix **X** with a dimension of $s \times f$ where $s = k \times l$ is the number of samples and $f = \mathrm{b}$ is the number of features in the data. Each row in **X** represents the spectral vector of a pixel in the hyperspectral data cube.

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1f} \\ x_{21} & x_{22} & \cdots & x_{2f} \\ \vdots & \vdots & \ddots & \vdots \\ x_{s1} & x_{s2} & \cdots & x_{sf} \end{bmatrix} \qquad (2.1)$$

Each column (feature) in $\boldsymbol{X}$ can be represented as $x_e = [x_{1e}, x_{2e}, \ldots . x_{Se}]^T$ where $\epsilon\, [1, f]$. If the mean of $x_e$ is denoted as $c_e$, each feature can be centralised as in (2.2).

$$\bar{x}_e = x_e - c_e \qquad (2.2)$$

If each column in $\boldsymbol{X}$ is replaced with the corresponding centralised $\bar{x}_e$, $\boldsymbol{X}$ can be mean-centered and formulated as in (2.3).

$$\overline{\boldsymbol{X}} = \begin{bmatrix} \bar{x}_{11} & \bar{x}_{12} & \cdots & \bar{x}_{1f} \\ \bar{x}_{21} & \bar{x}_{22} & \cdots & \bar{x}_{2f} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{x}_{s1} & \bar{x}_{s2} & \cdots & \bar{x}_{sf} \end{bmatrix} \tag{2.3}$$

The covariance matrix of the data matrix, $\boldsymbol{W}$ can be computed using (2.4).

$$\boldsymbol{W} = \frac{1}{S}\overline{\boldsymbol{X}}^T\overline{\boldsymbol{X}} \tag{2.4}$$

2. Eigenvalues and eigenvectors computation

The covariance matrix can be decomposed into three different matrices as in (2.5).

$$\boldsymbol{W} = v\lambda v^T \tag{2.5}$$

where $v = \left[v_1, v_2, v_3, \ldots . v_f\right]$ is an orthonormal matrix that contains the eigenvectors and $\lambda = [\lambda_1, \lambda_2, \lambda_3, \ldots . \lambda_f]$ is a diagonal matrix containing the eigenvalues.

3. Data projection.

The data can be projected using (2.6). If the eigenvectors are ranked in accordance with their corresponding eigenvalues in descending order, a sub matrix containing the first $d$ eigenvectors, represented as $\boldsymbol{V_d}$, can be selected to transform the data matrix into a lower dimensional space $Y_T$ as in (2.7).

$$Y = \overline{\boldsymbol{X}}v \tag{2.6}$$

$$Y = \overline{\boldsymbol{X}}\boldsymbol{V}_d \tag{2.7}$$

#### 2.2.3.2. Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) performs data dimensionality reduction by computing a transformation matrix to maximize the between class variance and minimize the within class variance in the data. LDA therefore goes further than PCA by considering the classes in the data. Hence, LDA is classified as a supervised dimensionality reduction technique. LDA can be broken down into the following three steps: (1) Computation of the within-class variance and between-class variance, (2) Computation of the transformation matrix and eigenvectors, and (3) Data projection. Implementation steps of the conventional LDA and its applications for feature extraction and dimensionality reduction of hyperspectral data are fully described below:

1. Computation of the within-class and between-class variance

Before LDA can be applied on the hyperspectral data cube, the data cube has to be converted into a data matrix $\boldsymbol{X}$ as illustrated in Figure 2.5. The dimension of $\boldsymbol{X}$ is given as $s \times f$ where $s = k \times l$ and $f = b$ represent the number of samples (pixels) and features (bands) in the hyperspectral data respectively. Each sample in $\boldsymbol{X}$ is a spectral vector of a pixel in the hyperspectral data cube and can be represented by $\boldsymbol{x_n}$ where $n \in [1, s]$. If the number of classes and $j$th class in $\boldsymbol{X}$ are represented as $c$ and $c_j$ respectively, the number of samples in each class can be represented as $N_j$ and the $i$th sample in class $c_j$ represented as $x_{ij}$ where $i \in [1,\ N_j]$. The mean of the samples in each class $c_j$, denoted as $\boldsymbol{m_j}$ where $j \in [1,\ c]$ , and the mean of all the samples in the data matrix $\boldsymbol{X}$, denoted as $\boldsymbol{m}$, can then be computed using (2.8) and (2.9) below.

$$\boldsymbol{m_j} = \frac{1}{N_j} \sum_{i=1}^{N_j} x_{ij} \qquad (2.8)$$

$$\boldsymbol{m} = \sum_{j=1}^{c} \frac{N_j}{s} \boldsymbol{m_j} \quad (2.9)$$

The within-class variance $\boldsymbol{V_W}$ and the between-class variance $\boldsymbol{V_B}$ of $\boldsymbol{X}$ can be computed using (2.10) and (2.11) respectively.

$$\boldsymbol{V_W} = \sum_{j=1}^{c} \sum_{i=1}^{N_j} (x_{ij} - \boldsymbol{m_j})^T (x_{ij} - \boldsymbol{m_j}) \quad (2.10)$$

$$\boldsymbol{V_B} = \sum_{j=1}^{c} N_j (\mathbf{m_j} - \mathbf{m})^T (\mathbf{m_j} - \mathbf{m}) \quad (2.11)$$

where $\boldsymbol{V_W} \in \Re^{f \times f}$ and $\boldsymbol{V_B} \in \Re^{f \times f}$.

2. Computation of transformation matrix and eigenvectors computation, and data projection

The transformation matrix, $\boldsymbol{T}$, can be calculated to maximize the between-class variance, $\boldsymbol{V_B}$, and minimize the within-class variance, $\boldsymbol{V_W}$, using (2.12).

$$\boldsymbol{T} = \boldsymbol{V_W}^{-1} \boldsymbol{V_B}, \boldsymbol{T} \in \Re^{f \times f} \quad (2.12)$$

The eigenvalues $\lambda$ and the eigenvectors $\boldsymbol{v}$ are computed from the EigenValue Decomposition (EVD) of $\boldsymbol{T}$. The eigenvectors $\boldsymbol{v}$ with a dimension of $f * f$ are then ranked using their corresponding eigenvalues (starting from the highest to the lowest). By selecting the first $d$ columns of the ranked $\boldsymbol{v}$ and discarding the remaining eigenvectors (with small eigenvalues), the ranked $\boldsymbol{v}$ is reduced to $\boldsymbol{V_d}$ with a dimension of $f \times d$. It is noteworthy that the value of $d$ is bound by the number of non-zero eigenvalues, which is given as $c - 1$. This limits the number of components that can be selected by the LDA to $c - 1$ which is also the rank of the between-class variance matrix, $\boldsymbol{V_B}$ [52],[53],[54].

The data matrix $\boldsymbol{X}$ can then be projected into a lower dimensional space $Y$ using (2.13).

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{V_d},\ \boldsymbol{X} \in \Re^{s \times f}, \boldsymbol{V_d} \in \Re^{f \times d} \tag{2.13}$$

where $\boldsymbol{Y}$ is the projected data and $\boldsymbol{X}$ is the original data.

#### *2.2.3.3. Folded Principal Component Analysis (F-PCA)*

Folded PCA (F-PCA) is an extension of the conventional PCA. F-PCA folds each spectral vector in the data matrix into a 2D matrix. The data is now structured as a stack of 2D matrices (folded spectral vectors). F-PCA applies the three main steps of the PCA mentioned in Section 2.2.3.1 on the data, treating each 2D matrix in the stack as a sample. After projecting the data into a lower dimensional space, F-PCA unfolds the projected samples. Unlike PCA which applies the three key steps on the data matrix (a set of spectral vectors), F-PCA processes a set of 2D matrices (the stack). By folding each spectral vector, F-PCA can extract the local features in the hyperspectral data for improved classification performance. Also, since the dimension of the folded matrices are usually much smaller than the original data matrix, the computational costs and memory requirement for the F-PCA are also usually smaller than those of the PCA [37]. Different dimensions of the folded matrices are usually explored with the one that gives the best classification accuracy selected.

Let $\bar{x}_n = [\bar{x}_{n1}, \bar{x}_{n2}, \ldots . \bar{x}_{nf}]$ represents each row (spectral vector) in the data matrix $\overline{\boldsymbol{X}}$ where $n \in [1, s]$, each spectral vector is folded into a 2D matrix (folded sample), $\overline{\boldsymbol{X}}_n$. The covariance matrix of each folded vector is computed using (2.14). This is known as the partial covariance matrix [37]. All the partial covariance matrices are accumulated to form the main covariance matrix of the data matrix $\overline{\boldsymbol{X}}$ as in (2.15).

$$\boldsymbol{W}_n = \overline{\boldsymbol{X}}_{\boldsymbol{n}}{}^T \overline{\boldsymbol{X}}_n \tag{2.14}$$

$$\boldsymbol{W}_{FPCA} = \frac{1}{s} \sum_{n=1}^{s} \boldsymbol{W}_n = \frac{1}{s} \sum_{n=1}^{s} \bar{\boldsymbol{X}}_n{}^T \bar{\boldsymbol{X}}_n \qquad (2.15)$$

As in the PCA, EVD of the covariance is done to obtain the eigenvectors and their corresponding eigenvalues. Projection of each folded samples is done using a subset of the eigenvector matrix (first $d$ eigenvectors selected) using (2.16).

$$\boldsymbol{Y_n} = \bar{\boldsymbol{X}}_n \boldsymbol{V}_d \qquad (2.16)$$

The final step of the F-PCA is the unfolding of each projected sample matrix, $\boldsymbol{Y_n}$.

#### *2.2.3.4. Genetic Algorithms*

Genetic Algorithms (GAs) are optimization techniques which have found applications in data dimensionality reduction process, specifically feature selection. Genetic algorithms function as a wrapper-based feature selector. This is illustrated in Figure 2.6 where a Genetic Algorithm, like any other wrapper-based feature selectors, analyses the relationship between the entire feature sets in the raw data and classification model to select an optimal subset of features for increased classification performance.

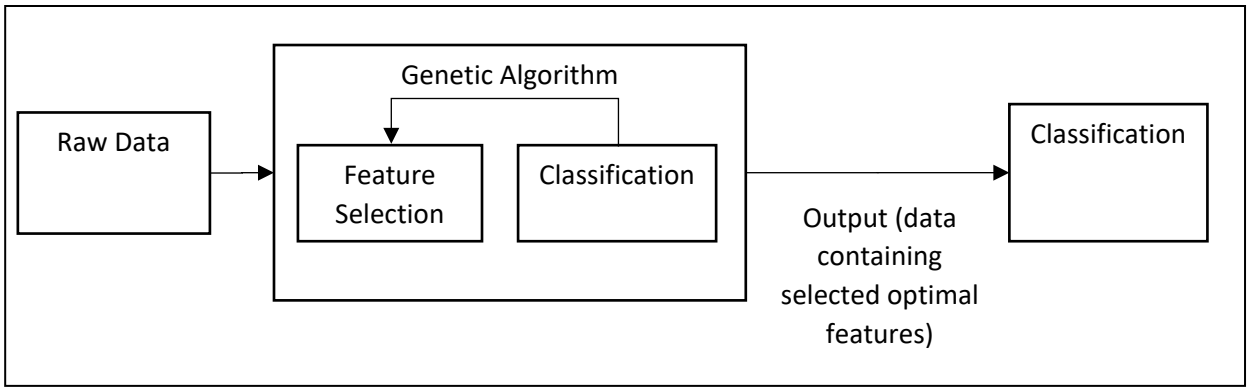



Figure 2.6 Selection of optimal features in the raw data using a Genetic Algorithm. The Genetic Algorithm considers several subsets of the original feature set and uses a classification model to select the optimal feature subset as its output.

GAs selects optimal feature subset from the main feature sets using the principle of natural selection, crossover, and mutation which are explained below:

1) Selection: GAs randomly initialize a set of chromosomes (possible solutions), also known as the candidate solutions. Each chromosome is usually encoded as a vector containing binary values (genes). Fitness of each chromosomes are estimated using a fitness function, also known as the objective function. Chromosomes which are considered unfit are discarded while those which are considered fit are selected. Selected chromosomes are mated using crossover and mutation to produce new offspring.
2) Crossover: This is an operator used by GAs to mate chromosomes which are considered fit to reproduce fitter chromosomes. New chromosomes which are produced by crossover usually contains genes which are present in at least one of the mating chromosomes.
3) Mutation: This is another operator used by GAs to produce new offspring but differ from the crossover because the offspring it produces contain genes which

are not present in the parent chromosomes. Reproduction by mutation provides a way for GAs to explore the search space and reach a global solution.

The selection, crossover, and mutation processes are repeated for many generations until certain conditions are met. The common approach is to terminate the process when the maximum number of generations set is reached or fitness of the solution has reached a peak level such that better results cannot be attained at further generations [55],[56]. Output of the feature selection process using Genetic Algorithm is a reduced dataset which contains selected optimal features. Performance of classification models can be improved by feeding such models with the reduced dataset instead of the data containing the full features.

### 2.2.4 Classification

At this stage, classification models are built to classify the hyperspectral data. To achieve this, the hyperspectral data is divided into training and testing sets, after which classification models are trained using the training set. The trained models can then be applied to classify samples (spectral vectors) which are present in the testing sets i.e. different classes are assigned to samples in the testing set. The models selected and implemented for classification of hyperspectral imaging data in this thesis are Random Forests [8] and Support Vector Machines [37]. Workings of the selected classification models and reasons for their selection are explained below:

1) Random Forest

Random Forest is an ensemble model which contains many decision trees. Each decision tree in the Random Forest, as its name implies, is a tree-like classifier which uses the concepts of flow-chart to learn the representation in data to make decision

when presented with new data. In each decision tree are nodes and branches. Each node in the tree shows the test of an attribute or feature. Each branch of the tree joins (extends) a node to the subsequent one and shows one of the possible test results on the attribute in the node it extends [57]. Each branch therefore provides an alternative route from the previous node it extends to a new node. Starting with the topmost node (which is the parent or root node), each decision tree in the Random Forest outlines all possible routes or branches to the final nodes (which are also known as the leaf nodes). The leaf nodes show the outcomes (targeted classes or categories in the data). Each decision tree is constructed from a bootstrapped (sampled with replacement) dataset and allowed to grow to a maximum extent from a random subset of variables at each nodes of the trees. Each decision tree in the Random Forest classification model is illustrated in Figure 2.7 where the final nodes show the targeted classes (possible decisions) [58]. Each decision tree in the Random Forest is allowed to make a decision and accumulation of the decisions reached by the trees are used by the Random Forest to make predictions on new data (as illustrated in Figure 2.8).

Random Forests can effectively process large datasets and overcome the problem of overfitting through the use of many decision trees. Random Forests is selected for the tasks in Chapter 3 and 6 of this thesis where new data fusion and dimensionality reduction techniques are proposed for hyperspectral data classification and hyperspectral images of rice seeds are used to evaluate performance of the proposed techniques. The selection of Random Forests is due to the promising results achieved by the Random Forests model when applied in related papers on hyperspectral imaging data of rice seeds [8],[9],[59].

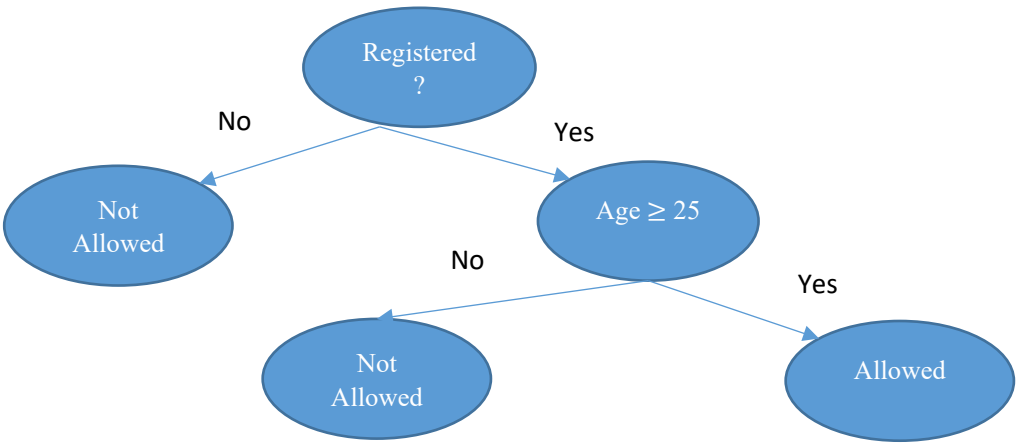



Figure 2.7 A decision tree to classify people as ‘allowed’ or ‘not allowed’ (to enter a venue). Based on this tree, only those who are registered and are aged 25 or above will be classified as allowed.

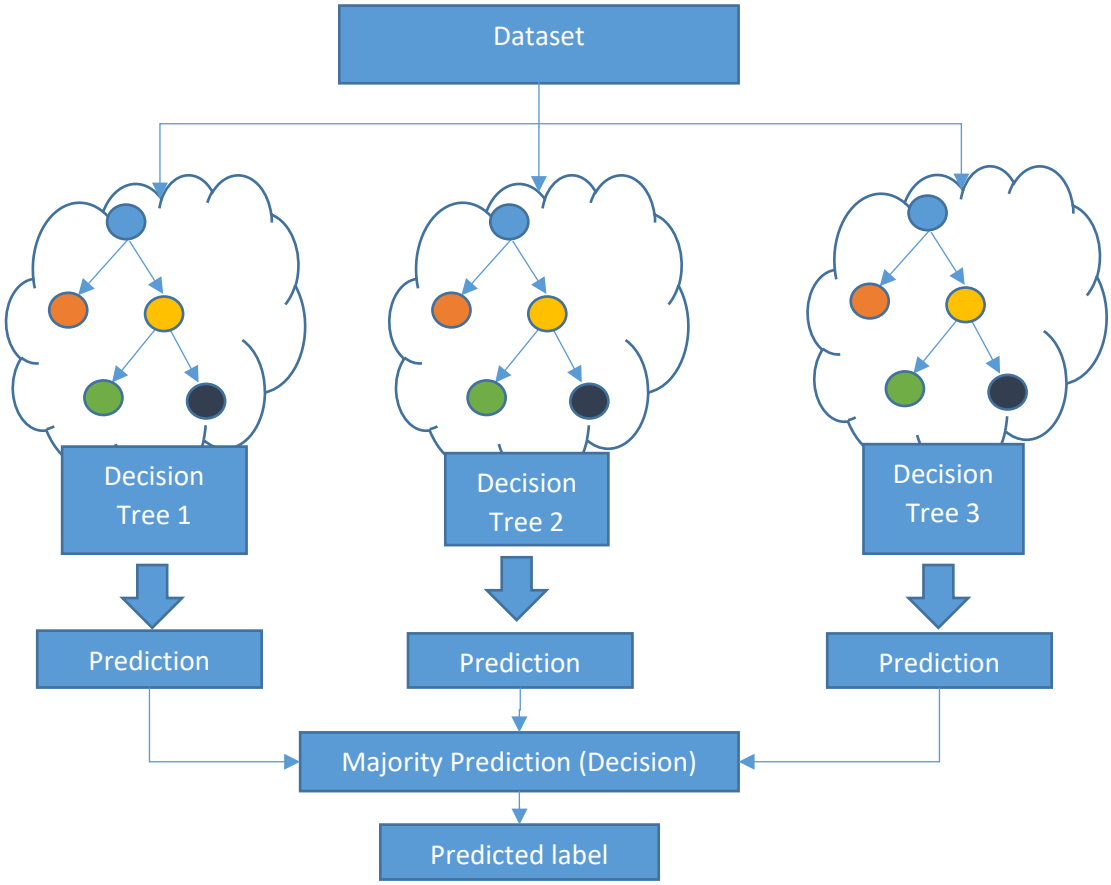



Figure 2.8 A random forest classification model containing three decision trees where the aggregate of the decisions reached by each tree is used to make a final prediction on the data.

2) Support Vector Machines (SVM)

SVM is a classification model which uses a kernel function to transform data into a higher dimensional space where it finds an optimal hyperplane (as illustrated in Figure 2.9) to discriminate the different classes in the data. One kernel that is commonly adopted for use in SVM when applied on hyperspectral imaging data is Radial Bias Functions (RBF), which uses two parameters namely penalty and gamma [60],[61]. The first step in using SVM is to optimize parameters of the kernel function. The optimal parameter selection can be achieved at the training stage using a grid search

and the training dataset. The SVM model with the optimized parameters can then be applied on the test dataset for final evaluation. SVM is adopted for the tasks in Chapter 5 of this thesis where a new dimensionality reduction approach is proposed for hyperspectral data classification and remotely sensed hyperspectral images are used to evaluate performance of the proposed approach. The satisfactory classification performance achieved by the SVM model when applied in related papers [60],[61] on remotely sensed hyperspectral data motivated its selection.

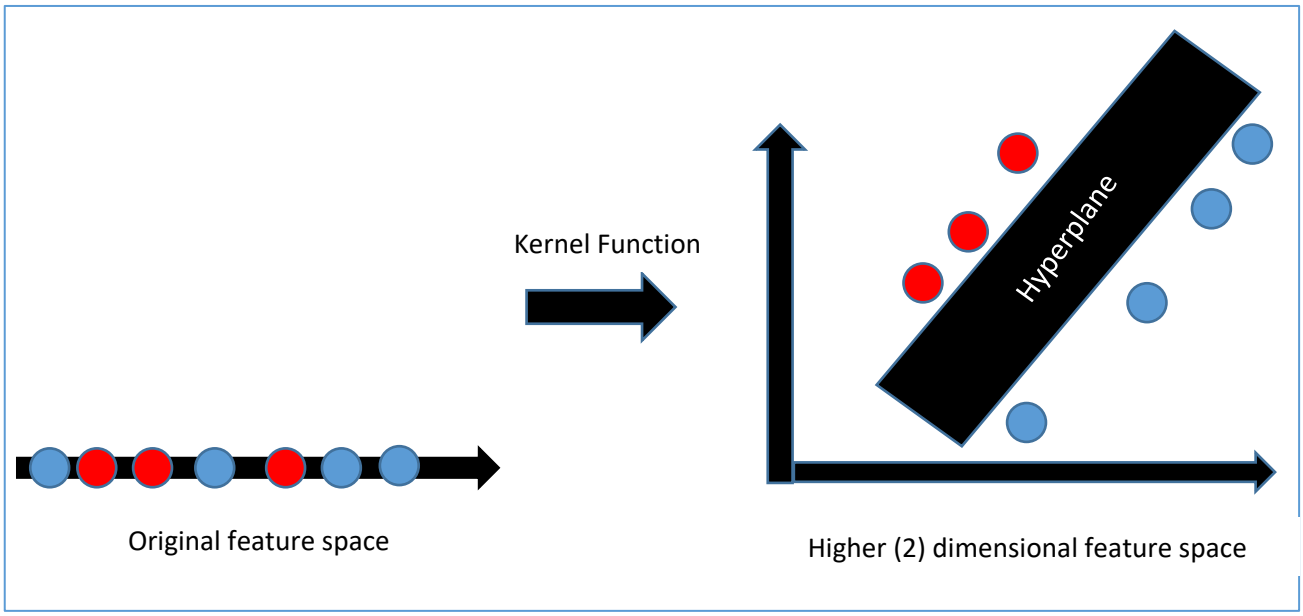



Figure 2.9 Transformation of the data by SVM to a higher dimensional space using a kernel function. The two categories become separable in the new space.

### 2.2.5. Performance Evaluation

Performance metrics are usually computed to evaluate how well the model has performed in classifying the hyperspectral data. It is worth noting here that the model performance can be influenced by the type of pre-processing techniques, dimensionality reduction techniques and classification models adopted. This work is focused on developing new dimensionality reduction approaches for hyperspectral

imaging data and so it will be given utmost attention when performance evaluation of model is being carried out. Description of the various performance metrics which will be used at different stages of this thesis is provided below:

1) Classification Accuracy

Accuracy of a classification model can be computed by comparing the predicted labels and the targeted labels in the test set. The Overall Accuracy (OA) can be computed as the ratio of the number of correct predictions to the total number of predictions. Mathematically, OA can be defined using (2.17). If the accuracy for each class in the data, $A_j$ is calculated using (2.17), then the Average Accuracy (AA) can be calculated as the average of all $A_j$ for the data. AA is mathematically defined in (2.18).

$$\text{OA} = \frac{\text{number of correctly predicted labels}}{\text{total number of predictions}} \qquad (2.17)$$

$$\text{AA} = \frac{\sum_j^c A_j}{c} \qquad (2.18)$$

where $A_i$ is the accuracy for class $i$ and $c$ is the number of classes in the data.

2) Precision, Recall and $F_1$ score

Precision is a measure of the fraction of positive predictions that are true positives while recall is a measure of the fraction of the actual positives that are true positives. $F_1$ score is the harmonic mean of precision and recall. It aims to find a balance between the precision and recall. Precision, recall and $F_1$ score are mathematically defined in (2.19), (2.20) and (2.21).

$$P = \frac{t_p}{t_p + f_p} \qquad (2.19)$$

$$R = \frac{t_p}{t_p + f_n} \tag{2.20}$$

$$F_1 \text{ score} = 2 * \frac{P * R}{P + R} \tag{2.21}$$

where $t_p$ is the number of true positives, $f_p$ is the number of false positives, $t_n$ is the number of true negatives and $f_n$ is the number of false negatives.

Macro averaging and micro averaging are two common approaches for computing the average of *P, R* and $F_1$ score. In macro averaging, the *P, R* and $F_1$ for each class are computed and the mean of *P, R* and $F_1$ for all classes are reported as the macro averaged *P, R* and $F_1$. In micro averaging, true positives, false positives, true negatives and false negatives are computed for each of the classes. Macro averaged and macro averaged *P, R* and $F_1$ score are mathematically defined in (2.22) – (2.27), where c is the number of classes in the data.

$$P_{macro\ averaged} = \frac{P_1 + P_2 + \cdots + P_C}{C} \tag{2.22}$$

$$R_{macro\ averaged} = \frac{R_1 + R_2 + \cdots + R_C}{C} \tag{2.23}$$

$$P_{macro\ averaged} = \frac{F_1\ \text{score}_1 + F_1\ \text{score}_2 + \cdots + F_1\ \text{score}_C}{C} \tag{2.24}$$

$$P_{micro\ averaged} = \frac{t_{p1} + t_{p2} + \cdots t_{pc}}{t_{p1} + t_{p2} + \cdots t_{pc} + f_{p1} + f_{p2} + \cdots + f_{pc}} \tag{2.25}$$

$$R_{micro\ averaged} = \frac{t_{p1} + t_{p2} + \cdots t_{pc}}{t_{p1} + t_{p2} + \cdots t_{pc} + f_{n1} + f_{n2} + \cdots + f_{nc}} \tag{2.26}$$

$$F_1\ \text{score}_{micro\ averaged} = 2 * \frac{P_{micro\ averaged} * R_{micro\ averaged}}{P_{micro\ averaged} + RR_{micro\ averaged}} \tag{2.27}$$

3) Kappa Coefficient

Kappa coefficient ($kc$) is a statistical tool used as a performance metric in classification to quantify the degree of agreement between the predicted labels and the actual labels [61],[62]. Kappa coefficient can be calculated using (2.28) and takes both the observed and chance agreements into account. Higher Kappa coefficient implies a better classification performance [21].

$$kc = \frac{P_o - P_e}{1 - P_e} \tag{2.28}$$

where $P_o$ is the observed agreement and $P_e$ is the probability of chance agreement.

# 3. Related Work

Review of publications which are related to the current study is presented in this chapter. Specifically, 3.1 presents a review of dimensionality reduction techniques for hyperspectral imaging data in classification applications. 3.2 reviews computer vision based techniques used by various authors in the literature for classification purposes with a focus on the potential of hyperspectral imaging data in rice seed classification. A review of various techniques which are used for dimensionality reduction of hyperspectral imaging data when applied for classification of rice seeds is presented in 3.3.

## 3.1. A Review of Dimensionality Reduction Tools for Hyperspectral Data Classification

Hyperspectral data usually contains hundreds of spectral bands, many of which are highly correlated [63],[64],[65]. High correlation among the spectral bands in the hyperspectral data results in data redundancy and noise [65],[66]. Also, enough samples are usually not available in the hyperspectral data for training [63],[67]. Insufficient training samples and high dimensionality of hyperspectral data often result in Hughes phenomenon [24],[68],[69]. The Hughes phenomenon is usually responsible for the decline in the classification accuracy as the number of spectral bands increases after the accuracy has initially risen to a peak [17],[18],[70]. Hence, the classification performance (classification accuracy and computational complexity) of traditional machine learning models can be degraded by Hughes phenomenon when the number of training samples are limited and the data dimensionality is high. Dimensionality reduction of hyperspectral data through feature extraction techniques can therefore resolve the problem posed by the Hughes phenomenon. Feature

extraction techniques can be applied on hyperspectral data to transform it into a lower dimensional space where features are de-correlated and reduced in numbers. Hence, the classification accuracy is increased, and the computational complexity and memory requirements are reduced as well.

Deep learning models which incorporate layers for feature extraction prior to classification have been applied for hyperspectral data classification. In [68], a three dimensional Convolutional Neural Network (3D CNN) was proposed for the classification of hyperspectral images. In the proposed 3D CNN, the convolution layers and the feature maps are three dimensional to capture both the spatial and spectral information in the hyperspectral images. A pooling layer was included in the network by the authors in [68] to further reduce the size of the feature maps. The authors in [68] used Pavia University dataset to evaluate performance of the 3D CNN and showed that the 3D CNN can outperform CNN when applied to classify hyperspectral data in terms of classification accuracy.

In another paper [24], four different deep learning models namely two dimensional CNN (2D-CNN), three dimensional CNN (3D-CNN), recurrent 2D-CNN (R-2D-CNN) and recurrent 3D-CNN (R-3D-CNN) were proposed for classification of hyperspectral data. In the 2D-CNN, the authors implemented a 2D convolution layers for feature extraction and excluded the pooling layers (which can further reduce dimensionality of the feature maps) from the network to prevent them from affecting the network's classification accuracy. The authors went further to develop a recurrent 2D CNN (R-2D CNN) which is capable of shrinking the small patch at the centre of each pixel gradually and concentrating them at the central pixel to reduce any noise which can arise in the 2D CNN. Similar approaches were implemented to develop the

3D CNN and the recurrent 3D CNN wherein the three dimensional convolutional layers were implemented to allow the network to capture both the spatial and the spectral context. The authors observed that the R-2D-CNN and R-3D-CNN gave higher classification accuracy than the 2D-CNN and 3D-CNN respectively and that R-3D-CNN gave the best classification performance when applied on Indian Pines, Botswana, Salinas, Pavia Center, Pavia University and Kennedy Space hyperspectral datasets.

While the deep learning classifiers achieved promising results, they require complex parameter tuning. Unlike deep learning based models, traditional machine learning classifiers require simple parameter tuning and outputs of a separate feature extraction technique [24],[71]. These feature extraction techniques can be divided into supervised and unsupervised techniques. This division depends on whether the labels in the data are used in the feature extraction process or not. Supervised techniques make use of the labels in the feature extraction process while unsupervised techniques do not. Several supervised feature extraction techniques have been applied for dimensionality reduction of hyperspectral data.

In [72], dimensionality reduction of Indian Pine hyperspectral data was performed using Linear Discriminant Analysis (LDA), a supervised feature extraction technique. The authors in [72] fed the features extracted from the hyperspectral data to a SVM classification model and observed that data dimensionality reduction using LDA can improve performance of classification models.

Generalized Discriminant Analysis (GDA) is another supervised technique which was applied in [39] for feature reduction of hyperspectral data acquired using

AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) and PHI (Pushbroom Hyperspectral Imager) sensors . In [39], GDA was implemented as a non-linear version of the LDA by using a kernel trick to map all the samples in the hyperspectral data to a new high dimensional space. The use of polynomial kernels, RBF kernels and neural network kernels were considered. Subsequently, the traditional steps in the linear version (LDA) were applied on the data in the new space for non-linear feature extraction. The significant improvement on the classification performance of the full spectral bands and the LDA achieved by the GDA using the RBF kernel demonstrated the potential of GDA for feature extraction of hyperspectral data.

Similarly, dimensionality reduction of hyperspectral data have been carried out using several unsupervised feature extraction techniques. In [73], dimensions of different hyperspectral datasets namely Indian Pines, Salinas, and Washington DC hyperspectral datasets were reduced using Principal Component Analysis (PCA), an unsupervised feature extraction technique. PCA transformed the original features into linearly uncorrelated principal components – the components which explain the most variance in the data are usually retained. The performance of PCA in [73] are promising and demonstrates its potential in feature extraction of hyperspectral data prior to classification.

Singular Spectral Analysis (SSA) is another unsupervised feature extraction techniques which was applied in [74] on two hyperspectral datasets (captured using AVIRIS) namely 92AV3C and Salinas C datasets. In [74], SSA decomposed each pixel into several components and reconstructed the pixel while discarding the noisy and less representative ones based on EigenValue Decomposition (EVD). By so doing, SSA achieved improvements in the classification accuracy.

Recent research has focused more on extending and proposing different versions of some of these techniques for improved classification performance. For instance, the paper in [41] applied a non-linear version of PCA to transform the Pavia University, Pavia Center and Washington DC hyperspectral datasets into a linearly separable feature space to capture higher order statistics while extracting the principal components. By capturing higher order statistics, the authors in [41] achieved improvement in the classification accuracy. Also, the authors in [75] improve the performance of SSA, which was evaluated on 92AV3C and Salinas C hyperspectral datasets, by applying Singular Value Decomposition (SVD) on a representative pixel. This is different from the conventional SSA where SVD is applied on every pixel in the hyperspectral data. The application of SVD on a representative pixel by the authors in [75] led to reduction in computational complexity.

The paper in [37] proposed an extension of the PCA where each of the spectral vectors in the data matrix was folded into feature matrix. The authors in [37] then computed the covariance matrix and eigenvectors using the converted matrix (folded vectors) and unfolded the projected samples for classification of hyperspectral data which resulted in higher classification accuracy and reduced computational complexity. Improvements in the performance of the classification model (SVM) achieved in [37] when applied on Indian Pine hyperspectral and Synthetic Aperture Radar (SAR) datasets motivates the work in Chapter 5 of this thesis where LDA is extended using a similar innovative step.

LDA transforms the data into a lower dimensional space by computing a transformation matrix to maximize the between class variance and minimize the within class variance. The transformation matrix produced by LDA can maximize the

separability among the different classes in the data. The number of features ($f$) which are present in the data determines the size of the between class matrix, the within class matrix and the transformation matrix (which is given as $f * f$). Since hyperspectral data usually contains many features (usually in hundreds), the dimension of these matrices can become very large and huge memory and high computational cost will be required by LDA to store and process such matrices [76],[77],[78]. The maximum number of features that can be extracted by LDA is $c - 1$, where $c$ is the number of classes in the hyperspectral data. This limitation is imposed on LDA by the rank of the between class variance matrix which is given as $c - 1$ [52],[53],[54]. Also, LDA gives below par performance when applied on small training samples which explains why its usage on hyperspectral data is limited since the availability of hyperspectral data samples for training is usually limited [26].

To solve the small sample size problem, the papers in [79],[80],[81] used two dimensional (2D) LDA in face recognition applications. In the 2D-LDA proposed by the authors in [79], each sample was represented using a matrix and so the data was treated as a collection of matrices. This is different from the common data representation where each sample is a vector and the data is a data matrix. The authors in [79] went on to propose another approach named 2D-LDA + LDA, where LDA is applied on the outputs of the 2D-LDA to further reduce the features produced by the 2D-LDA. Performance of the two proposed approaches were evaluated on three publicly available face datasets namely PIX, ORL, and PIE. The results obtained by the authors in [79] demonstrate the potential of the 2D-LDA in improving the performance of the traditional LDA when used in small sample size scenarios. The

authors in [79] also observed that 2D-LDA + LDA gave the highest classification accuracy though it was slower than the 2D LDA.

The papers in [80] and [81] also applied 2D-LDA but on different face datasets. The authors in [80] applied 2D-LDA on ORL and Yale face database B while ORL face database was used for performance evaluation of 2D-LDA by the authors in [81]. Potential of 2D LDA in solving the problem of small sample size (which limits the traditional LDA) was demonstrated, according to the experimental results obtained in [80] and [81].

In a more related paper, 2D LDA was applied in [38] for feature extraction of four different hyperspectral datasets namely Indian Pines, Pavia University, Kennedy Space Center (KSC) and Botswana hyperspectral datasets. In [38], each of the feature vectors in the data was converted into a feature matrix to overcome the influence of small sample size on classification results. However, the authors in [38] did not consider the concept of 'folding of the pixel', which was introduced for PCA in [37]. Hence, unfolding of the projected samples was also not considered. Instead, the eigenvectors were combined into a single projection vector using a weighted sum. By using a single projection vector, the number of features that can be extracted by 2D LDA becomes limited to the number of columns in the feature matrix. Also, the performance of 2D LDA was not compared with that of the Full Spectral Bands (FSBs). It is therefore difficult to verify the effectiveness of 2D LDA in solving the small sample size problem since feature extraction and data reduction of hyperspectral data are aimed at enhancing the performance of classifiers on FSBs. Finally, computational complexity analysis and experiments to illustrate how the feature vector to matrix conversion can be limited were not performed.

## 3.2. A Review of Computer Vision Based Approaches for Classification Tasks with A Focus on the Potential of Hyperspectral Imaging Data in Rice Seed Classification

While hyperspectral imaging data continues to be limited by its high dimensional nature, it has become very useful in classification of Agri-tech products, specifically rice seeds [39], using computer vision based approaches. This section will review the use of computer vision based approaches for classification tasks with a focus on the potential of hyperspectral imaging data when applied in rice seed classification.

Computer vision systems have been used in a range of food quality assessment applications [82], [83], [12], [21]. Recent research has focused on creating new techniques to automatically inspect and assess food quality by combining image analysis and machine learning techniques. The paper in [84] explored different rice seeds (polished) classification and quality control tasks. Specifically, the paper in [84] presents a comprehensive survey of various computer vision techniques, physical property measurements, compound content and distributions of rice grains for seed quality control.

Classification is an important task in rice seed quality control and assessment process [15]. Rice seed classification using computer vision systems are usually implemented in several key steps, some of which are image data collection, data dimensionality reduction and feature representations via models using pattern recognition algorithms or multivariate analysis techniques. Appearance-based approaches usually apply one or combination of the following features: morphological, colour, and textural. Table 3.1 presents a summary of existing techniques used by various authors for rice seed classification using computer vision based systems.

Table 3.1 A Survey on Rice Seed Classification Techniques Using Computer Vision Approaches

| Ref. | Number of Varieties | Extracted Features | Sensing modality | DRT | Classifier | Performance | Dataset Public | Year |
|---|---|---|---|---|---|---|---|---|
| Huang et al. [85] | 3 | Shape-based | RGB | - | Back-propagation neural network (BPNN) | 95.56% | No | 2017 |
| Hong et al. [59] | 6 | Morphological, colour, texture, GIST, and SIFT | RGB | - | RF, SVM | 90.54% | No | 2015 |
| Lui et al. [86] | 6 | Colour, and morphological | RGB | PCA | Neural network | 84.33% | No | 2005 |
| Wang et al. [15] | 3 | Spectral and morphological | HSI | PCA | PCA, back-propagation neural network (BPNN) | 89.18% - 94.45% | No | 2014 |
| Vu et al. [9] | 6 | Spectral and morphological | HSI | PCA | RF, SVM | 84% | No | 2016 |
| Kuo et al. [32] | 30 | Morphological, colour, and texture | RGB | - | Sparse coding | 89.1% | No | 2016 |
| OuYang et al. [87] | 5 | Colour | RGB | - | Back-propagation neural network (BPNN) | 93.66% | No | 2010 |
| Aznan et al. [88] | 5 | Morphological | RGB | - | Discriminant function analysis | 96% | No | 2016 |
| Pazoki et al. [89] | 5 | Colour, morphological, and shape | RGB | UTA algorithm | Multi-layer perceptron, neuro-fuzzy neural networks | 98.40% - 99.73 % | No | 2014 |
| Sun et al. [10] | 4 | Spectral, texture and morphological | HSI | PCA | SVM | 91.67% | No | 2015 |
| Singh et al. [90] | 4 | Colour, texture, and wavelet | RGB | Mean feature vector similarity | Back-propagation neural network (BPNN) | 96.25 - 100% | No | 2016 |
| Shwetank et al. [16] | 5 | Spectral | HSI | PCA, Segmented PCA | SAM | 82.61 % | No | 2010 |
| Liu et al. [91] | 2 | Spectral | HSI | PCA | probabilistic neural network (PNN) | 100% | No | 2011 |
| Gilanie et al. [92] | 7 | Deep | RGB | - | CNN | 100% | No | 2021 |
| Kiratiratanapruk [22] | 14 | Shape-based, colour, and texture | RGB | PCA | Logistic Regression (LR), Linear Discrimination Analysis (LDA), k-Nearest Neighbours (KNN), SVM and CNN | 95.14% | No | 2020 |
| Joshi et al. [93] | 4,7 | Deep | Optical Coherence Tomography (OCT) | - | CNN | 89.6% | No | 2021 |
| Sun et al. [94] | 5 | Spectral, colour, texture, and morphological | HSI | Bootstrapping soft shrinkage | SVM | 99.44% | No | 2021 |
| Yong et al. [95] | 2 | Spectral, and deep | HSI | PCA | CNN, Residual Neural Network (RNN), Partial Least Squares Discrimination Analysis (PLS-DA), SVM | 99.50% | No | 2020 |

Lai et al. [96] manually classified cereal grains. Seven types of grain namely corn, soybeans, sorghum, white rice, brown rice, wheat and barley were selected by the authors in [96] for the classification tasks. Basic physical parameters (area, perimeter, width, length, ferret diameters, projected height, projected width, volume, and convex perimeter) measured using image analysis were initially employed by the authors to create patterns for each of the grains. After finding out that the measured physical parameters were not effective enough in classifying the grains, different shape-based functions were then derived from the basic physical parameters by the authors to enhance the process. For instance, aspect ratio ($\frac{length}{width}$), $\frac{area}{volume}$, and circularity were adopted by the authors to define the pattern for sorghum to differentiate them from other types of grains. Convex perimeter and $\frac{area}{volume}$ were selected as the criteria for identifying brown rice. The pattern for differentiating soybeans from other types of grains were defined using area and circularity, as selected by the authors. To differentiate corn, white rice, wheat and barley from other types of grains, two, four, three and nine criteria were adopted respectively.

Similarly, Sakai et al. [97] manually differentiated brown rice from polished rice grains. The authors created a population of polished rice grains by polishing a selected number of brown rice using polishing machines. For the classification task, the authors extracted physical dimensions and shape functions of 4 varieties of rice grains using two-dimensional image analysis. The dimensions extracted are area, perimeter, maximum length and maximum width while compactness and elongation are the shape functions which were derived from the dimensions. For instance, elongation was computed by the authors as $\frac{maximum\ length}{maximum\ width}$. By using the extracted

dimensions and the shape functions as criteria, the authors in [97] achieved 95.4% as the probability of discriminating between brown rice and polished rice.

More recent papers [59], [86] and [87] focused on automatic classification of rice seeds. The authors in [59] extracted colour, texture, GIST, SIFT and shape based (area, length, width, $\frac{length}{width}$, major axis length, minor axis length, convex hull area and convex hull perimeter) features to classify 6 varieties of rice seeds using Random Forests, K-Nearest Neighbours and Support Vector Machines models. In [59], the Random Forests model gave the highest average accuracy of 90.54%. The authors in [86] extracted colour and morphological (area, length, width, major axis length, minor axis length, thinness ratio, aspect ratio, etc.) features to classify 6 varieties of rice seeds. Due to high correlation among the extracted features, the authors in [86] selected 17 out of the 21 features extracted. The feature selection process in [86] was based on correlation coefficients. PCA was applied on the rice seed data by the authors to further reduce the remaining 17 features to 4 principal components. Output of the PCA in [86] was then used to train a neural network model for classification of the rice seeds. The authors in [86] achieved an average classification accuracy of 84.33%. In a quest for automatic classification of rice seeds using appearance-based features of the seeds, the authors in [87] extracted colour and basic shape-based (length, area, etc.) features of 5 varieties of rice seeds and used the extracted features to train a neural network and achieved a classification accuracy of 93.66%.

Huang et al. [85] analysed shape descriptors which go beyond the commonly applied features in the literature namely, chaff tip (width, height), depth of concavities of rice kernels, etc. The paper in [85] achieved a classification accuracy of 95.56% when discriminating similar rice seed varieties using a neural network. The authors

also compared the performance of the neural network in [85] with that of a Bayes classifier and they both achieved comparable results. However, evaluation in [85] is limited to only 3 varieties. Kuo et al. [32] extracted and employed morphological, colour, and textural features to classify 30 varieties of rice seeds using multi-focus image fusion and sparse representation classification. The authors in [32] achieved an accuracy of 89.1%. Although, the authors in [32] briefly acknowledged the use of limited number of rice seed species by majority of the literature, they did not illustrate how this would affect the discrimination ability of classifiers. Instead, their paper focused on detailed region of interest (e.g., sterile lemmas) on the grains.

Recently, HSI data have been applied and produced promising classification performance in food and agriculture engineering. Wang et al. [15] extracted spectral information in the VIS/NIR range of 400-1000 nm to classify 3 rice seed varieties. The authors in [15] combined the extracted spectral features with the degree of chalkiness and shape-based features which were also extracted from the acquired HSI images. PCA was applied on the spectral data to reduce its dimensionality and the extracted principal components were used to train a neural network which attained a classification accuracy of 94.45%.

In [98], the authors showed that combining Least Squares Support Vector Machine (LS-SVM) regression method and VIS/NIR spectroscopy at a range of 325-1075 nm can be successfully applied to monitor nitrogen status in rice. More recently, the authors in [8] used HSI data to identify four rice seed cultivars. The authors in [8] utilised full spectral bands in the range 1,039-1,612 nm, and achieved a classification accuracy of up to 100% using a Random Forest (RF) classifier. However, the authors

in [8] hybridized four cultivars from other species to attain that level of performance, hence, it is not clear how the inter/intra class varies among the cultivars.

In a search for optimal feature combination, the authors in [9] and [10] extracted the following different feature combination schemes: spectral and texture features; morphological, texture and spectral features; and morphological and texture features. The authors in [9] applied PCA on a combination of spectral and spatial features extracted from a dataset of 6 rice seed species and used the output of PCA to train Random Forest and Support Vector Machines models. The authors in [9] observed that the Random Forest model achieved the best precision of 84%. The authors in [10] also applied PCA on a combination of spectral, texture and morphological features extracted from a dataset of 4 rice seed species and used the extracted features to train a Support Vector Machines model. The author in [10] observed that the combination of spectral, morphological and textural features (classification accuracy of 91.67%, with four polished rice species) gave the best classification results.

Finally, it is worth noting that Table 3.1 includes new publications, which are related to the current study and some of which have cited [34] (which is the result of work that will be presented in Chapter 4 of this thesis).

## 3.3. A Review of Dimensionality Reduction Techniques for Hyperspectral Imaging Data When Applied in Classification of Agri-tech Products (With a Special Focus on Rice Seeds)

Hyperspectral imaging systems have produced very promising results when applied in the classification of Agri-tech products such as tea [21], lamb [13], and more specifically rice seeds [15].

In [21], a maximum likelihood classifier and artificial neural network were used to classify 5 types of tea samples using hyperspectral imaging data which were acquired at a visible light range of 400 – 800 nm. PCA was used for dimensionality reduction of the data prior to classification. The authors in [21] achieved promising classification results and also observed that artificial neural networks performed better than maximum likelihood classifiers.

The authors in [13] classified 3 types of lamb muscles using hyperspectral imaging data which was acquired at a NIR range of 900 – 1700 nm. PCA was applied on the data in [13] to reduce its dimensions. Optimal spectral features were selected in [13] by analysing PCA loadings and finding wavelengths which correspond to the two most significant principal components. Selected features were presented to a Linear Discriminant Analysis model for classification. Overall classification accuracy of 100% achieved by the authors in [13] demonstrates the potential of hyperspectral imaging in lamb muscles classification.

A more related paper [15] trained an artificial neural network to classify 3 varieties of rice seeds using hyperspectral images which were acquired at a VIS/NIR range of 400-1000 nm. PCA was used by the authors in [15] for dimensionality

reduction of the spectral features extracted from the hyperspectral data. The degree of chalkiness and shape-based features, which were also extracted from the acquired hyperspectral images, were fused with the outputs of PCA to train the classification model which attained promising classification results (accuracy of 94.45%).

However, hyperspectral imaging still faces some challenges that continue to limit its potential for classification and quality inspection of rice seeds. One of these limitations is the presence of very high number of spectral features, usually in hundreds, in the acquired hyperspectral images of rice seeds. Conventional classification models suffer from curse of dimensionality due to the presence of a large number of features, and this degrades classifier performance [17],[16],[18]. Hence, the development of innovative data dimensionality reduction techniques for redundant data removal while retaining important information becomes imperative for enhanced classification performance.

PCA and LDA are two well-known tools for dimensionality reduction of hyperspectral imaging data [99],[73]. PCA is a conventional technique used for the dimensionality reduction of data [16] and is noted from the summary of the literature review into dimensionality reduction techniques presented in Table 3.1 to be the most applied for dimensionality reduction in HSI-based rice seed classifications. The paper in [64] which will form a part of Chapter 4 in this thesis carried out a comparative analysis of the performance of LDA and PCA and observed that LDA can perform better, as a dimensionality reduction tool, than PCA for hyperspectral imaging data.

Genetic Algorithms (GAs), are another dimensionality reduction tool which function as a wrapper-based feature selector. Like any other wrapper-based feature

selectors, GAs analyse the relationship between the entire feature sets and classification models to select an optimal subset of features for increased classification performance [100], [101]. GAs have been applied for feature selection and reduction and produced very promising results in a host of applications.

GA was applied in [100] for feature selection in a spam detection application. Features which were relevant for spam detection were selected by the GA to solve the problem of high number of features (attributes) which characterize spam messages. The outputs of the GA were then presented to a random weight network model for classification. Promising classification results were obtained by the authors in [100] and demonstrated the potential of feature selection using GA in spam detection.

In [101], GA was used to select an optimal feature subset from a set of features which were extracted from a vowel speech signal. The selected features were presented to a KNN classifier to classify Turkish vowels. The authors in [101] achieved a classification accuracy of 100% which demonstrates the potential of GA as a feature selector in vowel classification.

The authors in [102] applied GA as a part of a two-stage gene selection approach for cancer data classification. The first stage involved selection of genes which contain cancer-related information. This was followed by selection of an optimal gene subset from many subsets of the genes which were selected in the first stage. Outputs of the second stage was then presented to a SVM model for classification. The authors in [102] observed that the proposed two-stage approach achieved the highest classification accuracy when compared with other gene selection

approaches (clustered gene selection, non-clustered gene selection, GA gene selection and mutual information gene selection).

It is noted that none of the techniques reported in Table 3.1 utilised GA for feature reduction when designing dimensionality reduction techniques for automated rice seed inspection using HSI data. It is also noted from the review of the literature that very promising results were produced by the hybridization of GA and PCA [28],[29], and GA and LDA [30],[31] in other applications.

In [28], GA and PCA were combined for dimensionality reduction of ear biometrics data. The authors in [28] used GA to select an optimal feature subset in the data and PCA for feature extraction and further dimensionality reduction of the selected subset. The combined GA and PCA achieved accuracy which are comparable with those of PCA and the full feature system with reduced feature size.

GA and PCA were also combined in [29] to reduce the dimensions of a time series data for human activity recognition using a 2-level Hidden Markov Model (HMM) classifier. As in [28], selection of optimal features was followed by feature extraction using GA and PCA respectively. The combined GA and PCA achieved the highest classification accuracy in [29] when its performance was compared with that of PCA and the full feature system.

The paper in [30] combined GA and LDA for dimensionality reduction in image retrieval systems using K-Nearest Neighbour model (for similarity measure between the features of an image database and that of the image that is being retrieved). GA was applied to select optimal colour and textural features from dermatological images. Features selected by the GA were then fed to LDA for feature extraction and

further dimensionality reduction of the data. When compared with other techniques like LDA, PCA and Independent Components Analysis (ICA) in [30], the combined GA and LDA gave the best performance which demonstrate its effectiveness in reducing the dimensions of the data for image retrieval.

Similar to the paper in [30] is the one carried out by the authors in [31] where dimensionality of hyperspectral imaging and Synthetic Aperture Radar data was reduced using the combination of GA and LDA. Promising results achieved by the authors in [31] further demonstrate the effectiveness of optimal feature selection using GA prior to feature extraction using LDA.

### 3.4. Summary, Discussion and Proposed Contributions

This chapter presented the review of publications which are related to the current study. Firstly, this chapter reviewed the various techniques which were used in the literature for dimensionality reduction of hyperspectral imaging data in classification applications. It was noted in the review that despite the great potential shown by deep learning models in classifying hyperspectral imaging data, such models are still limited by their requirements for complex parameter tuning. It was also noted that, unlike deep learning models, traditional machine learning classifiers require simple parameter tuning and a separate data dimensionality reduction tools (deep learning models incorporate these tools). Hence, the review focused more on dimensionality reduction tools applied on hyperspectral data prior to classification using traditional machine learning models. Specifically, the review noted the limited use of LDA in reducing the dimensionality of hyperspectral data (which is characterised by the presence of limited samples for training) due to the below par performance given by LDA when applied on small data samples and the limit on the number of features that can be extracted by

LDA. 2D LDA was then noted to have been applied to solve the problem of small sample size in a recent paper [38] where spectral vectors in the hyperspectral data were converted to matrices and data projection was performed using a single projection vector. These continue to limit the number of features that can be extracted.

In Chapter 5 of this thesis, a folded LDA (F-LDA), an extension and improved version of LDA, will be proposed for dimensionality of hyperspectral data. The proposed approach is motivated by the concept of 'folding the pixels' which was introduced for PCA in a related paper [37], where improved classification performance was achieved and noted in the review. By folding each pixel in the hyperspectral data, the proposed F-LDA can focus on each spectral vector and will be able to extract the local features within the data [75],[37]. It is expected that the proposed F-LDA will outperform the conventional LDA in terms of classification accuracy. It is also expected that reduced computational complexity and memory requirement will be achieved due to the size of the various matrices, which will be processed and stored at different stages of the proposed F-LDA, will be smaller than those in the conventional LDA. While the proposed F-LDA shares the concept of feature vector – feature matrix conversion with the 2D LDA, it is further expected that the proposed F-LDA will attain higher classification accuracy than the 2D LDA because of the individual treatment of the eigenvectors and unfolding of the projected samples in the proposed F-LDA.

Secondly, this chapter also reviewed various computer vision based techniques used in the literature with a special focus on the potential of hyperspectral imaging data in rice seed classification. It was noted that many papers utilised spatial information extracted from RGB images to classify rice seeds. Some more recent papers were also noted to have utilised spectral information (and in some cases, spectral and spatial

information) extracted from hyperspectral data for rice seed classification and achieved promising results. Some research gaps can also be noted from the review of existing rice seed data classification techniques. Firstly, the performance of related techniques presented in Table 3.1 varies significantly and there are no common datasets to compare or benchmark the performance of those techniques with one another. Secondly, although the classification performance attained in those studies are promising, most authors used a relatively limited number of different rice seed species to evaluate performance of the related techniques. In practical applications, the presence of different rice seed varieties poses a challenge to automated system. For instance, the authors in [33] observed the presence of a very large number of rice accessions (120,000+ accessions) in TT Chang Genetic Resources Centre at the International Rice Research Institute (IRRI), Philippines. They argued that shape-based features of cultivated rice accessions can be helpful in new rice seed authentication. Kuo et al. [32] also observed the cultivation of rice grains of hundreds of varieties. These make the proposition of robust non-destructive approaches to authenticate large number of rice seed varieties imperative. In this thesis, the rice seed dataset which consists of ninety rice seed varieties will be utilised. The dataset of 90 seed varieties is also made publicly available. Both the high spatial resolution and spectral information are essential when developing and deploying robust classification models and will be exploited for rice seed species classification (an important step in automated rice seed screening exercises for quality control and inspection).

Finally, this chapter reviewed various techniques which are used for dimensionality reduction of hyperspectral imaging data when applied for classification of Agri-tech products with a special focus on rice seeds. It was noted that PCA is a more commonly

applied dimensionality reduction tool for hyperspectral data than LDA and that LDA can perform better, as a dimensionality reduction tool, than PCA for hyperspectral imaging data as will be shown later in chapter 4 [64]. The paper in [64], which presents a comparative analysis of the performance of LDA and PCA on hyperspectral data of rice seeds, will form a part of Chapter 4 in this thesis. The review also noted the promising results achieved by GA in many applications as an optimal feature selection technique and in papers where GA was used to select optimal features prior to feature extraction using PCA in [28],[29], and LDA in [30],[31] for further data dimensionality. No attempt was made in any of the related papers in Table 3.1 to apply the hybridized approach for dimensionality reduction in HSI-based rice seed classification. This conceptual gap coupled with the LDA's superiority to PCA as a dimensionality reduction technique [64] therefore inspires the hybridization of GA and F-LDA (GA+F-LDA) which will be proposed for dimensionality reduction of hyperspectral imaging data in Chapter 6.

## 4. Hyperspectral Imaging Data Classification: Evaluating the Effectiveness of Combining Spectral Features from Hyperspectral Images and Spatial Features from RGB Images

Chapter 2 reviewed various computer vision-based techniques which were proposed in the literature for classification. It can be noted from the review of related techniques presented in Chapter 2 that several approaches relied on the use of appearance-based information (such as morphological, textural or colour features) extracted from RGB images to train machine learning models for data classification [103],[59],[86],[33]. It can also be noted from the review that recent approaches focused more on the use of an alternative sensing modality named, hyperspectral imaging which offers simultaneous extraction of both the appearance-based and spectra information [15],[82],[9] to enhance the performance of machine learning models [8]. However, hyperspectral imaging, when compared to RGB imaging, provides a decreased pixel density due to spatial binning and to enhance robustness of acquired data [19],[20]. Consequently, fidelity in the appearance-based features is reduced especially when small objects such as rice seeds are being discriminated. This downside of hyperspectral imaging is addressed in this chapter by evaluating the performance of a system that combines spatial (appearance based) information extracted from high spatial resolution images (RGB) and spectral information extracted from high spectral resolution images (hyperspectral) to enhance the classification process.

Furthermore, while the application of hyperspectral imaging for data classification can produce very promising results [15],[10], the high number of spectral features which are present in acquired hyperspectral images is responsible for the problem of curse of dimensionality which degrades the performance of classification models. This limits

the potential of hyperspectral imaging data in classification. It is therefore necessary to reduce the number of features by removing redundant and noisy information from the hyperspectral data with minimal loss through the application of dimensionality reduction techniques. From the review of literature presented in Chapter 3, it can be noted that Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are examples of data dimensionality reduction techniques and that PCA is a more commonly applied dimensionality reduction technique for hyperspectral imaging data than LDA [15],[21],[10],[13]. This chapter therefore also aims to carry out a comparative analysis of the performance of PCA and LDA when applied to reduce the dimension of spectral data in classification applications.

It is also worth noting, as established in Chapter 1 and 2, that classification using computer vision-based approaches has become very useful in rice seed screening exercises. Such exercises, which can be carried out manually by experts using visual inspection, is aimed at removing weeds and off-types from the seed batch. There is a continued need to enhance the screening process by employing automatic systems to reduce the time and efforts required for visual inspection [15],[22],[23]. Hence, the performance of the approach proposed in this chapter is evaluated on a dataset of rice seeds. From the review of related papers on rice seeds presented in Chapter 2, it can be noted that most of the approaches presented in those papers are implemented using RGB images. It can be further noted that performance of most of the approaches are evaluated using only a small number of different rice seed species with varying degrees of accuracy achieved. It is therefore not clear whether superiority of algorithms employed, effectiveness of feature descriptors used to train the models, or, differences in the inter-class or intra-class variation of species used in each paper is responsible

for differences in classification performance among existing techniques. Nevertheless, Kue et al. [32] and Peralta et al. [33] evaluated the performance of their methods using 30 and 754 species (relatively large number of species) and achieved accuracy of up to 89.1% and 44.87% respectively. The performance reported by the authors in [32] and [33] are not as strong as those reported in some of the other methods in the related papers. This lends credence to the hypothesis that other methods do not necessarily used superior algorithms or more effective feature descriptors. Instead, it is likely that less challenging datasets containing a smaller number of species which exhibit favourable intra-class and inter-class variation were used for performance evaluation of those methods. However, it is difficult to confirm or reject this hypothesis since the datasets used in those papers are not publicly available.

This chapter therefore makes the following contributions:

1) The effectiveness of an innovative framework for combining spatial and spectral features is evaluated and it is shown that the combined features improve discrimination ability of the selected classifier.
2) Comparative study of LDA and PCA as dimensionality reduction techniques for hyperspectral data is conducted and it is shown that LDA can perform better than PCA when applied to reduce dimensionality of spectral data.
3) Performance evaluation of the proposed approach is carried out on a large, diverse dataset of 90 rice seed species. Experimental results show that performance of classification models can be impacted by varying the number of species in the datasets. Hence, similarity assessment of species is recommended in rice seed classification.

4) The large hyperspectral dataset of rice seeds is made publicly available [34],[35] to the community to assist in the benchmarking of the proposed approach and feature combination. The rice seed dataset, which has been downloaded 6,205 times as at 27th June 2022, was made available at an online research repository named Zenodo [36] in January 2020. However, the information available at Zenodo [36] shows that the dataset is yet to be cited.

### 4.1. System Setup

HSI and RGB images of the rice seeds were captured during a research exchange funded by the British Council and Newton Fund (grant number: NRCP1516/1/65). All images (HSI and RGB) were gathered in University of Strathclyde's Hyperspectral Imaging Centre by Dr Hai Vu and Dr Dao Trung Kien of Hanoi University of Science and Technology - co-authors on [34] as follows: Images of rice seeds were acquired using a system which comprised of a high resolution RGB camera and a hyperspectral imaging system. A photo of the acquisition system is shown in Figure 4.1. RGB images were collected at 4,896 × 3,264 pixels using a Fujifilm X-M1 with a 35mm/F2.0 lens as the digital camera. A hyperspectral imaging system was deployed to collect hyperspectral images at a Visible - Near Infrared (VIS/NIR) range of ~ (385 – 1000) nm. Classification models can exploit the difference in colour variation of species in the Visible region ~ (385 - 700) nm and chemical composition in the NIR region ~ (700 - 1000) nm to discriminate the seed species [23],[7],[104]. Hence, the VIS/NIR range is used in this work. The hyperspectral imaging system consisted of a Specim V10E Imaging Spectrograph and Hamamatsu ORCA-05G CCD camera. Two halogen bulbs were included in the system to provide illumination. The bulbs were accurately positioned to ensure that balanced lighting are provided across the scene. For stability,

after switching the bulbs on, they were allowed to reach constant temperature. This was done before the data can be acquired in a dark room to minimise any other sources of illumination variance. The Fujifilm RGB digital camera was operated in a manual mode with an ISO and a shutter speed of 400 and 16 ms respectively. Acquired images were saved in JPEG format with no automatic adjustment (e.g., white balance) applied. The hyperspectral imaging system is a push-broom system which captures HSI data using a line by line approach. Hence, a motorised translational stage was placed directly below the imager to allow scanning. Three important parameters of the hyperspectral imaging system were adjusted as follows:

1) The exposure time of the camera versus the speed of movement of the translational stage. This was calibrated to avoid spatial distortions;
2) A trade-off between the exposure time and the aperture of camera (f=18). This was done to ensure a suitable light intensity;
3) Setting of the height between the lens and the stage. This was done to allow the camera field of view to capture the entire area containing all seeds in each data cube.

Before the images of rice seeds were acquired, images of a flat checkerboard patterns were collected for calibrating planar and lens distortion effects for the RGB and HSI sensors. This was performed once and the rotation and transformation parameters (camera calibration parameters) were stored in a XML file for alignment and registration of the images acquired from both the RGB and hyperspectral imaging systems. The image registration, devised by Dr Hai Vu and Dr Dao Trung Kien of Hanoi University of Science and Technology along with co-authors at the University of Strathclyde on [34], was performed using the following method: To register (warp)

the images from RGB to HSI plane, firstly, the corner points in chessboard images collected by HSI and RGB cameras were detected using harris corners operator (detector). Affine transformation was then applied on both images to determine the scale, translation and rotation matrices. The HSI images and the RGB images of rice seeds were multiplied using the affine transformation operator (scale, rotation and translation parameters). During registration, it was necessary to align the HSI and RGB images. To achieve this, firstly, canny detector was used to get the edge of each rice seed species. Then, coordinates of the HSI images was determined. This was followed by application of the transformation matrix to align the two images. [34],[35].

### 4.2. Description and Processing of Dataset

Ninety known rice seed varieties which were provided by the National Center of Protection of New Varieties and Goods of Plants (NCPNVGGP) in Vietnam are used in this study. These rice seed varieties are frequently planted in Vietnam to cultivate rice for consumption and exportation and so were chosen for use in this work. Experienced technical staff at NCPNVGGP manually screened the selected samples in the traditional way to ensure that only seeds which belong to the 90 species to be analysed are contained in each sample population. A single kernel from each of the 90 varieties is illustrated in Figure 4.2.

96 individual rice seeds were provided for each of the 90 species considered. The 96 kernels of each of the 90 species were divided into 2 batches with each batch containing 48 individual rice seed samples. HSI system and the digital RGB camera were then deployed to acquire images of each batch of 48 seeds which was placed on a white sheet of paper in a (6 x 8) matrix structure. Hence, 2 hyperspectral data cubes and 2 high resolution RGB images (each containing 48 different seeds) of each of the

90 species of rice seeds were captured. In total, the dataset is made up of fully registered images (RGB and hyperspectral) of 8640 seeds i.e. 90 varieties x 96 seeds. In this study, manual positioning of the seeds on the white sheet was done to avoid overlaps or touching boundaries between them. In practice, an arrangement of a conveyor belt where seeds are mechanically and individually spread under appropriate and consistent illumination can be conceived for imaging.

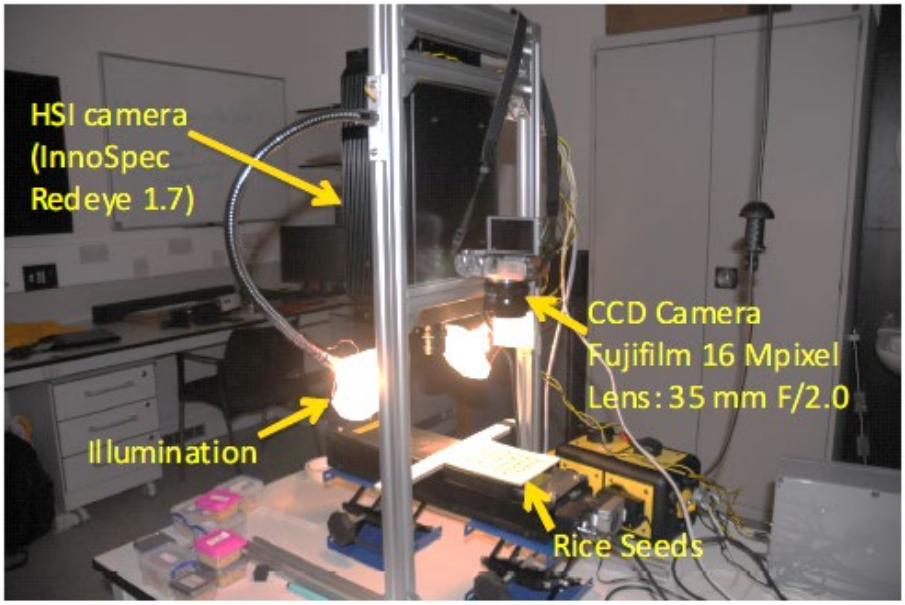



Figure 4.1 A photo of the HSI and CCD cameras setup for data acquisition.

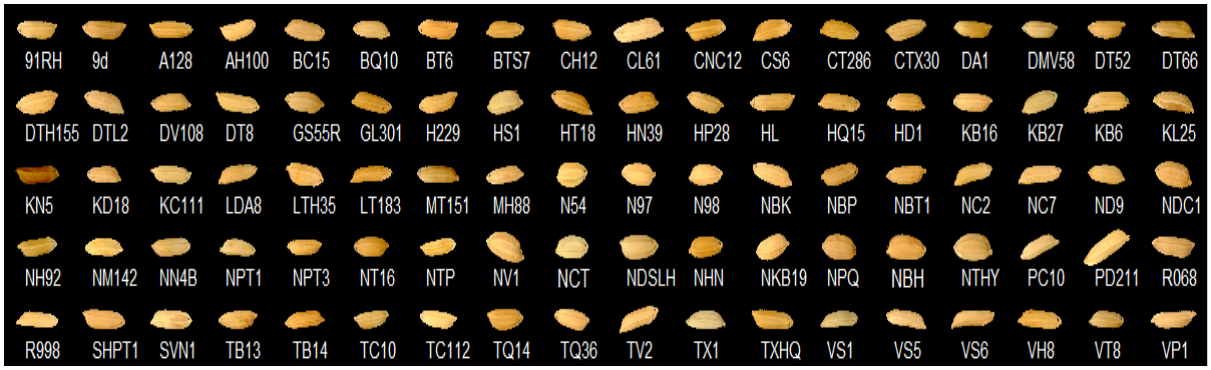



Figure 4.2 Photos of rice seed samples of 90 species. The short name of each species is given beneath each kernel. © 2020 IEEE

The processing steps used for spatial and spectral features extraction is illustrated in Figure 4.3. Initially, rotation and transformation matrix obtained from the checkerboard pattern (as described in Section 4.1) are used to calibrate both image modalities for lens and planar effects. Normalisation of the HSI data cube is then performed as described in Section 4.4. After this, the processing paths take a slightly different direction. Segmentation of the RGB data is performed using the process described in Section 4.3. High resolution spatial features are then extracted using binary masks for each rice seed in the RGB image as described in Section 4.4.

To segment the rice seeds in the HSI data, transformation of the masks obtained from the RGB segmentation to the HSI space is then performed using the calibration matrix obtained from imaging the checkboard pattern. Spectral features are subsequently extracted using the segmented seeds as described in Section 4.5.

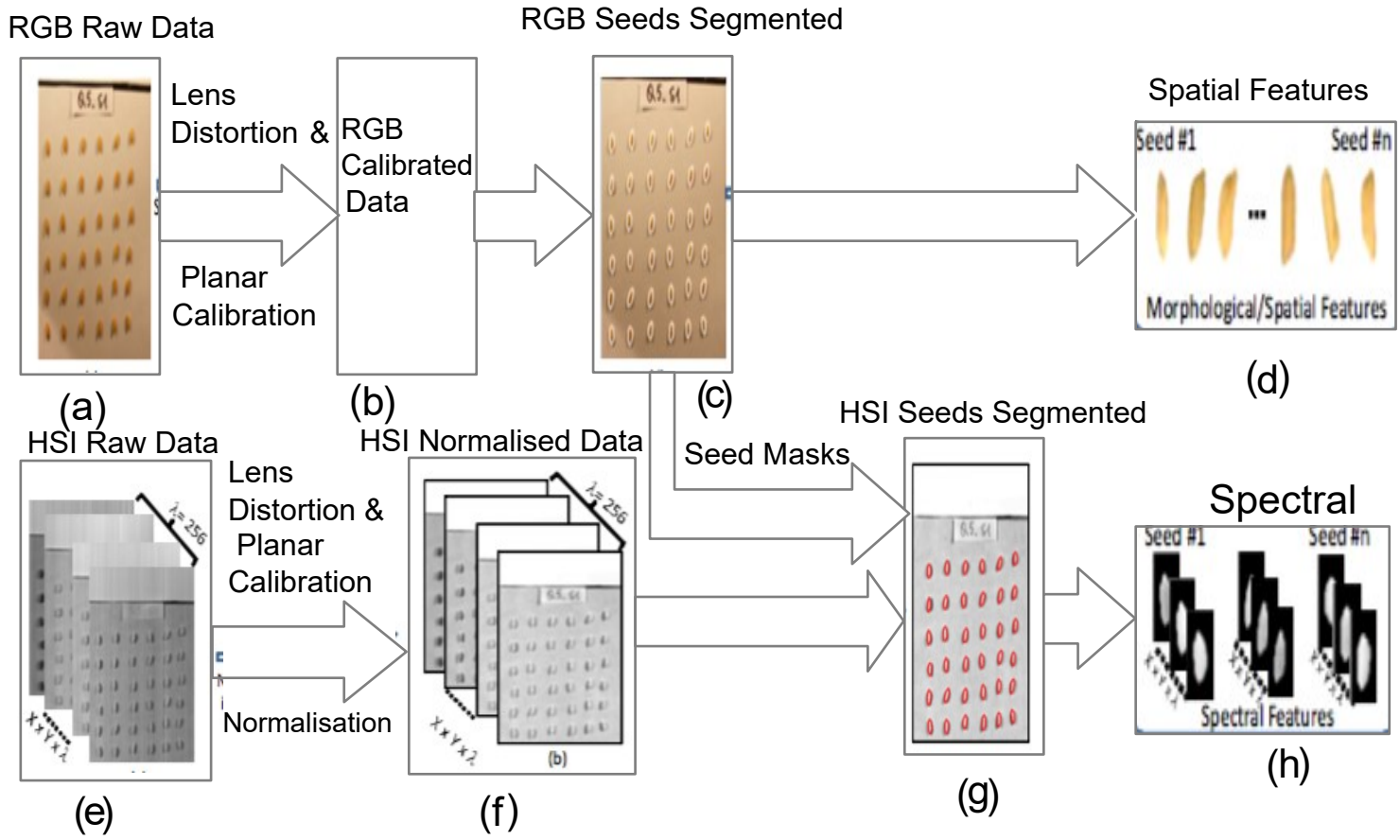



Figure 4.3 A diagram showing the stages of the data processing task: (a) Raw RGB image obtained by the Fujifilm digital camera. (b) Lens distortion and planer camera calibration. (c) Rice seed segmentation and mask extraction. (d) Computation of spatial features for each of the rice seed masks. (e) Raw HSI data. (f) HSI normalisation, lens distortion and planer camera calibration. (g) The rice seed masks from step (c) are translated to the HSI plane and used to segment the rice seeds in the HSI image. (h) Computation of the spectral features.

### 4.3. Calibration Procedures and Segmentation of Rice Seeds

The rice seed image segmentation, devised by Dr Hai Vu and Dr Dao Trung Kien of Hanoi University of Science and Technology along with co-authors at the University of Strathclyde on [34], was performed as follows: The rice seed is segmented on the high spatial resolution RGB images to ensure that the kernel is completely captured. The proposed procedure is implemented in the following steps:

(1) Extraction of the R-channel of the RGB image; the R-Channel is selected since it offers the highest contrast to the background.

(2) A Morphological opening [105] is applied to the R-channel image to generate a background image. Subtraction of the generated background image from the R-channel is then performed. This is followed by the application of a thresholding operator to obtain the binary images of seeds [106],[107].

### 4.4. Extraction of Spatial Features

Trained personnel manually screen rice seeds by analysing their spatial features. In this study, spatial features of rice seeds are extracted from RGB image masks to capture the know-how of the trained personnel. The features extracted are selected due to their effectiveness in species discrimination, as shown in related papers, [59] and [86]. The following six parameters are computed for a single rice seed kernel to set up a morphological feature vector $f$:

1) $f_1$ (Area) : refers to the number of pixels inside a rice seed kernel
2) $f_2$, $f_3$ (Major Axis Length and Minor Axis Length) : refer to the length (in pixels) of the major and minor axis of the ellipse that has the same normalised second central moment at the region of the sample seed

3) $f_4$ (Aspect Ratio) : refers to the ratio of Minor Axis Length over the Major Axis length i.e $(\frac{f_3}{f_2})$

4) $f_5$: refers to the Perimeter over Area Ratio $\frac{Perimter}{Area}$, where Perimeter is the number of pixels along the seed boundary; and Area is the same as $f_1$

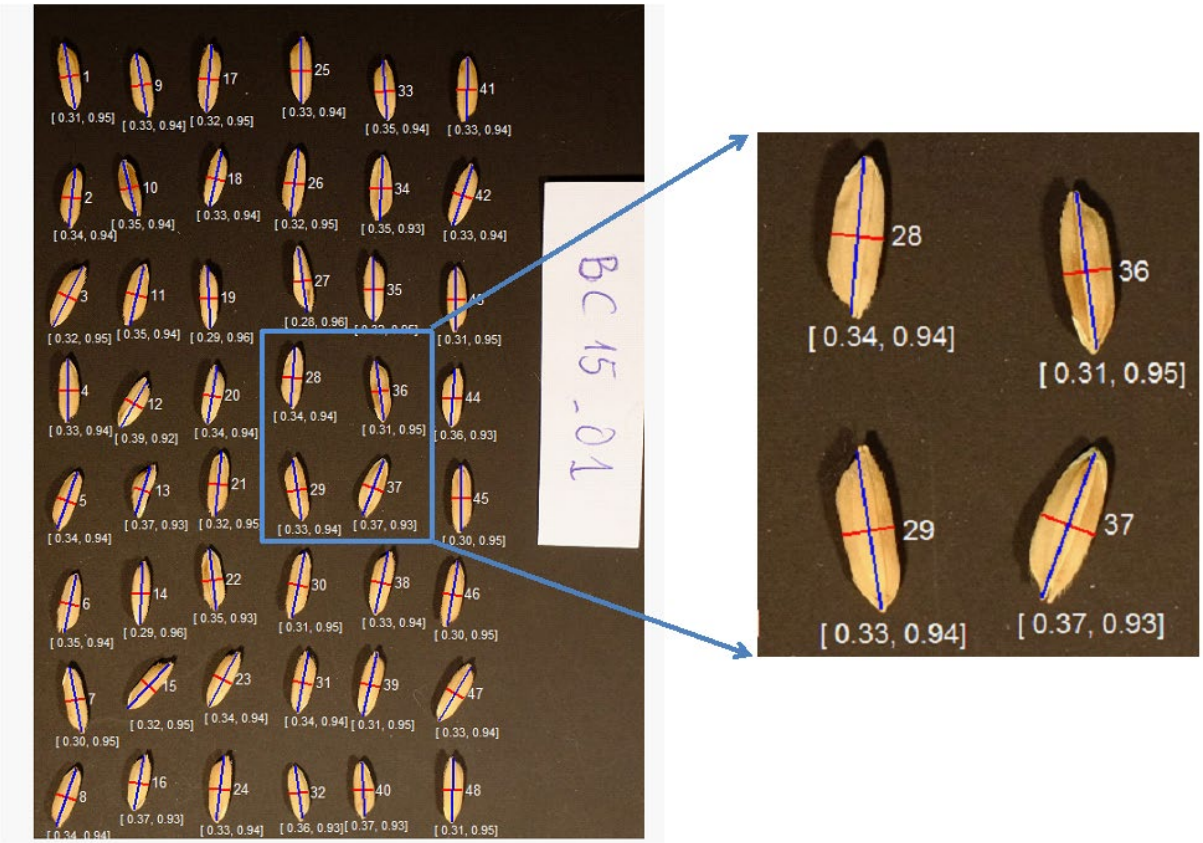



Figure 4.4 Results of the spatial feature extraction. The identification number of each seed is shown to its right. For illustration purposes, only features $f_4$ and $f_6$ are shown below each kernel. The MajorAxisLength $f_2$ and MinorAxisLength $f_3$ are marked with a red and a blue line, respectively. © 2020 IEEE

5) $f_6$ (Eccentricity): refers to the distance between two foci of the ellipse, and the major axis length of the ellipse (i.e. feature) i.e. $\frac{FociDistance}{MajorAxisLength}$ .

Figure 4.4 illustrates results of the spatial feature extraction process using a sample image containing 48 rice seeds. As shown in Figure 4.4, the spatial features are expected to be more accurate than those reported in [9], [10] (where they were extracted from the HSI system which has lower spatial resolution). This is because they have been extracted from the high spatial resolution images.

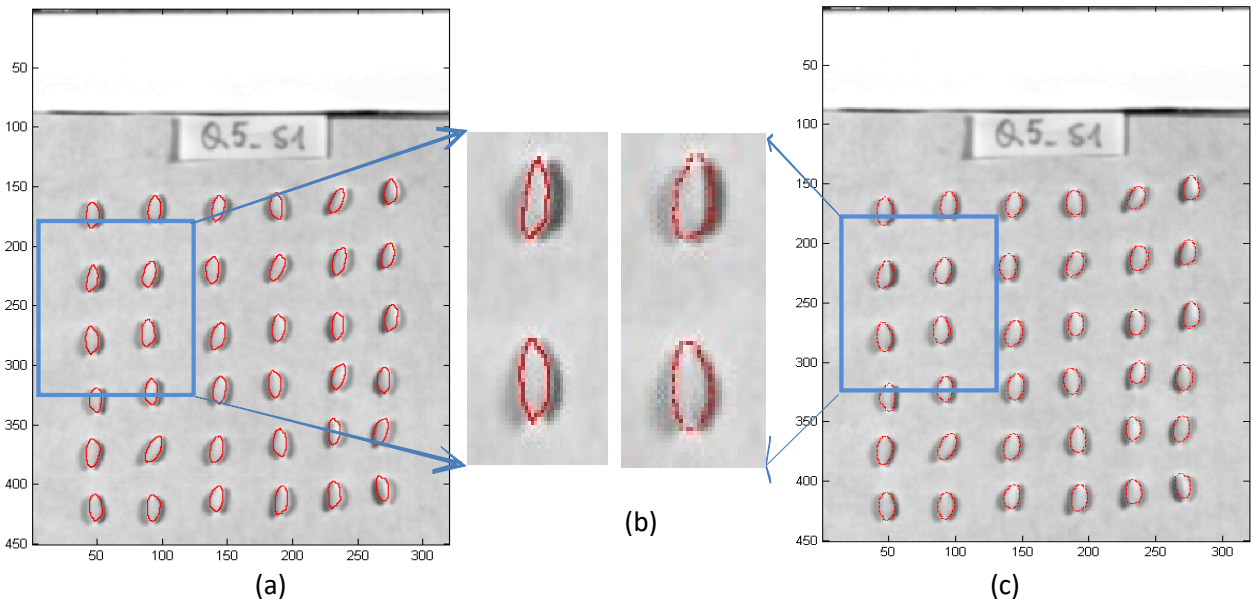



Figure 4.5 Comparing segmentation results of rice seeds on HSI images. (a) Using RGB images for reference (b) A Zoom-in version of two segmentation results (Red and white points are the results from (a) and (c) respectively) (c) Without using RGB images. © 2020 IEEE

## 4.5. Extraction of Spectral Feature Extraction

The process of extracting spectral information from the data can be divided into the following two key stages (which are illustrated in Figure 4.3): data correction, and feature extraction. Firstly, correction of the collected data is carried out following the approach described in [108] to reduce the variation in the acquired reflectance values

among measurements. Let $y$ represents a hyperspectral data cube which consist of reflectance values as a two-parameter set:

$$y_w(x), x \in X, w \in \Lambda \quad (4.1)$$

where $w$ represents a wavelength which belong to $\Lambda$ (a set of wavelengths) and $x$ represents a pixel in $I$ where $I$ is a 2D image (of row $k$ and column $l$). Hence, the array of reflectance values at each wavelength can be represented as an image where relationships between the pixel reflectance values in the spatial domain have meanings. Different lighting conditions could cause the raw reflectance value at each $x$ to vary. In order to reduce the variation of the acquired reflectance values, the data was scaled in relation to a known max reflectance value; (4.2) is used to normalise the raw data:

$$y_w(x) := \frac{y_{raw,w}(x) - b(n,w)}{w_r(n,w) - b(n,w)}, w \in \Lambda \quad (4.2)$$

where $b(n,w)$ and $w_r(n,w)$ are the reflectance values of reference dark and white images respectively. The lens-cap was covered to acquire the dark reference images. For the white reference, it is a 100% reflective spectralon tile. This is a highly reflective Lambertian scatter which is commonly used for HSI systems calibration. The average of $b(n)$ and $w_r(n)$ on reflectance values at column $n$ along the height dimension of the white tile is computed for each wavelength ($w$). After the spectral data is normalised, transformation of the rice seed masks (obtained from the above description of the rice seed segmentation procedure) to the HSI plane is performed using a transformation matrix (including the rotation and translation operators) to segment the seed samples on the spectral images, as illustrated in Figure 4.3. The advantages of using the proposed segmentation method compared to the conventional work (e.g., in [9], [10] where the authors of those papers directly extracted the seed segments from the low

spatial resolution images acquired by HSI system) is clearly illustrated in Figure 4.5. When the seed segmentation is performed using only the HSI data, in many cases, some pixels at boundary regions of the shadow are included in the segments and not the pure spectra of the seeds themselves. Consequently, extracting the morphological features using only spectral image segmentation could become inaccurate and cause the inclusion of spectra of non-rice-seed pixels in the analysis. Hence, in this study, segmentation of the seeds is performed from the high resolution RGB images. This will ensure the correct inclusion of both the spatial and spectral features in the analysis.

Spectral data from every pixel of the seed regions is extracted using the segmented seed samples on a hyperspectral data. The spectrum of all the pixels in each seed is averaged and the result obtained is used to determine the spectral features for the seed. As stated in (4.2), a raw spectral feature vector in the hyperspectral data cube is a set of $y_w$ where $w$ is one of the 256 bands which belong to $\Lambda$ (a set of spectral wavelengths or bands) at a range of ~ (385 – 1000) nm for the utilised VIS/NIR hyperspectral imaging system.

### 4.6. Dimensionality Reduction

The acquired hyperspectral data contains many spectral features (originally 256). PCA and LDA are feature extraction techniques which can be used to reduce the dimensionality of data [109]. One of the objectives of the study presented in this chapter is to compare the performance of LDA with the commonly used PCA when applied on spectral data. In line with this objective, PCA and LDA are applied on the spectral data to reduce its dimension in a preliminary analysis conducted. The outputs of the PCA and the LDA are used separately to train a classifier for performance comparison of the two dimensionality reduction techniques. The technique which

gives a better performance is then selected for use at a later stage where analysis on the RGB and hyperspectral imaging system is conducted. When PCA is applied on the data, the number of principal components selected is varied starting from 1 up to $f$, where $f = 256$ is the number of features in the dataset. The number of features extracted by LDA is varied starting from 1 up to $c - 1$ where $c$ is the number of species in the dataset. For the PCA and LDA, the classification results obtained using the optimal principal components and features are reported respectively.

### 4.7. Classification

The classification of rice seed variety is aimed at detecting seeds within a batch which do not belong to the species that are expected to be in the batch. Classification models are trained to perform this task and can directly make use of the full-band wavelengths or only the outputs of feature extraction or dimensionality reduction techniques applied on the spectral data, and/or spatial features. Hence, the features extracted from the rice seed data are utilised in the following four different ways:

1) Spatial features only
2) Spectral features only
3) A combination of spectral features and spatial features
4) A combination of features extracted from the application of dimensionality reduction tool on the spectral data and the spatial features.

A Random Forest (RF) classifier, which has produced better classification results than many other classifiers including support vector machines and k-Nearest Neighbor in many related papers [9],[59],[8], is the classification model adopted for rice seed

variety classification. The number of decision trees used in the Random Forest and the ratio of training to testing samples selected are 500 and 4:1 respectively.

### 4.8. Results and Analysis

In this thesis, Precision, $P$, Recall, $R$, and $F_1$ score are selected as performance metrics to evaluate the RF classifier's performance and the effectiveness of the proposed approach.

In accordance with the objectives of this thesis, first, 100 sub-datasets which contains 256 spectral features and 20 randomly selected species was extracted from the dataset of 90 species for performance comparison of PCA and LDA as dimensionality reduction techniques for hyperspectral imaging data. Second, using the collected datasets of 90 rice seed species, analysis on the proposed fusion of spatial features extracted from RGB images and spectral features extracted from hyperspectral images are carried out and presented under the following three separate circumstances:

1) Training the RF classifier and evaluating the performance of the proposed approach using all 90 species.
2) Training the classifier separately using 6 different subsets of the dataset of 90 species. There are 6 species (greater than or similar to those used in related papers – see Chapter 3) in each of the 6 subsets which are drawn randomly from the available 90 species. This is motivated by the need to compare performance of the proposed approach to state-of-the-art techniques which are more inclined to evaluating performance of proposed techniques on datasets containing small number of species.

3) Finally, another 6 subsets (different from the ones used in the second circumstance) of the dataset of 90 species is selected by varying the number of species in the new subsets from 6 – 90. Each sub-dataset is then used to train the RF classifier. This is done to explore how the classifier's performance can be affected when the number of species in the dataset is increased.

The results obtained from the preliminary analysis (comparing the performance of PCA and LDA) and those obtained when considering three separate circumstances for the new and fused RGB and hyperspectral imaging system are reported and analysed in the following sub-sections.

### 4.8.1. Comparative Analysis of the Performance of PCA and LDA on A Sub Dataset of 20 Species

In this experiment, the RF classifier was trained using 100 sub datasets of 20 randomly selected species and 256 spectral features before the application of the two dimensionality techniques (PCA and LDA). PCA and LDA are then applied separately to reduce the dimensionality of the sub datasets. The features extracted by PCA and LDA are used to train the RF classifier to determine which of the two techniques will be more effective in reducing the dimensions of hyperspectral data. Classification results obtained from the use of raw spectral features and features extracted by PCA and LDA on the RF classifier and 100 selected sub datasets are averaged and presented in Table 4.1. From Table 4.1, it can be observed that the lowest average precision, average recall and average *F1* score were given by the RF classifier when trained using the full raw spectral features in the sub dataset. Also, from Table 4.1, it can be observed that performance of the RF classifier can be improved by applying dimensionality reduction techniques on the data prior to classification. Specifically, the RF classifier

gave higher average precision, average recall and average $F1$ scores when trained using the PCA outputs. It can be further observed that the LDA features gave the highest average precision, average recall and average $F1$ score overall. The results obtained therefore show that LDA can give better performance than PCA when used for feature extraction and dimensionality reduction of hyperspectral imaging data.

Table 4.1 Classification results obtained from the comparative analysis of the performance of PCA and LDA

| Feature Schemes | Average precision (%) | Average Recall (%) | Average $F_1$ score (%) |
|---|---|---|---|
| Spectral | 59.65 ± 5.14 | 59.90 ± 5.01 | 58.93 ± 5.11 |
| PCA outputs | 81.67 ± 3.21 | 81.69 ± 3.33 | 80.87 ± 3.41 |
| LDA outputs | **85.88 ± 3.31** | **85.90 ± 3.30** | **85.52 ± 3.35** |

### 4.8.2. Performance Analysis on All 90 Species

In this experiment, 4 different RF models are trained using the following different feature types and combinations of features extracted from the dataset of 90 species:

1) Spatial features
2) Spectral features
3) Combination of spatial features and spectral features on full bands
4) Combination of spatial features and features extracted by LDA when applied on the spectral features.

The classification results obtained are presented in Table 4.2. From Table 4.2, it can be observed that the lowest average precision, average recall and average $F1$ score

were achieved when the RF model was trained using only spatial features. Training the RF model using only spectral features improved the average precision, average recall and average $F1$ score. As expected, the RF model gave higher average precision, average recall and average $F1$ scores when trained using the combined spatial and spectral features. LDA is then applied on the spectral features (originally 256) in the dataset to reduce dimensionality of the spectral data. The spatial features are combined with the features extracted by the LDA to train the RF model. Classification results obtained for each approach are presented in Table 4.2 and Figure 4.6. The classification results obtained when using the spatial features and LDA features combined is an improvement on those obtained when the other feature types and feature combinations are used. The usefulness of the approach is therefore validated. From Figure 4.6, it can be observed that the spatial features and the first 85 LDA components gives the best classification results with average precision, average recall and average $F_1$ score of 79.64%, 78.80% and 78.27% respectively for all 90 species. While the average precision, recall and $F1$ score have been discussed here, the individual results for each of the 90 species are also presented in Table 4.3. Interestingly, as can be seen in Table 4.3, the $F1$ scores vary between 80-100%, and 90-100% for 45 and 22 varieties respectively. These classification results show $F1$ scores which are very high and comparable with those reported in Chapter 3. These results demonstrate the potential of the proposed imaging modality (combining spatial features extracted from RGB images and spectral features extracted from hyperspectral images) for rice seed species discrimination even when the number of species is fairly large. However, as can be seen in Table 4.3, this technique performs better for some species than others when used on the entire dataset of 90 species. Further analyses are

carried out and presented in Section 4.8.4 to identify what might be responsible for this.

Table 4.2 Classification results with and without dimensionality reduction. © 2020 IEEE

| Feature Schemes | Average precision (%) | Average Recall (%) | Average $F_1$ score (%) |
|---|---|---|---|
| Spatial | 16.33 | 16.57 | 15.96 |
| Spectral | 34.93 | 35.86 | 34.46 |
| Spatial + spectral on full bands | 51.66 | 51.49 | 50.51 |
| Spatial + 85 LDA Components from Spectral (based on Figure 4.6) | **79.64** | **78.80** | **78.27** |

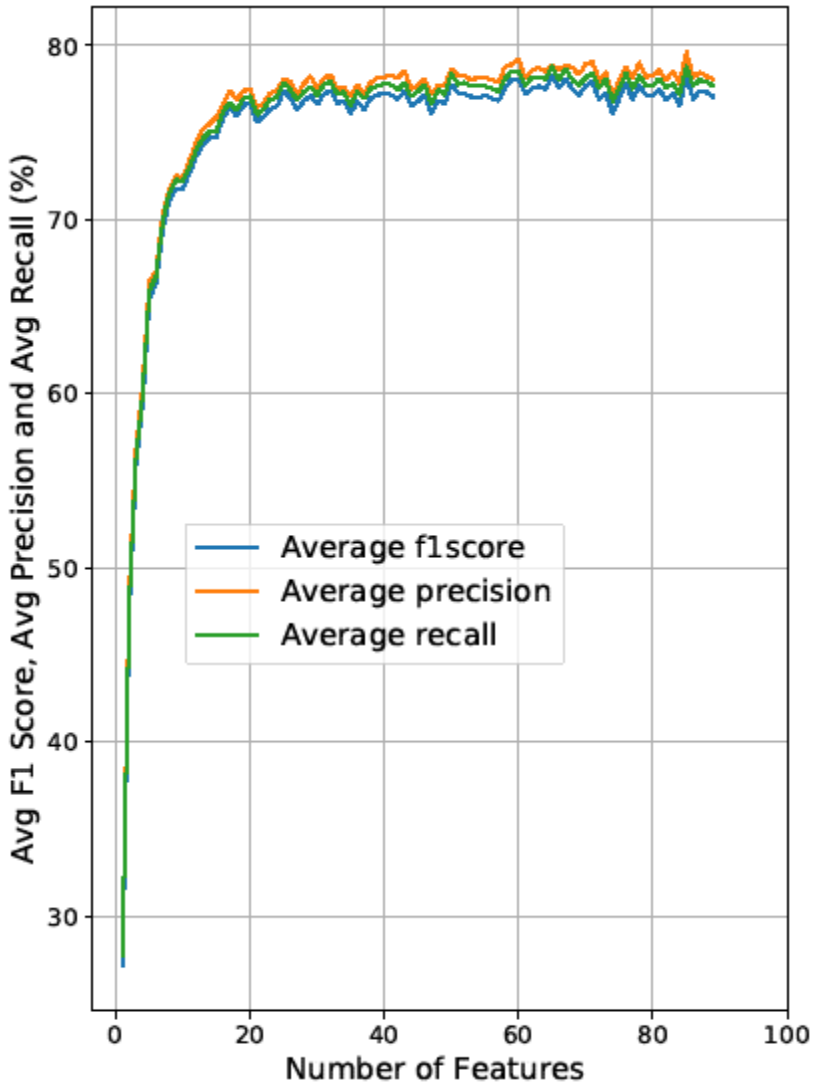



Figure 4.6 Classification results (average precision, average recall and average $F_1$ *score*) using the spatial + 85 LDA features, and 90 species. © 2020 IEEE

Table 4.3 Classification results with spatial + 85 LDA features, and 90 species. © 2020 IEEE

| Labels | Precision | Recall | $F_1$ score | Labels | Precision | Recall | $F_1$ score | Labels | Precision | Recall | $F_1$ score |
|---|---|---|---|---|---|---|---|---|---|---|---|
| GS55R | 100.00 | 100.00 | 100.00 | LTH35 | 89.47 | 80.95 | 85.00 | KN5 | 62.50 | 90.91 | 74.07 |
| TC10 | 100.00 | 100.00 | 100.00 | PD211 | 82.61 | 86.36 | 84.44 | TV2 | 78.57 | 68.75 | 73.33 |
| NV1 | 100.00 | 100.00 | 100.00 | LT183 | 75.00 | 95.45 | 84.00 | DTH155 | 81.25 | 65.00 | 72.22 |
| KL25 | 100.00 | 100.00 | 100.00 | PC10 | 81.82 | 85.71 | 83.72 | ND9 | 68.18 | 75.00 | 71.43 |
| N97 | 100.00 | 100.00 | 100.00 | HQ15 | 73.91 | 94.44 | 82.93 | KB6 | 73.33 | 68.75 | 70.97 |
| NPQ | 96.00 | 100.00 | 97.96 | KD18 | 87.50 | 77.78 | 82.35 | NTBH | 77.27 | 65.38 | 70.83 |
| NT16 | 95.24 | 100.00 | 97.56 | TB14 | 94.12 | 72.73 | 82.05 | GL301 | 60.87 | 82.35 | 70.00 |
| BT6 | 100.00 | 94.74 | 97.30 | NTP | 72.73 | 94.12 | 82.05 | NH92 | 66.67 | 71.43 | 68.97 |
| HS1 | 100.00 | 94.12 | 96.97 | NTHY | 76.47 | 86.67 | 81.25 | DMV58 | 66.67 | 71.43 | 68.97 |
| TXHQ | 93.75 | 100.00 | 96.77 | NDSLH | 76.47 | 86.67 | 81.25 | NBP | 75.00 | 62.50 | 68.18 |
| NCT | 100.00 | 92.86 | 96.30 | VS6 | 78.95 | 83.33 | 81.08 | NM142 | 66.67 | 66.67 | 66.67 |
| N54 | 92.31 | 100.00 | 96.00 | DA1 | 80.00 | 80.00 | 80.00 | LDA8 | 55.17 | 84.21 | 66.67 |
| CS6 | 93.33 | 93.33 | 93.33 | NC7 | 87.50 | 73.68 | 80.00 | NC2 | 54.17 | 81.25 | 65.00 |
| NPT3 | 93.33 | 93.33 | 93.33 | MH88 | 73.68 | 87.50 | 80.00 | HD1 | 60.00 | 70.59 | 64.86 |
| CL61 | 89.47 | 94.44 | 91.89 | NN4B | 79.17 | 79.17 | 79.17 | VP1 | 66.67 | 62.50 | 64.52 |
| DT66 | 85.00 | 100.00 | 91.89 | AH1000 | 70.83 | 89.47 | 79.07 | VS1 | 51.61 | 84.21 | 64.00 |
| VT8 | 100.00 | 84.21 | 91.43 | SVN1 | 81.82 | 75.00 | 78.26 | DT8 | 57.89 | 68.75 | 62.86 |
| SHPT1 | 95.45 | 87.50 | 91.30 | A128 | 84.21 | 72.73 | 78.05 | H229 | 69.23 | 56.25 | 62.07 |
| VH8 | 95.24 | 86.96 | 90.91 | CT286 | 80.00 | 76.19 | 78.05 | TX1 | 100.00 | 44.44 | 61.54 |
| N98 | 95.00 | 86.36 | 90.48 | CH12 | 76.19 | 80.00 | 78.05 | HP28 | 71.43 | 52.63 | 60.61 |
| NHN | 95.00 | 86.36 | 90.48 | TQ14 | 72.73 | 84.21 | 78.05 | MT151 | 100.00 | 42.86 | 60.00 |
| R068 | 91.67 | 88.00 | 89.80 | KB27 | 73.68 | 82.35 | 77.78 | BC15 | 56.25 | 64.29 | 60.00 |
| NDC1 | 95.45 | 84.00 | 89.36 | 91RH | 100.00 | 62.50 | 76.92 | CNC12 | 100.00 | 42.11 | 59.26 |
| BQ10 | 95.45 | 84.00 | 89.36 | CTX30 | 83.33 | 71.43 | 76.92 | KC111 | 72.73 | 50.00 | 59.26 |
| DTL2 | 87.50 | 91.30 | 89.36 | R998KBL | 66.67 | 90.00 | 76.60 | NBK | 69.23 | 47.37 | 56.25 |
| HN39 | 84.21 | 94.12 | 88.89 | HT18 | 81.25 | 72.22 | 76.47 | DT52 | 64.29 | 50.00 | 56.25 |
| HL | 85.71 | 90.00 | 87.80 | TC112 | 81.25 | 72.22 | 76.47 | NBT1 | 50.00 | 50.00 | 50.00 |
| DV108 | 80.00 | 94.12 | 86.49 | BTS7 | 66.67 | 88.89 | 76.19 | NPT1 | 40.00 | 40.00 | 40.00 |
| 9d | 86.36 | 86.36 | 86.36 | TQ36 | 64.00 | 94.12 | 76.19 | TB13 | 42.86 | 35.29 | 38.71 |
| NKB19 | 82.14 | 88.46 | 85.19 | VS5 | 76.19 | 76.19 | 76.19 | KB16 | 33.33 | 30.77 | 32.00 |

### 4.8.3. Performance Analysis on Selected Subsets of 6 Species

In this experiment, 6 subsets are selected from dataset of 90 species in order to compare the proposed approach with state-of-the-art techniques which (as shown in Chapter 3) tend to be evaluated on data with small variety of species; usually, 5-6 with the exception of [32] (30 species) and [33] (754 species). 5 of the 6 selected subsets, each consisted of 6 species which are randomly selected while the 6$^{th}$ subset consists of species that showed the worst classification performance in Table 4.3. A summary of all the subsets is presented in Table 4.4. LDA is applied on the spectral data of each subset to extract features which are combined with corresponding spatial features. The RF classifier is trained separately using the 6 subsets and the classification results obtained are presented in Table 4.5. It is also noted that the time required for computation (also considering both the segmentation and classification tasks) is 0.53 s on a commodity hardware (Intel Core i7). When compared to a manual rice seed screening which normally takes minutes, the computation time (0.53 s) is negligible for screening such seeds in practical applications. It was observed that the average precision and average recall were significantly improved for the first 5 subsets when compared to the classification results obtained for the dataset of 90 species. The proposed approach gave very high average $F1$ scores and outperformed equivalent scores attained by the state-of-the-art techniques reviewed in Chapter 3 with the exception of those reported in [8],[89],[90]. These experimental results demonstrate that, in line with state-of-the-art techniques for rice seed classification, taking advantage of spatial features from high spatial resolution images and combining them with spectral features from hyperspectral images can achieve very good classification results and removal of impure species from rice seed samples. For the 6$^{th}$ subset of

species, the average $F_1$ score obtained when employing the first model trained on the dataset of 90 species is 45.54%. On the other hand, the $F_1$ score rises to 61.29% when the 6 species selected was used specifically to train the RF model. This increase in the $F_1$ score becomes possible due to the dataset which is now smaller and suggests that the use of a targeted model would be more appropriate when the species are known a-priori.

### 4.8.4. Performance Analysis on Subsets with Varying Species Sizes

It is clear from the results presented in both Sections 4.8.2 and 4.8.3 that the size of the dataset and variety of species considered in each experiment directly affects the performance of the RF model. To further explore this, another 6 subsets of data are selected from the dataset of 90 species. This time, the number of species in each subset of data is varied to include 6, 20, 40, 60, 80 and 90 different species to explore exactly how the performance of the classifier in a study can be influenced by varying number of species. LDA is applied on each of the data subsets of increasing size and the features extracted are combined with the corresponding spatial features. The RF model is then trained using the spatial features combined with the features extracted from LDA (starting from 1 LDA features up to $c - 1$, where $c$ is the number of species in the dataset). A plot of average $F1$ score against the number of LDA features used to train the classifier is obtained for each subset of the data considered. These plots are illustrated in Figure 4.7. The maximum average $F_1$ score is also obtained from each of the plots in Figure 4.7 and used to obtain plots of the maximum average $F1$ scores against the number of species considered as illustrated in Figure 4.8. From Figure 4.8, it can be observed that the RF model's performance is influenced by the number of species used in this study. The average $F1$ scores decreased significantly from 98.17%

for 6 species to 78.27% for 90 species. One reason for this decrease in the performance of the classifier could be due to an increase in the similarity level among the rice seed species as the number of species increases.

Consequently, investigating the influence of similarities among the species or classes which are present in the data of rice seeds on the classifiers' performance becomes important. Also, it was reported in the previous subsection that the methods employed in [8],[89],[90] attained better classification results than the approach used in this chapter. Assessing the similarity level among the species of rice seeds will also be helpful in clarifying whether the use of better feature combination schemes, better algorithms or inter/intra class variation among species themselves is responsible for the higher performances reported in those papers. Though, in general, the approach used in this chapter performs very well, it is not without some limitations. This can be illustrated by analysing situations where the approach performs well and where it does not. For instance, spectral profiles of some species which gave good classification results in Table 4.3 are illustrated in Figure 4.9 and Figure 4.10. It is clear from Figure 4.9 and Figure 4.10 that the spectral profiles of those species are not identical which was responsible for the good classification results reported for them. Spectral profiles of some species which gave poor classification results in Table 4.3 are also illustrated in Figure 4.11 and Figure 4.12. As can be seen in Figure 4.11, the spectral profiles of species TB13, KB16, NBK and NPT1 are virtually identical. Similarly, spectral profiles of species KC111, BC15 and MT15 are also virtually identical as can be seen in Figure 4.12. This led to the misclassification of some seeds by the proposed approach which was responsible for the poor classification results reported for those species. It is therefore necessary to assess how similar the species or classes of rice

seeds are and to overcome the negative effects on the performance of the classifiers due to the similarity levels. While it is believed that misclassifications occur due to species in the data showing similar properties, it is worth noting that the observed decrease in classification performance could also be due to the limited number of rice seed samples available against what is needed to cover the large feature space. It is also necessary to explore the effects of varying the number of rice seed samples in each class.

Table 4.4 Species contained in each sub-dataset. © 2020 IEEE

| Subset | Species |
|---|---|
| 1 | HS1, CH12, AH1000, SVN1, 91RH, DT8 |
| 2 | TB14, N54, NKB19, HQ15, BT6, NC7 |
| 3 | KB6, AH1000, HQ15, TQ14, KL25, NHN |
| 4 | TC10, DTL2, KB16, BT6, KB27, CNC12 |
| *5* | CL61, NKB19, VH8, TX1, MT15, HL |
| 6 | NBK, DT52, NBT1, NPT1, TB13, KB16 |

Table 4.5 Classification results using the output of LDA (combined with the spatial features) and randomly drawn 6 species. © 2020 IEEE

| Subsets | Average precision (%) | Average Recall (%) | Average $F_1$ score (%) | Min–Max $F_1$ score (%) |
|---|---|---|---|---|
| 1 | 96.03 | 96.46 | 96.18 | 89.66-100 |
| 2 | 96.23 | 96.31 | 96.21 | 89.47-100 |
| 3 | 98.59 | 98.33 | 98.42 | 97.30-100 |
| 4 | 98.55 | 97.93 | 98.17 | 95.45-100 |
| 5 | 96.39 | 96.73 | 96.52 | 92.86-100 |
| 6 | 61.99 | 61.12 | 61.29 | 35.71 -85.71 |

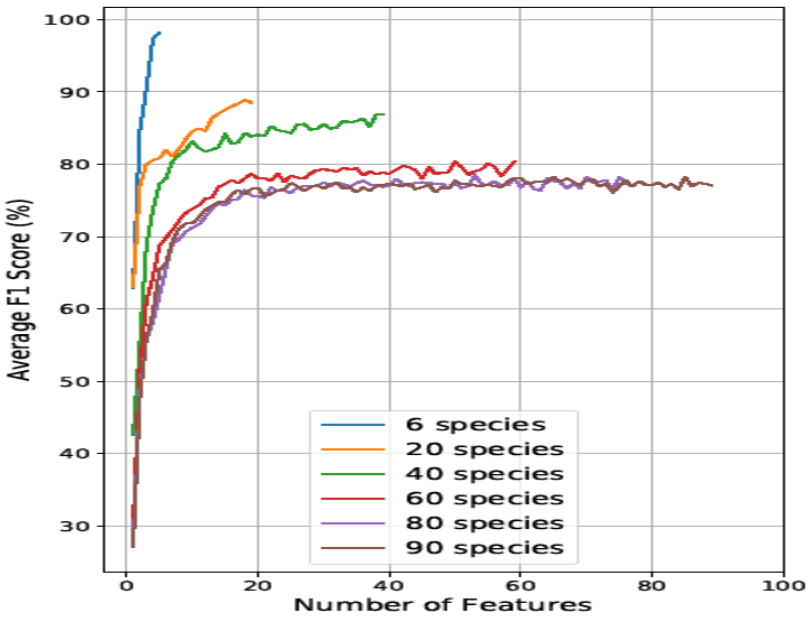



Figure 4.7 Plots of average *$F_1$ score* against number of features for sub-datasets with species sizes of 6, 20, 40, 80, and 90. © 2020 IEEE

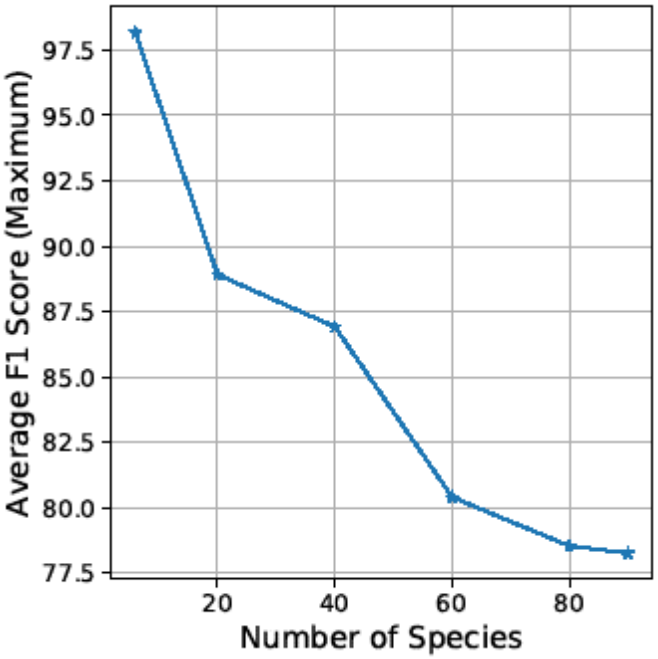



Figure 4.8 A plot of maximum average *F1 score* against number of species. © 2020 IEEE

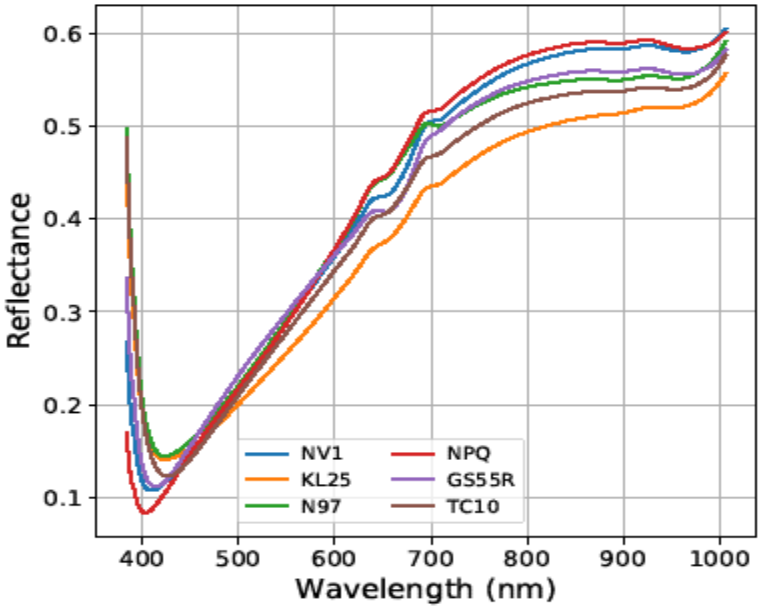



Figure 4.9 Average spectral profiles of some species with good classification results in Table 4.3. © 2020 IEEE

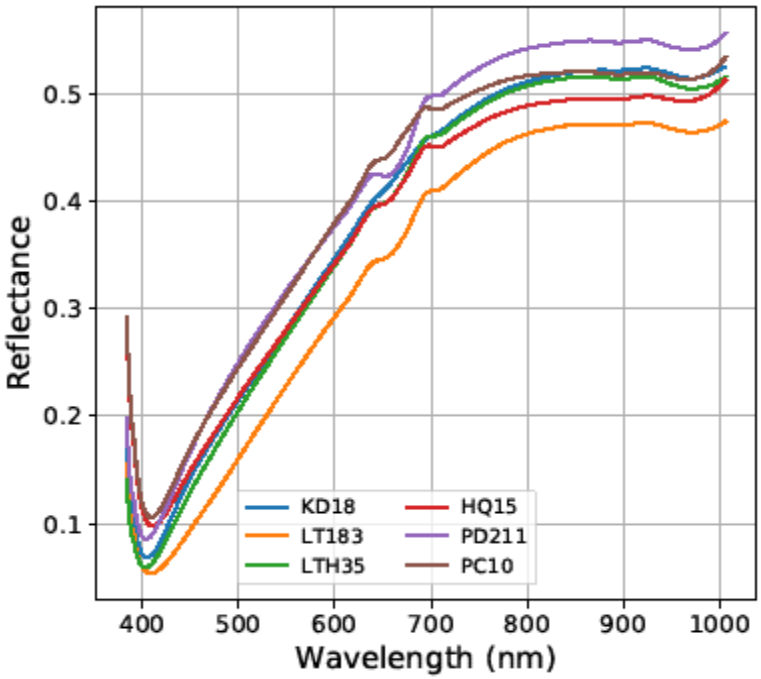



Figure 4.10 Average spectral profiles of some species with good classification results in Table 4.3. © 2020 IEEE

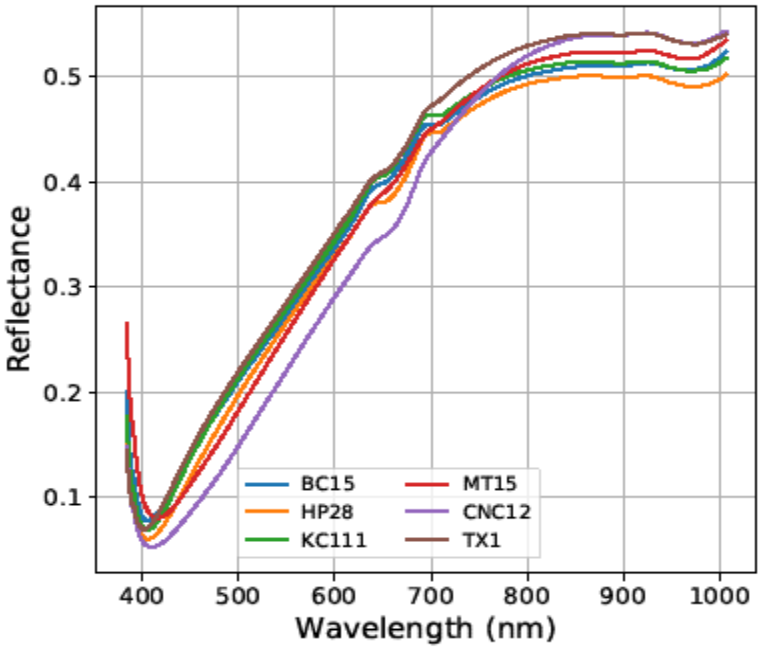



Figure 4.11 Average spectral profiles of some species with poor classification results in Table 4.3. © 2020 IEEE

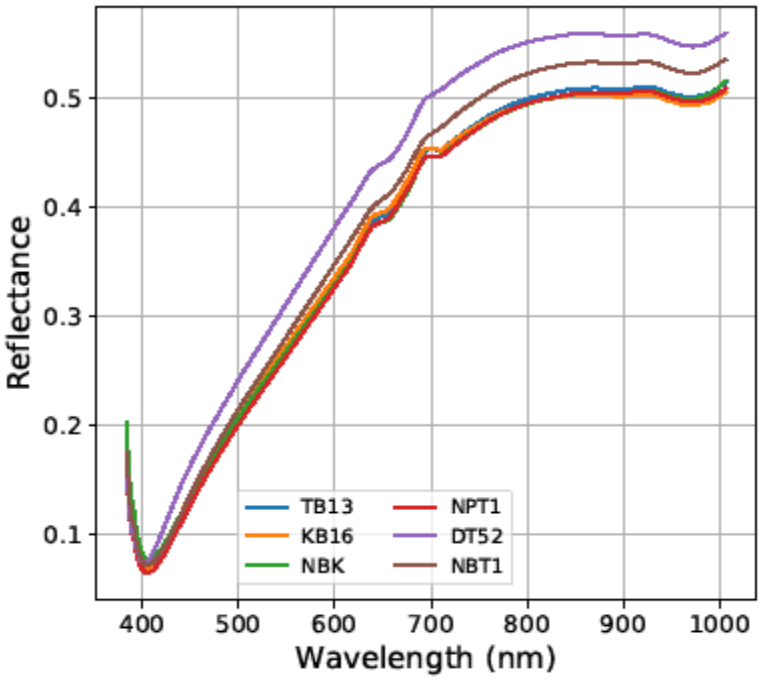



Figure 4.12 Average spectral profiles of some species with poor classification results in Table 4.3. © 2020 IEEE

## 4.9. Summary

Comparison of the performance of LDA and PCA for dimensionality reduction of hyperspectral imaging data has been presented in this chapter. Experimental results obtained show that LDA can perform better than PCA as a dimensionality reduction technique for hyperspectral imaging data. This chapter also evaluated the performance of a new, fused, RGB and HSI system for hyperspectral data classification. The system was used to acquire RGB images and hyperspectral image data cubes which offer high spatial and spectral resolution respectively. Spatial features extracted from the acquired RGB images and spectral features extracted from the acquired hyperspectral data cubes constitute the dataset which was used in this work to evaluate performance of the RGB and hyperspectral imaging system. The dataset has been made publicly available [34],[35]. Experimental results demonstrate that taking advantage of spatial

features from high spatial resolution images and combining them with spectral features from hyperspectral data cubes can achieve very good classification results and removal of impure species from the data. Suboptimal performance was reported for some species and this was linked to the use of large number of species and similarities among the species.

## 5. Folded Linear Discriminant Analysis: A Novel Technique for Feature Extraction and Dimensionality Reduction of Hyperspectral Imaging Data

From the review of related papers presented in Chapter 3, it can be noted that PCA is a more commonly applied dimensionality reduction tool for hyperspectral imaging than LDA [15],[21],[10],[13]. In Chapter 4, LDA was used to reduce the dimensionality of hyperspectral imaging data prior to classification. LDA achieved promising results in Chapter 4 and gave better performance than the commonly applied PCA. In [37], an extension of PCA, named folded-PCA, was proposed for feature extraction and dimensionality reduction of hyperspectral imaging data. The folded-PCA was implemented by folding each of the spectral vectors in the hyperspectral data matrix, applying the conventional PCA steps on the resulting data (a set of 2D matrices) and unfolding the projected samples for classification. Promising results achieved by LDA as a dimensionality reduction technique for hyperspectral data in Chapter 4 and the increased accuracy and reduced computational complexity achieved by the authors in [37] motivate the extension of LDA for hyperspectral data in this chapter using a mathematical step which is similar to the one used in [37].

It can also be noted from the review of related papers presented in Chapter 3 that LDA gives sub-optimal performance on small training samples and that the number of features that can be extracted by LDA is limited to $c - 1$ where $c$ is the number of classes in the data. These constraints often limit the use of LDA on hyperspectral data since sufficient samples are not usually available in such data for training. To solve these problems, 2D LDA was applied in a recent paper [38] where pixels in the HSI data were converted to a matrix and a single vector was computed for projection. While

this is a nice extension, the approach used continues to limit the number of features that can be extracted to the number of columns in the converted matrix.

This chapter therefore proposes a new Folded-LDA (F-LDA), an improved version of the traditional LDA transform. The proposed F-LDA folds each spectral vector (samples) in the hyperspectral data into a matrix. Different dimensions (configurations) of the folded samples are exploited and extensive experiments carried out to illustrate the folding limits, especially when the dimensions of the folded samples is set to $f * 1$ or $1 * f$ where $f$ is the number of features in the hyperspectral data. Eigenvectors are processed individually and the projected samples unfolded to extract the final features. Therefore, the number of features that can be extracted is no longer limited to $c - 1$ but is now given as the product of the number of columns in the folded samples (converted matrices) and the rank of the between-class variance matrix. This allows the extraction of many more discriminant features by the proposed F-LDA which makes it more flexible than the conventional LDA. The proposed F-LDA is therefore capable of extracting more informative features (capturing local structure in the hyperspectral data through the use of folded samples), and targeting higher classification accuracy than the conventional LDA, 2D LDA and the full feature space. Computational complexity of the proposed approach is analysed to illustrate its additional benefits (when compared with the traditional LDA) which are summarized as follows:

1) The complexity of computing the within-class variance, between-class variance, transformation matrix, and eigenvectors is reduced in the proposed approach.

2) The complexity of projecting the hyperspectral data into a lower dimensional space is reduced.
3) The proposed approach requires much less contiguous memory than the conventional LDA.

## 5.1. Proposed Folded-LDA

### 5.1.1. Concepts of the Proposed F-LDA

The proposed F-LDA folds each sample (spectral vector) in the hyperspectral data matrix, $\boldsymbol{X}$, into a matrix, as illustrated in Figure 5.1. This approach is different from that of the conventional LDA where samples are treated as spectral vectors. Folding of the spectral vectors into matrices by the proposed F-LDA provides an alternative way of generating variance matrices, where local structure (information across contiguous spectral bands) in the data are captured [75],[37].

Following the conversion of all the spectral vectors in the data matrix $\boldsymbol{X}$ into matrices of the same configuration (size) in the proposed F-LDA, data samples in a class are represented as a stack of matrices (folded samples) which belong to that class. Using the new data representation, the mean of each class and the overall mean of the data matrix are computed as matrices, which can then be used to compute the within-class variance, between-class variance and the transformation matrix.

Each spectral vector, whose length is given as $f$ (the number of features in the data matrix), can be folded in such a way that the size of the folded samples is given as $G \times B$ where $G$ is the number of groups and $B$ is number of bands in each group, which is the same for all groups for simplicity. $G = f$ will be shown later in Section 5.3 as a special case of the F-LDA which simplifies it to the conventional LDA since

the number of groups in the folded samples is set to $f$. It will also be shown that $G = 1$ is another special case of the F-LDA which simplifies it to the original data matrix since the number of groups in the folded samples is 1. The last step of the proposed F-LDA is to fold the projected samples, which can be fed to models for classification.

Computing the transformation matrix using the stack of folded samples facilitates the extraction of features (which are now many more than $c - 1$) from the HSI data in an alternative but more effective way. This leads to improvement in classification performance (higher classification accuracy, less computation complexity and reduced contiguous memory requirement). It is noteworthy here that this improvement depends on the configuration of the folded samples, $G \times B$.

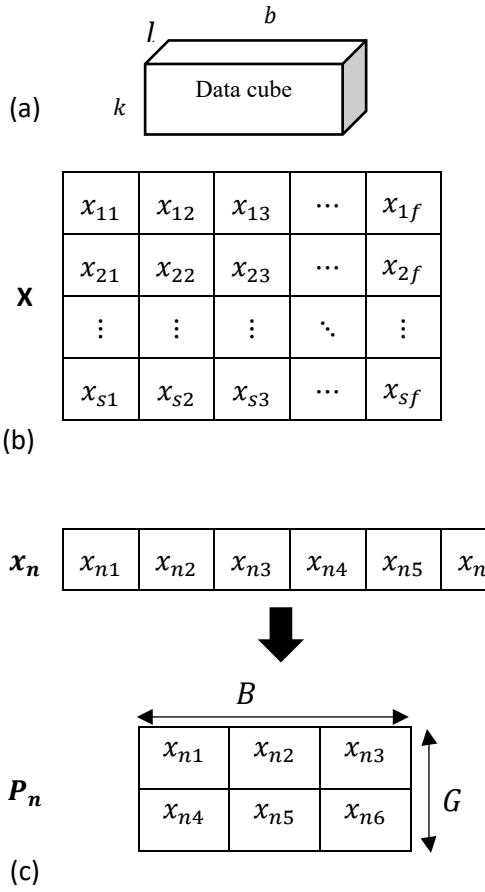



Figure 5.1 (a) the hyperspectral data (b) the data matrix **X** where each row depicts a spectra vector (sample), $x_n = [x_{n1}\ x_{n2}\ x_{n3} \dots\dots\dots x_{nf}]$ (c) The spectral vector $\boldsymbol{x_n}$ is folded to form a 2D matrix, $\boldsymbol{P_n}$ where $n \in [1, s]$, $G = 2$, $B = 3$ and $f = G \times B = 6$. © 2020 IEEE

### 5.1.2. Implementation of the Proposed F-LDA

If a spectral vector in the data matrix is denoted as $\boldsymbol{x_n} = [x_{n1}\ x_{n2}\ x_{n3} \dots\dots\dots x_{nf}]$ as shown in Figure 5.1 where $n \in [1, s]$, a folded sample (converted matrix), $\boldsymbol{P_n}$, of this vector can be denoted using (5.1).

$$\boldsymbol{P_n} = \begin{bmatrix} p_{n(1,1)} & \cdots & p_{n(1,B))} \\ \vdots & \ddots & \vdots \\ p_{n(G,1)} & \cdots & p_{n(G,B)} \end{bmatrix} \tag{5.1}$$

and each element in the matrix $P_n$, can be denoted as $p_{n(h+1,i)}$ and computed using (5.2).

$$p_{n(h+1,i)} = x_{(h*B)+i} \tag{5.2}$$

where $h \epsilon [0, G-1]$ and $i \epsilon [1, B]$.

The mean of the folded samples in each class $c_j$, denoted as $\boldsymbol{M_j}$ where $j \epsilon [1,\ c]$ , and the overall mean of all the folded samples, denoted as $\boldsymbol{M}$, can be calculated using (5.3) and (5.4).

$$\boldsymbol{M_j} = \frac{1}{N_j} \sum_{i=1}^{N_j} P_{ij}\ \ \boldsymbol{M_j} \epsilon \Re^{G \times B} \tag{5.3}$$

$$\boldsymbol{M} = \sum_{j=1}^{c} \frac{N_j}{s} M_j\ ,\ \ \boldsymbol{M} \epsilon \Re^{G \times B} \tag{5.4}$$

where $\boldsymbol{P_{ij}}$ is the ith converted matrix in class $c_j$ and $i \epsilon [1,\ N_j]$.

The within-class variance $\boldsymbol{V_{PW}}$ and the between-class variance $\boldsymbol{V_{PB}}$ of the data matrix can be computed using (5.5) and (5.6) respectively.

$$\boldsymbol{V_{PW}} = \sum_{j=1}^{c} \sum_{i=1}^{N_j} (\boldsymbol{P_{ij}} - \boldsymbol{M_j})(\boldsymbol{P_{ij}} - \boldsymbol{M_j})^T \tag{5.5}$$

$$\boldsymbol{V_{PB}} = \sum_{j=1}^{c} N_j (\mathbf{M_j} - \mathbf{M})(\mathbf{M_j} - \mathbf{M})^T \tag{5.6}$$

where $\boldsymbol{V_{PW}} \epsilon \Re^{G \times G}$ and $\boldsymbol{V_{PB}} \epsilon \Re^{G \times G}$.

Using the between-class variance, $\boldsymbol{V_{PB}}$, and the within-class variance, $\boldsymbol{V_{PW}}$, the transformation matrix, $\boldsymbol{T_P}$, the eigenvalues, the eigenvectors, and the selected eigenvectors, $\boldsymbol{V_{Pd}}$ can be computed by applying the same approach in the conventional

LDA. The data can then be projected into a lower dimensional space using (5.8) where $\boldsymbol{Y_n}$ is the projected matrix of each sample.

$$\boldsymbol{T_P} = \boldsymbol{V_{PW}}^{-1}\boldsymbol{V_{PB}}, \boldsymbol{T_P} \in \Re^{G \times G} \tag{5.7}$$

$$\boldsymbol{Y_n} = \boldsymbol{P_n}^{\boldsymbol{T}}\boldsymbol{V_{Pd}}, \boldsymbol{V_{Pd}} \in \Re^{G \times d\prime}, \ \boldsymbol{Y_n} \in \Re^{G \times d\prime} \tag{5.8}$$

Finally, the algorithmic step code of the proposed F-LDA is presented in Table 5.1. The size of the between-class variance $\boldsymbol{V_{PB}}$, and the within-class variance, $\boldsymbol{V_{PW}}$, is $G \times G$ while the size of the between-class variance, $\boldsymbol{V_B}$, and the within-class variance, $\boldsymbol{V_W}$, computed using the conventional LDA is $f \times f$ or $GB \times GB$. Similarly, the size of the transformation matrix, $\boldsymbol{T_P}$ is $G \times G$ while that of the transformation matrix computed using the conventional LDA is $f \times f$ or $GB \times GB$. This therefore leads to significant reduction in the complexity of computing the within-class variance, between-class variance, transformation matrix, eigenvalues and eigenvectors as will be shown in Section 5.3.2. Furthermore, data projection is done by the multiplication of two smaller matrices, $\boldsymbol{P_n}^{\boldsymbol{T}}$ and $\boldsymbol{V_{Pd}}$ which are of size $B \times G$ and $G \times d$ respectively. This also contributes to the significant reduction in computational complexity.

Table 5.1 Algorithmic step code of the proposed F-LDA. © 2020 IEEE

| Step | Algorithmic code |
|---|---|
| 1. | Convert the hyperspectral data cube to data matrix $\boldsymbol{X}$ |
| 2. | Fold each spectral vector $\boldsymbol{x_n}$ in the data matrix to a 2D matrix $\boldsymbol{P_n}$ to form a set of 2D matrices |
| 3. | Compute the mean $\boldsymbol{M_j}$ of all $\boldsymbol{P_n}$ (folded samples) in each class |
| 4. | Compute the mean $\boldsymbol{M}$ of all $\boldsymbol{P_n}$ (folded samples) in the data |
| 5. | Use matrices $\boldsymbol{M}$ and $\boldsymbol{M_j}$ to compute the within-class variance matrix $\boldsymbol{V_{PW}}$ and between-class variance matrix $\boldsymbol{V_{PB}}$ |
| 6. | Compute the transformation matrix $\boldsymbol{T_P}$ using (5.7) |
| 7. | Compute the eigenvectors and eigenvalues of $\boldsymbol{T_P}$ |
| 8. | Rank the eigenvectors in descending order according to their eigenvalues |
| 9. | Use the first $k$ eigenvectors to project the data into a lower dimensional space as in (5.8) |
| 10. | Unfold the projected matrices |

### 5.1.3. Extraction of Local Structures Using the Proposed F-LDA

This subsection provides explanation on how the proposed F-LDA can capture the local structure in the spectral vectors. If each row in each of the folded matrices $P_{ij} = P_n$ in (5.1) is denoted as $p_{ijk}$ where $k \in [1, G]$, $P_{ij}$ can be formulated as (5.9).

$$P_{ij} = \begin{bmatrix} p_{ij1} \\ p_{ij2} \\ \vdots \\ p_{ijG} \end{bmatrix} = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_G \end{bmatrix} \tag{5.9}$$

Each spectral vector in the data matrix can then be expressed as $x_{ij} = x_n = [p_{n1} \quad p_{n2} \quad \cdots \quad p_{nG}]$ . If $\boldsymbol{m_j}$, the overall mean of each class in the conventional LDA, is folded into a $G \times B$ matrix, $\boldsymbol{m_j}$ can also be formulated using (5.10).

$$\boldsymbol{M_{mJ}} = \begin{bmatrix} m_{j1} \\ m_{J2} \\ \vdots \\ m_{JG} \end{bmatrix} \tag{5.10}$$

The within-class variance $\boldsymbol{V_W}$ as used in the conventional LDA can then be expressed using (5.11)

$$\boldsymbol{V_W} = \sum_{j=1}^{c} \sum_{i=1}^{N_j} \begin{bmatrix} (p_1 - m_{j1})(p_1 - m_{j1}) & (p_1 - m_{j1})(p_2 - m_{j2}) & \ldots & (p_1 - m_{j1})(p_G - m_{jG}) \\ (p_2 - m_{j2})(p_1 - m_{j1}) & (p_2 - m_{j2})(p_2 - m_{j2}) & \cdots & (p_2 - m_{j2})(p_G - m_{jG}) \\ \vdots & \vdots & \ddots & \vdots \\ (p_G - m_{jG})(p_1 - m_{j1}) & (p_G - m_{jG})(p_2 - m_{j2}) & \cdots & (p_G - m_{jG})(p_G - m_{jG}) \end{bmatrix} \tag{5.11}$$

Also, if each row in $\boldsymbol{M_j}$ is denoted as $t_{ji}$ where $i \in [1, G]$, $\boldsymbol{M_J}$ can be formulated as (5.12).

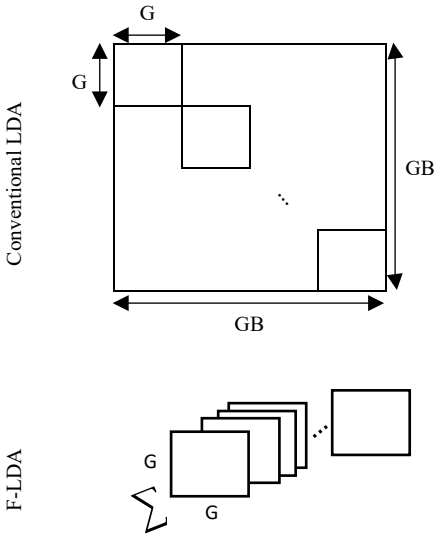



Figure 5.2 Comparing the within-class variance matrices constructed using the F-LDA and the conventional LDA. The F-LDA matrix is based on the accumulation of those blocks across the main diagonal of the conventional LDA matrix, leading to a local extraction of features. © 2020 IEEE

$$\boldsymbol{M_J} = \begin{bmatrix} t_{j1} \\ t_{J2} \\ \vdots \\ t_{JG} \end{bmatrix} = \begin{bmatrix} m_{j1} \\ m_{J2} \\ \vdots \\ m_{JG} \end{bmatrix} = \boldsymbol{M_{m_J}} \tag{5.12}$$

Finally, the within-class variance $\boldsymbol{V_{PW}}$ in (5.5) as used in the proposed F-LDA can then be expressed using (5.13).

$$\boldsymbol{V_{PW}} = \sum_{j=1}^{c} \sum_{i=1}^{N_j} \big[ (p_1 - t_{j1})(p_1 - t_{j1}) + (p_2 - t_{j2})(p_2 - t_{j2}) + \cdots + (p_G - t_{jG})(p_G - t_{jG}) \big] \tag{5.13}$$

Accumulation of the diagonal elements of $\boldsymbol{V_W}$ in the conventional LDA are used to construct the within-class variance matrix, $\boldsymbol{V_{PW}}$ in (5.5) as shown in (5.13) and illustrated in Figure 5.2. The proposed F-LDA therefore captures the local structures within the group bands and extract features that improve discrimination.

### 5.1.4. Different Configurations and Their Implications

Different configurations ($G \times B$) of the folded samples can be exploited in the proposed F-LDA. The factors of $f$, the number of features in the data matrix, is used in selecting the different configurations to be exploited. The total number of features that can be extracted by F-LDA in each case is $d$ and is given as $B \times d_{\text{EVD}}$, where $d_{\text{EVD}}$ is the number of extracted components at the EigenValue Decomposition (EVD) of the transformation matrix, $\boldsymbol{T_P}$. The number of discriminant components that can be extracted is limited to the number of non-zero eigenvalues, which is also the rank of the between-class variance matrix, $\boldsymbol{V_{PB}}$ [52],[53],[54]. This implies that the value of $d_{\text{EVD}}$ can only be varied from 1 to $r$ where $r$ is the rank of $\boldsymbol{V_{PB}}$. The value of $r$ is not the same for different configurations as shown in Table 5.2 – 5.16. It can be seen in Table 5.2 – 5.16 that the value of $r$ is equal to $c - 1$ whenever the configuration of the folded sample is set to $G \times 1$ where $c$ is the number of classes in the data. This is a special case of the F-LDA which simplifies it to the conventional LDA. In all the cases considered, the configuration ($G \times B$) that gives the best classification results is selected as the optimal configuration.

The configuration of the folded samples can only be set to $f \times 1$ and $1 \times f$ in a case where $f$ is a prime number. To explore other configurations ($G \times B$) in the proposed F-LDA, extra feature vectors of zeros can be added to the data matrix to fill the empty spaces in the folded samples with zeros [37].

### 5.1.5. Classification

To compare the performance of the proposed and the traditional approaches, a Support Vector Machine (SVM) model is adopted for classification and implemented in this work using the RBF kernel because of its satisfactory performance in related papers

[60],[61]. The SVM model was trained using k-fold cross validation ($k$ = 5) to optimize parameters (penalty ($pc$) and gamma ($g$)) of the RBF kernel using a grid search. The training was carried out k times. In each case, the training set was divided into k folds, one of which was held out for validation and the rest for training. Classification results computed for all the k cases were then averaged and used to optimize parameters of the SVM. The SVM model with the optimal value of $pc$ and $g$ was used for final evaluation on the test set. This process was carried out 10 times and the classification results obtained in all the cases were averaged and reported.

## 5.2. Datasets and Experimental Settings

### 5.2.1. Datasets

Five hyperspectral datasets which are publicly available [50] and widely used in related papers [24],[26],[27] were selected to evaluate the performance of the proposed F-LDA. Full description of the selected hyperspectral datasets (Botswana, Salinas – A, Pavia Center, Pavia University, and Indian Pine data) are provided in Chapter 2 of this thesis. Each of the selected datasets was divided into training and testing sets. The size of the training and testing sets was selected to simulate a small sample size scenario for training [27],[110]. Specifically, the Botswana data was divided into training (5%) and testing (95%) sets. For the Salinas – A, Pavia Center, and Pavia University data, 220 samples were selected for training while the rest were used for testing. The Indian Pine hyperspectral data was split into training (6%) and testing (94%) sets.

### 5.2.2. Experimental Settings

Overall Accuracy (OA), Average Accuracy (AA) and kappa coefficient ($kc$) are widely used as metrics to evaluate performance of proposed techniques on the selected five

datasets in related papers [73],[41],[110]. Hence, OA, AA and *kc* are adopted in this chapter as metrics for performance evaluation of the proposed approach. The following three different feature schemes are initially used to train the SVM model on each of the datasets: 1) Full feature space; 2) LDA features; and 3) F-LDA features. The number of features extracted by the LDA, $d$, from each dataset is varied from 1 up to $c-1$ (where $c$ is the number of class in the dataset). For the proposed F-LDA, the number of features $d$ that can be extracted depends on the values of $B$ and $d_{\mathrm{EVD}}$ and is given as $B \times d_{\mathrm{EVD}}$ where $d_{\mathrm{EVD}}$ is the number of EVD components selected for projection. For F-LDA, the number of extracted features is also varied from 1 up to $r$, where $r$ is the rank of the between-class variance matrix, $\boldsymbol{V_{PB}}$. It is worth noting that the number of features in the Pavia University dataset is 103. This is a prime number which limits the size of the converted matrices that can be obtained from the data to $1 \times 103$ and $103 \times 1$. Extra zeros are added to the data so that the empty spaces in the converted matrices can be filled and the proposed F-LDA can be applied using other configurations [75],[37].

The performance of the proposed F-LDA is compared with that of three supervised techniques namely 2D LDA [38], GDA [39] and NWFE [40]. These techniques (2D LDA, GDA and NWFE) are selected for fair comparisons since the proposed approach is also a supervised technique. The proposed F-LDA is further compared with two unsupervised techniques namely KPCA [41] and F-PCA (the technique that motivated this work) [37] for wider comparison. Gaussian kernel is selected as the kernel function for GDA and KPCA. Parameter (width) of the Gaussian kernel is optimized in the range $[10^1, 10^2 \ldots 10^5]$. Different configurations of the folded samples are also exploited when applying the 2D LDA and F-PCA on the selected hyperspectral

datasets for comparison with the proposed F-LDA. In the 2D LDA, the data was projected using $y_{Pd} = \boldsymbol{P_n}^{T}\boldsymbol{v_{Pd}}$, $\boldsymbol{P_n} \in \Re^{G\times B}$, $\boldsymbol{v_{Pd}} \in \Re^{G\times 1}$, where $\boldsymbol{v_{Pd}}$ is the single projection vector. Hence, the number of features that can be extracted by the 2D LDA is limited to $B$ (the number of columns in the folded samples) [38].

## 5.3. Experimental Results and Analysis

In this section, effects of the proposed method on the classification accuracy, computational complexity, and contiguous memory requirement for the selected datasets are investigated. The experimental results and analysis are therefore presented in the following three separate sub-sections addressing each of the following aspects: classification accuracy, computational complexity, and contiguous memory requirement).

### 5.3.1. Effect on Classification Accuracy

Firstly, the SVM model is trained using the full feature space available in each of the datasets. Secondly, traditional LDA is used to reduce the dimensionality of the datasets and extract features therefrom to train the SVM model. Plots of the OA and AA against the number of features extracted by LDA are obtained and the results illustrated in some figures for each dataset. The maximum OA and maximum AA are obtained from these plots and presented alongside the results obtained when the full feature space is used in Table 5.2 – 5.16. The maximum kappa coefficients (*kc*) are also obtained and reported in Table 5.2 – 5.16 in each of the considered cases. As shown in Table 5.2 – 5.16, the classification accuracy attained by the SVM model is lower when it was trained using the LDA features than when the full feature space was used to train it for all the datasets considered. These results are expected since LDA is known for giving below par performance when used in SSS scenarios [26],[111],[27],[110],[112].

Finally, the proposed F-LDA was used to reduce the dimensionality of the datasets and extract features therefrom to train the SVM model.

1. Classification Accuracy for the Botswana Dataset

F-LDA is applied on the Botswana dataset and while exploring different configurations ($G \times B$) of the converted matrices. Plots of the OA and AA against the number of components at the EVD, $d_{\mathrm{EVD}}$ are obtained and illustrated in Figure 5.3 for each of the configurations. The maximum OA and AA are then obtained from each of these plots and the classification results presented in Table 5.2. From Table 5.2, it can be observed that OA, AA, and *kc* were lowest when the configuration was set to $145 \times 1$. This is a special case of the F-LDA where $G = f$, which simplifies the F-LDA to the conventional LDA. Also, from Table 5.2, it can be observed that the maximum OA, AA and *kc* was highest when the F-LDA configuration was set to $29 \times 5$. This is an improvement on the classification accuracy obtained when the SVM model was trained using: (1) the LDA features and (2) the full feature space. One can also see that another special case of the F-LDA where $G = 1$ simplifies the F-LDA to the full feature space. Further steps are taken to train the SVM model using the features extracted by the 2D LDA, GDA, NWFE, KPCA and F-PCA when applied on the Botswana dataset. Again, the best OA, AA and *kc* are obtained and presented in Table 5.2. From Table 5.2, it can be seen that the proposed F-LDA consistently produce better classification performance (higher OA, AA and *kc)* than the other techniques used to benchmark its performance. When considering the standard deviation, from Table 5.2, it can be observed that the lower bands of the OA, AA and kc achieved by the proposed F-LDA ($91.17 \pm 1.06$, $91.69 \pm 1.15$ and $90.43 \pm 0.01$ respectively) are higher than the average classification results reported for the other

techniques except for KPCA (89.97%, 90.41% and 89.13% respectively) and F-PCA (89.96%, 90.10% and 89.13 respectively) where the reported average OA, AA and kc are comparable to those reported for the proposed approach. The improvement on the performance of the classifier (when fed with the outputs of F-PCA and KPCA) achieved by the proposed approach is not much (when considering the standard deviation reported in each case) and therefore considered not significant.

The classifier's performance is also reported in terms of Precision ($P$), Recall ($R$) and $F_1$ score ($F_1$) in Table 5.3 and Table 5.4. From Table 5.3 and Table 5.4, it can be observed that the proposed F-LDA continues to give the best classification performance in terms of $P$, $R$ and $F_1$. It can be noted in Table 5.3 that $P$, $R$ and $F_1$ all have the same value. This is usually the case when their average is computed using micro averaging [113]. It is also worth noting in Table 5.4, which shows the results of macro averaging, that some of the results in $P$, $R$ and $F_1$ are 'nan'. This occurs due to division by zero when computing $P$, $R$ and $F_1$ for each class [114]. Division by zero can occur when the following is equal to zero for any of the classes in the dataset (1) sum of true positives and false positives in the formula for computing $P$ (2) sum of true positives and false negatives in the formula for computing $R$. $F_1$ for a class will be set to 'nan' when either the recall or precision for that class is already 'nan'. Because macro averaging directly computes the average of all the classification results recorded for all classes, 'nan' is reported as the overall average since at least one of the values to be averaged is 'nan'. It should also be noted that in some of the cases reported, macro averaged $F_1$ is 'nan' while both the macro averaged precision and macro averaged recall are not. The reason for this is explained as follows: when the precision for any of the class is zero, the average precision can still be a valid number. This is

also the case for recall i.e. when the recall for any of the class is zero, the average recall can still be a valid number. But if both recall and precision are zero for a particular class, then the average $F_1$ for that class will be set to 'nan'. This will again set the macro averaged $F_1$ to 'nan' since at least one of the values to be averaged is 'nan'.

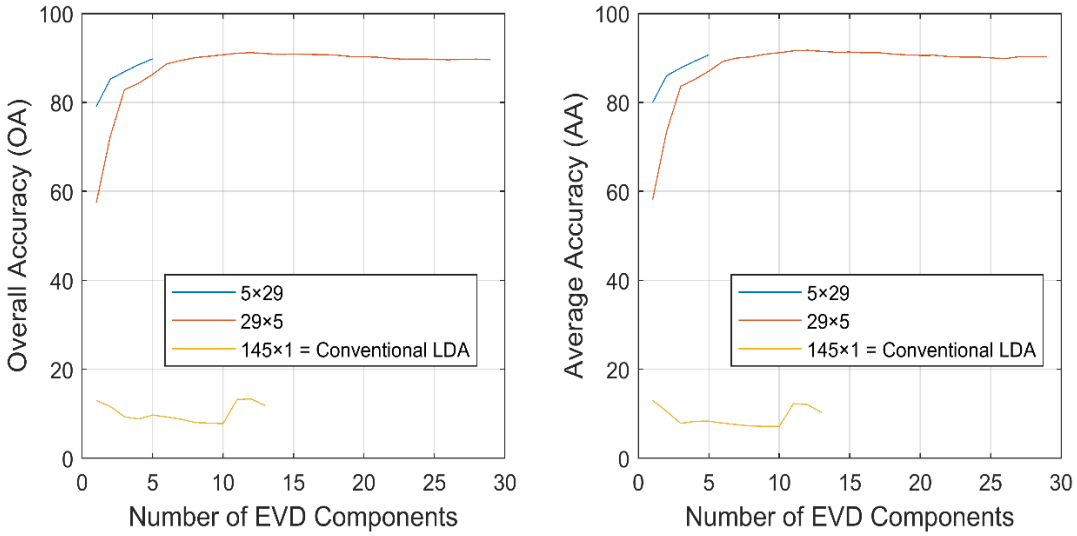



Figure 5.3 Classification results for the Botswana dataset using F-LDA. © 2020 IEEE

Table 5.2 Classification Results (Best Cases) for the Botswana Dataset (14 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA. © 2020 IEEE

| **Sample Shape** | **$V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$)** | **Best $d_{EVD}$** | **Number of Features ($d_{TOTAL}$)** | **OA (%)** | **AA (%)** | ***kc* (%)_** |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 145 | N/A | N/A | 145 | 87.60 ± 0.73 | 88.54 ± 0.75 | 86.56 ± 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 145 | 13 | 12 | 12 | 13.31 ± 5.14 | 13.04 ± 2.68 | 6.07 ± 0.03 |
| **Folded-LDA** | | | | | | |
| 1 × 145 | N/A | N/A | 145 | 87.60 ± 0.73 | 88.54 ± 0.75 | 86.56 ± 0.01 |
| 5 × 29 | 5 | 5 | 145 | 89.79 ± 1.03 | 90.68 ± 1.08 | 88.94 ± 0.01 |
| **29 × 5** | **29** | **12** | **60** | **91.17 ± 1.06** | **91.69 ± 1.15** | **90.43 ± 0.01** |
| 145 × 1 | 13 | 12 | 12 | 13.31 ± 5.14 | 13.04 ± 2.68 | 6.07 ± 0.03 |
| **2D-LDA** | | | | | | |
| **1 × 145** | **N/A** | **N/A** | **145** | **87.60 ± 0.73** | **88.54 ± 0.75** | **86.56 ± 0.01** |
| 5 × 29 | 5 | N/A | 29 | 79.46 ± 2.59 | 80.50 ± 2.91 | 77.75 ± 0.03 |
| 29 × 5 | 29 | N/A | 5 | 69.81 ± 5.59 | 70.67 ± 5.42 | 67.29 ± 0.06 |
| 145 × 1 | 13 | N/A | 1 | 14.15 ± 3.37 | 14.40 ± 3.38 | 7.44 ± 0.04 |
| **GDA** | | | | | | |
| 1 × 145 | 13 | 13 | 13 | 85.72 ± 1.04 | 85.97 ± 1.85 | 84.52 ± 0.01 |
| **NWFE** | | | | | | |
| 1 × 145 | 20 | 5 | 5 | 87.57 ± 1.77 | 88.10 ± 2.29 | 86.53 ± 0.02 |
| **KPCA** | | | | | | |
| 1 × 145 | 10 | 7 | 7 | 89.97 ± 1.15 | 90.41 ± 1.51 | 89.13 ± 0.01 |
| **Folded PCA** | | | | | | |
| **1 × 145** | **5** | **5** | **5** | **89.96 ± 0.91** | **90.10 ± 1.01** | **89.13 ± 0.01** |
| 5 × 29 | 5 | 2 | 10 | 87.27 ± 1.74 | 88.45 ± 1.85 | 86.21 ± 0.02 |
| 29 × 5 | 5 | 1 | 29 | 87.47 ± 1.07 | 88.47 ± 1.05 | 86.43 ± 0.01 |
| 145 × 1 | N/A | N/A | 145 | 87.60 ± 0.73 | 88.54 ± 0.75 | 86.56 ± 0.01 |

Table 5.3 Classification Results (Precision, Recall and $F_1$ Score using Micro Averaging) for the Botswana Dataset (14 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA.

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 145 | N/A | N/A | 145 | 87.60 + 0.01 | 87.60 + 0.01 | 87.60 + 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 145 | 13 | 12 | | 13.31 + 0.05 | 13.31 + 0.05 | 13.31 + 0.05 |
| **Folded-LDA** | | | | | | |
| 1 × 145 | N/A | N/A | 145 | 87.60 + 0.01 | 87.60 + 0.01 | 87.60 + 0.01 |
| 5 × 29 | 5 | 5 | 145 | 89.79 + 0.01 | 89.79 + 0.01 | 89.79 + 0.01 |
| **29 × 5** | **29** | **12** | **60** | **91.17 + 0.01** | **91.17 + 0.01** | **91.17 + 0.01** |
| 145 × 1 | 13 | 12 | 12 | 13.31 + 0.05 | 13.31 + 0.05 | 13.31 + 0.05 |
| **2D-LDA** | | | | | | |
| **1 × 145** | **N/A** | **N/A** | **145** | **87.60 + 0.01** | **87.60 + 0.01** | **87.60 +0.01** |
| 5 × 29 | 5 | N/A | 29 | 79.46 + 0.03 | 79.46 + 0.03 | 79.46 + 0.03 |
| 29 × 5 | 29 | N/A | 5 | 69.81 + 0.06 | 69.81 + 0.06 | 69.81 + 0.06 |
| 145 × 1 | 13 | N/A | 1 | 14.15 + 0.03 | 14.15 + 0.03 | 14.15 + 0.03 |
| **GDA** | | | | | | |
| 1 × 145 | 13 | 13 | 13 | 85.72 + 0.01 | 85.72 + 0.01 | 85.72 + 0.01 |
| **NWFE** | | | | | | |
| 1 × 145 | 20 | 5 | 5 | 87.57 + 0.02 | 87.57 + 0.02 | 87.57 + 0.02 |
| **KPCA** | | | | | | |
| 1 × 145 | 10 | 7 | 7 | 89.97 + 0.01 | 89.97 + 0.01 | 89.97 + 0.01 |
| **Folded PCA** | | | | | | |
| **1 × 145** | **5** | **5** | **5** | **89.96 + 0.01** | **89.96 + 0.01** | **89.96 + 0.01** |
| 5 × 29 | 5 | 2 | 10 | 87.27 + 0.02 | 87.27 + 0.02 | 87.27 + 0.02 |
| 29 × 5 | 5 | 1 | 29 | 87.47 + 0.01 | 87.47 + 0.01 | 87.47 + 0.01 |
| 145 × 1 | N/A | N/A | 145 | 87.60 + 0.01 | 87.60 + 0.01 | 87.60 + 0.01 |

Table 5.4 Classification Results (Precision, Recall and $F_1$ Score Using Macro Averaging) for the Botswana Dataset (14 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA[1].

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 145 | N/A | N/A | 145 | 88.81 + 0.01 | 88.54 + 0.01 | 88.46 + 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 145 | 13 | 2 | 2 | 23.16 + 0.05 | 13.04 + 0.03 | - |
| **Folded-LDA** | | | | | | |
| 1 × 145 | N/A | N/A | 145 | 88.81 + 0.01 | 88.54 + 0.01 | 88.46 + 0.01 |
| 5 × 29 | 5 | 5 | 145 | 90.71 + 0.01 | 90.68 + 0.01 | 90.50 + 0.01 |
| **29 × 5** | **29** | **12** | **60** | **92.35 + 0.01** | **91.69 + 0.01** | **91.82 + 0.01** |
| 145 × 1 | 13 | 2 | 2 | 23.16 + 0.05 | 13.04 + 0.03 | - |
| **2D-LDA** | | | | | | |
| **1 × 145** | **N/A** | **N/A** | **145** | **88.81 + 0.01** | **88.54 + 0.01** | **88.46 + 0.01** |
| 5 × 29 | 5 | N/A | 29 | 80.96 + 0.03 | 80.50 + 0.03 | 80.28 + 0.03 |
| 29 × 5 | 29 | N/A | 5 | 72.07 + 0.05 | 70.67 + 0.05 | 70.60 + 0.06 |
| 145 × 1 | 13 | N/A | 1 | - | 14.40 + 0.03 | - |
| **GDA** | | | | | | |
| 1 × 145 | 13 | 13 | 13 | 87.85 + 0.01 | 85.97 + 0.02 | 86.49 + 0.01 |
| **NWFE** | | | | | | |
| 1 × 145 | 20 | 5 | 5 | 88.87 +0.01 | 88.10 + 0.02 | 88.16 + 0.02 |
| **KPCA** | | | | | | |
| 1 × 145 | 10 | 9 | 9 | 91.02 + 0.01 | 90.41 + 0.02 | 90.50 + 0.01 |
| **Folded PCA** | | | | | | |
| **1 × 145** | **5** | **5** | **5** | **90.78 +0.01** | **90.10 + 0.01** | **90.24 + 0.01** |
| 5 × 29 | 5 | 2 | 10 | 88.49 + 0.02 | 88.45 +0.02 | 88.25 + 0.02 |
| 29 × 5 | 5 | 1 | 29 | 88.71 + 0.01 | 88.47 + 0.01 | 88.37 + 0.01 |
| 145 × 1 | N/A | N/A | 145 | 88.81 + 0.01 | 88.54 + 0.01 | 88.46 + 0.01 |

[1] A dash is used in Table 5.4 where the result is not a number i.e. it cannot be computed. This happens because either the true positives and false positives are both zero or the true positives and false negatives are both zero.

2. Classification Accuracy for the Pavia Center Dataset

The F-LDA operation is also carried out on the Pavia Center dataset. The plots of OA and AA against $d_{\mathrm{EVD}}$ are obtained for each of the configurations considered and presented in Figure 5.4. The maximum OA and AA are extracted from Figure 5.4 and the classification results, including the maximum *kc* presented in Table 5.5. It can be observed that the highest OA, OA and *kc* are obtained when the configuration is set to $17 \times 6$. This is an improvement on the classification accuracy obtained when the SVM model was trained using the full feature space. The case $102 \times 1$ simplifies the proposed F-LDA to the conventional LDA, while another case $G = 1$ (i.e. $1 \times 102$) simplifies it to the full feature space. The outputs of 2D LDA, GDA, NWFE, KPCA and F-PCA when applied on the Pavia Center dataset are also used to train the SVM model. The classification results obtained are presented in Table 5.5. From Table 5.5, it can be observed that the proposed F-LDA continues to give the best classification performance in terms of OA, AA and *kc*. When considering the standard deviation, from Table 5.5, it can be observed that the lower bands of the OA, AA and kc achieved by the proposed F-LDA (96.63 ± 0.37, 88.06 ± 1.51 and 95.23 ± 0.01) are higher than the average classification results reported for the other techniques except for F-PCA where the reported average OA (96.17) and kc (94.58) are comparable to those reported for the proposed approach. Also, the average AA (86.87) reported for F-PCA is slightly higher than the lower band of the AA (88.06 ± 1.51) achieved by the proposed approach. The improvement on the performance of the classifier (when fed with the output of F-PCA) achieved by the proposed approach can also be considered not significant in this case.

The classifier's performance is also reported in terms of *P, R* and $F_1$ in Table 5.6 and Table 5.7. From Table 5.6 and Table 5.7, it can be seen that the proposed F-LDA consistently produce better classification performance (higher *P, R* and $F_1$) than the other techniques used to benchmark its performance. It can be noted in Table 5.6 that *P, R* and $F_1$ all have the same value. This is usually the case when their average is computed using micro averaging [113]. It is also worth noting in Table 5.7 that some of the results in *P, R* and $F_1$ are 'nan'. This occurs due to division by zero when computing them [114].

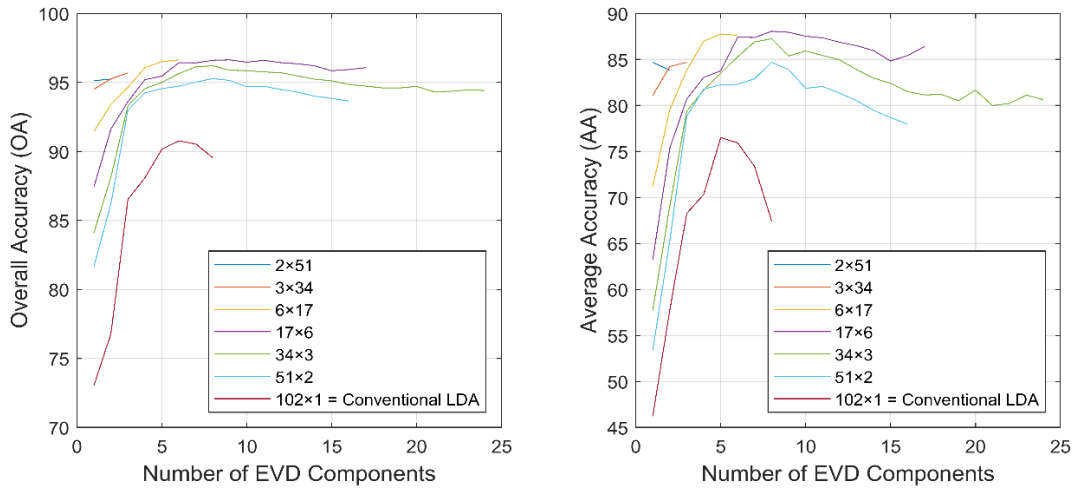



Figure 5.4 Classification results for the Pavia Center dataset using F-LDA. © 2020 IEEE

Table 5.5 Classification Results (Best Cases) for the Pavia Center Dataset (9 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA. © 2020 IEEE

| **Sample Shape** | **$V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$)** | **Best $d_{EVD}$** | **Number of Features ($d_{TOTAL}$)** | **OA (%)** | **AA (%)** | ***kc* (%)_** |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 102 | N/A | N/A | 102 | 95.44 ± 0.53 | 84.85 ± 1.51 | 93.54 ± 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 102 | 8 | 6 | 6 | 90.76 ± 1.63 | 76.50 ± 3.33 | 86.92 ± 0.02 |
| **Folded-LDA** | | | | | | |
| 1×102 | N/A | N/A | 102 | 95.44 ± 0.53 | 84.85 ± 1.51 | 93.54 ± 0.01 |
| 2×51 | 2 | 2 | 102 | 95.23 ± 0.65 | 84.69 ± 1.90 | 93.24 ± 0.01 |
| 3×34 | 3 | 3 | 102 | 95.69 ± 0.56 | 84.69 ± 3.04 | 93.90 ± 0.01 |
| 6×17 | 6 | 6 | 102 | 96.63 ± 0.39 | 87.76 ± 2.11 | 95.23 ± 0.01 |
| **17×6** | **17** | **9** | **54** | **96.63 ± 0.37** | **88.06 ± 1.51** | **95.23 ± 0.01** |
| 34×3 | 24 | 8 | 24 | 96.21 ± 0.26 | 87.25 ± 0.85 | 94.63 ± 0.00 |
| 51×2 | 16 | 8 | 16 | 95.28 ± 0.53 | 84.69 ± 1.54 | 93.30 ± 0.01 |
| 102×1 | 8 | 6 | 6 | 90.76 ± 1.63 | 76.50 ± 3.33 | 86.92 ± 0.02 |
| **2D-LDA** | | | | | | |
| **1×102** | **N/A** | **N/A** | **102** | **95.44 ± 0.53** | **84.85 ± 1.51** | **93.54 ± 0.01** |
| 2×51 | 2 | N/A | 51 | 93.49 ± 1.45 | 81.31 ± 3.20 | 90.74 ± 0.02 |
| 3×34 | 3 | N/A | 34 | 92.13 ± 0.68 | 74.40 ± 2.72 | 88.77 ± 0.01 |
| 6×17 | 6 | N/A | 17 | 88.90 ± 4.23 | 66.30 ± 10.03 | 83.90 ± 0.06 |
| 17×6 | 17 | N/A | 6 | 86.27 ± 3.27 | 61.88 ± 8.42 | 80.22 ± 0.05 |
| 34×3 | 24 | N/A | 3 | 77.92 ± 7.11 | 47.60 ± 9.42 | 66.89 ± 0.12 |
| 51×2 | 16 | N/A | 2 | 75.64 ± 7.91 | 44.38 ± 8.65 | 63.51 ± 0.13 |
| 102×1 | 8 | N/A | 1 | 66.20 ± 8.87 | 37.62 ± 5.40 | 52.00 ± 0.13 |
| **GDA** | | | | | | |
| 1 × 102 | 8 | 8 | 8 | 95.73 ± 0.42 | 85.35 ± 1.93 | 93.95 ± 0.01 |
| **NWFE** | | | | | | |
| 1 × 102 | 20 | 4 | 4 | 94.73 ± 0.58 | 81.71 ± 2.98 | 92.52 ± 0.01 |
| **KPCA** | | | | | | |
| 1 × 102 | 10 | 10 | 10 | 95.49 ± 0.51 | 84.60 ± 2.13 | 93.61 ± 0.01 |
| **Folded PCA** | | | | | | |
| 1×102 | 5 | 3 | 3 | 95.18 ± 0.46 | 83.26 ± 2.48 | 93.17 ± 0.01 |
| 2×51 | 5 | 3 | 6 | 95.98 ± 0.44 | 86.09 ± 1.42 | 94.31 ± 0.01 |
| **3×34** | **5** | **3** | **9** | **96.17 ± 0.51** | **86.87 ± 2.18** | **94.58 ± 0.01** |
| 6×17 | 5 | 2 | 12 | 95.96 ± 0.56 | 86.35 ± 2.00 | 94.27 ± 0.01 |
| 17×6 | 5 | 2 | 34 | 95.50 ± 0.45 | 84.96 ± 1.13 | 93.63 ± 0.01 |
| 34×3 | 5 | 2 | 68 | 95.08 ± 0.39 | 83.50 ± 3.24 | 93.02 ± 0.01 |
| 51×2 | 5 | 1 | 51 | 95.06 ± 0.93 | 83.58 ± 3.21 | 92.99 ± 0.01 |
| 102×1 | N/A | N/A | 102 | 95.44 ± 0.53 | 84.85 ± 1.51 | 93.54 ± 0.01 |

Table 5.6 Classification Results (Precision, Recall and $F_1$ Score Using Micro Averaging) for the Pavia Center Dataset (9 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA.

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 102 | N/A | N/A | 102 | 95.44 + 0.01 | 95.44 + 0.01 | 95.44 + 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 102 | 8 | 6 | 6 | 90.76 + 0.02 | 90.76 + 0.02 | 90.76 + 0.02 |
| **Folded-LDA** | | | | | | |
| 1×102 | N/A | N/A | 102 | 95.44 + 0.01 | 95.44 + 0.01 | 95.44 + 0.01 |
| 2×51 | 2 | 2 | 102 | 95.23 + 0.01 | 95.23 + 0.01 | 95.23 + 0.01 |
| 3×34 | 3 | 3 | 102 | 95.69 + 0.01 | 95.69 + 0.01 | 95.69 + 0.01 |
| **6×17** | **6** | **6** | **102** | **96.63 + 0.00** | **96.63 + 0.00** | **96.63 + 0.00** |
| **17×6** | **17** | **9** | **54** | **96.63 + 0.00** | **96.63 + 0.00** | **96.63 + 0.00** |
| 34×3 | 24 | 8 | 24 | 96.21 + 0.00 | 96.21 + 0.00 | 96.21 + 0.00 |
| 51×2 | 16 | 8 | 16 | 95.28 + 0.01 | 95.28 + 0.01 | 95.28 + 0.01 |
| 102×1 | 8 | 6 | 6 | 90.76 + 0.02 | 90.76 + 0.02 | 90.76 + 0.02 |
| **2D-LDA** | | | | | | |
| **1×102** | **N/A** | **N/A** | **102** | **95.44 + 0.01** | **95.44 + 0.01** | **95.44 + 0.01** |
| 2×51 | 2 | N/A | 51 | 93.49 + 0.01 | 93.49 + 0.01 | 93.49 + 0.01 |
| 3×34 | 3 | N/A | 34 | 92.13 + 0.01 | 92.13 + 0.01 | 92.13 + 0.01 |
| 6×17 | 6 | N/A | 17 | 88.90 + 0.04 | 88.90 + 0.04 | 88.90 + 0.04 |
| 17×6 | 17 | N/A | 6 | 86.27 + 0.03 | 86.27 + 0.03 | 86.27 + 0.03 |
| 34×3 | 24 | N/A | 3 | 77.92 + 0.07 | 77.92 + 0.07 | 77.92 + 0.07 |
| 51×2 | 16 | N/A | 2 | 75.64 + 0.08 | 75.64 + 0.08 | 75.64 + 0.08 |
| 102×1 | 8 | N/A | 1 | 66.20 + 0.09 | 66.20 + 0.09 | 66.20 + 0.09 |
| **GDA** | | | | | | |
| 1 × 102 | 8 | 8 | 8 | 95.73 + 8.00 | 95.73 + 8.00 | 95.73 + 8.00 |
| **NWFE** | | | | | | |
| 1 × 102 | 20 | 4 | 4 | 94.73 + 0.01 | 94.73 + 0.01 | 94.73 + 0.01 |
| **KPCA** | | | | | | |
| 1 × 102 | 10 | 10 | 10 | 95.49 + 0.01 | 95.49 + 0.01 | 95.49 + 0.01 |
| **Folded PCA** | | | | | | |
| 1×102 | 5 | 3 | 3 | 95.18 + 0.00 | 95.18 + 0.00 | 95.18 + 0.00 |
| 2×51 | 5 | 3 | 6 | 95.98 + 0.00 | 95.98 + 0.00 | 95.98 + 0.00 |
| **3×34** | **5** | **3** | **9** | **96.17 + 0.01** | **96.17 + 0.01** | **96.17 + 0.01** |
| 6×17 | 5 | 2 | 12 | 95.96 + 0.01 | 95.96 + 0.01 | 95.96 + 0.01 |
| 17×6 | 5 | 2 | 34 | 95.50 + 0.00 | 95.50 + 0.00 | 95.50 + 0.00 |
| 34×3 | 5 | 2 | 68 | 95.08 + 0.00 | 95.08 + 0.00 | 95.08 + 0.00 |
| 51×2 | 5 | 1 | 51 | 95.06 + 0.01 | 95.06 + 0.01 | 95.06 + 0.01 |
| 102×1 | N/A | N/A | 102 | 95.44 + 0.01 | 95.44 + 0.01 | 95.44 + 0.01 |

Table 5.7 Classification Results (Precision, Recall and $F_1$ Score Using Macro Averaging) for the Pavia Center Dataset (9 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA[2].

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 102 | N/A | N/A | 102 | 86.82 + 0.02 | 84.85 + 0.02 | 85.41 + 0.02 |
| **Conventional LDA** | | | | | | |
| 1 × 102 | 8 | 7 | 7 | 77.18 + 0.03 | 76.50 + 0.03 | 75.61 + 0.04 |
| **Folded-LDA** | | | | | | |
| 1×102 | N/A | N/A | 102 | 86.82 + 0.02 | 84.85 + 0.02 | 85.41 + 0.02 |
| 2×51 | 2 | 1 | 51 | 87.36 + 0.01 | 84.69 + 0.02 | 85.58 + 0.02 |
| 3×34 | 3 | 2 | 68 | 86.31 + 0.02 | 84.69 + 0.03 | 84.68 + 0.03 |
| 6×17 | 6 | 6 | 102 | 89.33 + 0.01 | 87.76 + 0.02 | 87.98 + 0.02 |
| **17×6** | **17** | **11** | **66** | **89.86 + 0.01** | **88.06 + 0.02** | **88.24 + 0.02** |
| 34×3 | 24 | 7 | 21 | 88.03 + 0.02 | 87.25 + 0.01 | 87.37 + 0.01 |
| 51×2 | 16 | 7 | 14 | 87.38 + 0.02 | 84.69 + 0.02 | 85.18 + 0.02 |
| 102×1 | 2 | 7 | 7 | 77.18 + 0.03 | 76.50 + 0.03 | 75.61 + 0.04 |
| **2D-LDA** | | | | | | |
| **1×102** | **N/A** | **N/A** | **102** | **86.82 + 0.02** | **84.85 +0.02** | **85.41 + 0.02** |
| 2×51 | 2 | N/A | 51 | 85.60 + 0.03 | 81.31 + 0.03 | 82.91 + 0.03 |
| 3×34 | 3 | N/A | 34 | 78.82 + 0.03 | 74.40 + 0.03 | - |
| 6×17 | 6 | N/A | 17 | - | 66.30 +0.10 | - |
| 17×6 | 17 | N/A | 6 | - | 61.88 + 0.08 | - |
| 34×3 | 24 | N/A | 3 | - | 47.60 + 0.09 | - |
| 51×2 | 16 | N/A | 2 | - | 44.38 + 0.09 | - |
| 102×1 | 8 | N/A | 1 | - | 37.62 + 0.05 | - |
| **GDA** | | | | | | |
| 1 × 102 | 8 | 8 | 8 | 87.89 + 0.01 | 85.35 + 0.02 | 86.12 + 0.02 |
| **NWFE** | | | | | | |
| 1 × 102 | 20 | 4 | 4 | 84.53 + 0.02 | 81.71 + 0.03 | 82.34 + 0.02 |
| **KPCA** | | | | | | |
| 1 × 102 | 10 | 6 | 6 | 86.45 + 0.02 | 84.60 + 0.02 | 84.83 + 0.02 |
| **Folded PCA** | | | | | | |
| 1×102 | 5 | 3 | 3 | 85.63 + 0.01 | 83.26 + 0.02 | 83.68 + 0.02 |
| 2×51 | 5 | 4 | 8 | 87.60 + 0.01 | 86.09 + 0.01 | 86.51 + 0.01 |
| **3×34** | **5** | **3** | **9** | **88.59 + 0.01** | **86.87 + 0.02** | **87.32 + 0.02** |
| 6×17 | 5 | 2 | 12 | 88.04 + 0.02 | 86.35 + 0.02 | 86.89 + 0.02 |
| 17×6 | 5 | 2 | 34 | 86.19 + 0.01 | 84.96 + 0.01 | 85.35 + 0.01 |
| 34×3 | 5 | 1 | 34 | 86.25 + 0.02 | 83.50 +0.03 | 84.23 + 0.03 |
| 51×2 | 5 | 1 | 51 | 86.42 + 0.01 | 83.58 + 0.03 | 84.32 + 0.02 |
| 102×1 | N/A | N/A | 102 | 86.82 + 0.02 | 84.85 + 0.02 | 85.41 + 0.02 |

[2] A dash is used in Table 5.7 where the result is not a number i.e. it cannot be computed. This happens because either the true positives and false positives are both zero or the true positives and false negatives are both zero.

3. Classification Accuracy for the Salinas-A Dataset

The proposed F-LDA is also applied on the Salinas-A data and the plots of OA and AA against $d_{\mathrm{EVD}}$ obtained for each of the configurations considered are presented in Figure 5.5. The maximum OA and AA are extracted from these plots and the classification results, including the maximum *kc* presented in Table 5.8. One can see in Table 5.8 that the highest OA, AA and *kc* are achieved when the configuration was set to $17 \times 12$ (99.19%). This is higher than the 98.60% attained when the full feature space was used to train the SVM model. Once more, the proposed F-LDA simplifies to the conventional LDA and the full feature space when the configuration is set to $204 \times 1$ (i.e., when $G = f$), and when $G = 1$ respectively. The proposed F-LDA gives the best classification performance in terms of OA, AA, and *kc* as can be seen in Table 5.8. When considering the standard deviation, from Table 5.8, it can be observed that the lower bands of the OA, AA and kc achieved by the proposed F-LDA (99.19 ± 0.29, 98.94 ± 0.45 and 98.98 ± 0.00) are comparable to (and slightly lower in some cases) those achieved by the other techniques namely F-PCA (99.06, 98.86 and 98.82), KPCA (99.01, 98.85 and 98.76), GDA (98.77, 98.49 and 98.46), and 2D LDA (98.60, 98.38 and 98.24). Based on these, the improvement on the performance of the classifier achieved (when fed with the outputs of other techniques) by the proposed approach can be considered not significant.

The classifier's performance is also reported in terms of *P, R* and $F_1$ in Table 5.9 and Table 5.10. From Table 5.9 and Table 5.10, it can be seen that the proposed F-LDA consistently produce better classification performance (higher *P, R* and $F_1$) than the other techniques used to benchmark its performance. It can be noted in Table 5.9 that *P, R* and $F_1$ all have the same value. This is usually the case when their average is

computed using micro averaging [113]. It is also worth noting in Table 5.10 that some of the results in $P$, $R$ and $F_1$ are 'nan'. This occur due to division by zero when computing them [114].

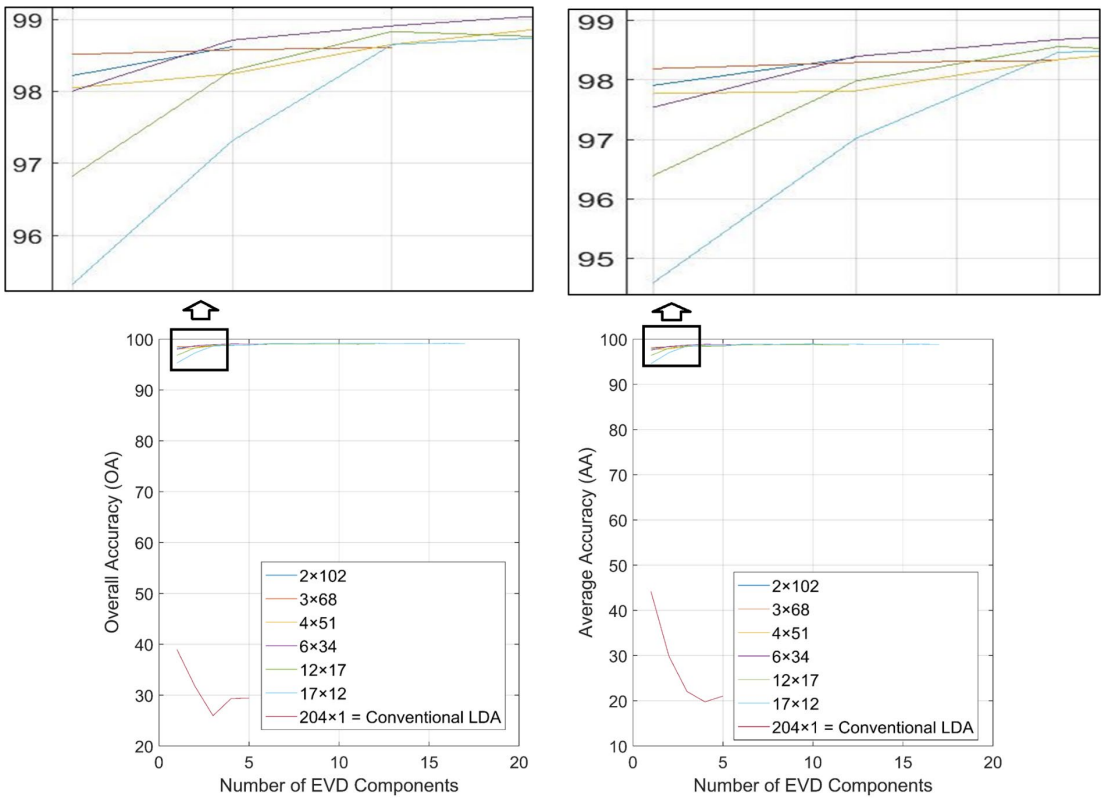



Figure 5.5 Classification results for the Salinas-A dataset using F-LDA

Table 5.8 Classification Results (Best Cases) for the Salinas-A Dataset (6 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA

| **Sample Shape** | **$V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$)** | **Best $d_{EVD}$** | **Number of Features ($d_{TOTAL}$)** | **OA (%)** | **AA (%)** | ***kc* (%)_** |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 204 | N/A | N/A | 204 | 98.60 ± 0.21 | 98.38 ± 0.25 | 98.24 ± 0.00 |
| **Conventional LDA** | | | | | | |
| 1 × 204 | 5 | 1 | 1 | 38.95 ± 8.01 | 44.20 ± 7.04 | 25.97 ± 0.09 |
| **Folded-LDA** | | | | | | |
| 1 × 204 | N/A | N/A | 204 | 98.60 ± 0.21 | 98.38 ± 0.25 | 98.24 ± 0.00 |
| 2 × 102 | 2 | 2 | 204 | 98.62 ± 0.23 | 98.38 ± 0.28 | 98.27 ± 0.00 |
| 3 × 68 | 3 | 3 | 204 | 98.61 ± 0.32 | 98.33 ± 0.40 | 98.26 ± 0.00 |
| 4 × 51 | 4 | 4 | 204 | 98.88 ± 0.27 | 98.66 ± 0.40 | 98.60 ± 0.00 |
| 6 × 34 | 6 | 4 | 136 | 99.06 ± 0.34 | 98.86 ± 0.49 | 98.82 ± 0.00 |
| 12 × 17 | 12 | 10 | 170 | 99.05 ± 0.24 | 98.81 ± 0.37 | 98.81 ± 0.00 |
| **17 × 12** | **17** | **10** | **120** | **99.19 ± 0.29** | **98.94 ± 0.45** | **98.98 ± 0.00** |
| 204 × 1 | 5 | 1 | 1 | 38.95 ± 8.01 | 44.20 ± 7.04 | 25.97 ± 0.09 |
| **2D-LDA** | | | | | | |
| **1 × 204** | **N/A** | **N/A** | **204** | **98.60 ± 0.21** | **98.38 ± 0.25** | **98.24 ± 0.00** |
| 2 × 102 | 2 | N/A | 102 | 98.21 ± 0.33 | 97.91 ± 0.41 | 97.75 ± 0.00 |
| 3 × 68 | 3 | N/A | 68 | 98.46 ± 0.30 | 98.17 ± 0.37 | 98.07 ± 0.00 |
| 4 × 51 | 4 | N/A | 51 | 97.94 ± 0.41 | 97.67 ± 0.51 | 97.42 ± 0.01 |
| 6 × 34 | 6 | N/A | 34 | 98.32 ± 0.52 | 98.06 ± 0.62 | 97.90 ± 0.01 |
| 12 × 17 | 12 | N/A | 17 | 97.29 ± 0.61 | 96.89 ± 0.90 | 96.61 ± 0.01 |
| 17 × 12 | 17 | N/A | 12 | 95.83 ± 1.96 | 95.24 ± 2.32 | 94.76 ± 0.02 |
| 204 × 1 | 5 | N/A | 1 | 39.05 ± 8.09 | 43.41 ± 7.62 | 25.33 ± 0.09 |
| **GDA** | | | | | | |
| 1 × 204 | 5 | 4 | 4 | 98.77 ± 0.31 | 98.49 ± 0.46 | 98.46 ± 0.00 |
| **NWFE** | | | | | | |
| 1 × 204 | 20 | 20 | 20 | 98.71 ± 0.33 | 98.38 ± 0.48 | 98.39 ± 0.00 |
| **KPCA** | | | | | | |
| 1 × 204 | 10 | 7 | 7 | 99.01 ± 0.25 | 98.85 ± 0.33 | 98.76 ± 0.00 |
| **Folded PCA** | | | | | | |
| 1 × 204 | 5 | 5 | 5 | 98.85 ± 0.24 | 98.61 ± 0.36 | 98.56 ± 0.00 |
| 2 × 102 | 5 | 4 | 8 | 98.93 ± 0.25 | 98.66 ± 0.37 | 98.66 ± 0.00 |
| 3 × 68 | 5 | 4 | 12 | 98.63 ± 0.47 | 98.38 ± 0.40 | 98.28 ± 0.01 |
| 4 × 51 | 5 | 4 | 16 | 98.97 ± 0.24 | 98.74 ± 0.23 | 98.71 ± 0.00 |
| **6 × 34** | **5** | **4** | **24** | **99.06 ± 0.32** | **98.86 ± 0.39** | **98.82 ± 0.00** |
| 12 × 17 | 5 | 3 | 36 | 99.04 ± 0.16 | 98.79 ± 0.25 | 98.80 ± 0.00 |
| 17 × 12 | 5 | 3 | 51 | 98.86 ± 0.26 | 98.59 ± 0.37 | 98.56 ± 0.00 |
| 204 × 1 | N/A | N/A | 204 | 98.60 ± 0.21 | 98.38 ± 0.25 | 98.24 ± 0.00 |

Table 5.9 Classification Results (Precision, Recall and $F_1$ Score Using Micro Averaging) for the Salinas-A Dataset (6 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 204 | N/A | N/A | 204 | 98.60 + 0.00 | 98.60 + 0.00 | 98.60 + 0.00 |
| **Conventional LDA** | | | | | | |
| 1 × 204 | 5 | 1 | 1 | 38.95 + 0.08 | 38.95 + 0.08 | 38.95 + 0.08 |
| **Folded-LDA** | | | | | | |
| 1 × 204 | N/A | N/A | 204 | 98.60 + 0.00 | 98.60 + 0.00 | 98.60 + 0.00 |
| 2 × 102 | 2 | 2 | 204 | 98.62 + 0.00 | 98.62 + 0.00 | 98.62 + 0.00 |
| 3 × 68 | 3 | 3 | 204 | 98.61 + 0.00 | 98.61 + 0.00 | 98.61 + 0.00 |
| 4 × 51 | 4 | 4 | 204 | 98.88 + 0.00 | 98.88 + 0.00 | 98.88 + 0.00 |
| 6 × 34 | 6 | 4 | 136 | 99.06 + 0.00 | 99.06 + 0.00 | 99.06 + 0.00 |
| 12 × 17 | 12 | 10 | 170 | 99.05 + 0.00 | 99.05 + 0.00 | 99.05 + 0.00 |
| **17 × 12** | **17** | **10** | **120** | **99.19 + 0.00** | **99.19 + 0.00** | **99.19 + 0.00** |
| 204 × 1 | 5 | 1 | 1 | 38.95 + 0.08 | 38.95 + 0.08 | 38.95 + 0.08 |
| **2D-LDA** | | | | | | |
| **1 × 204** | **N/A** | **N/A** | **204** | **98.60 + 0.00** | **98.60 + 0.00** | **98.60 + 0.00** |
| 2 × 102 | 2 | N/A | 102 | 98.21 + 0.00 | 98.21 + 0.00 | 98.21 + 0.00 |
| 3 × 68 | 3 | N/A | 68 | 98.46 + 0.00 | 98.46 + 0.00 | 98.46 + 0.00 |
| 4 × 51 | 4 | N/A | 51 | 97.94 + 0.00 | 97.94 + 0.00 | 97.94 + 0.00 |
| 6 × 34 | 6 | N/A | 34 | 98.32 + 0.01 | 98.32 + 0.01 | 98.32 + 0.01 |
| 12 × 17 | 12 | N/A | 17 | 97.29 + 0.01 | 97.29 + 0.01 | 97.29 + 0.01 |
| 17 × 12 | 17 | N/A | 12 | 95.83 + 0.02 | 95.83 + 0.02 | 95.83 + 0.02 |
| 204 × 1 | 5 | N/A | 1 | 39.05 + 0.08 | 39.05 + 0.08 | 39.05 + 0.08 |
| **GDA** | | | | | | |
| 1 × 204 | 5 | 4 | 4 | 98.77 + 0.00 | 98.77 + 0.00 | 98.77 + 0.00 |
| **NWFE** | | | | | | |
| 1 × 204 | 20 | 1 | 1 | 25.11 + 0.00 | 25.11 + 0.00 | 25.11 + 0.00 |
| **KPCA** | | | | | | |
| 1 × 204 | 10 | 7 | 7 | 99.01 + 0.00 | 99.01 + 0.00 | 99.01 + 0.00 |
| **Folded PCA** | | | | | | |
| 1 × 204 | 5 | 5 | 5 | 98.85 + 0.00 | 98.85 + 0.00 | 98.85 + 0.00 |
| 2 × 102 | 5 | 4 | 8 | 98.93 + 0.00 | 98.93 + 0.00 | 98.93 + 0.00 |
| 3 × 68 | 5 | 4 | 12 | 98.63 + 0.00 | 98.63 + 0.00 | 98.63 + 0.00 |
| 4 × 51 | 5 | 4 | 16 | 98.97 + 0.00 | 98.97 + 0.00 | 98.97 + 0.00 |
| **6 × 34** | **5** | **4** | **24** | **99.06 + 0.00** | **99.06 + 0.00** | **99.06 + 0.00** |
| 12 × 17 | 5 | 3 | 36 | 99.04 + 0.00 | 99.04 + 0.00 | 99.04 + 0.00 |
| 17 × 12 | 5 | 3 | 51 | 98.86 + 0.00 | 98.86 + 0.00 | 98.86 + 0.00 |
| 204 × 1 | N/A | N/A | 204 | 98.60 + 0.00 | 98.60 + 0.00 | 98.60 + 0.00 |

Table 5.10 Classification Results (Precision, Recall and $F_1$ Score Using Macro Averaging) for the Salinas-A Dataset (6 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA[3].

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 204 | N/A | N/A | 204 | 98.84 + 0.00 | 98.38 + 0.00 | 98.60 + 0.00 |
| **Conventional LDA** | | | | | | |
| 1 × 204 | 5 | 4 | 4 | 86.31 + 0.02 | 44.20 + 0.07 | 29.52 + 0.05 |
| **Folded-LDA** | | | | | | |
| 1 × 204 | N/A | N/A | 204 | 98.84 + 0.00 | 98.38 + 0.00 | 98.60 + 0.00 |
| 2 × 102 | 2 | 2 | 204 | 98.84 + 0.00 | 98.38 + 0.00 | 98.60 + 0.00 |
| 3 × 68 | 3 | 3 | 204 | 98.86 + 0.00 | 98.33 + 0.00 | 98.58 + 0.00 |
| 4 × 51 | 4 | 4 | 204 | 99.06 + 0.00 | 98.66 + 0.00 | 98.85 + 0.00 |
| 6 × 34 | 6 | 5 | 170 | 99.20 + 0.00 | 98.86 + 0.00 | 99.00 + 0.00 |
| 12 × 17 | 12 | 8 | 136 | 99.22 + 0.00 | 98.81 + 0.00 | 99.00 + 0.00 |
| **17 × 12** | **17** | **14** | **168** | **99.36 + 0.00** | **98.94 + 0.00** | **99.13 + 0.00** |
| 204 × 1 | 5 | 4 | 4 | 86.31 + 0.02 | 44.20 + 0.07 | 29.52 + 0.05 |
| **2D-LDA** | | | | | | |
| **1 × 204** | **N/A** | **N/A** | **204** | **98.84 + 0.00** | **98.38 + 0.00** | **98.60 + 0.00** |
| 2 × 102 | 2 | N/A | 102 | 98.50 + 0.00 | 97.91 + 0.00 | 98.18 + 0.00 |
| 3 × 68 | 3 | N/A | 68 | 98.67 + 0.00 | 98.17 + 0.00 | 98.40 + 0.00 |
| 4 × 51 | 4 | N/A | 51 | 98.12 + 0.00 | 97.67 + 0.01 | 97.88 + 0.00 |
| 6 × 34 | 6 | N/A | 34 | 98.47 + 0.01 | 98.06 + 0.01 | 98.25 + 0.01 |
| 12 × 17 | 12 | N/A | 17 | 97.35 + 0.01 | 96.89 + 0.01 | 97.09 + 0.01 |
| 17 × 12 | 17 | N/A | 12 | 95.92 + 0.02 | 95.24 + 0.02 | 95.50 + 0.02 |
| 204 × 1 | 5 | N/A | 1 | 43.97 + 0.05 | 43.41 + 0.08 | 41.71 + 0.07 |
| **GDA** | | | | | | |
| 1 × 204 | 5 | 4 | 4 | 98.79 + 0.00 | 98.49 + 0.00 | 98.62 + 0.00 |
| **NWFE** | | | | | | |
| 1 × 204 | 20 | 1 | 1 | - | 16.67 + 0.00 | - |
| **KPCA** | | | | | | |
| 1 × 204 | 10 | 7 | 7 | 99.09 + 0.00 | 98.85 + 0.00 | 98.96 + 0.00 |
| **Folded PCA** | | | | | | |
| 1 × 204 | 5 | 5 | 5 | 98.98 + 0.00 | 98.61 + 0.00 | 98.78 + 0.00 |
| 2 × 102 | 5 | 4 | 8 | 99.13 + 0.00 | 98.66 + 0.00 | 98.88 + 0.00 |
| 3 × 68 | 5 | 4 | 12 | 98.83 + 0.00 | 98.38 + 0.00 | 98.58 + 0.00 |
| 4 × 51 | 5 | 4 | 16 | 99.15 + 0.00 | 98.74 + 0.00 | 98.93 + 0.00 |
| **6 × 34** | 5 | 4 | 24 | 99.20 + 0.00 | **98.86 + 0.00** | **99.02 + 0.00** |
| **12 × 17** | 5 | 4 | 48 | **99.26 + 0.00** | 98.79 + 0.00 | 99.01 + 0.00 |
| 17 × 12 | 5 | 3 | 51 | 99.14 + 0.00 | 98.59 + 0.00 | 98.85 + 0.00 |
| 204 × 1 | N/A | N/A | 204 | 98.84 + 0.00 | 98.38 + 0.00 | 98.60 + 0.00 |

[3] A dash is used in Table 5.10 where the result is not a number i.e. it cannot be computed. This happens because either the true positives and false positives are both zero or the true positives and false negatives are both zero.

4. Classification Accuracy for the Indian Pine Dataset

Plots of OA and AA against $d_{\mathrm{EVD}}$ obtained for each of the configurations considered when the proposed F-LDA is applied on the Indian Pine dataset and related classification results are presented in Figure 5.6 and Table 5.11 respectively. Lowest OA, AA and *kc* are obtained when the configuration is set to $200 \times 1$. The highest OA and AA are attained when the configurations are set to $20 \times 10$ and $8 \times 25$, respectively. It can be observed that the configurations $20 \times 10$ and $8 \times 25$ both produce the highest *kc*. The highest classification results (OA, AA and *kc*) reported for the proposed F-LDA are much higher than those attained by the SVM model when it was trained using the LDA features. Again, the case $200 \times 1$ (i.e., when $G = f$) simplifies the proposed F-LDA to the conventional LDA while another case $1 \times 200$ ($G = f$) simplifies it to the full feature space. The 2D LDA, GDA, NWFE, KPCA and F-PCA are also applied to reduce the dimensionality of the Indian Pine dataset, and the proposed F-LDA achieves the best classification performance again. When considering the standard deviation, from Table 5.11, it can be observed that the lower bands of the OA, AA and kc achieved by the proposed F-LDA ($85.57 \pm 0.98$, $80.24 \pm 3.22$ and $83.49 \pm 0.01$) are higher than the average classification results reported for the other techniques except for F-PCA where the reported average OA (84.10) is comparable to those reported for the proposed approach. Also, the average AA (79.02) reported for F-PCA can be seen to be higher than the lower band of the AA (81.80) achieved by the proposed approach. The improvement on the performance of the classifier (when fed with the output of F-PCA) achieved by the proposed approach can also be considered not significant in this case.

The classifier's performance is also reported in terms of *P, R* and $F_1$ in Table 5.12 and Table 5.13. From Table 5.12 and Table 5.13, it can be observed that the proposed F-LDA continues to give the best classification performance in terms of *P, R* and $F_1$. It can be noted in Table 5.12 that *P, R* and $F_1$ all have the same value. This is usually the case when their average is computed using micro averaging [113]. It is also worth noting in Table 5.13 that some of the results in *P, R* and $F_1$ are 'nan'. This occur due to division by zero when computing them [114].

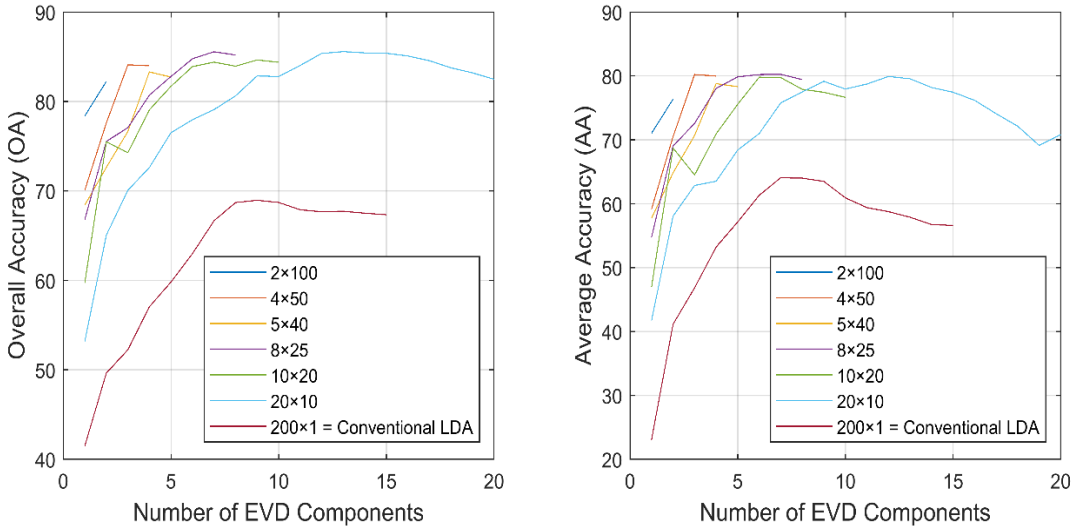



Figure 5.6 Classification results for the Indian Pine dataset using F-LDA

Table 5.11 Classification Results (Best Cases) for the Indian Pine Dataset (16 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA

| **Sample Shape** | **$V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$)** | **Best $d_{EVD}$** | **Number of Features ($d_{TOTAL}$)** | **OA (%)** | **AA (%)** | ***kc* (%)** |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 200 | N/A | N/A | 200 | 80.03 ± 0.62 | 73.04 ± 2.24 | 77.10 ± 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 200 | 15 | 9 | 9 | 68.95 ± 0.94 | 64.06 ± 2.45 | 64.49 ± 0.01 |
| **Folded-LDA** | | | | | | |
| 1×200 | N/A | N/A | 200 | 80.03 ± 0.62 | 73.04 ± 2.24 | 77.10 ± 0.01 |
| 2×100 | 2 | 2 | 200 | 82.23 ± 0.71 | 76.36 ± 2.34 | 79.66 ± 0.01 |
| 4×50 | 4 | 3 | 150 | 84.09 ± 0.95 | 80.22 ± 2.55 | 81.82 ± 0.01 |
| 5×40 | 5 | 4 | 160 | 83.30 ± 0.88 | 78.78 ± 3.21 | 80.91 ± 0.01 |
| **8×25** | 8 | 7 | 175 | 85.55 ± 0.81 | **80.24 ± 3.22** | **83.49 ± 0.01** |
| 10×20 | 10 | 9 | 180 | 84.64 ± 0.92 | 79.78 ± 2.68 | 82.41 ± 0.01 |
| **20×10** | 20 | 13 | 130 | **85.57 ± 0.98** | 79.90 ± 2.81 | **83.49 ± 0.01** |
| 200×1 | 15 | 9 | 9 | 68.95 ± 0.94 | 64.06 ± 2.45 | 64.49 ± 0.01 |
| **2D-LDA** | | | | | | |
| **1×200** | **N/A** | **N/A** | **200** | **80.03 ± 0.62** | **73.04 ± 2.24** | **77.10 ± 0.01** |
| 2×100 | 2 | N/A | 100 | 76.33 ± 1.32 | 70.50 ± 4.85 | 72.85 ± 0.02 |
| 4×50 | 4 | N/A | 50 | 71.97 ± 2.32 | 63.62 ± 3.40 | 67.80 ± 0.03 |
| 5×40 | 5 | N/A | 40 | 69.22 ± 2.63 | 58.37 ± 4.82 | 64.54 ± 0.03 |
| 8×25 | 8 | N/A | 25 | 68.39 ± 1.85 | 56.03 ± 4.26 | 63.53 ± 0.02 |
| 10×20 | 10 | N/A | 20 | 68.59 ± 2.37 | 58.23 ± 4.80 | 63.69 ± 0.03 |
| 20×10 | 20 | N/A | 10 | 58.18 ± 1.86 | 44.35 ± 2.76 | 51.32 ± 0.02 |
| 200×1 | 15 | N/A | 1 | 42.64 ± 2.00 | 25.94 ± 2.91 | 32.19 ± 0.02 |
| **GDA** | | | | | | |
| 1×200 | 15 | 14 | 14 | 79.12 ± 0.74 | 73.10 ± 1.92 | 76.07 ± 0.01 |
| **NWFE** | | | | | | |
| 1×200 | 20 | 12 | 12 | 81.23 ± 0.85 | 75.56 ± 4.40 | 78.48 ± 0.01 |
| **KPCA** | | | | | | |
| 1×200 | 10 | 6 | 6 | 79.01 ± 0.90 | 75.67 ± 2.06 | 75.98 ± 0.01 |
| **Folded PCA** | | | | | | |
| 1×200 | 5 | 5 | 5 | 75.09 ± 0.85 | 71.32 ± 2.58 | 71.45 ± 0.01 |
| 2×100 | 5 | 5 | 10 | 83.16 ± 0.67 | 79.59 ± 2.05 | 80.78 ± 0.01 |
| 4×50 | 5 | 5 | 20 | 82.06 ± 0.43 | 77.41 ± 2.15 | 79.45 ± 0.01 |
| 5×40 | 5 | 5 | 25 | 83.68 ± 0.70 | 78.77 ± 2.37 | 81.33 ± 0.01 |
| **8×25** | **5** | **4** | **32** | **84.10 ± 0.97** | **79.02 ± 4.11** | **81.80 ± 0.01** |
| 10×20 | 5 | 3 | 30 | 82.39 ± 1.05 | 77.82 ± 2.39 | 79.86 ± 0.01 |
| 20×10 | 5 | 2 | 40 | 83.62 ± 0.68 | 79.31 ± 3.00 | 81.26 ± 0.01 |
| 200×1 | N/A | N/A | 200 | 80.03 ± 0.62 | 73.04 ± 2.24 | 77.10 ± 0.01 |

Table 5.12 Classification Results (Precision, Recall and $F_1$ Score Using Micro Averaging) for the Indian Pine Dataset (16 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 200 | N/A | N/A | 200 | 80.03+ 0.01 | 80.03+ 0.01 | 80.03+ 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 200 | 15 | 9 | 9 | 68.95 + 0.01 | 68.95 + 0.01 | 68.95 + 0.01 |
| **Folded-LDA** | | | | | | |
| 1×200 | N/A | N/A | 200 | 80.03 + 0.01 | 80.03 + 0.01 | 80.03 + 0.01 |
| 2×100 | 2 | 2 | 200 | 82.23 + 0.01 | 82.23 + 0.01 | 82.23 + 0.01 |
| 4×50 | 4 | 3 | 150 | 84.09 + 0.01 | 84.09 + 0.01 | 84.09 + 0.01 |
| 5×40 | 5 | 4 | 160 | 83.30 + 0.01 | 83.30 + 0.01 | 83.30 + 0.01 |
| 8×25 | 8 | 7 | 175 | 85.55 + 0.01 | 85.55 + 0.01 | 85.55 + 0.01 |
| 10×20 | 10 | 9 | 180 | 84.64 + 0.01 | 84.64 + 0.01 | 84.64 + 0.01 |
| **20×10** | 20 | 13 | 130 | **85.57 + 0.01** | **85.57 + 0.01** | **85.57 + 0.01** |
| 200×1 | 15 | 9 | 9 | 68.95 + 0.01 | 68.95 + 0.01 | 68.95 + 0.01 |
| **2D-LDA** | | | | | | |
| **1×200** | **N/A** | **N/A** | **200** | **80.03 + 0.01** | **80.03 + 0.01** | **80.03 + 0.01** |
| 2×100 | 2 | N/A | 100 | 76.33 + 0.01 | 76.33 + 0.01 | 76.33 + 0.01 |
| 4×50 | 4 | N/A | 50 | 71.97 + 0.02 | 71.97 + 0.02 | 71.97 + 0.02 |
| 5×40 | 5 | N/A | 40 | 69.22 + 0.03 | 69.22 + 0.03 | 69.22 + 0.03 |
| 8×25 | 8 | N/A | 25 | 68.39 + 0.02 | 68.39 + 0.02 | 68.39 + 0.02 |
| 10×20 | 10 | N/A | 20 | 68.59 + 0.02 | 68.59 + 0.02 | 68.59 + 0.02 |
| 20×10 | 20 | N/A | 10 | 58.18 + 0.02 | 58.18 + 0.02 | 58.18 + 0.02 |
| 200×1 | 15 | N/A | 1 | 42.64 + 0.02 | 42.64 + 0.02 | 42.64 + 0.02 |
| **GDA** | | | | | | |
| 1×200 | 15 | | | | | |
| **NWFE** | | | | | | |
| 1×200 | 20 | 12 | 12 | 81.23 + 0.01 | 81.23 + 0.01 | 81.23 + 0.01 |
| **KPCA** | | | | | | |
| 1×200 | 10 | 6 | 6 | 79.01 + 0.01 | 79.01 + 0.01 | 79.01 + 0.01 |
| **Folded PCA** | | | | | | |
| 1×200 | 5 | 5 | 5 | 75.09 + 0.01 | 75.09 + 0.01 | 75.09 + 0.01 |
| 2×100 | 5 | 5 | 10 | 83.16 + 0.01 | 83.16 + 0.01 | 83.16 + 0.01 |
| 4×50 | 5 | 5 | 20 | 82.06 + 0.00 | 82.06 + 0.00 | 82.06 + 0.00 |
| 5×40 | 5 | 5 | 25 | 83.68 + 0.01 | 83.68 + 0.01 | 83.68 + 0.01 |
| **8×25** | **5** | **4** | **24** | **84.10 + 0.01** | **84.10 + 0.01** | **84.10 + 0.01** |
| 10×20 | 5 | 3 | 30 | 82.39 + 0.01 | 82.39 + 0.01 | 82.39 + 0.01 |
| 20×10 | 5 | 2 | 40 | 83.62 + 0.01 | 83.62 + 0.01 | 83.62 + 0.01 |
| 200×1 | N/A | N/A | 200 | 80.03 + 0.01 | 80.03 + 0.01 | 80.03 + 0.01 |

Table 5.13 Classification Results (Precision, Recall and $F_1$ Score Using Macro Averaging) for the Indian Pine Dataset (16 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA[4].

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 200 | N/A | N/A | 200 | 79.50 + 0.02 | 73.04 + 0.02 | 75.10 + 0.02 |
| **Conventional LDA** | | | | | | |
| 1 × 200 | 15 | 9 | 9 | 69.13 + 0.02 | 64.06 + 0.02 | 64.23 + 0.02 |
| **Folded-LDA** | | | | | | |
| 1×200 | N/A | N/A | 200 | 79.50 + 0.02 | 73.04 + 0.02 | 75.10 + 0.02 |
| 2×100 | 2 | 2 | 200 | 81.66 + 0.01 | 76.36 + 0.02 | 78.09 + 0.02 |
| 4×50 | 4 | 3 | 150 | 85.27 + 0.02 | 80.22 + 0.03 | 81.91 + 0.02 |
| 5×40 | 5 | 4 | 160 | 85.24 + 0.02 | 78.78 + 0.03 | 80.87 + 0.03 |
| 8×25 | 8 | 7 | 175 | 87.25 + 0.01 | 80.24 + 0.03 | 82.69 + 82.69 |
| 10×20 | 10 | 9 | 180 | 87.76 + 0.02 | 79.78 + 0.03 | 82.17 + 0.02 |
| **20×10** | **20** | **18** | **180** | **89.34 + 0.01** | **79.90 + 0.03** | **82.89 + 0.02** |
| 200×1 | 15 | 9 | 9 | 69.13 + 0.02 | 64.06 + 0.02 | 64.23 + 0.02 |
| **2D-LDA** | | | | | | |
| **1×200** | **N/A** | **N/A** | **200** | **79.50 + 0.02** | **73.04 + 0.02** | **75.10 + 0.02** |
| 2×100 | 2 | N/A | 100 | - | 70.50 + 0.05 | - |
| 4×50 | 4 | N/A | 50 | 69.90 + 0.03 | 63.62 + 0.03 | 65.12 + 0.03 |
| 5×40 | 5 | N/A | 40 | - | 58.37 + 0.05 | - |
| 8×25 | 8 | N/A | 25 | - | 56.03 + 0.04 | - |
| 10×20 | 10 | N/A | 20 | - | 58.23 + 0.05 | - |
| 20×10 | 20 | N/A | 10 | - | 44.35 + 0.03 | - |
| 200×1 | 15 | N/A | 1 | - | 25.94 + 0.03 | - |
| **GDA** | | | | | | |
| 1×200 | 15 | 13 | 13 | 84.67 + 0.01 | 73.10 + 0.02 | 76.50 + 0.02 |
| **NWFE** | | | | | | |
| 1×200 | 20 | 12 | 12 | 81.96 + 0.03 | 75.56 + 0.04 | 76.55 + 0.04 |
| **KPCA** | | | | | | |
| 1×200 | 10 | 6 | 6 | 77.87 + 0.02 | 75.67 + 0.02 | 76.28 + 0.02 |
| **Folded PCA** | | | | | | |
| 1×200 | 5 | 5 | 5 | 74.74 + 0.01 | 71.32 + 0.03 | 72.30 + 0.02 |
| **2×100** | 5 | 5 | 10 | 83.58 + 0.02 | **79.59 + 0.02** | 80.81 + 0.02 |
| 4×50 | 5 | 5 | 20 | 82.75 + 0.01 | 77.41 + 0.02 | 78.98 + 0.02 |
| 5×40 | 5 | 5 | 25 | 83.94 + 0.01 | 78.77 + 0.02 | 80.58 + 0.02 |
| 8×25 | 5 | 3 | 24 | 83.28 + 0.01 | 79.02 + 0.04 | 79.23 + 0.02 |
| **10×20** | 5 | 5 | 50 | **84.14 +0.02** | 77.82 + 0.02 | 79.75 + 0.02 |
| **20×10** | 5 | 2 | 40 | 83.95 + 0.01 | 79.31 + 0.03 | **80.87 + 0.02** |
| 200×1 | N/A | N/A | 200 | 79.50 + 0.02 | 73.04 + 0.02 | 75.10 + 0.02 |

[4] A dash is used in Table 5.13 where the result is not a number i.e. it cannot be computed. This happens because either the true positives and false positives are both zero or the true positives and false negatives are both zero.

5. Classification Accuracy for the Pavia University Dataset

Finally, the outputs of the F-LDA is applied on the Pavia University dataset to train the SVM model. The classification results attained are then used to obtain the plots of OA and AA against $d_{\mathrm{EVD}}$ which are presented in Figure 5.7. The highest classification values are extracted from these plots and presented in Table 5.14. The highest OA, AA and *kc* are obtained when the configuration is set to $8 \times 13$. This is an improvement on the accuracies from the full feature space. As can be seen in Table 5.14, configurations $103 \times 1$ and $1 \times 103$ simplifies the proposed F-LDA to the conventional LDA and the full feature space respectively. The features extracted by the 2D LDA, GDA, NWFE, KPCA and F-PCA when applied on the Pavia Center dataset are also used to train the SVM model. The classification results are obtained and presented in Table 5.14. From Table 5.14, it can be observed that the proposed F-LDA continues to give the best classification performance (highest OA, AA and *kc*). When considering the standard deviation, from Table 5.14, it can be observed that the lower bands of the OA, AA and kc achieved by the proposed F-LDA (86.43 ± 1.16, 81.05 ± 2.42 and 81.77 ± 0.02) are higher than the average classification results reported for the other techniques except for F-PCA (85.02, 79.61 and 79.84) and 2D LDA (85.02, 79.61 and 79.84) where the reported average OA, AA and kc are comparable (slightly higher in some cases) to those reported for the proposed approach. The improvement on the performance of the classifier (when fed with the outputs of F-PCA and 2D LDA) achieved by the proposed approach can be considered not significant in these cases.

The classifier's performance is also reported in terms of *P, R* and $F_1$ in Table 5.15 and Table 5.16. From Table 5.15 and Table 5.16, it can be observed that the proposed F-

LDA continues to give the best classification performance in terms of *P, R* and $F_1$. It can be noted in 5.15 that *P, R* and $F_1$ all have the same value. This is usually the case when their average is computed using micro averaging [113]. It is also worth noting in Table 5.16 that some of the results in *P, R* and $F_1$ are 'nan'. This occur due to division by zero when computing them [114].

Finally, the classification maps of the five selected hyperspectral data are presented in Figure 5.8 - 5.12 to qualitatively compare the proposed F-LDA approach to conventional LDA when used to classify the hyperspectral data. From Figure 5.8 - 5.12, it can be observed that the classification maps generated when the F-LDA is applied are generally smoother than the ones generated when the LDA is applied for all the selected datasets. Classification maps obtained when F-PCA was used are also presented in Figure 5.8 – 5.12 for comparison with F-LDA and interpretation of which pixels are misclassified. While in general F-LDA performs better than F-PCA, from the classification maps, it can be observed that F-PCA classified pixels in some classes better than F-LDA. In Figure 5.11, it can be seen that the model with the F-PCA classified the grass-pastures and grass-trees pixels better than the model with the F-LDA. Also, in Figure 5.12, it can be seen that the model with the F-PCA classified the gravel pixels better than the model with the F-LDA.

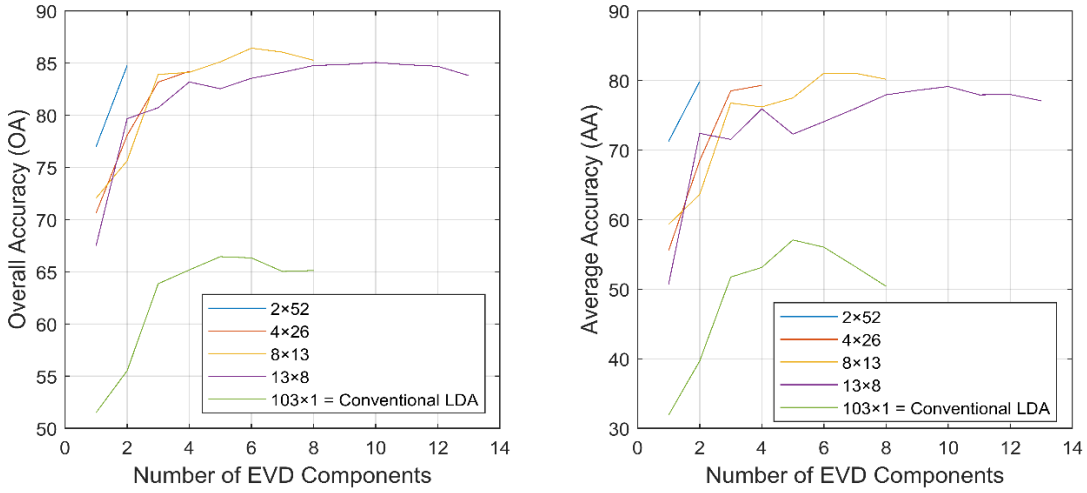



Figure 5.7 Classification results for the Pavia University dataset using F-LDA. © 2020 IEEE

Table 5.14 Classification Results (Best Cases) for the Pavia University Dataset (9 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA. © 2020 IEEE

| **Sample Shape** | **$V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$)** | **Best $d_{EVD}$** | **Number of Features ($d_{TOTAL}$)** | **OA (%)** | **AA (%)** | ***kc* (%)_** |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 103 | N/A | N/A | 103 | 85.02 ± 1.47 | 79.61 ± 3.20 | 79.84 ± 0.02 |
| **Conventional LDA** | | | | | | |
| 1 × 103 | 8 | 5 | 5 | 66.45 ± 1.66 | 57.07 ± 3.80 | 55.86 ± 0.02 |
| **Folded-LDA** | | | | | | |
| 1 × 103 | N/A | N/A | 103 | 85.02 ± 1.47 | 79.61 ± 3.20 | 79.84 ± 0.02 |
| 2 × 52 | 2 | 2 | 104 | 84.77 ± 1.30 | 79.85 ± 2.61 | 79.49 ± 0.02 |
| 4 × 26 | 4 | 4 | 104 | 84.21 ± 0.98 | 79.30 ± 3.06 | 78.74 ± 0.01 |
| **8 × 13** | **8** | **6** | **78** | **86.43 ± 1.16** | **81.05 ± 2.42** | **81.77 ± 0.02** |
| 13 × 8 | 13 | 10 | 80 | 85.06 ± 1.49 | 79.14 ± 3.20 | 79.95 ± 0.02 |
| 103 × 1 | 8 | 5 | 5 | 66.45 ± 1.66 | 57.07 ± 3.80 | 55.86 ± 0.02 |
| **2D-LDA** | | | | | | |
| **1 × 103** | **N/A** | **N/A** | **103** | **85.02 ± 1.47** | **79.61 ± 3.20** | **79.84 ± 0.02** |
| 2 × 52 | 2 | N/A | 52 | 76.52 ± 1.26 | 66.61 ± 3.44 | 67.41 ± 0.02 |
| 4 × 26 | 4 | N/A | 26 | 69.76 ± 2.99 | 56.12 ± 6.71 | 57.72 ± 0.05 |
| 8 × 13 | 8 | N/A | 13 | 74.42 ± 1.80 | 64.31 ± 5.96 | 64.76 ± 0.03 |
| 13 × 8 | 13 | N/A | 8 | 69.95 ± 3.07 | 58.99 ± 5.57 | 58.65 ± 0.04 |
| 103 × 1 | 8 | N/A | 1 | 51.49 ± 6.01 | 33.80 ± 5.85 | 32.30 ± 0.11 |
| **GDA** | | | | | | |
| 1 × 103 | 8 | 7 | 7 | 77.76 ± 1.93 | 70.91 ± 2.30 | 70.28 ± 0.02 |
| **NWFE** | | | | | | |
| 1 × 103 | 20 | 17 | 17 | 80.32 ± 1.29 | 70.21 ± 2.42 | 73.28 ± 0.02 |
| **KPCA** | | | | | | |
| 1 × 103 | 10 | 9 | 9 | 80.33 ± 2.19 | 70.96 ± 3.77 | 73.22 ± 0.02 |
| **Folded PCA** | | | | | | |
| 1 × 103 | 5 | 4 | 4 | 78.32 ± 1.39 | 68.13 ± 3.39 | 69.95 ± 0.02 |
| 2 × 52 | 5 | 4 | 8 | 82.01 ± 1.13 | 75.57 ± 5.02 | 75.61 ± 0.02 |
| 4 × 26 | 5 | 2 | 8 | 82.06 ± 0.99 | 75.33 ± 3.26 | 75.68 ± 0.02 |
| 8 × 13 | 5 | 2 | 16 | 82.24 ± 1.01 | 77.50 ± 3.20 | 76.12 ± 2.00 |
| 13 × 8 | 5 | 3 | 39 | 83.86 ± 0.99 | 76.92 ± 4.99 | 78.19 ± 0.01 |
| **103 × 1** | **N/A** | N/A | **103** | **85.02 ± 1.47** | **79.61 ± 3.20** | **79.84 ± 0.02** |

Table 5.15 Classification Results (Precision, Recall and $F_1$ Score Using Micro Averaging) for the Pavia University Dataset (9 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA.

| **Sample Shape** | **$V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$)** | **Best $d_{EVD}$** | **Number of Features ($d_{TOTAL}$)** | **Precision (%)** | **Recall (%)** | **$F_1$ Score (%)** |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 103 | N/A | N/A | 103 | 85.02 + 0.01 | 85.02 + 0.01 | 85.02 + 0.01 |
| **Conventional LDA** | | | | | | |
| 1 × 103 | 8 | 5 | 5 | 66.45 + 0.02 | 66.45 + 0.02 | 66.45 + 0.02 |
| **Folded-LDA** | | | | | | |
| 1 × 103 | N/A | N/A | 103 | 85.02 + 0.01 | 85.02 + 0.01 | 85.02 + 0.01 |
| 2 × 52 | 2 | 2 | 104 | 84.77 + 0.01 | 84.77 + 0.01 | 84.77 + 0.01 |
| 4 × 26 | 4 | 4 | 104 | 84.21 + 0.01 | 84.21 + 0.01 | 84.21 + 0.01 |
| **8 × 13** | **8** | **6** | **78** | **86.43 + 0.01** | **86.43 + 0.01** | **86.43 + 0.01** |
| 13 × 8 | 13 | 10 | 80 | 85.06 + 0.01 | 85.06 + 0.01 | 85.06 + 0.01 |
| 103 × 1 | 8 | 5 | 5 | 66.45 + 0.02 | 66.45 + 0.02 | 66.45 + 0.02 |
| **2D-LDA** | | | | | | |
| **1 × 103** | **N/A** | **N/A** | **103** | **85.02 + 0.01** | **85.02 + 0.01** | **85.02 + 0.01** |
| 2 × 52 | 2 | N/A | 52 | 76.52 + 0.01 | 76.52 + 0.01 | 76.52 + 0.01 |
| 4 × 26 | 4 | N/A | 26 | 69.76 + 0.03 | 69.76 + 0.03 | 69.76 + 0.03 |
| 8 × 13 | 8 | N/A | 13 | 74.42 + 0.02 | 74.42 + 0.02 | 74.42 + 0.02 |
| 13 × 8 | 13 | N/A | 8 | 69.95 + 0.03 | 69.95 + 0.03 | 69.95 + 0.03 |
| 103 × 1 | 8 | N/A | 1 | 51.49 + 0.06 | 51.49 + 0.06 | 51.49 + 0.06 |
| **GDA** | | | | | | |
| 1 × 103 | 8 | 7 | 7 | 77.76 + 0.02 | 77.76 + 0.02 | 77.76 + 0.02 |
| **NWFE** | | | | | | |
| 1 × 103 | 20 | 17 | 17 | 80.32 + 0.01 | 80.32 + 0.01 | 80.32 + 0.01 |
| **KPCA** | | | | | | |
| 1 × 103 | 10 | 9 | 9 | 80.33 + 0.02 | 80.33 + 0.02 | 80.33 + 0.02 |
| **Folded PCA** | | | | | | |
| 1 × 103 | 5 | 4 | 4 | 78.32 + 0.01 | 78.32 + 0.01 | 78.32 + 0.01 |
| 2 × 52 | 5 | 4 | 8 | 82.01 + 0.01 | 82.01 + 0.01 | 82.01 + 0.01 |
| 4 × 26 | 5 | 2 | 8 | 82.06 + 0.01 | 82.06 + 0.01 | 82.06 + 0.01 |
| 8 × 13 | 5 | 2 | 16 | 82.24 + 0.01 | 82.24 + 0.01 | 82.24 + 0.01 |
| 13 × 8 | 5 | 3 | 39 | 83.86 + 0.01 | 83.86 + 0.01 | 83.86 + 0.01 |
| **103 × 1** | **N/A** | **N/A** | **103** | **85.02 + 0.01** | **85.02 + 0.01** | **85.02 + 0.01** |

Table 5.16 Classification Results (Precision, Recall and $F_1$ Score Using Macro Averaging) for the Pavia University Dataset (9 Classes) Using Original Feature Space, Conventional LDA, F-LDA (with Different Configurations), 2D LDA, GDA, NWFE, KPCA and F-PCA[5].

| Sample Shape | $V_B$ Matrix Rank (r)/ EVD Components ($d_{EVD}$) | Best $d_{EVD}$ | Number of Features ($d_{TOTAL}$) | Precision (%) | Recall (%) | $F_1$ Score (%) |
|---|---|---|---|---|---|---|
| **Original Feature Space** | | | | | | |
| 1 × 103 | N/A | N/A | 103 | 83.06 + 0.02 | 79.61 + 0.03 | 80.60 + 0.03 |
| **Conventional LDA** | | | | | | |
| 1 × 103 | 8 | 6 | 6 | 61.06 + 0.03 | 57.07 + 0.04 | 57.66 + 0.03 |
| **Folded-LDA** | | | | | | |
| 1 × 103 | N/A | N/A | 103 | 83.06 + 0.02 | 79.61 + 0.03 | 80.60 + 0.03 |
| 2 × 52 | 2 | 2 | 104 | 82.90 + 0.02 | 79.85 + 0.03 | 80.84 + 0.02 |
| 4 × 26 | 4 | 4 | 104 | 83.24 + 0.02 | 79.30 + 0.03 | 80.57 + 0.02 |
| **8 × 13** | **8** | **6** | **78** | **85.44 + 0.02** | **81.05 + 0.02** | **82.48 + 0.02** |
| 13 × 8 | 13 | 8 | 64 | 82.40 + 0.01 | 79.14 + 0.03 | 79.84 + 0.02 |
| 103 × 1 | 8 | 6 | 6 | 61.06 + 0.03 | 57.07 + 0.04 | 57.66 + 0.03 |
| **2D-LDA** | | | | | | |
| **1 × 103** | **N/A** | **N/A** | **103** | **83.06 + 0.02** | **79.61 +0.03** | **80.60 + 0.03** |
| 2 × 52 | 2 | N/A | 52 | 75.27 + 0.03 | 66.61 + 0.03 | - |
| 4 × 26 | 4 | N/A | 26 | 63.83 + 0.05 | 56.12 +0.07 | 56.92 + 0.07 |
| 8 × 13 | 8 | N/A | 13 | - | 64.31 + 0.06 | - |
| 13 × 8 | 13 | N/A | 8 | - | 58.99 +0.06 | - |
| 103 × 1 | 8 | N/A | 1 | - | 33.80 + 0.06 | - |
| **GDA** | | | | | | |
| 1 × 103 | 8 | 7 | 7 | 75.42 + 0.03 | 70.91 + 0.02 | 72.48 + 0.02 |
| **NWFE** | | | | | | |
| 1 × 103 | 20 | 17 | 17 | 77.37 + 0.02 | 70.21 + 0.02 | 72.36 + 0.02 |
| **KPCA** | | | | | | |
| 1 × 103 | 10 | 10 | 10 | 76.95 + 0.04 | 70.96 + 0.04 | 71.86 +0.04 |
| **Folded PCA** | | | | | | |
| 1 × 103 | 5 | 4 | 4 | - | 68.13 + 0.03 | - |
| 2 × 52 | 5 | 5 | 10 | 80.20 + 0.03 | 75.57 + 0.05 | - |
| 4 × 26 | 5 | 4 | 16 | 79.41 + 0.02 | 75.33 + 0.03 | 76.29 + 0.03 |
| 8 × 13 | 5 | 2 | 16 | 79.88 + 0.03 | 77.50 + 0.03 | 77.31 + 0.01 |
| 13 × 8 | 5 | 2 | 26 | 80.65 + 0.04 | 76.92 + 0.05 | 75.83 + 0.01 |
| **103 × 1** | **N/A** | **N/A** | **103** | **83.06 + 0.02** | **79.61 + 0.03** | **80.60 + 0.03** |

[5] A dash is used in Table 5.16 where the result is not a number i.e. it cannot be computed. This happens because either the true positives and false positives are both zero or the true positives and false negatives are both zero.

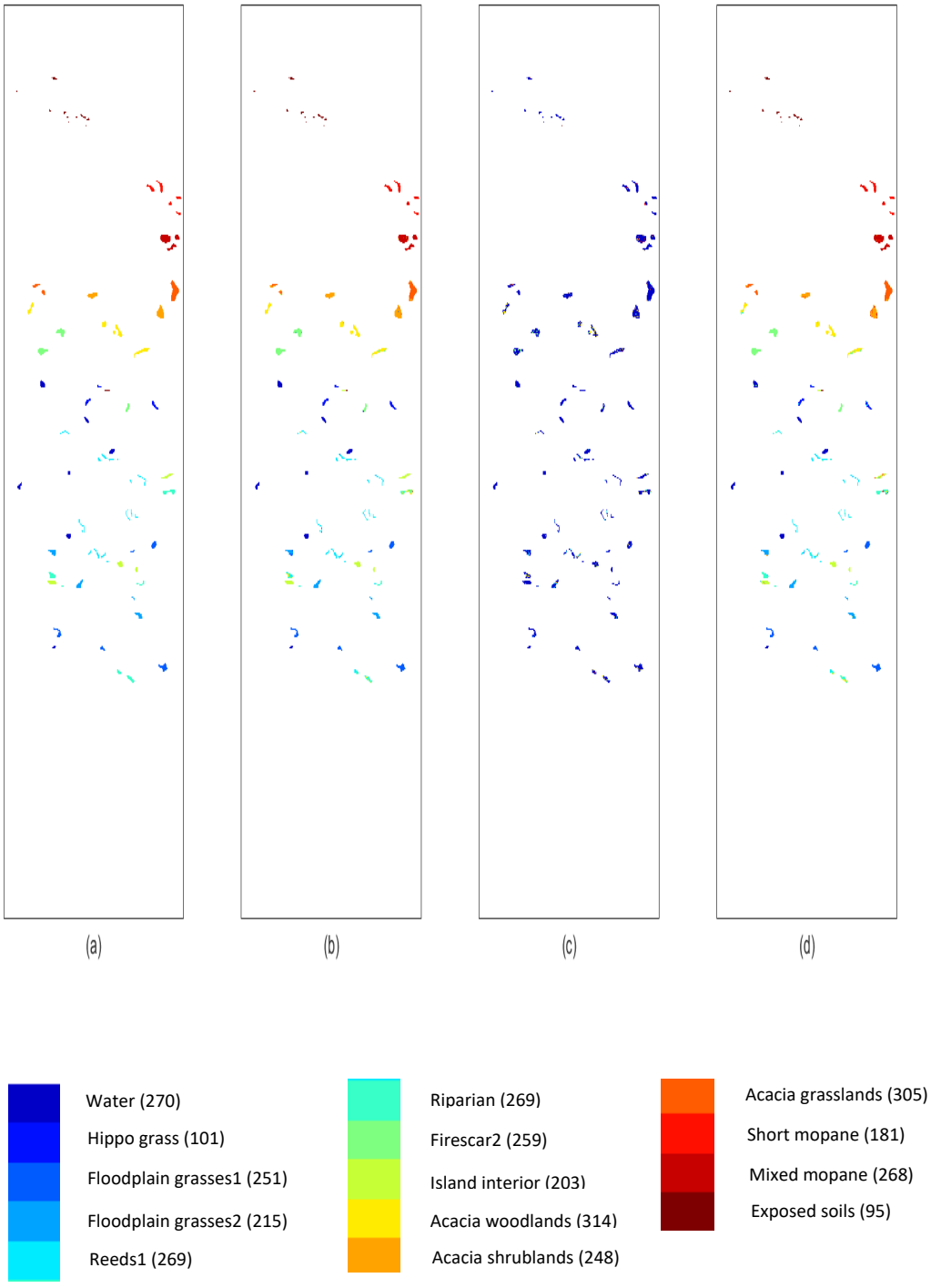



Figure 5.8 Botswana data's (a) Ground truth image (b) Classification map using F-LDA (29 × 5) (c) Classification map using LDA (d) Classification map using F-PCA (1 × 145) (also showing the number of samples in each class).

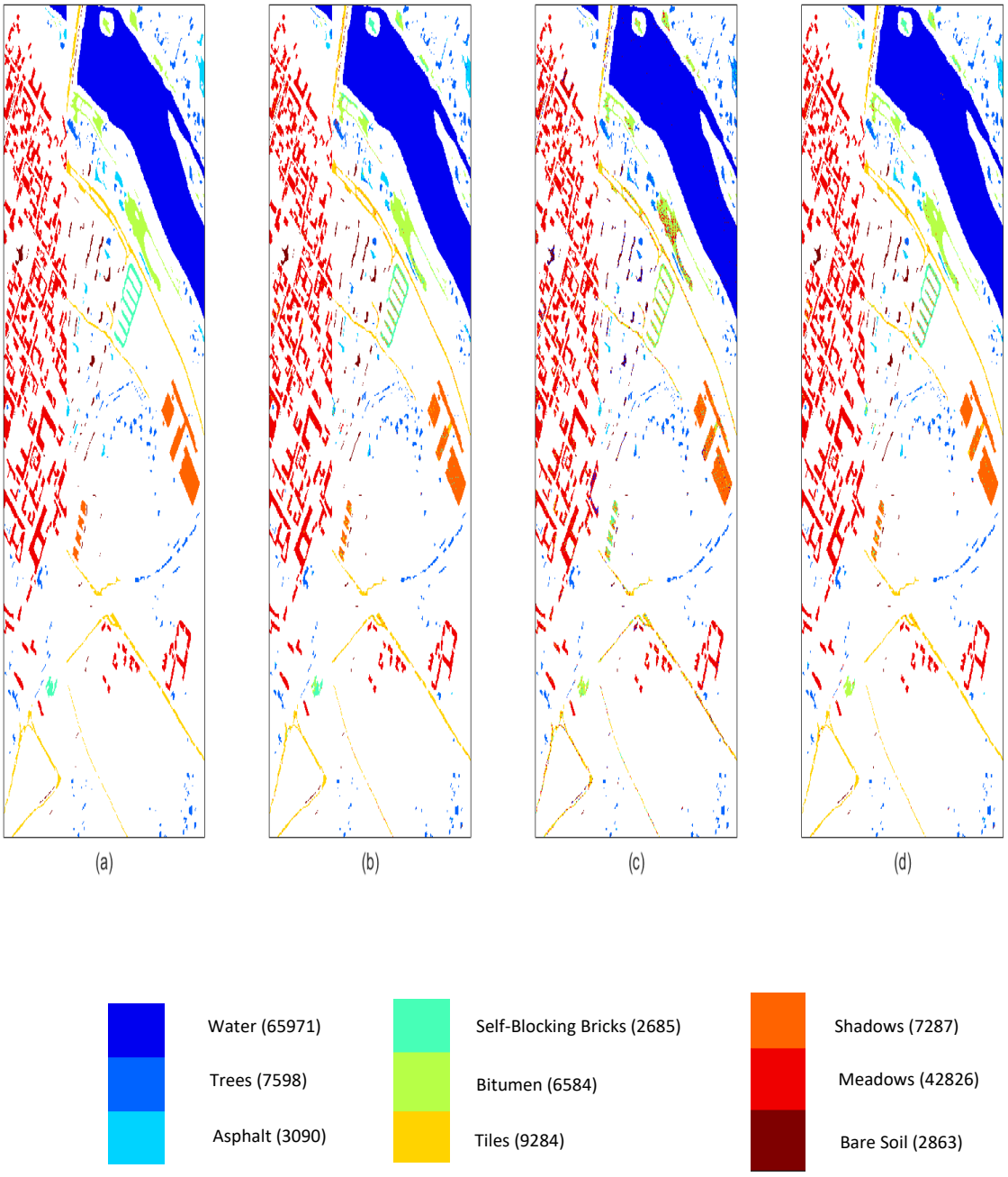



Figure 5.9 Pavia Center data's (a) Ground truth image (b) Classification map using F-LDA (17×6) (c) Classification map using LDA (d) Classification map using F-PCA (3×34) (also showing the number of samples in each class).

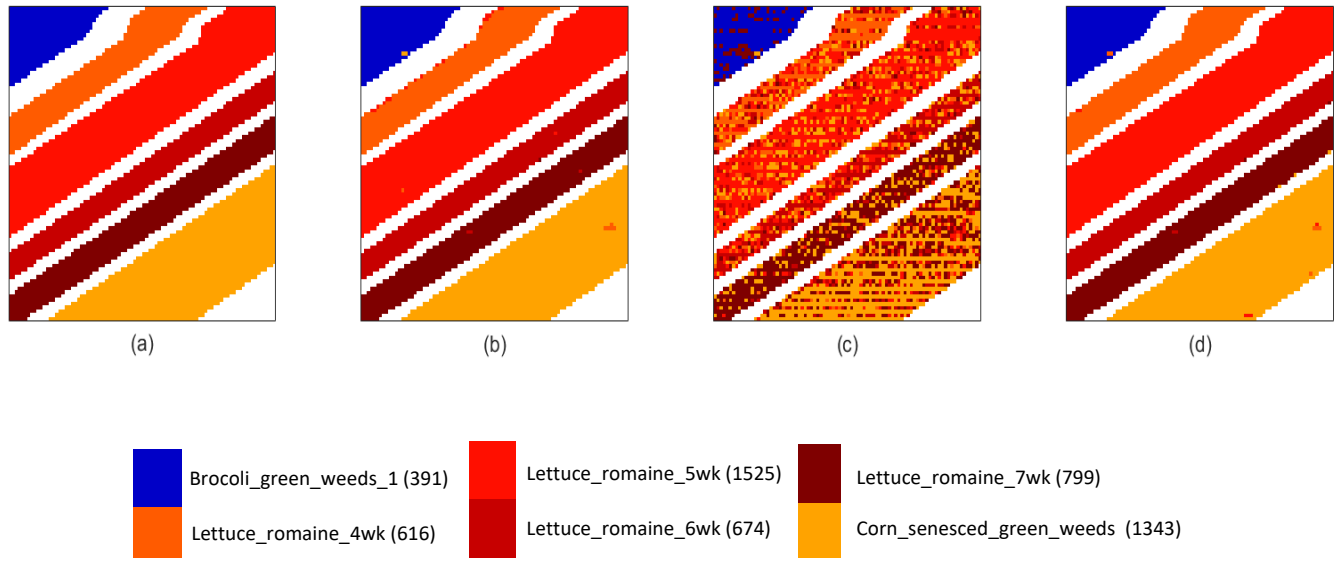



Figure 5.10 Salina-A data's (a) Ground truth image (b) Classification map using F-LDA (17 × 12) (c) Classification map using LDA (d) Classification map using F-PCA (6 × 34) (also showing the number of samples in each class).

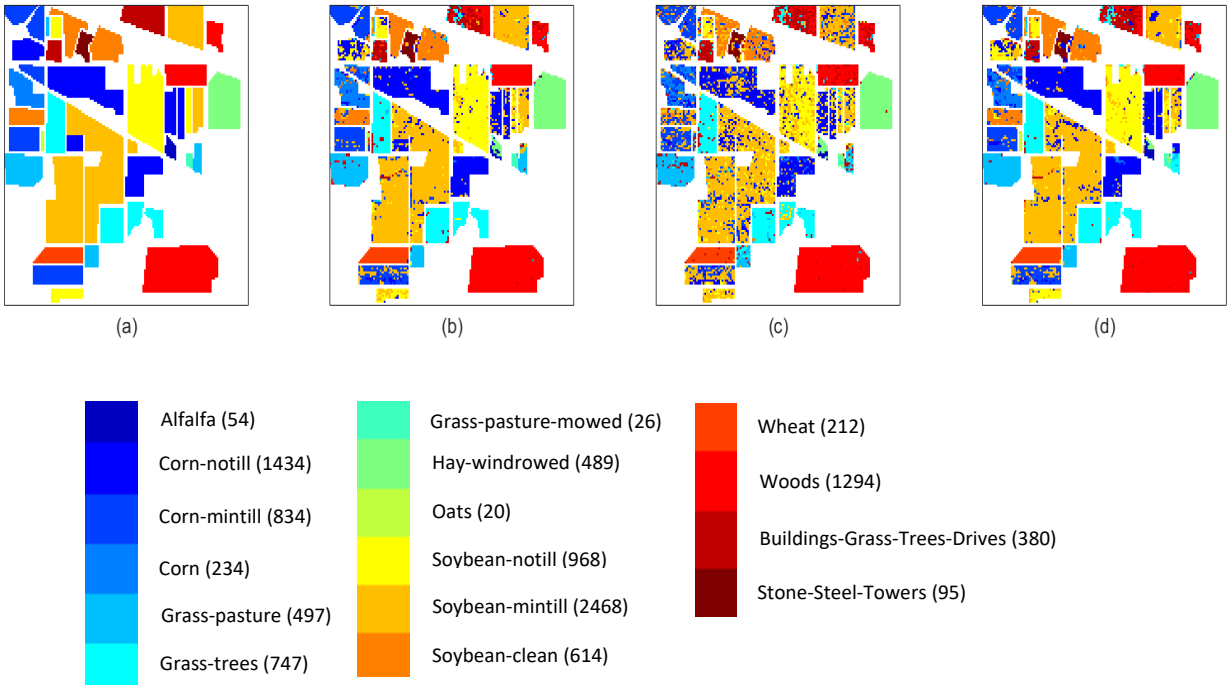



Figure 5.11 Indian Pine data's (a) Ground truth image (b) Classification map using F-LDA (8×25) (c) Classification map using LDA (d) Classification map using F-PCA (8×25) (also showing the number of samples in each class).

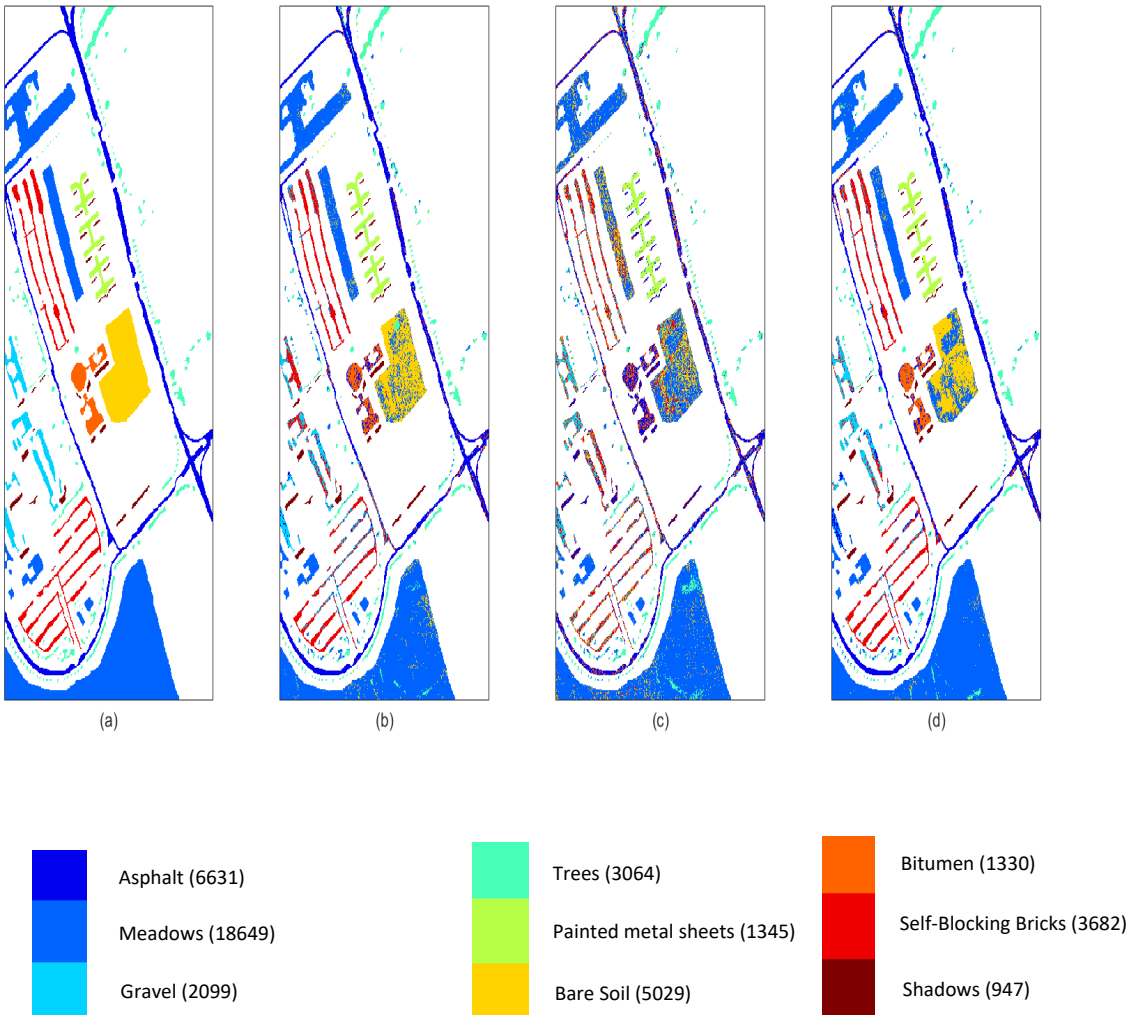



Figure 5.12 Pavia University data's (a) Ground truth image (b) Classification map using F-LDA (8 × 13) (c) Classification map using LDA (d) Classification map using F-PCA (103 × 1) (also showing the number of samples in each class).

### 5.3.2. Effect on Computational Complexity

The computational complexity of the different stages of the conventional LDA and the proposed F-LDA are illustrated and compared in Table 5.17 where $c$, $N_j$ and $d$

represent the number of classes in the data, the number of samples in each class and the number of features extracted respectively.

In the conventional LDA, the complexity of computing the within-class variance $\boldsymbol{V_W}$ and between-class variance $\boldsymbol{V_B}$ matrices are $o(cN_jG^2B^2)$ and $o(cG^2B^2)$ respectively where $\boldsymbol{V_W} \in \Re^{GB \times GB}$, $\boldsymbol{V_B} \in \Re^{GB \times GB}$, $\left(x_{ij} - \boldsymbol{m_j}\right) \in \Re^{1 \times GB}$, and $\left(\mathbf{m_j} - \mathbf{m}\right) \in \Re^{1 \times GB}$. The cost of claculating $\boldsymbol{V_W}^{-1}$ is $o(G^3B^3)$. $o(G^3B^3)$ is the complexity of multiplying $\boldsymbol{V_W}^{-1}$ and $\boldsymbol{V_B}$. Hence, the required complexity of computing the transformation matrix $\boldsymbol{T}$ and eigenvectors are $o(2G^3B^3)$ and $o(G^3B^3)$ , respectively. The complexity of projecting the data is $o(sGBd)$.

In the proposed F-LDA, the complexity of computing the within-class variance $\boldsymbol{V_{PW}}$ and between-class variance $\boldsymbol{V_{PB}}$ matrices are $o(cN_jBG^2)$ and $o(cBG^2)$ respectively where $\boldsymbol{V_{PW}} \in \Re^{G \times G}$, $\boldsymbol{V_{PB}} \in \Re^{G \times G}$, $\left(\boldsymbol{P_{ij}} - \boldsymbol{M_j}\right) \in \Re^{G \times B}$ and $\left(\mathbf{M_j} - \mathbf{M}\right) \in \Re^{G \times B}$. The complexity of computing $\boldsymbol{V_{PW}}^{-1}$ is o $(G^3)$. The computational complexity required to calculate the product of $\boldsymbol{V_{PW}}^{-1}$ and $\boldsymbol{V_{PB}}$, is o $(G^3)$. Hence, the complexity of computing the transformation matrix $\boldsymbol{T_P}$ and eigenvectors are $o(2G^3)$ and $o(G^3)$ respectively. The complexity of projecting each sample is $Gd$ where $d = B \times d_{\mathrm{EVD}}$. Hence, the complexity of projecting all the samples is $sGd$.

The content consumption for the selected datasets are also computed using the computational complexities and presented in Table 5.18. One can see in Table 5.17 and Table 5.18 that F-LDA requires less computational complexity to implement all stages. The complexity of computing the within-class variance and between-class variance is reduced by a saving factor of $B$. A reduction in the cost of computing the transformation matrix and eigenvectors by $B^3$ is reported. A saving factor of $B$ is

reported for the data projection. Reduced computational complexity is achieved by the proposed F-LDA thanks to the dimensions of its within-class variance, between-class variance, and the transformation matrices which are now smaller than those in the traditional LDA.

Table 5.17 Computational complexity for the different stages of the proposed F-LDA and the conventional LDA. © 2020 IEEE

| Stages | LDA | F-LDA | Saving factors |
|---|---|---|---|
| Within-class variance matrix | $o(cN_jG^2B^2)$ | $o(cN_jG^2B)$ | $B$ |
| Between-class variance matrix | $o(cG^2B^2)$ | $o(cG^2B)$ | $B$ |
| Transformation matrix | $o(2G^3B^3)$ | $o(2G^3)$ | $B^3$ |
| Eigen problem | $o(G^3B^3)$ | $o(G^3)$ | $B^3$ |
| Data projection | $o(sGBd)$ | $o(sGd)$ | $B$ |

Table 5.18 Computational complexity (content consumption) for the different stages of the proposed F-LDA and the conventional LDA. © 2020 IEEE

| Dataset | | Best configuration ($G \times B$) | Within-class variance matrix | Between-class variance matrix | Transformation matrix | Eigen problem | Data projection |
|---|---|---|---|---|---|---|---|
| Botswana | LDA | N/A | o($294350N_j$) | o(294350) | o(6097250) | o(3048625) | o($145\ sd$) |
| | F-LDA | 29 × 5 | o($58870N_j$) | o(58870) | o(48778) | o(24389) | o($29\ sd$) |
| | Saving factors | - | 5 | 5 | 125 | 125 | 5 |
| Pavia Center | LDA | N/A | o($93636N_j$) | o(93636) | o(2122416) | o(1061208) | o($102\ sd$) |
| | F-LDA | 17×6 | o($15606N_j$) | o(15606) | o(9826) | o(4913) | o($17\ sd$) |
| | Saving factors | - | 6 | 6 | 216 | 216 | 6 |
| Salinas-A | LDA | N/A | o($249696N_j$) | o(249696) | o(16979328) | o(8489664) | o($204\ sd$) |
| | F-LDA | 17 × 12 | o($20808N_j$) | o(20808) | o(9826) | o(4913) | o($17\ sd$) |
| | Saving factors | - | 12 | 12 | 1728 | 1728 | 12 |
| Indian Pine | LDA | N/A | o($640000N_j$) | o(640000) | o(16000000) | o(8000000) | o($200\ sd$) |
| | F-LDA | 8×25 | o($25600N_j$) | o(25600) | o(1024) | o(512) | o($8\ sd$) |
| | Saving factors | - | 25 | 25 | 15625 | 15625 | 25 |
| Pavia University | LDA | N/A | o($97344N_j$) | o(97344) | o(2249728) | o(1124864) | o($104\ sd$) |
| | F-LDA | 8 × 13 | o($7488N_j$) | o(7488) | o(1024) | o(512) | o($8\ sd$) |
| | Saving factors | - | 13 | 13 | 2197 | 2197 | 13 |

The feature extraction time of the proposed F-LDA on the datasets (selecting the first five EVD components where applicable) are also presented and compared to the other techniques in Table 5.19. In all cases, one can see from Table 5.19 that both the 2D LDA and F-LDA are slower than the conventional LDA. This can be attributed to the extra time used by the 2D LDA and F-LDA to convert each spectral vector into a feature matrix. It is worth noting that while the proposed approach F-LDA is slower

than the conventional LDA, the proposed F-LDA requires less computational complexity because it processes relatively smaller matrices at key stages as shown in the earlier parts of this section. One can also see that the F-LDA is slightly slower than the 2D LDA. This can also be attributed to the extra time required for individual treatment of the eigenvectors and unfolding of the projected samples. In most cases, while F-LDA can be found to be slower than K-PCA, it is faster than NWFE, GDA and F-PCA. Generally, the range of the time required by the proposed F-LDA for feature extraction is 0.084 - 2.059s. While this is slightly slower than the time required by its counterparts, this trade-off is offset by the benefits of the proposed F-LDA (higher classification accuracy, reduced computational complexity and reduced contiguous memory requirement).

Table 5.19 Feature extraction time (s) of different techniques (using the first five EVD components when applicable, F-LDA and 2D LDA include related configuration). © 2020 IEEE

| Techniques | Botswana | Pavia Center | Salinas-A | Indian Pine | Pavia University |
|---|---|---|---|---|---|
| LDA | 0.012 | 0.072 | 0.023 | 0.029 | 0.029 |
| F-LDA | 0.084 (29×5) | 2.059 (17×6) | 0.100 (17×12) | 0.159 (8×25) | 0.432 (8×13) |
| 2D LDA | 0.032 (1×145) | 0.929 (1×102) | 0.059 (1×204) | 0.118 (1×200) | 0.292 (1×103) |
| NWFE | 0.111 | 0.085 | 0.110 | 0.544 | 0.076 |
| GDA | 4.846 | 267.021 | 10.646 | 60.171 | 76.052 |
| F-PCA | 0.179 | 1.654 | 0.130 | 0.228 | 1.028 |
| KPCA | 0.021 | 0.860 | 0.037 | 0.291 | 0.250 |

### 5.3.3. Effect on Contiguous Memory Requirement

The contiguous memory requirement at the different stages of the conventional LDA and the proposed F-LDA are illustrated in Table 5.20. Content consumption for each of the five datasets are also computed and illustrated in Table 5.21 using the results in Table 5.20. From Table 5.20 and Table 5.21, it can be seen that the within-class variance matrix, between-class variance matrix and transformation matrix in the F-LDA require less memory than those in the conventional LDA by a saving factor of $B^2$. Similarly, saving factors of $S$ and $B^2$ are reported for the original data matrix and the projected data matrix respectively when the proposed F-LDA is applied instead of the conventional LDA. Overall, the F-LDA required much less contiguous memory than the conventional LDA. This can be attributed to the size of the matrices at the different stages of the proposed F-LDA which are now smaller than their counterparts in the conventional LDA.

Table 5.20 Different stages of the F-LDA and LDA and corresponding memory requirements. © 2020 IEEE

| Stages | LDA | F-LDA | Saving factor |
|---|---|---|---|
| Data matrix size | $s \times GB$ | $G \times B$ | s |
| Within-class variance matrix size | $GB \times GB$ | $G \times G$ | $B^2$ |
| Between-class variance matrix size | $GB \times GB$ | $G \times G$ | $B^2$ |
| Transformation matrix size | $GB \times GB$ | $G \times G$ | $B^2$ |
| Projection matrix size | $GB \times d$ | $G \times {}^{d}/_{B}$ | $B^2$ |

Table 5.21 Different stages of the F-LDA and LDA and corresponding memory requirements (content consumption). © 2020 IEEE

| Dataset | | Best Configuration ($G \times B$) | Data matrix size | Within-class variance matrix size | Between-class variance matrix size | Transformation matrix size | Projection matrix size |
|---|---|---|---|---|---|---|---|
| Botswana | LDA | - | $145s$ | 21025 | 21025 | 21025 | $145d$ |
| | F-LDA | $29 \times 5$ | 145 | 841 | 841 | 841 | $5.8d$ |
| | Saving factors | - | $s$ | 25 | 25 | 25 | 25 |
| Pavia Center | LDA | - | $102s$ | 10404 | 10404 | 10404 | $102d$ |
| | F-LDA | 17×6 | 102 | 289 | 289 | 289 | $2.8333d$ |
| | Saving factors | - | $s$ | 36 | 36 | 36 | 36 |
| Salinas-A | LDA | - | $204s$ | 41616 | 41616 | 41616 | $204d$ |
| | F-LDA | $17 \times 12$ | 204 | 289 | 289 | 289 | $1.4167 d$ |
| | Saving factors | - | $s$ | 144 | 144 | 144 | 144 |
| Indian Pine | LDA | - | $200s$ | 40000 | 40000 | 40000 | $200d$ |
| | F-LDA | 8×25 | 200 | 64 | 64 | 64 | $0.32d$ |
| | Saving factors | - | $s$ | 625 | 625 | 625 | 625 |
| Pavia Univ. | LDA | - | $104s$ | 10816 | 10816 | 10816 | $104d$ |
| | F-LDA | $8 \times 13$ | 104 | 64 | 64 | 64 | $0.6154 d$ |
| | Saving factors | - | $s$ | 169 | 169 | 169 | 169 |

## 5.4. Summary

A new Folded Linear Discriminant Analysis (F-LDA) for effective and efficient feature extraction and dimensionality reduction of hyperspectral data in Small Sample Size (SSS) scenarios is presented in this chapter. By replicating a simple but effective mathematical 'trick' (folding the pixels) which was motivated by a previous work to

extend PCA [37], the proposed, new, F-LDA can produce more informative features (higher classification accuracy than the original feature space, conventional LDA, 2D LDA, and other state-of-the-art methods) with reduced contiguous memory requirement and less computational complexity. Performance of the proposed F-LDA is evaluated on five publicly available hyperspectral datasets from different sensors (AVIRIS, ROSIS, Hyperion). Experimental results showed that the proposed approach is superior to the traditional approach when applied in SSS scenarios.

## 6. Hybridizing GA and F-LDA for Dimensionality Reduction of Hyperspectral Data

It was established from the review of literature presented in Chapter 3 that PCA is a more commonly applied dimensionality reduction tool for hyperspectral imaging than LDA [15],[21],[10],[13]. In Chapter 4, LDA was applied on hyperspectral data to transform it to a lower dimensional space while extracting useful features to better classify the data. The results obtained showed that LDA can perform better when applied on hyperspectral data than the more commonly applied PCA.

From the review of related papers presented in Chapter 3, it can be noted that the performance of LDA as a feature extraction and dimensionality technique for hyperspectral imaging data can be limited by the lack of enough samples in the hyperspectral data for training. In Chapter 5, an extended and improved version of the LDA, named Folded LDA (F-LDA) was proposed to effectively reduce the dimensionality reduction of hyperspectral data in small training sample size scenario. The proposed F-LDA outperformed the traditional LDA in terms of classification accuracy, computational complexity and memory requirement when used in that scenario. It was also established from the review of related papers presented in Chapter 3 that Genetic Algorithm (GA) has become a useful feature selection technique for data dimensionality reduction in many applications. It was further established that using GA to select optimal feature subset from the original feature set prior to feature extraction using PCA [29],[28] or LDA [31],[30] can improve performance of classification models.

This chapter is thus aimed at exploring the effectiveness of a new hybridized GA and F-LDA (GA + F-LDA) for data dimensionality reduction in classification of hyperspectral data. The proposed approach is evaluated on two hyperspectral datasets

of rice seeds containing 256 spectral features and another hyperspectral dataset of sugar containing 160 spectral features. Data classification is an important step in rice seed screening exercises and there is a continued need to automate and enhance such screening exercises [15],[22],[23]. These make rice seed classification a suitable area of application where F-LDA and the proposed (GA + F-LDA) would be deployed to reduce the dimensionality of hyperspectral data of rice seeds thus enhancing the classification tasks and the screening exercises. Hence, rice seed dataset was selected for this study. For the sugar dataset, it was selected to demonstrate the potential of the proposed approach for dimensionality reduction of hyperspectral data of other Agri-tech products. Experimental results show that applying F-LDA on reduced spectral datasets (datasets containing optimal features selected by GA) can achieve reduction in computational complexity and memory requirement.

## 6.1. Methods and Materials

### 6.1.1. Data Acquisition and Description

Two groups of rice seed datasets are used in this work to evaluate performance of the proposed approach. There are 10 different rice seed sub datasets in the first group with each sub dataset containing 10 varieties. The second group contains another 10 different rice seed sub datasets with each sub dataset containing 20 varieties. The rice seed datasets in the two groups, which are subsets of the main rice seed datasets of 90 varieties used in Chapter 4, are randomly selected to demonstrate the consistency of the proposed approach in achieving improved classification performance. Average of classification results obtained using the sub datasets in each group are computed and reported. The rice seed datasets contain 256 spectra features which were extracted from hyperspectral images collected using a Visible-Near Infrared (VIS/NIR) range

HSI system operating at ~ (385 – 1000) nm and 6 spatial features (area, major axis length, minor axis length, aspect ratio, perimeter over area ratio and eccentricity) which were extracted from RGB images collected using a Fujifilm X-M1 with a 35mm/F2.0 lens. The procedure and processing steps used to extract the spatial and spectral features are described in Chapter 4. While this chapter is focused on addressing the high dimensionality problem in the spectral domain, spatial features are also included in the rice seed analysis to show their ability to further improve classification results when fused with spectral features.

In order to demonstrate the potential of the proposed technique for other applications, a hyperspectral dataset of sugar containing 9 different varieties (Sugar Ester S170, Sugar Ester S770, Sugar Ester S1570, Sugar Ester P1570, D-Mannitol, D-Sorbitol, D-Glucose, D-Galactose and D-Fructose) was utilised. The hyperspectral images of the sugar were collected [115] using the Neo VNIR-1600 sensor and has a spectral range of ~ 400 – 1000 nm. The number of samples per variety in the sugar dataset is 125. In total, the sugar dataset has 1125 samples. The sugar dataset was published in Machine Learning Reports [116] and made available at an online research repository named, 4TU.ResearchData and the University of Groningen research pure portal in 2016. The dataset is yet to be cited based on the information available at 4TU.ResearchData and pure portal. According to the information available on pure portal, the dataset has been downloaded 214 times as at 27$^{th}$ June 2022. The information available on 4TU.ResearchData shows that the data has been downloaded 256 times as at 27$^{th}$ June 2022.

### 6.1.2. Spectral Reflectance Profiles of Rice Seeds

The average spectra, in the VIS/NIR range of ~ (385 – 1000) nm, of samples of each of the species in two rice seed datasets containing 10 and 20 varieties are plotted and illustrated in Figure 6.1 and Figure 6.2. The average spectral profiles show differences which can be attributed to the differences in chemical property (composition) of the species of rice seeds in the NIR region ~ (700 - 1000) nm and physical property (colour variation) in the Visible region ~ (385 - 700) nm [23],[7],[104]. Hence, models are built to exploit the spectral differences in these regions, reduce the dimensionality of the data by retaining only key features and discriminate the rice seed species. The average spectra, in the VIS/NIR of 429.19 – 986.32 nm, of the 9 species in the sugar data sets are also plotted and illustrated in Figure 6.3.

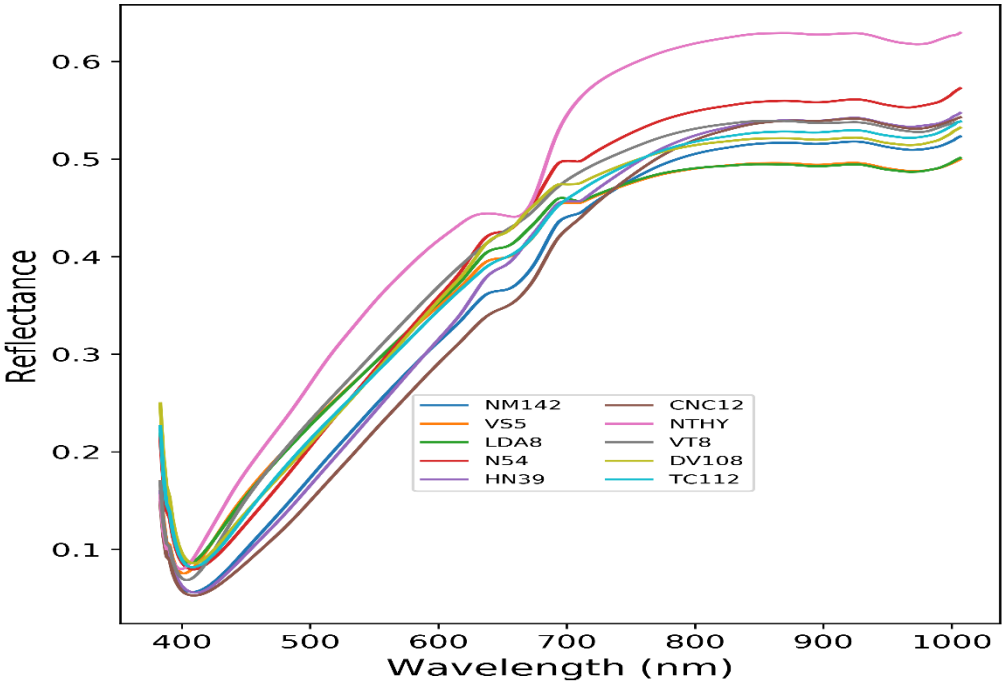



Figure 6.1 The average spectral profiles of 10 rice seed species.

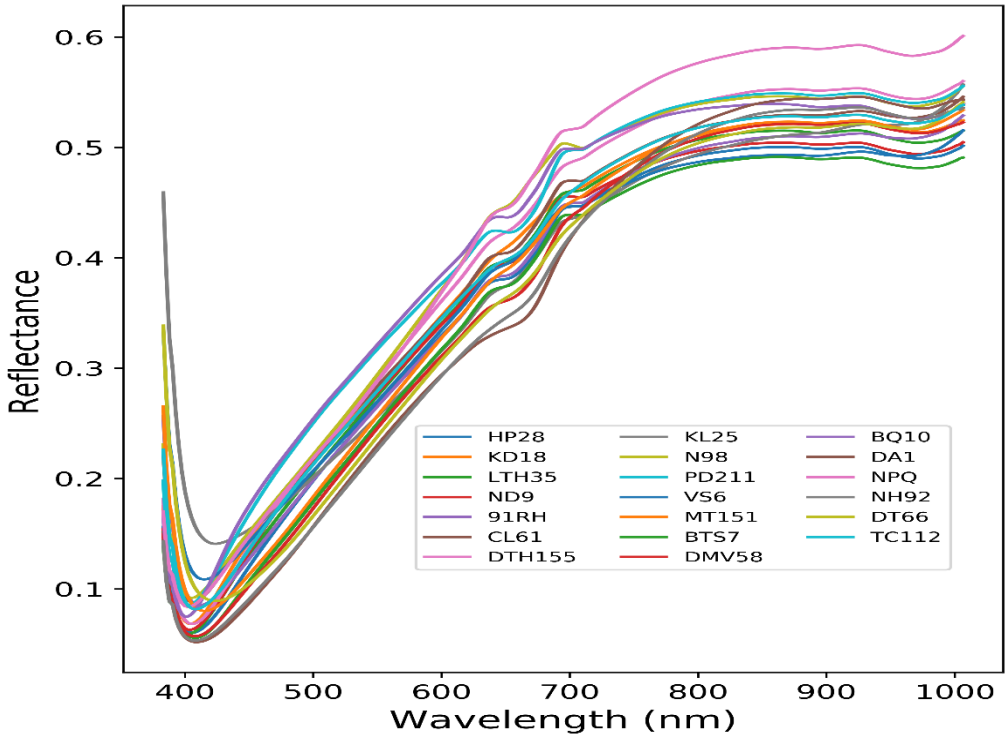



Figure 6.2 The average spectral profiles of 20 rice seed species.

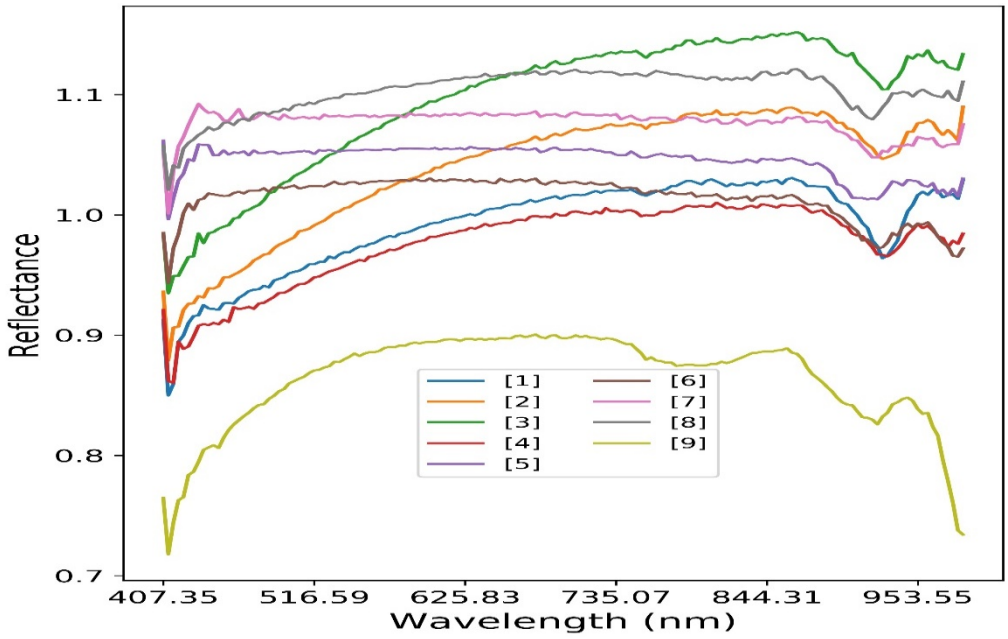



Figure 6.3 The average spectral profiles of 9 sugar species

### 6.1.3. Proposed Approach

#### *6.1.3.1. Concepts*

The proposed approach (GA + F-LDA) consists of the following three important steps, which are illustrated, in Figure 6.4: data acquisition, dimensionality reduction (DR), and classification. The proposed approach differs from those reported in the review of techniques used by various authors to classify hyperspectral images of rice seeds presented in Chapter 3 in the way the dimensionality reduction stage is realized. In the GA + F-LDA, dimensionality reduction is performed in two steps: 1) optimal feature subset selection using a Genetic Algorithm and 2) feature extraction using F-LDA. A

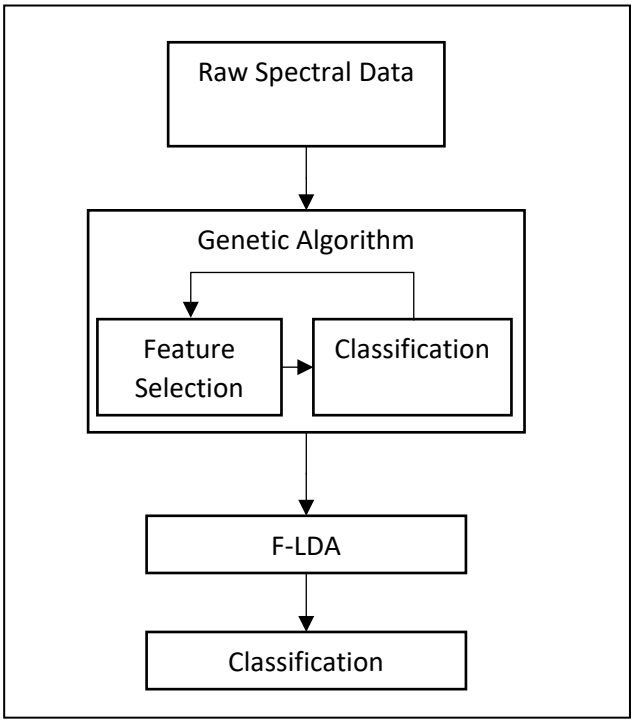



Figure 6.4 Proposed technique

data subset is extracted from each of the original spectral datasets through the selection of optimal spectral features using the Genetic Algorithm. The features of the extracted data subsets are then reduced further using F-LDA. This allow the application of F-LDA on a reduced dataset (number of features is now much less than what is in the

original dataset) while also improving the classification performance of the conventional LDA. It is expected that, in principle, the use of reduced number of features will further reduce the computational complexity and memory requirement of F-LDA. Implementation details of the dimensionality reduction stage and classification step are explained in the next subsection.

#### *6.1.3.2. Implementation Details*

First, the genetic algorithm is implemented in 3 key steps which are motivated by the principles of natural selection and genetics: selection, crossover and mutation[101],[117],[118],[102]. The algorithm randomly initiates a population of candidate solutions, uses an objective function to estimate the fitness of the current set of candidate solutions, discards the candidate solutions which are considered unfit and mates the fitter ones by crossover and mutation to produce the next generation offspring solutions (a new population of candidate solutions). This process is repeated over many generations until either an optimal solution is achieved or the maximum generation set is reached. When offspring solutions are produced by crossover, they contain genes which are present in the parents' chromosomes (each solution is considered a chromosome with a set of genes). Offspring solutions produced by mutations contain genes which are not present in both parents. During mutation, the search space is explored to achieve a global optimal solution. In this thesis, the crossover and the mutation was set to a probability of 0.6 and 0.2 respectively while the initial population of candidate solutions and maximum number of generations were set to 100 [118],[102]. The algorithm was implemented in PYTHON using the scikit-learn module, sklearn-genetic [36] to select the feature subset that maximizes the

objective function. The classifier and the fitness (objective) function utilised are Random Forest and accuracy respectively.

Second, optimal spectral features of the extracted data subsets are further reduced using F-LDA. F-LDA converts each spectral vector in the hyperspectral data matrix in to a 2D matrix (folded sample or vector) of size $G \times B = f$, where $f$ is the length of each spectral vector and also the number of features in the hyperspectral data. F-LDA then applies the traditional LDA steps on the new data which is a now set of 2D matrices, project the data unto a lower dimensional space, unfolds the projected samples and present the unfolded samples to models for classification. Instead of processing a set of spectral vectors as in LDA, F-LDA handles a set of folded spectral vectors (2D matrices). Different dimensions (configurations) of the folded samples (converted matrices) are considered in this chapter and the one that produces the best classification results is selected as the best and reported. A full explanation of the theoretical and mathematical concepts of both the LDA and F-LDA can be found in Chapter 2 and 5 of this thesis respectively.

Finally, a Random Forest (RF) model is trained for data classification. Random Forest model is used in this work due to the promising results it achieved when applied on hyperspectral imaging data of rice seeds in related papers [8],[9],[59]. For the Random forest classification, the data is split into training and testing sets. The number of decision trees in the Random forest, D is varied from 100 to 1000 in step of 100. The optimal values of D are determined using a k-fold (k=5) cross validation on the training set. That is, the Random Forest classifier was trained and validated k times. In each case, the classifier was trained using k-1 of the folds and validated using the remaining fold. The cross validation then output the average of the accuracies recorded in all

cases. The classification results obtained during the final evaluation on the test set is reported in the next section.

## 6.2. Results and Discussion

Accuracy together with Precision, $P$, Recall, $R$, and $F_1$ score are adopted as metrics for performance evaluation and comparison of the proposed approach with different feature schemes using the RF classifier. Entire spectral feature sets or selected/reduced spectral feature sets resulting from a dimensionality reduction process are used for the training of classification models. Therefore, in this work, the following different feature schemes are utilised and their performances compared with that of the proposed GA+F-LDA features (F-LDA features extracted from the GA outputs):

1) Raw spectral features
2) An optimal feature subset selected from the raw spectral feature sets using the GA only
3) LDA features extracted from the raw spectral feature sets
4) PCA features extracted from the raw spectral feature sets
5) PCA features extracted from the GA outputs
6) LDA features extracted from the GA outputs
7) Features extracted by the standard F-LDA i.e., without GA

### 6.2.1. Spectra Features Selection

Using the GA, the number of spectral features selected varied for different datasets as illustrated in Table 6.1. For the group of rice seed datasets with 10 and 20 species, the number of spectral features selected ranges from 96 to 177 and 94 to 225 respectively. The number of spectral features selected when the sugar data was utilised is 64.

In order to visualize the regions of the spectral range that contain the selected spectral features and are most relevant for the classification tasks in this work, the average spectra of a sample species taken from each of the datasets considered is plotted and illustrated in Figure 6.5 to Figure 6.25. As shown in Figure 6.5, for the sugar dataset, features in both the visible region and the lower bands of the NIR region are useful in discriminating the sugar varieties. GA selected more informative features around the following three clusters: 429.19 – 534.79 nm, 574.85 – 753.28 nm and 811.54 – 986.32 nm regions. Selected features around 429.19 – 534.79 nm and 574.85 – 753.28 nm can be related to variation in the physical property of the varieties [23],[7],[104] while those selected around 811.54 – 986.32 nm can be related to the second and third overtones of O-H, C-H and N-H absorption bands [119],[104].

For the rice seed dataset, as can be seen in Figure 6.6 to Figure 6.25, the selected features can be seen to have spread across the spectral range and cannot be grouped into clusters in most of the cases. This indicates that all regions of the spectral range are relevant and that the differences in chemical property (composition) in the lower bands of the NIR region ~ (700 - 1000) nm and physical property (colour variation among the species) in the visible region ~ (385 - 700) nm of the spectral profiles are useful in discriminating the rice seed species [23],[7],[104]. While both Visible and NIR regions of the spectral range can be seen to be important, GA has been able to reduce redundancy (resulting from correlation among the spectral bands) in the datasets by selecting only the relevant spectral features in each dataset.

Table 6.1 Number of features in each dataset before and after the selection of optimal features using GA

| Dataset | Random subsets | Number of spectral features | Number of selected features |
|---|---|---|---|
| Rice seeds (10 species) | 1st | 256 | 139 |
| | 2nd | | 96 |
| | 3rd | | 114 |
| | 4th | | 177 |
| | 5th | | 145 |
| | 6th | | 147 |
| | 7th | | 132 |
| | 8th | | 110 |
| | 9th | | 111 |
| | 10th | | 103 |
| Rice seeds (20 species) | 1st | 256 | 155 |
| | 2nd | | 123 |
| | 3rd | | 183 |
| | 4th | | 225 |
| | 5th | | 212 |
| | 6th | | 157 |
| | 7th | | 137 |
| | 8th | | 94 |
| | 9th | | 140 |
| | 10th | | 128 |
| Sugar | N/A | 160 | 64 |

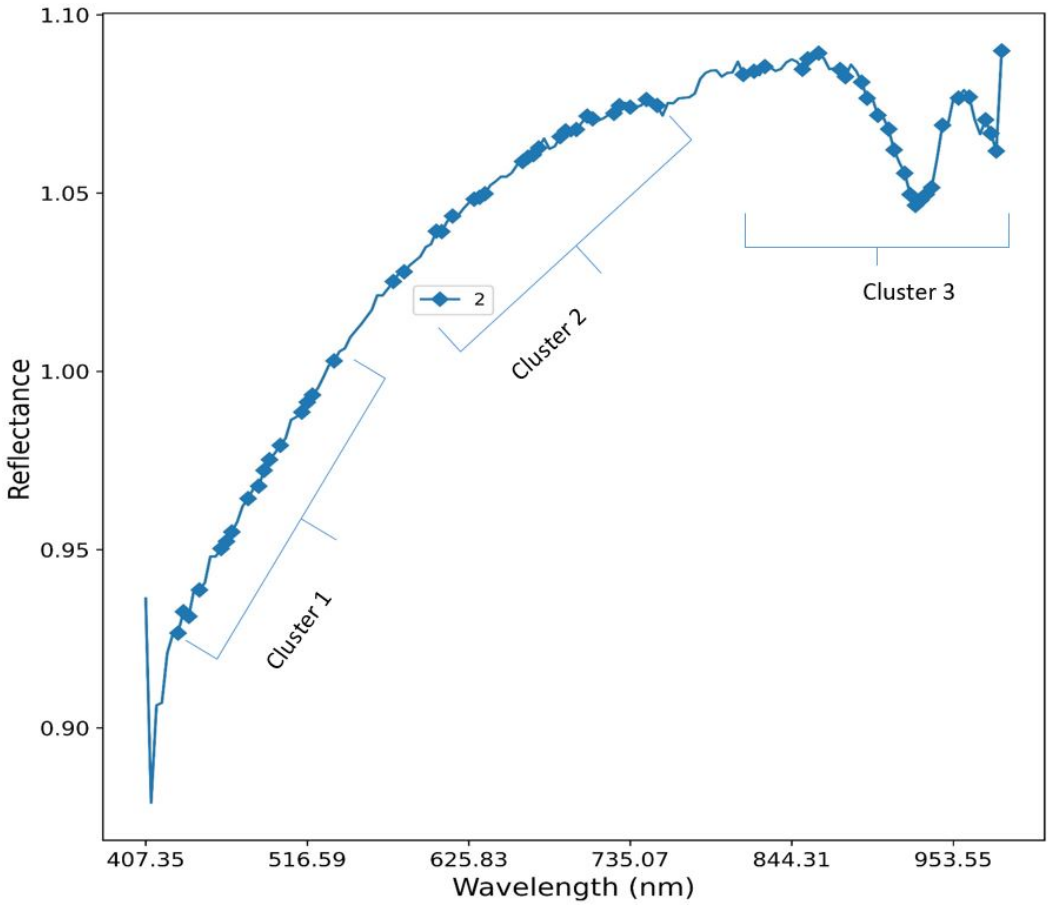



Figure 6.5 Average spectra of a sample species taken from the sugar dataset

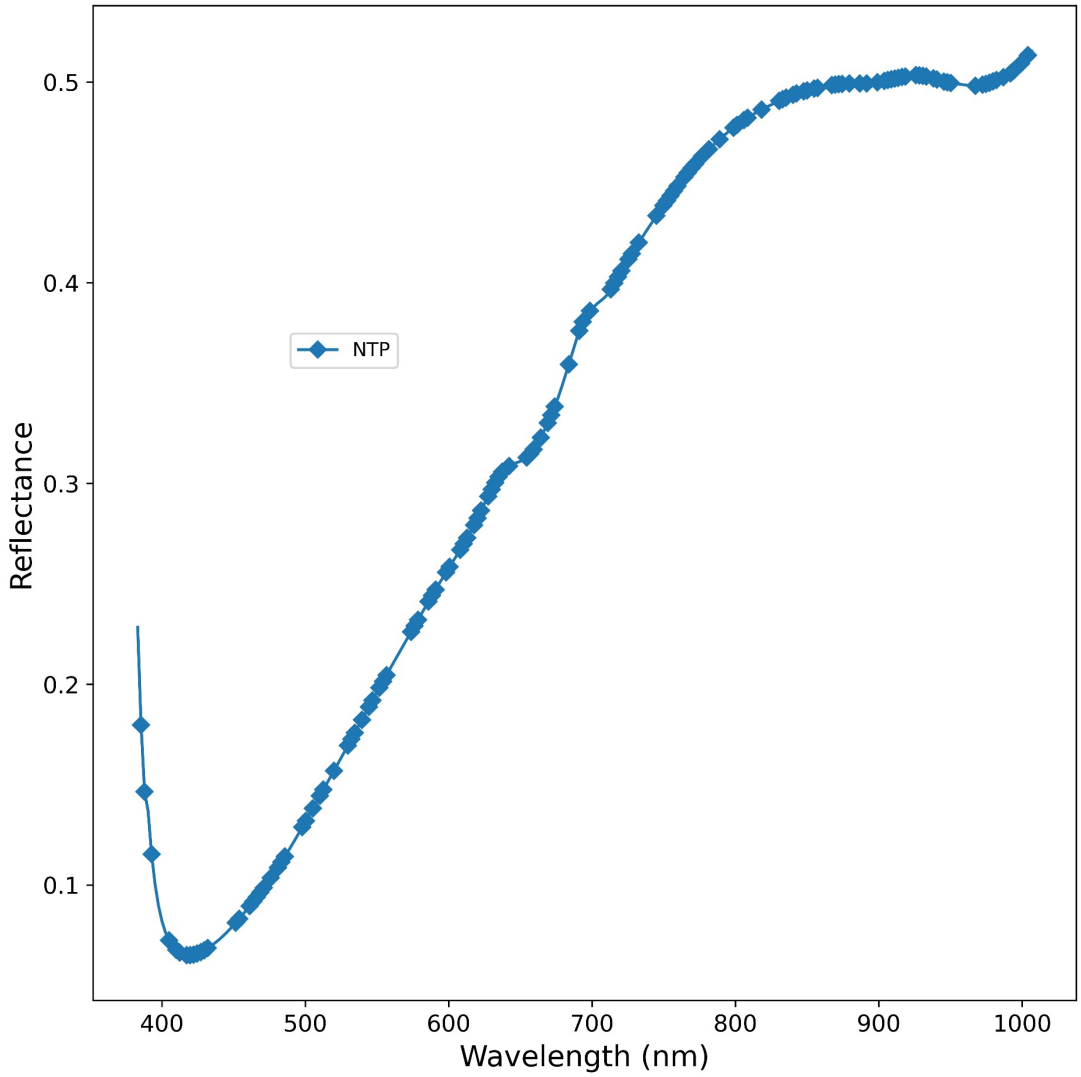



Figure 6.6 Average spectra of a sample species taken from the first data subset containing 10 rice seed species

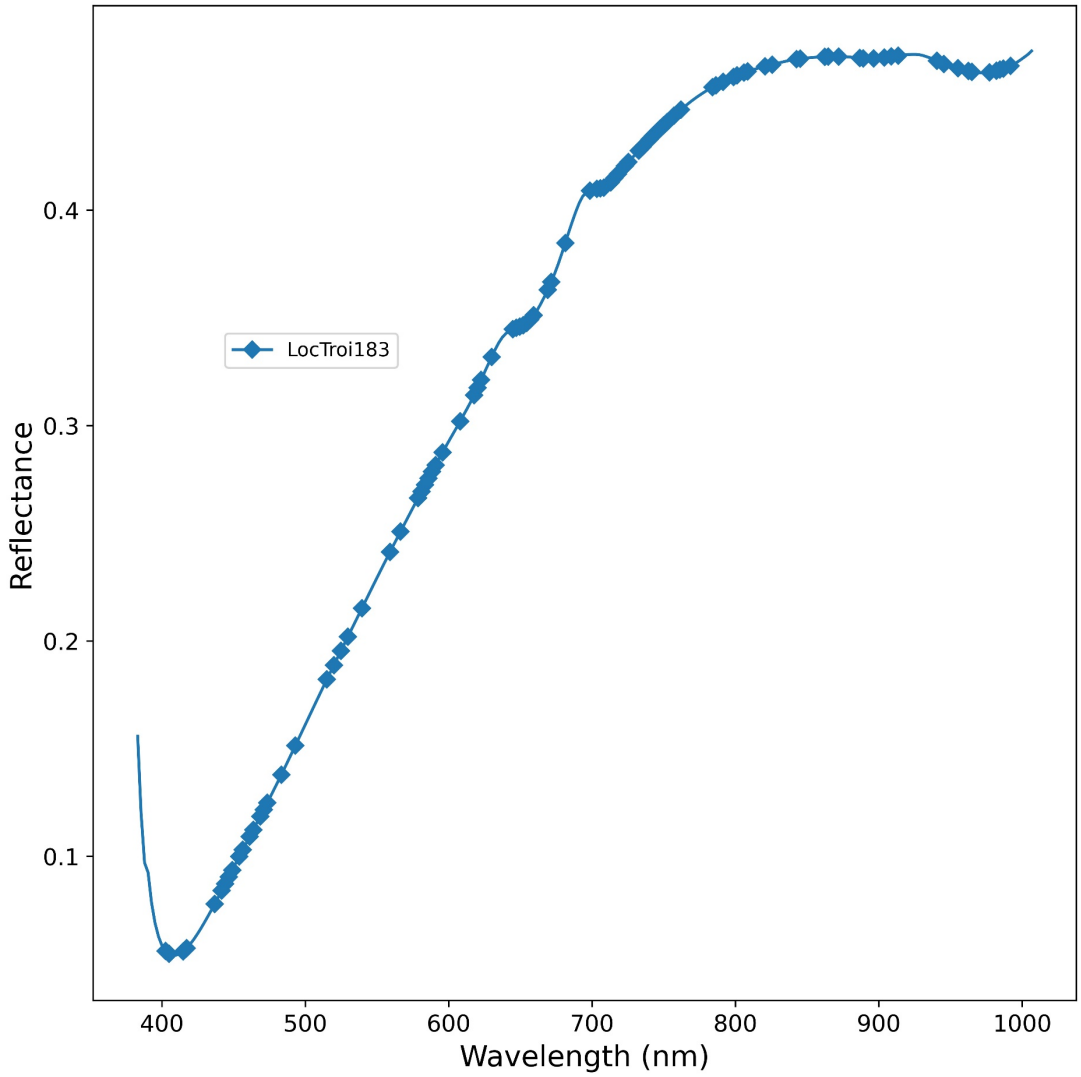



Figure 6.7 Average spectra of a sample species taken from the second data subset containing 10 rice seed species

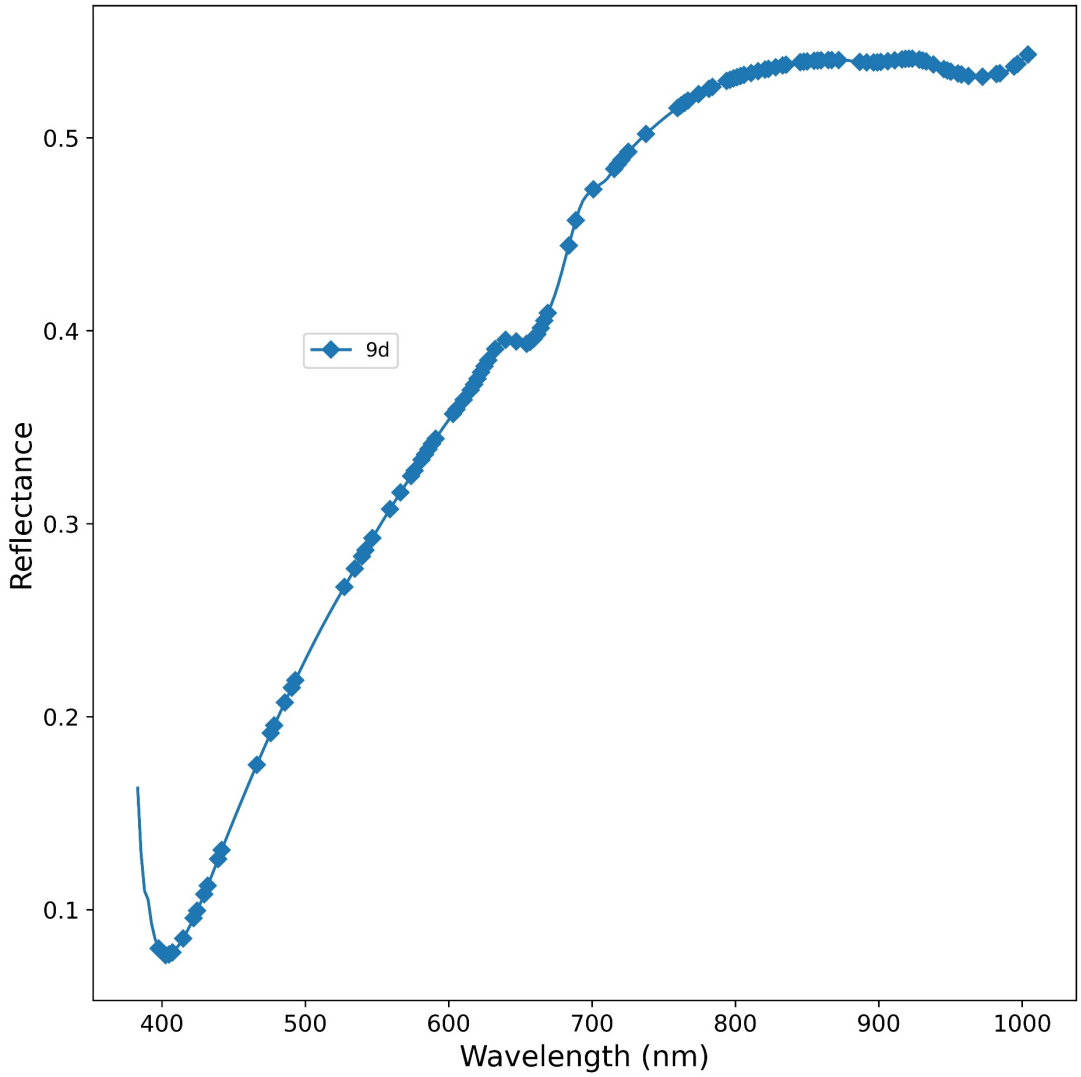



Figure 6.8 Average spectra of a sample species taken from the third data subset containing 10 rice seed species

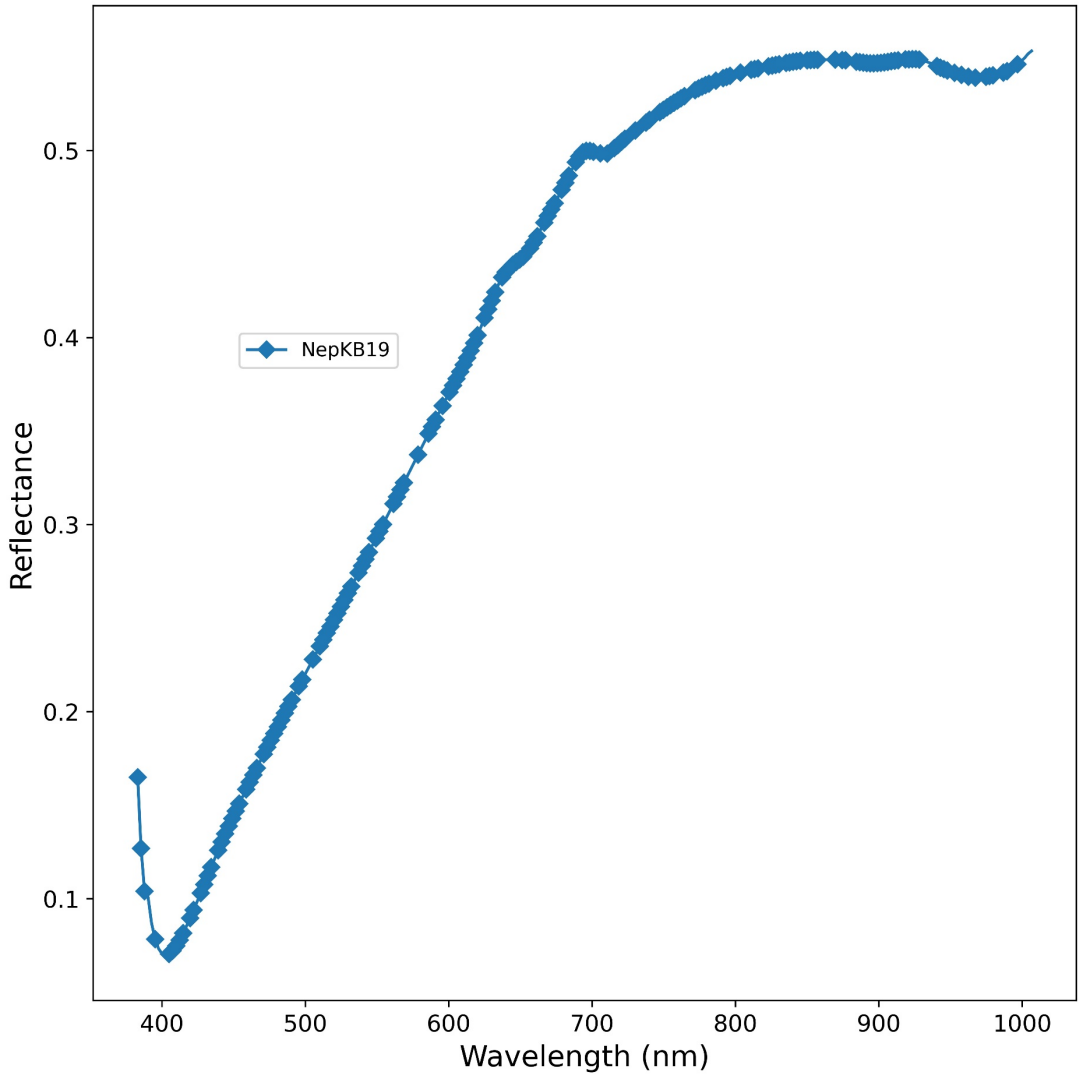



Figure 6.9 Average spectra of a sample species taken from the fourth data subset containing 10 rice seed species

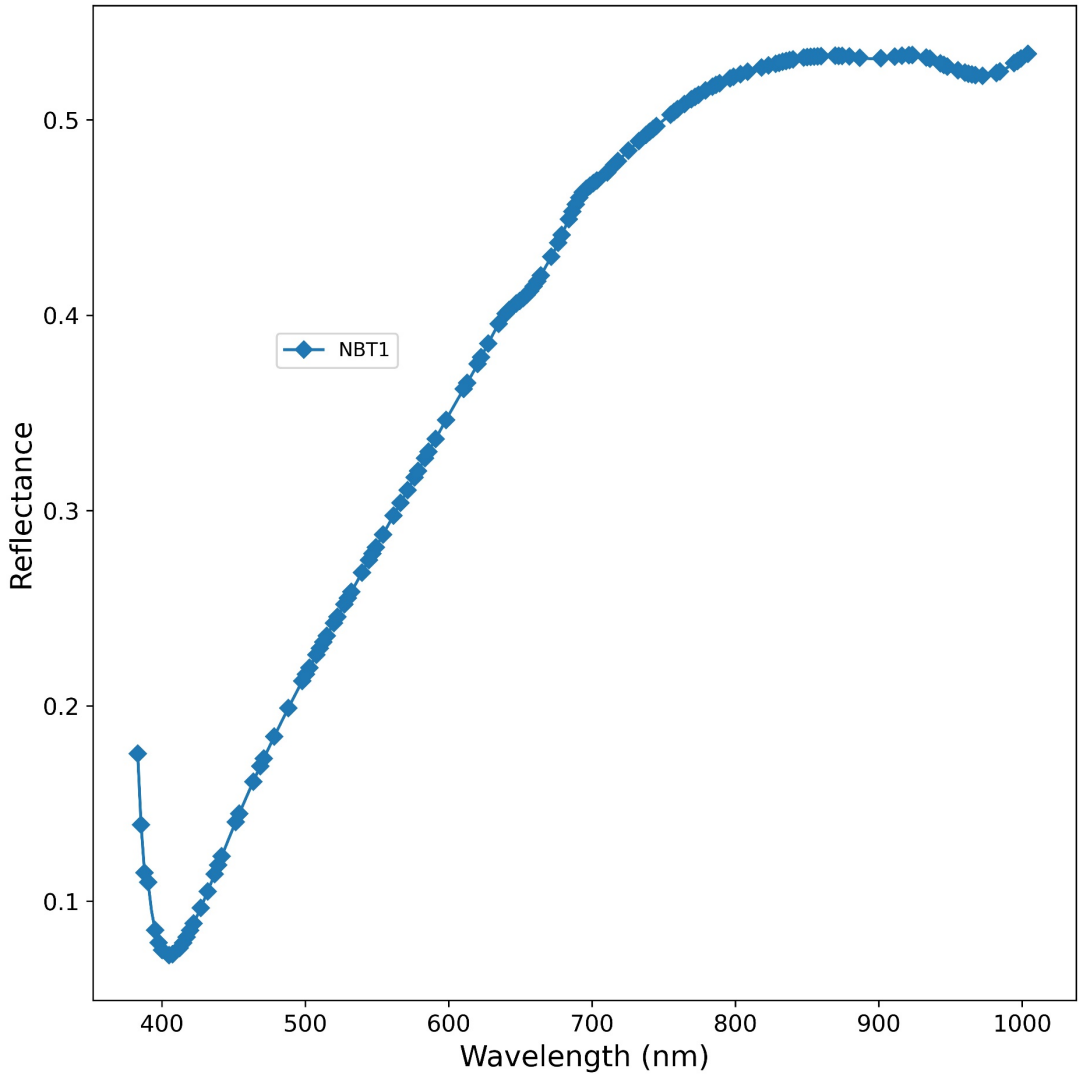



Figure 6.10 Average spectra of a sample species taken from the fifth data subset containing 10 rice seed species

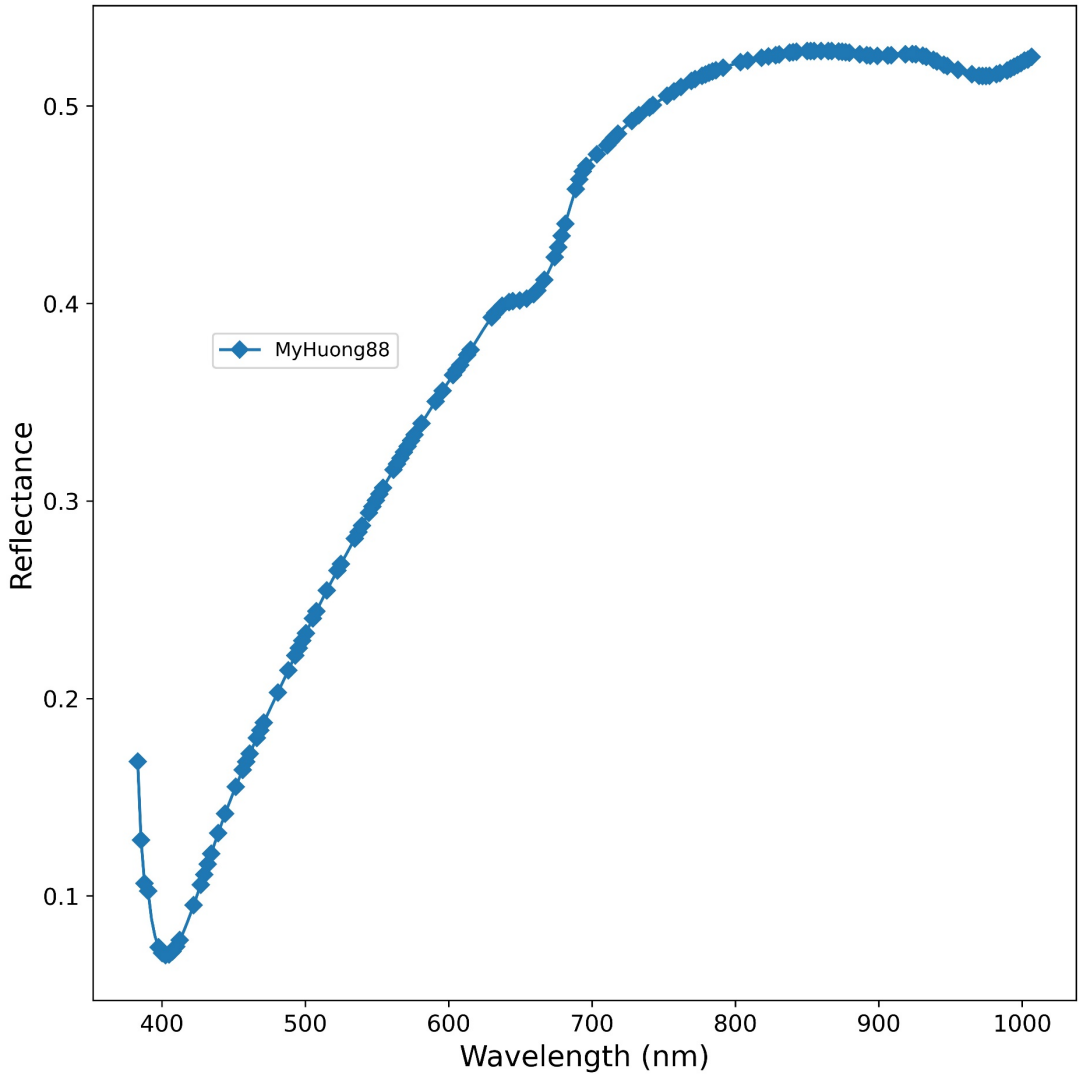



Figure 6.11 Average spectra of a sample species taken from the sixth data subset containing 10 rice seed species

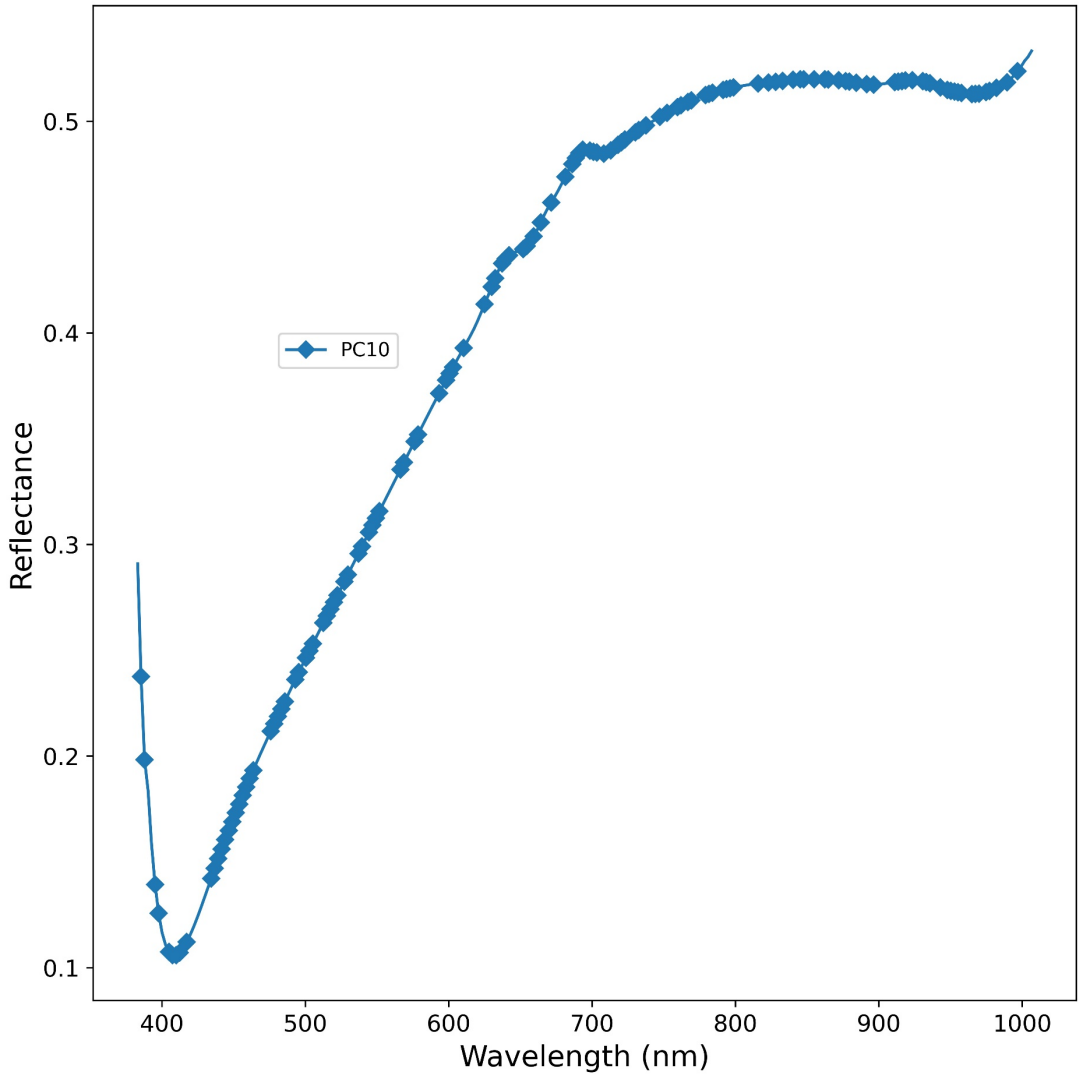



Figure 6.12 Average spectra of a sample species taken from the seventh data subset containing 10 rice seed species

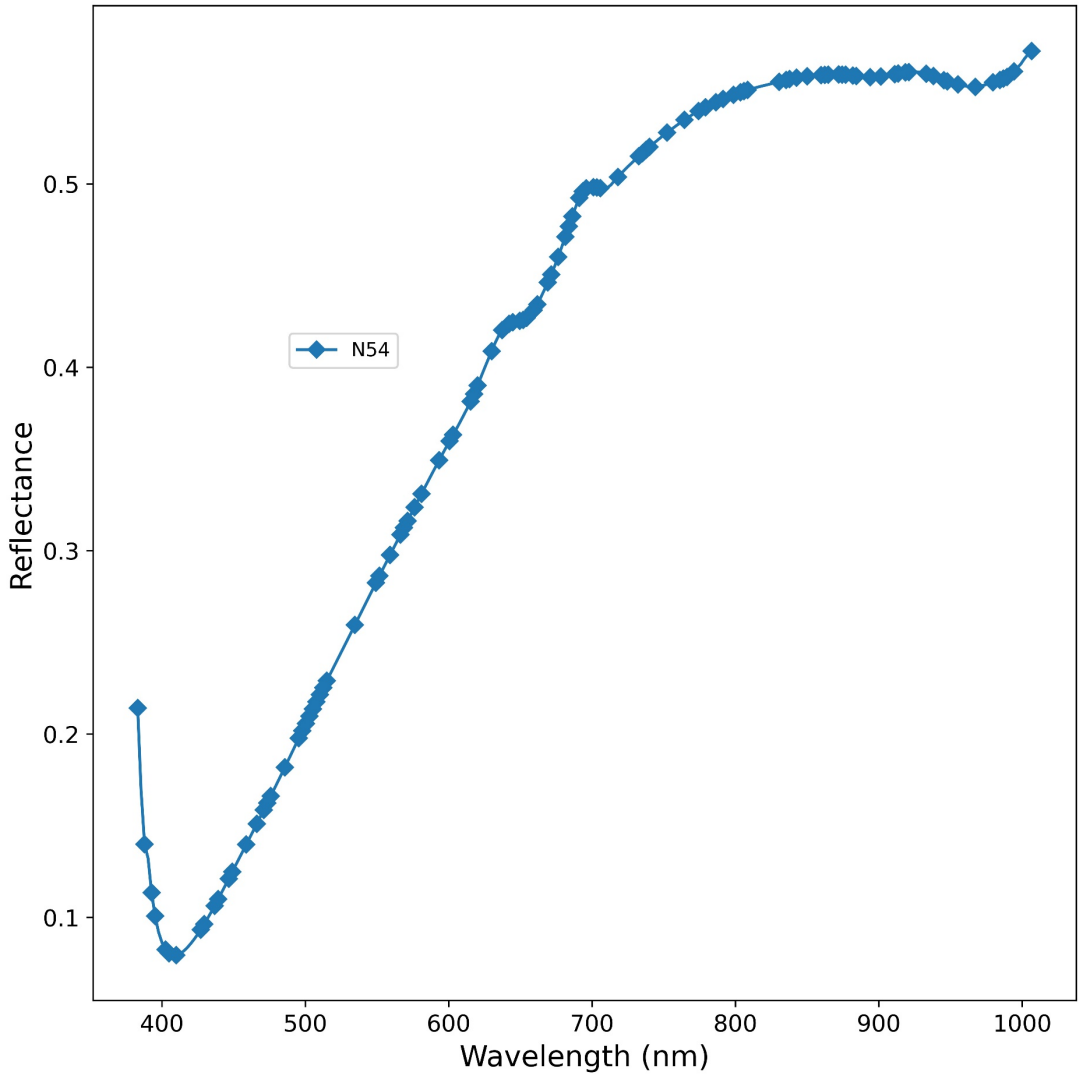



Figure 6.13 Average spectra of a sample species taken from the eight data subset containing 10 rice seed species

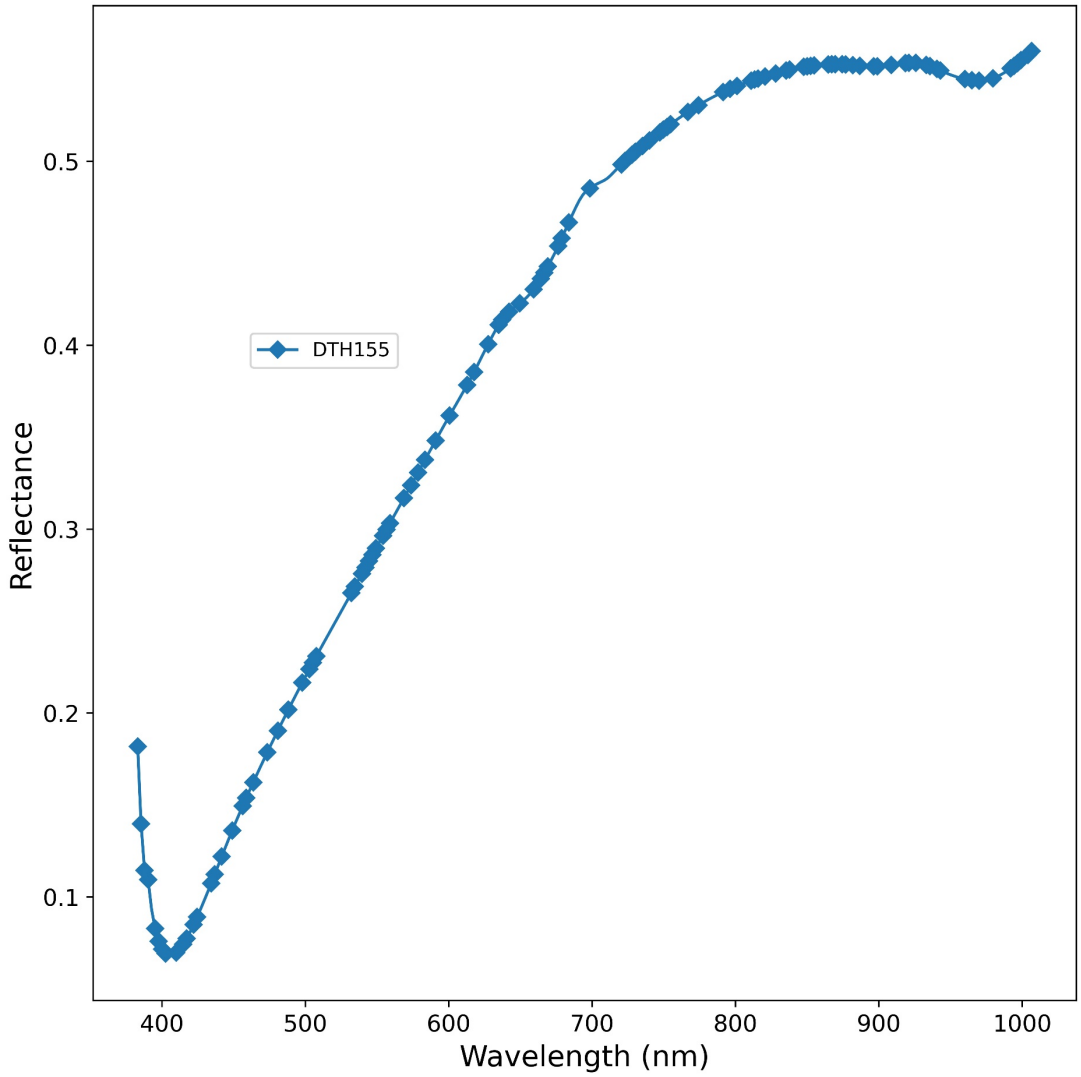



Figure 6.14 Average spectra of a sample species taken from the ninth data subset containing 10 rice seed species

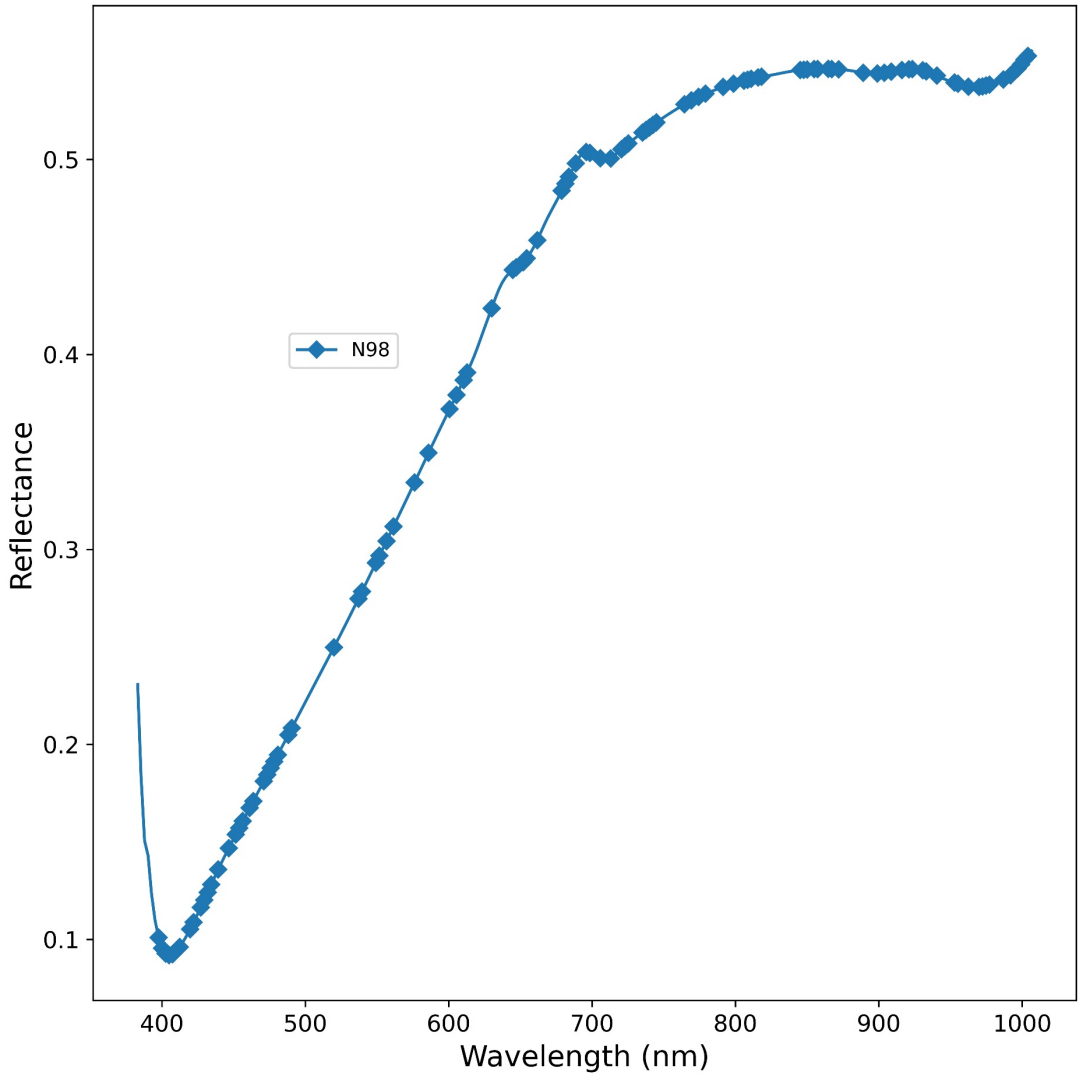



Figure 6.15 Average spectra of a sample species taken from the tenth data subset containing 10 rice seed species

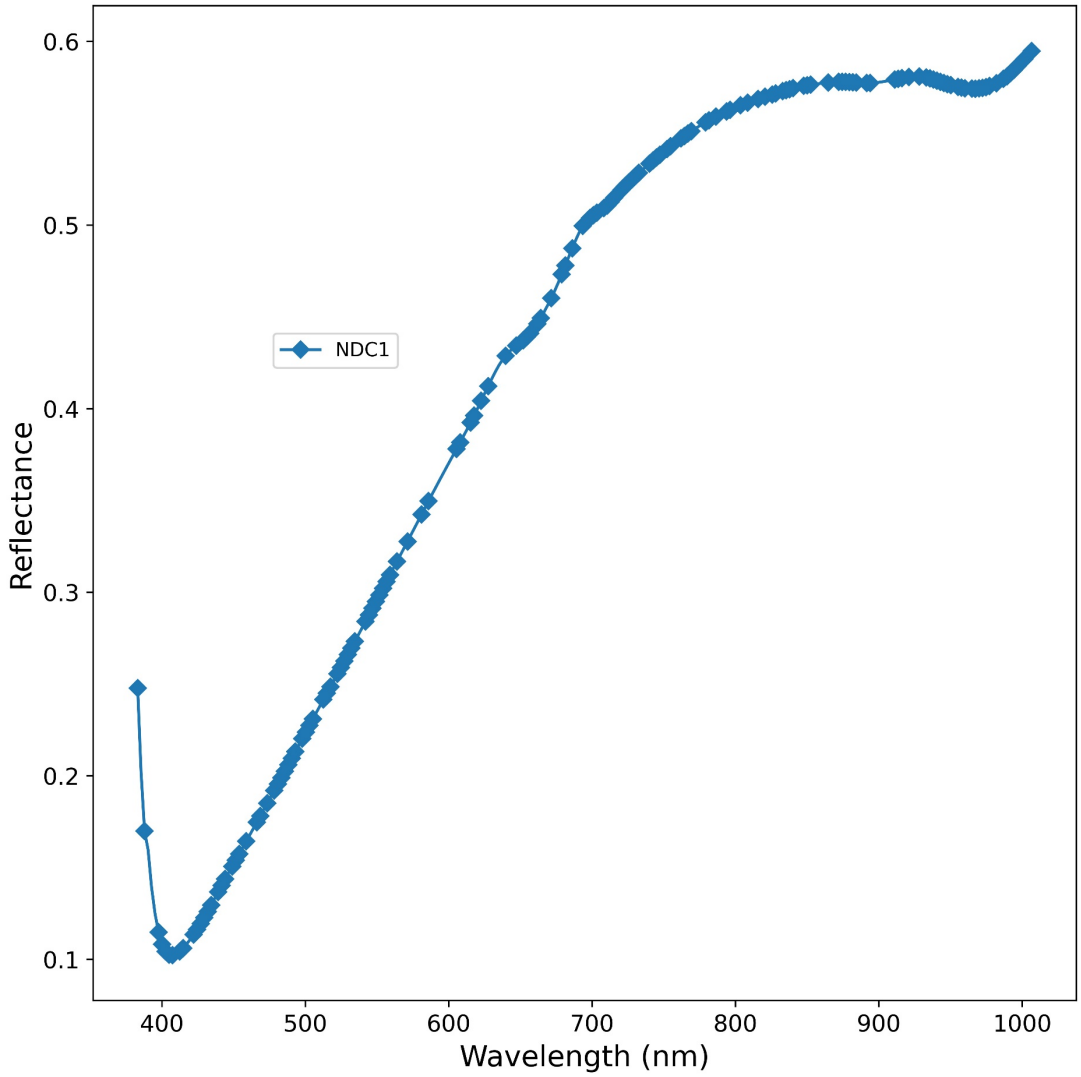



Figure 6.16 Average spectra of a sample species taken from the first data subset containing 20 rice seed species

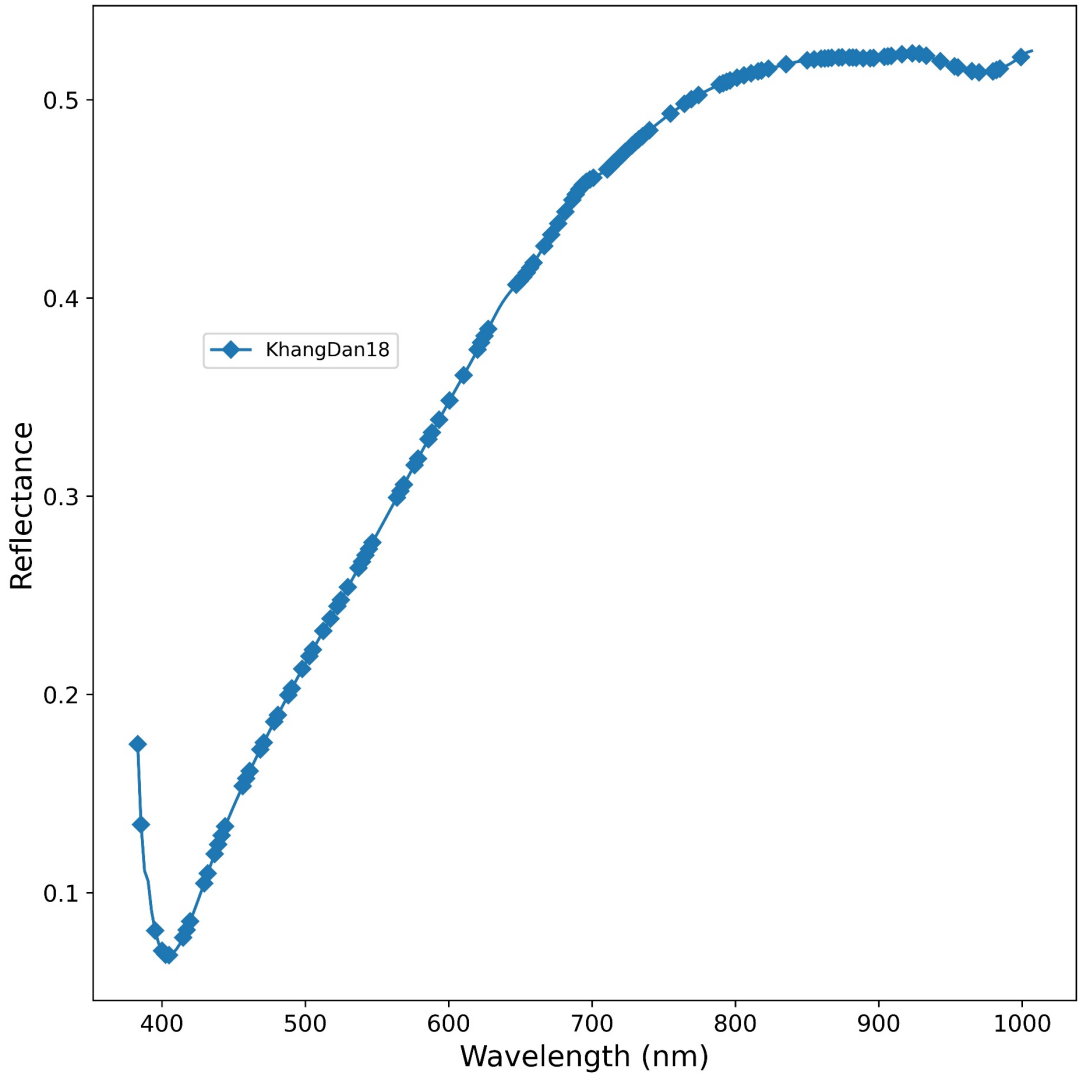



Figure 6.17 Average spectra of a sample species taken from the second data subset containing 20 rice seed species

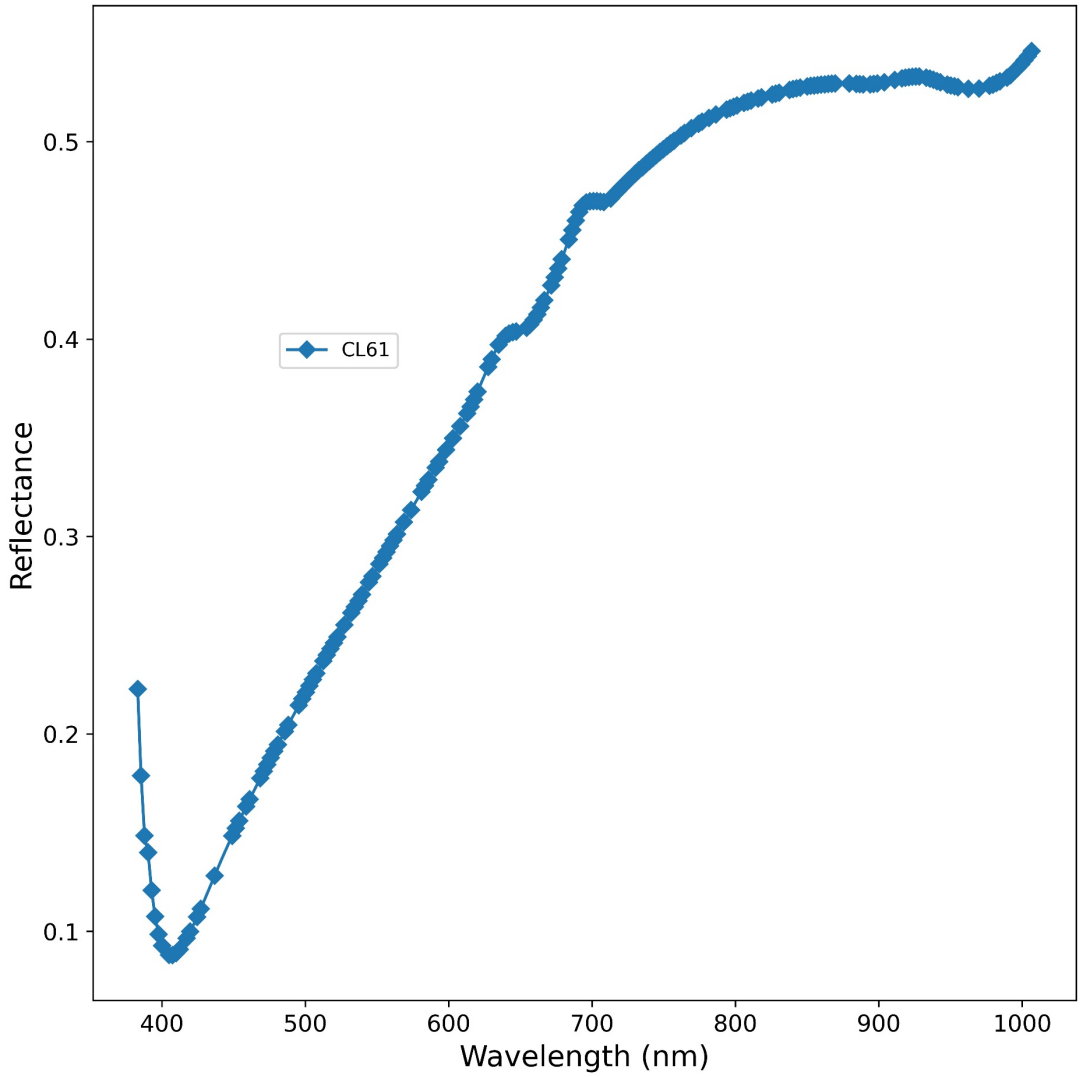



Figure 6.18 Average spectra of a sample species taken from the third data subset containing 20 rice seed species

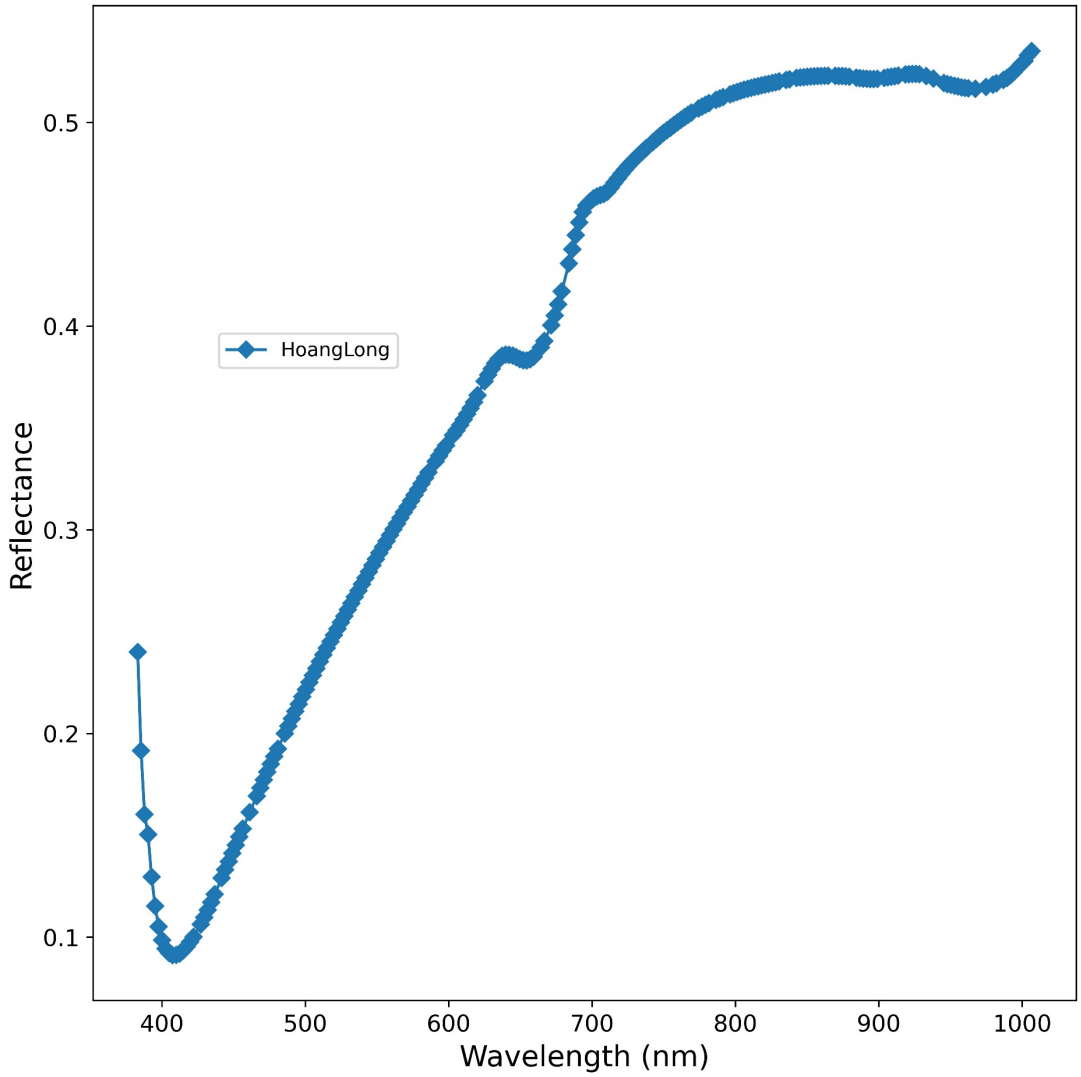



Figure 6.19 Average spectra of a sample species taken from the fourth data subset containing 20 rice seed species

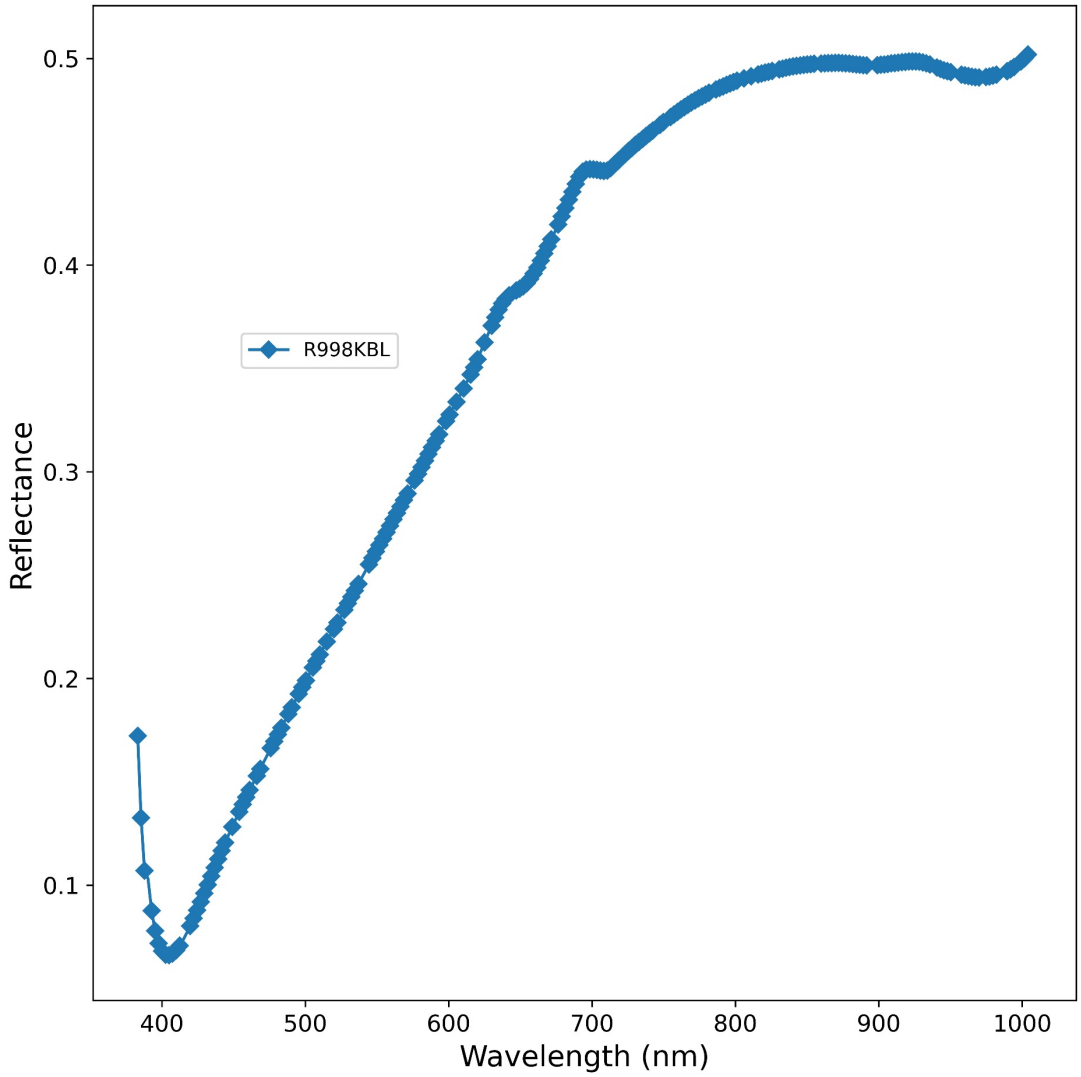



Figure 6.20 Average spectra of a sample species taken from the fifth data subset containing 20 rice seed species

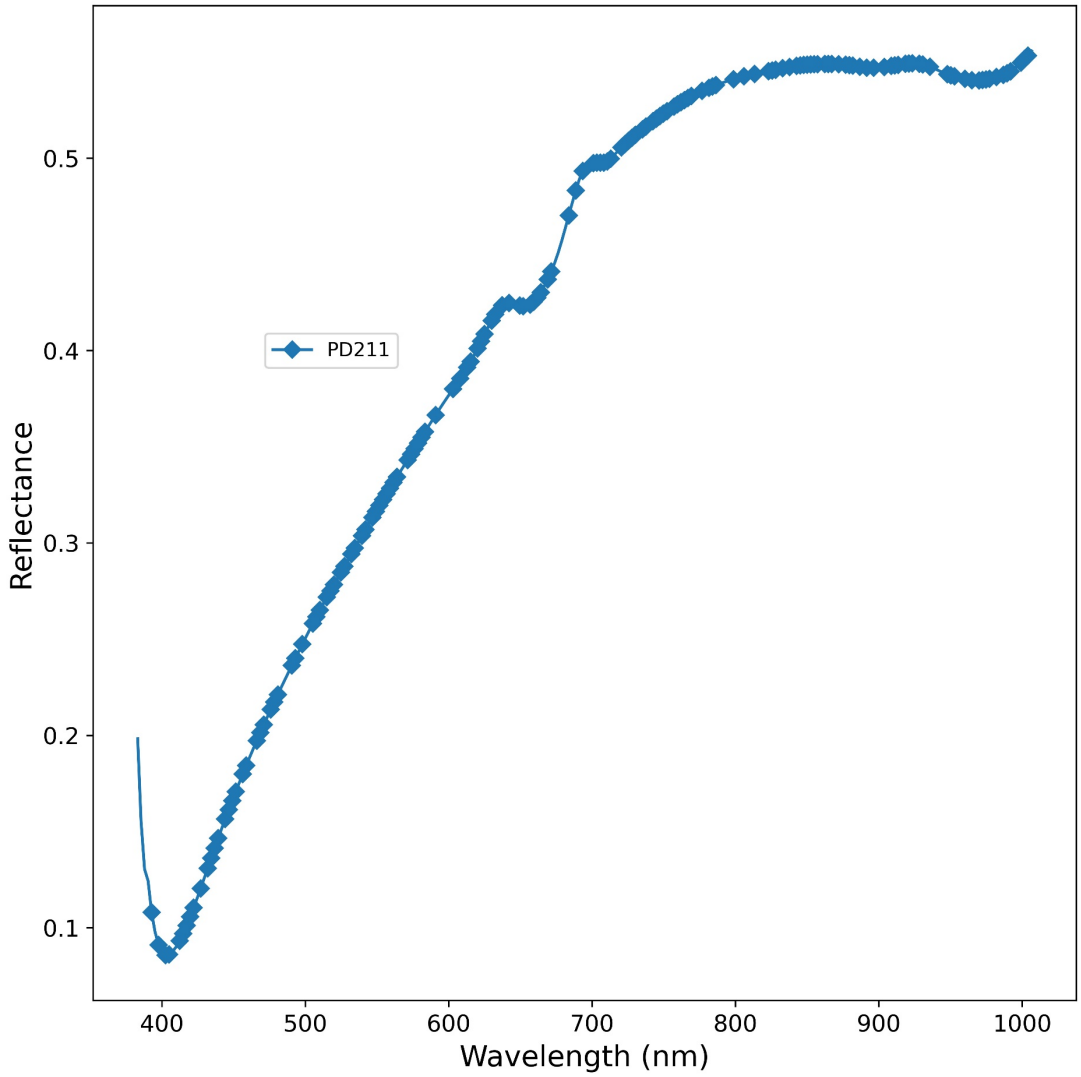



Figure 6.21 Average spectra of a sample species taken from the sixth data subset containing 20 rice seed species

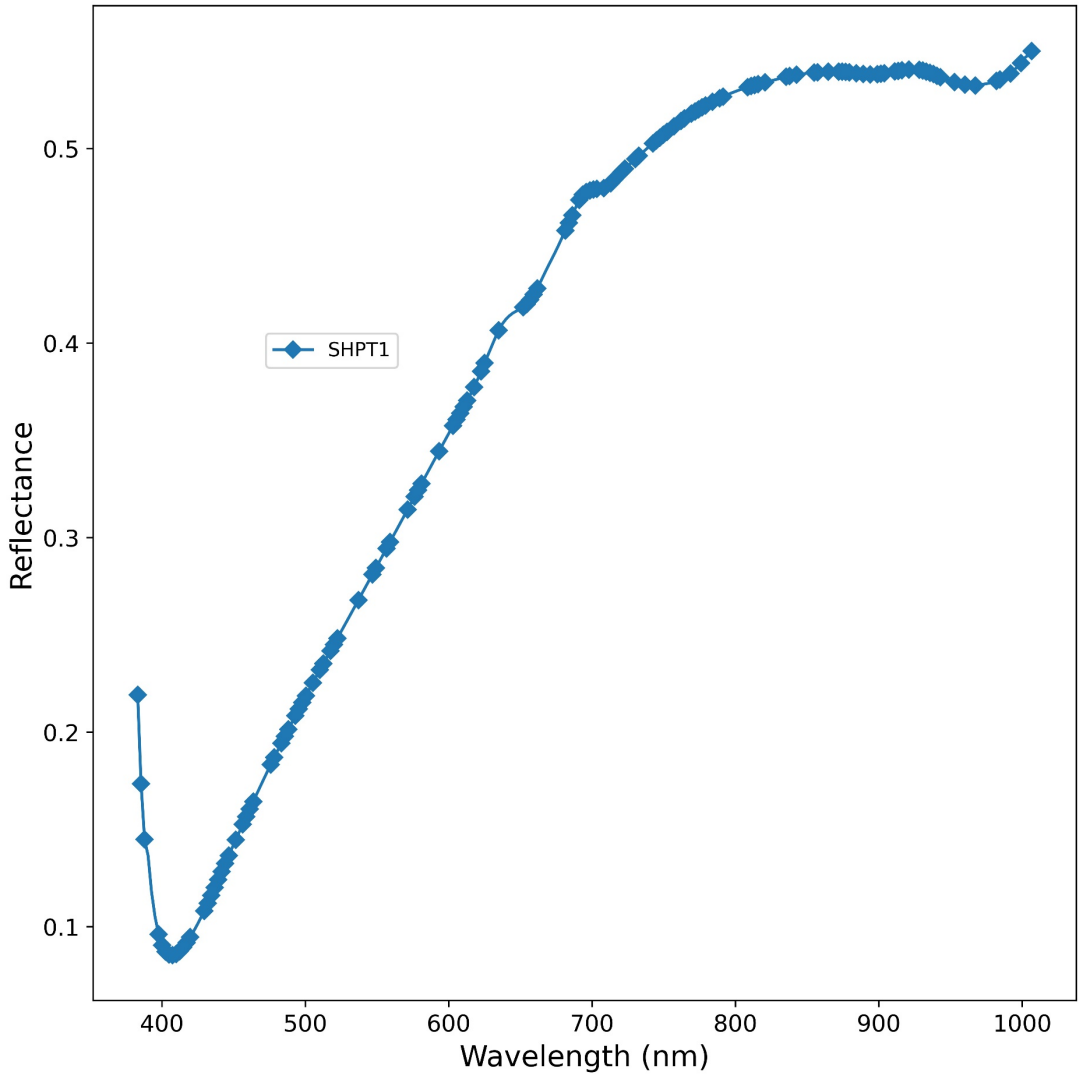



Figure 6.22 Average spectra of a sample species taken from the seventh data subset containing 20 rice seed species

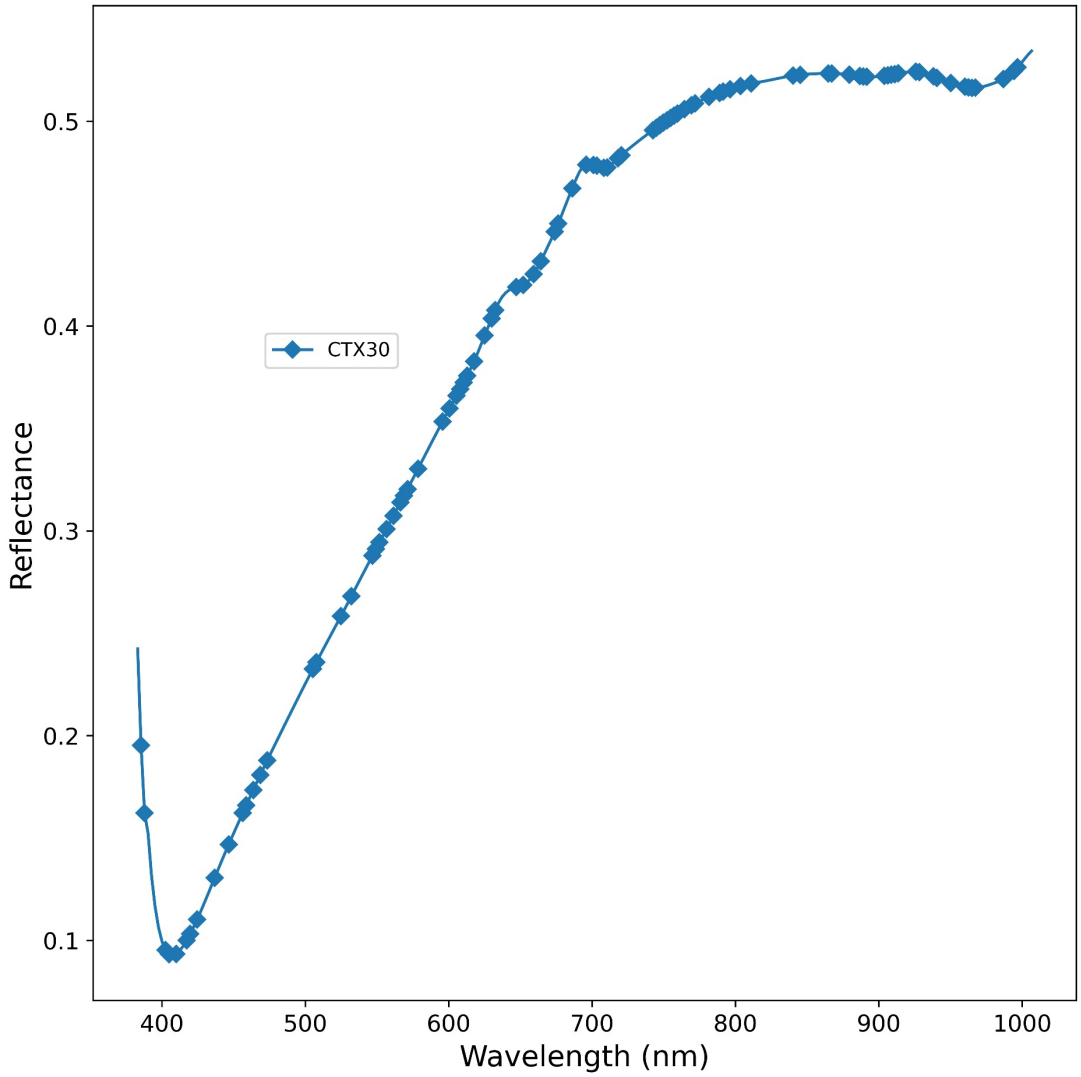



Figure 6.23 Average spectra of a sample species taken from the eight data subset containing 20 rice seed species

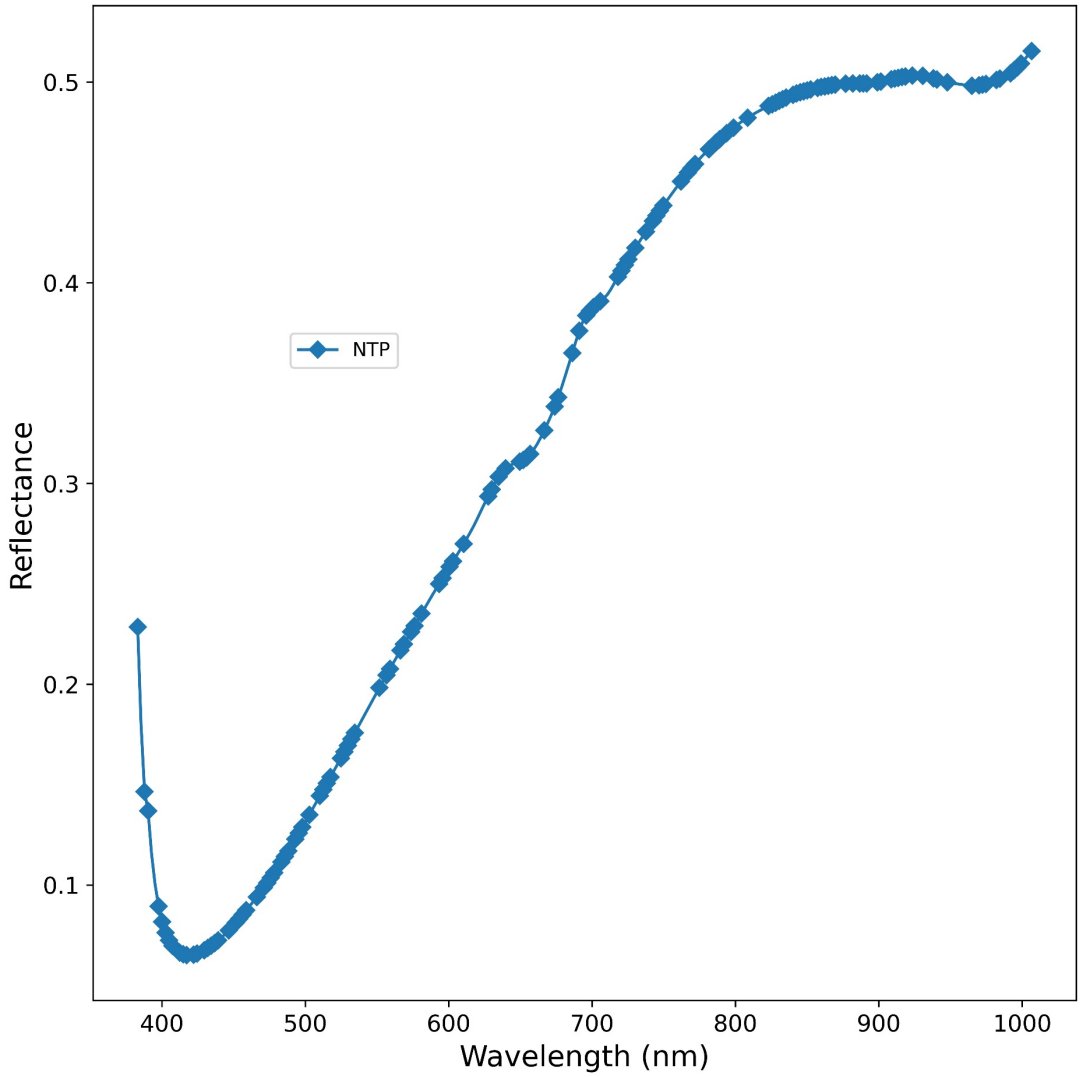



Figure 6.24 Average spectra of a sample species taken from the ninth data subset containing 20 rice seed species

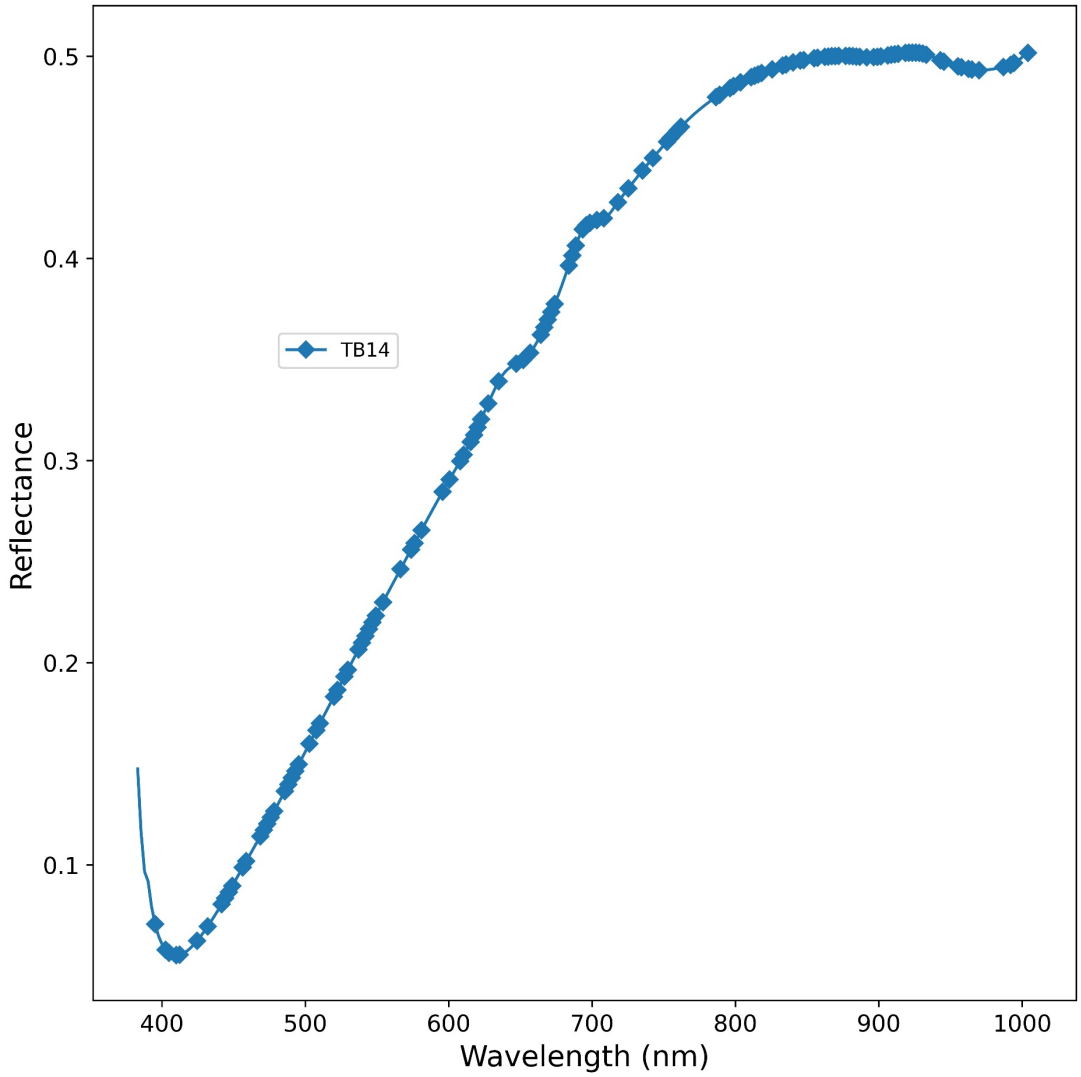



Figure 6.25 Average spectra of a sample species taken from the tenth data subset containing 20 rice seed species

### 6.2.2. Analysing the Performance on the Rice Seed Datasets

Firstly, before any dimensionality reduction of the rice seed spectral data is performed, the 10 random subsets of 10 rice seed varieties are used to train the RF classifier. Sub datasets with reduced feature subset are also selected from each of the random subsets using GA and used to train the RF classifier. F-LDA is applied on the raw spectral datasets and the datasets with the selected GA features. The outputs of the F-LDA from

both cases are used separately to train the RF classifier. In order to compare performance of the proposed approach, (GA+ F- LDA) with other techniques (LDA, GA + LDA, PCA and GA+PCA), firstly, outputs of PCA are applied on the spectral data and the data with the selected GA features, starting from 1 up to 10 principal components in the dataset. Secondly, outputs of LDA are applied on the spectral data and the data with the selected GA features, starting from 1 up to $c - 1$ where $c$ is the number of species in the dataset. Finally, outputs of F-LDA and GA+F-LDA are separately combined with 6 spatial features. In all the cases considered, the size of the training samples are varied and the average of classification results obtained using the 10 random subsets of rice seed varieties are computed and presented in Table 6.2 and Table 6.3.

From the classification results presented in Table 6.2 and Table 6.3, it can be observed that the accuracy and $F_1$ score with the spectral features only were improved when the RF was trained with the selected GA features. As expected, these results are significantly improved when the RF is trained with the output of PCA and LDA applied to the raw spectral data. Ultimately, as can be seen in Table 6.2 and Table 6.3, the classification accuracy and $F_1$ score obtained when LDA features extracted from the spectral data (with high training to testing samples ratio) are utilised is greater than the accuracy and $F_1$ score obtained when the PCA features are utilised. This validates the motivation for proposing the hybridization of GA and F-LDA for dimensionality reduction of hyperspectral data in rice seed classification.

It can also be observed that the classification accuracy and $F_1$ score increase with the size of the training sets when the RF classifier is trained with the outputs of the LDA and that GA+LDA, F-LDA and GA+FLDA feature schemes all compensate for the

inability of LDA to give comparable classification results when applied on small training samples. Furthermore, Table 6.2 and Table 6.3 show that GA + F-LDA features performed better than the features produced by PCA, LDA, GA+PCA and GA+LDA when used to train the classifier. It is also observed that the proposed GA+F-LDA gave accuracy and $F_1$ score which are lower than those given by F-LDA only as can be seen in Table 6.2 and Table 6.3. The reduction in classification performance of GA+F-LDA (when compared with F-LDA only) can be considered insignificant since the accuracy and $F_1$ score achieved by both GA+F-LDA and F-LDA only are comparable when the standard deviation reported in both cases are considered.. As can be seen in Table 6.2 and Table 6.3, classification performance of F-LDA and GA+F-LDA are improved when their outputs are combined with the spatial features.

While the proposed GA + F-LDA is slightly outperformed by the standard F-LDA in term of accuracy and $F_1$ score, the proposed GA + F-LDA can reduce the computational complexity and memory requirement at the different stages of F-LDA (in majority of the cases considered) as illustrated in Table 6.8 and Table 6.9. These reduction in computational complexity and memory requirement may be preferred in practice, though at the expense of slight reduction in classification performance (accuracy and $F_1$ score). The adoption of which technique (F-LDA or GA+FLDA) to use would then depend on the application. The important point is that F-LDA comes out well whether it is applied on the datasets with full spectral features or on reduced datasets (dataset containing optimal features selected by GA). The average feature extraction time (s) of F-LDA (when applied on the random datasets of 10 varieties) in both cases are also illustrated in Table 6.14. As can be seen in Table 6.14, F-LDA used less time in extracting the features when applied on the reduced datasets, though the

reduction in feature extraction time is not significant when considering the standard deviation reported for each case. Also, the time used by F-LDA in both cases can be seen in Table 6.14 to be less than 0.1s. The reduction in computational complexity and memory requirement attained by F-LDA when applied on the reduced datasets therefore demonstrates the potential of the proposed approach (GA+F-LDA) in feature extraction and dimensionality reduction of hyperspectral data.

The above process is repeated on the rice seed datasets of 20 varieties and the classification results presented in Table 6.4 and Table 6.5. Again, as can be seen in Table 6.4 and Table 6.5, GA+LDA, F-LDA and GA+F-LDA feature schemes continue to compensate for the inability of LDA to perform well on small training samples. One can also see in Table 6.4 and Table 6.5 that GA+F-LDA continues to outperform the other techniques namely PCA, LDA, GA+PCA, GA+LDA and gave $F_1$ score and an accuracy which are lower than those attained by F-LDA. Again, the reduction in classification performance of GA+F-LDA (when compared with F-LDA only) can be considered insignificant since the accuracy and $F_1$ score achieved by both GA+F-LDA and F-LDA only are comparable when the standard deviation reported in both cases are considered. As can be seen in Table 6.10 and Table 6.11, reduction in computational complexity and memory requirement are recorded for majority of the cases in F-LDA when applied on the reduced datasets (datasets containing the selected GA features). The average feature extraction time (s) of F-LDA when applied on the full and GA features is also illustrated in Table 6.14. As can be seen in Table 6.14, F-LDA used less time in extracting the features when applied on the reduced datasets, though the reduction in feature extraction time is not significant when considering the

standard deviation reported for each case. Also, the time used by F-LDA in both cases can be seen in Table 6.14 to be less than 0.25s.

It is worthy of note that the hyperspectral data of rice seeds used in this thesis were collected at a Visible - Near Infrared (VIS/NIR) range of ~ (385 – 1000) nm. To investigate whether the values below 400nm affect the results, some extra work was carried out and this is shown in Appendix A.

### 6.2.3. Analysing the Performance on the HSI Data of Sugar Dataset

In order to show the potential of the proposed technique for dimensionality reduction of hyperspectral data of other Agri-tech products, the process described in 6.2.1 is repeated on the sugar dataset of 9 varieties and the classification results illustrated in Table 6.6 and Table 6.7. Table 6.6 and Table 6.7 show that GA+F-LDA features performed better than those produced by PCA, LDA, GA+PCA and GA+LDA. Table 6.6 and Table 6.7 also show that the F-LDA features gave accuracy and $F_1$ score which are higher than those given by GA+F-LDA features when used to train the RF classifier. While the standard F-LDA gave a slightly better classification performance than the proposed GA+F-LDA in term of accuracy and $F_1$ score, as can be seen in Table 6.6 and Table 6.7, reduction in computational complexity and memory requirement was achieved at the different stages of the F-LDA when it was applied on the features selected by GA. This once again demonstrate the potential of applying F-LDA on reduced datasets (datasets containing optimal features selected by GA). It is also worth noting that, as illustrated in Table 6.14, F-LDA used less time in extracting the features when applied on the reduced dataset, though the time used by F-LDA in both cases can be seen to be less than 0.1s.

Table 6.2 Classification accuracies using the rice seed datasets with 10 varieties

| Train:Test Ratio | 30:70 | 40:60 | 50:50 | 60:40 | 70:30 | 80:20 |
|---|---|---|---|---|---|---|
| **Spectra** | 66.70 ± 4.54 | 69.85 ± 5.02 | 69.50 ± 5.11 | 71.45 ± 5.67 | **71.89 ± 5.53** | 71.10 ± 4.77 |
| **GA** | 66.40 ± 4.31 | 69.67 ± 5.03 | 68.85 ± 4.84 | 71.88 ± 4.77 | **72.38 ± 5.59** | 71.34 ± 4.14 |
| **PCA** | 74.40 ± 5.09 | 75.96 ± 4.12 | 76.46 ± 4.31 | 77.80 ± 4.93 | 78.66 ± 4.17 | **79.80 ± 5.55** |
| **GA + PCA** | 73.83 ± 5.29 | 75.72 ± 5.16 | 76.59 ± 4.96 | 77.82 ± 5.69 | 78.96 ± 5.13 | **79.18 ± 5.75** |
| **LDA** | 39.72 ± 6.06 | 66.66 ± 6.47 | 73.76 ± 4.50 | 80.19 ± 4.65 | 83.04 ± 5.03 | **84.54 ± 4.10** |
| **GA + LDA** | 76.41 ± 6.71 | 83.07 ± 3.94 | 85.59 ± 4.76 | 87.62 ± 3.61 | 88.35 ± 3.34 | **90.01 ± 3.64** |
| **F-LDA** | 88.72 ± 3.01 | 90.89 ± 2.15 | 91.82 ± 2.43 | 91.98 ± 2.76 | 92.50 ± 2.09 | **93.98 ± 2.77** |
| **GA + F-LDA** | 85.10 ± 5.28 | 87.85 ± 3.45 | 89.26 ± 3.67 | 89.59 ± 3.34 | 90.93 ± 2.02 | **91.93 ± 2.94** |
| **F-LDA + Spatial** | 92.96 ± 2.80 | 94.18 ± 1.84 | 94.94 ± 2.25 | 95.22 ± 2.39 | 95.81 ± 1.79 | **96.99 ± 1.84** |
| **GA + F-LDA + Spatial** | 91.46 ± 3.86 | 93.65 ± 2.93 | 93.66 ± 2.70 | 94.65 ± 2.17 | 95.17 ± 1.45 | **96.21 ± 1.45** |

Table 6.3 $F_1$ scores using the rice seed datasets with 10 varieties

| Train:Test Ratio | 30:70 | 40:60 | 50:50 | 60:40 | 70:30 | 80:20 |
|---|---|---|---|---|---|---|
| **Spectra** | 66.27 ± 4.58 | 69.28 ± 5.12 | 69.22 ± 5.08 | **71.14 ± 5.73** | 71.09 ± 5.84 | 70.22 ± 5.09 |
| **GA** | 66.00 ± 4.40 | 69.14 ± 5.07 | 68.60 ± 4.86 | **71.60 ± 4.97** | 71.50 ± 5.92 | 70.33 ± 4.43 |
| **PCA** | 74.03 ± 5.24 | 75.51 ± 4.21 | 76.44 ± 4.21 | 77.51 ± 4.98 | 78.01 ± 3.97 | **79.32 ± 5.44** |
| **GA + PCA** | 73.52 ± 5.40 | 75.30 ± 5.19 | 76.52 ± 4.82 | 77.53 ± 5.82 | **78.52 ± 4.81** | 78.51 ± 5.67 |
| **LDA** | 39.18 ± 6.10 | 66.42 ± 6.46 | 73.78 ± 4.39 | 80.15 ± 4.61 | 82.78 ± 4.83 | **84.05 ± 3.92** |
| **GA + LDA** | 76.39 ± 6.63 | 82.97 ± 3.95 | 85.66 ± 4.67 | 87.51 ± 3.64 | 88.25 ± 3.18 | **89.66 ± 3.84** |
| **F-LDA** | 88.66 ± 3.04 | 90.83 ± 2.15 | 91.84 ± 2.38 | 91.91 ± 2.76 | 92.40 ± 1.97 | **93.76 ± 3.05** |
| **GA + F-LDA** | 85.05 ± 5.30 | 87.79 ± 3.45 | 89.32 ± 3.73 | 89.51 ± 3.25 | 90.83 ± 2.03 | **91.66 ± 3.31** |
| **F-LDA + Spatial** | 93.00 ± 2.75 | 94.15 ± 1.77 | 94.97 ± 2.21 | 95.18 ± 2.43 | 95.74 ± 1.79 | **96.91 ± 1.83** |
| **GA + F-LDA + Spatial** | 90.94 ± 3.87 | 93.01 ± 2.91 | 93.06 ± 2.96 | 94.08 ± 2.72 | 94.73 ± 1.77 | **95.90 ± 1.37** |

Table 6.4 Classification accuracies using the rice seed datasets with 20 varieties

| **Train:Test Ratio** | 30:70 | 40:60 | 50:50 | 60:40 | 70:30 | 80:20 |
|---|---|---|---|---|---|---|
| **Spectra** | 54.68 ± 3.63 | 55.98 ± 3.37 | 57.58 ± 4.12 | 58.29 ± 3.30 | 59.63 ± 3.92 | **60.03 ± 3.77** |
| **GA** | 54.54 ± 3.81 | 56.17 ± 3.33 | 57.35 ± 4.33 | 58.00 ± 3.01 | 59.42 ± 3.92 | **59.99 ± 4.61** |
| **PCA** | 61.69 ± 4.12 | 64.06 ± 3.89 | 64.98 ± 3.76 | 65.17 ± 3.23 | 67.01 ± 3.72 | **68.09 ± 4.21** |
| **GA + PCA** | 62.30 ± 3.69 | 63.76 ± 4.26 | 64.93 ± 4.18 | 65.41 ± 3.02 | 66.67 ± 4.44 | **67.53 ± 4.41** |
| **LDA** | 64.92 ± 3.80 | 74.51 ± 2.70 | 77.55 ± 2.51 | 81.86 ± 2.53 | 83.30 ± 2.99 | **84.32 ± 2.19** |
| **GA + LDA** | 73.89 ± 3.47 | 78.83 ± 2.05 | 81.22 ± 3.05 | 83.93 ± 2.26 | 84.42 ± 3.32 | **85.73 ± 1.84** |
| **F-LDA** | 82.15 ± 2.49 | 84.32 ± 1.95 | 85.45 ± 2.24 | 87.16 ± 2.49 | 87.49 ± 2.97 | **88.39 ± 2.46** |
| **GA + F-LDA** | 79.97 ± 4.18 | 82.21 ± 3.05 | 83.44 ± 3.20 | 84.81 ± 2.50 | 85.84 ± 3.50 | **86.65 ± 2.46** |
| **F-LDA + Spatial** | 88.94 ± 2.22 | 90.24 ± 2.05 | 91.10 ± 1.35 | 92.26 ± 2.11 | 92.84 ± 2.14 | **93.02 ± 1.73** |
| **GA + F-LDA + Spatial** | 87.51 ± 2.93 | 88.65 ± 1.74 | 89.84 ± 1.39 | 90.62 ± 2.35 | **91.67 ± 2.38** | 91.64 ± 2.08 |

Table 6.5 $F_1$ scores using the rice seed datasets with 20 varieties

| **Train:Test Ratio** | 30:70 | 40:60 | 50:50 | 60:40 | 70:30 | 80:20 |
|---|---|---|---|---|---|---|
| **Spectra** | 54.04 ± 3.79 | 55.36 ± 3.57 | 56.90 ± 4.27 | 57.70 ± 3.28 | 58.97 ± 3.70 | **59.29 ± 3.94** |
| **GA** | 53.88 ± 3.94 | 55.52 ± 3.52 | 56.70 ± 4.44 | 57.32 ± 3.02 | 58.75 ± 3.73 | **59.16 ± 4.62** |
| **PCA** | 61.04 ± 4.35 | 63.37 ± 4.28 | 64.14 ± 3.96 | 64.56 ± 3.24 | 66.39 ± 3.92 | **67.34 ± 4.73** |
| **GA + PCA** | 61.69 ± 3.86 | 63.07 ± 4.64 | 64.09 ± 4.49 | 64.85 ± 3.10 | 66.02 ± 4.57 | **66.84 ± 4.79** |
| **LDA** | 64.83 ± 3.74 | 74.35 ± 2.74 | 77.38 ± 2.62 | 81.85 ± 2.54 | 83.05 ± 3.02 | **84.21 ± 2.23** |
| **GA + LDA** | 73.75 ± 3.40 | 78.58 ± 2.16 | 80.96 ± 3.22 | 83.86 ± 2.17 | 84.24 ± 3.45 | **85.49 ± 2.03** |
| **F-LDA** | 82.02 ± 2.46 | 84.01 ± 2.07 | 85.14 ± 2.40 | 87.10 ± 2.46 | 87.31 ± 3.03 | **88.25 ± 2.55** |
| **GA + F-LDA** | 79.77 ± 4.23 | 81.92 ± 3.16 | 83.16 ± 3.25 | 84.75 ± 2.49 | 85.59 ± 3.67 | **86.45 ± 2.64** |
| **F-LDA + Spatial** | 88.89 ± 2.22 | 90.12 ± 2.10 | 90.88 ± 1.48 | 92.24 ± 2.00 | 92.75 ± 2.15 | **93.03 ± 1.88** |
| **GA + F-LDA + Spatial** | 87.44 ± 2.93 | 88.61 ± 1.81 | 89.66 ± 1.39 | 90.55 ± 2.35 | 91.55 ± 2.39 | **91.60 ± 2.21** |

Table 6.6 Classification accuracies using the sugar data

| Train:Test Ratio | 30:70 | 40:60 | 50:50 | 60:40 | 70:30 | 80:20 |
|---|---|---|---|---|---|---|
| **Spectra** | 39.47 | 39.26 | 42.98 | 43.11 | 44.67 | **48.00** |
| **GA** | 39.72 | 39.26 | 45.29 | 46.22 | 45.86 | **48.00** |
| **PCA** | 46.83 | 46.96 | 47.78 | 50.22 | 51.48 | **53.78** |
| **GA + PCA** | 53.17 | 55.41 | 54.71 | 54.44 | **55.62** | 55.56 |
| **LDA** | 46.19 | 51.41 | 55.24 | 57.56 | 57.10 | **64.00** |
| **GA + LDA** | 55.20 | 56.74 | 57.37 | 60.00 | **62.43** | 60.89 |
| **F-LDA** | 62.94 | 63.26 | 66.96 | 67.11 | 69.53 | **70.67** |
| **GA + F-LDA** | 60.79 | 59.41 | 61.46 | 63.56 | 63.31 | **64.44** |

Table 6.7 $F_1$ scores using the sugar data

| Train:Test Ratio | 30:70 | 40:60 | 50:50 | 60:40 | 70:30 | 80:20 |
|---|---|---|---|---|---|---|
| **Spectra** | 38.18 | 38.48 | 42.29 | 42.18 | 42.67 | **46.55** |
| **GA** | 38.64 | 38.92 | 44.46 | 45.45 | 44.65 | **47.08** |
| **PCA** | 46.70 | 47.06 | 47.58 | 48.99 | 49.62 | **52.93** |
| **GA + PCA** | 53.00 | **55.56** | 54.10 | 54.05 | 55.23 | 54.15 |
| **LDA** | 46.18 | 51.76 | 55.19 | 57.02 | 56.77 | **61.85** |
| **GA + LDA** | 55.42 | 56.51 | 56.68 | 59.09 | **62.21** | 60.14 |
| **F-LDA** | 62.74 | 63.26 | 66.67 | 66.04 | **68.62** | 68.49 |
| **GA +F-LDA** | 60.38 | 59.15 | 61.33 | 62.78 | **63.07** | 62.97 |

Table 6.8 Computational complexity (content consumption) for the different stages of the F-LDA when applied separately on the full and selected GA features using the rice seed datasets of 10 species ($N_j$ and $d$ represent the number of samples per species and the number of features extracted by F-LDA respectively).

| Dataset (ratio of training to testing sample size used is 80:20) | | Best configuration based on the classifier's accuracy ($G \times B$) | Within-class variance matrix | Between-class variance matrix | Transformation matrix | Eigen problem | Data projection |
|---|---|---|---|---|---|---|---|
| Random Subsets | Computational complexity (formulated for F-LDA in Chapter 5) | - | $o(cN_jG^2B)$ | $o(cG^2B)$ | $o(G^3)$ | $o(G^3)$ | $o(sGd)$ |
| 1st | F-LDA (full features) | 128 * 2 | o(327680$N_j$) | o(327680) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 139 * 1 | o(193210$N_j$) | o(193210) | o(2685619) | o(2685619) | o(139 $sd$) |
| | Saving factors | - | **1.70** | **1.70** | 0.78 | 0.78 | 0.92 |
| 2nd | F-LDA (full features) | 64 * 4 | o(163840$N_j$) | o(163840) | o(262144) | o(262144) | o(64 $sd$) |
| | F-LDA (GA features) | 24 * 4 | o(23040$N_j$) | o(23040) | o(13824) | o(13824) | o(24 $sd$) |
| | Saving factors | - | **7.11** | **7.11** | **18.96** | **18.96** | **2.67** |
| 3rd | F-LDA (full features) | 32 * 8 | o(81920$N_j$) | o(81920) | o(32768) | o(32768) | o(32 $sd$) |
| | F-LDA (GA features) | 57 * 2 | o(64980$N_j$) | o(64980) | o(185193) | o(185193) | o(57 $sd$) |
| | Saving factors | - | **1.26** | **1.26** | 0.18 | 0.18 | 0.56 |
| 4th | F-LDA (full features) | 64 * 4 | o(163840$N_j$) | o(163840) | o(262144) | o(262144) | o(64 $sd$) |
| | F-LDA (GA features) | 59 * 3 | o(104430$N_j$) | o(104430) | o(205379) | o( 205379) | o(59 $sd$) |
| | Saving factors | - | **1.57** | **1.57** | **1.28** | **1.28** | **1.08** |
| 5th | F-LDA (full features) | 64 * 4 | o(163840$N_j$) | o(163840) | o(262144) | o(262144) | o(64 $sd$) |
| | F-LDA (GA features) | 29 * 5 | o(42050$N_j$) | o(42050) | o(42050) | o(42050) | o(29 $sd$) |
| | Saving factors | - | **3.90** | **3.90** | **6.23** | **6.23** | **2.21** |
| 6th | F-LDA (full features) | 64 * 4 | o(163840$N_j$) | o(163840) | o(262144) | o(262144) | o(64 $sd$) |
| | F-LDA (GA features) | 49 * 3 | o(72030$N_j$) | o(72030) | o(117649) | o(117649) | o(49 $sd$) |
| | Saving factors | - | **2.27** | **2.27** | **2.23** | **2.23** | **1.31** |
| 7th | F-LDA (full features) | 64 * 4 | o(163840$N_j$) | o(163840) | o(262144) | o(262144) | o(64 $sd$) |
| | F-LDA (GA features) | 66 * 2 | o(87120$N_j$) | o(87120) | o(287496) | o(287496) | o(66 $sd$) |
| | Saving factors | - | **1.88** | **1.88** | 0.91 | 0.91 | 0.97 |
| 8th | F-LDA (full features) | 128 * 2 | o(327680$N_j$) | o(327680) | o(2097152) | o(2097152) | o(128 $sd$) |
| | F-LDA (GA features) | 110 * 1 | o(121000$N_j$) | o(121000) | o(1331000) | o(1331000) | o(110 $sd$) |
| | Saving factors | - | **2.71** | **2.71** | **1.58** | **1.58** | **1.16** |
| 9th | F-LDA (full features) | 64 * 4 | o(163840$N_j$) | o(163840) | o(262144) | o(262144) | o(64 $sd$) |
| | F-LDA (GA features) | 37 * 3 | o(41070$N_j$) | o(41070) | o(50653) | o(50653) | o(37 $sd$) |
| | Saving factors | - | **3.99** | **3.99** | **5.18** | **5.18** | **1.73** |
| 10th | F-LDA (full features) | 128 * 2 | o(327680$N_j$) | o(327680) | o(2097152) | o(2097152) | o(128 $sd$) |
| | F-LDA (GA features) | 103 * 1 | o(106090$N_j$) | o(106090) | o(1092727) | o(1092727) | o(103 $sd$) |
| | Saving factors | - | **3.09** | **3.09** | **1.92** | **1.92** | **1.24** |

Table 6.9 Memory requirement (content consumption) at the different stages of the F-LDA when applied separately on the full and selected GA features using the rice seed datasets of 10 species ($N_j$ and $d$ represent the number of samples per species and the number of features extracted by F-LDA respectively).

| Dataset (ratio of training to testing sample size used is 80:20) | | Best configuration based on the classifier's accuracy ($G \times B$) | Data matrix size | Within-class variance matrix size | Between-class variance matrix size | Transformation matrix size | Projection matrix size |
|---|---|---|---|---|---|---|---|
| Random Subsets | Memory requirement (formulated for F-LDA in Chapter 5) | - | $G \times B$ | $G \times G$ | $G \times G$ | $G \times G$ | $G \times {}^{d}/_{B}$ |
| 1st | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 139 * 1 | 139 | 19321 | 19321 | 19321 | $139d$ |
| | Saving factors | - | **1.84** | 0.85 | 0.85 | 0.85 | 0.46 |
| 2nd | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | $16d$ |
| | F-LDA (GA features) | 24 * 4 | 96 | 576 | 576 | 576 | $6d$ |
| | Saving factors | - | **2.67** | **7.11** | **7.11** | **7.11** | **2.67** |
| 3rd | F-LDA (full features) | 32 * 8 | 256 | 1024 | 1024 | 1024 | $4d$ |
| | F-LDA (GA features) | 57 * 2 | 114 | 114 | 114 | 114 | $28.50d$ |
| | Saving factors | - | **2.25** | **8.98** | **8.98** | **8.98** | 0.14 |
| 4th | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | $16d$ |
| | F-LDA (GA features) | 59 * 3 | 177 | 3481 | 3481 | 3481 | $19.67d$ |
| | Saving factors | - | **1.45** | **1.18** | **1.18** | **1.18** | 0.81 |
| 5th | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | $16d$ |
| | F-LDA (GA features) | 29 * 5 | 145 | 841 | 841 | 841 | 5.8 |
| | Saving factors | - | **1.77** | **4.87** | **4.87** | **4.87** | **2.76** |
| 6th | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | $16d$ |
| | F-LDA (GA features) | 49 * 3 | 147 | 2401 | 2401 | 2401 | 16.33 |
| | Saving factors | - | **1.74** | **1.71** | **1.71** | **1.71** | 0.98 |
| 7th | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | 16 |
| | F-LDA (GA features) | 66 * 2 | 132 | 4356 | 4356 | 4356 | 4356 |
| | Saving factors | - | **1.94** | 0.94 | 0.94 | 0.94 | 0.00 |
| 8th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | 64 |
| | F-LDA (GA features) | 110 * 1 | 110 | 12100 | 12100 | 12100 | 110 |
| | Saving factors | - | **2.33** | **1.35** | **1.35** | **1.35** | 0.58 |
| 9th | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | $16d$ |
| | F-LDA (GA features) | 37 * 3 | 111 | 1369 | 1369 | 1369 | $12.33d$ |
| | Saving factors | - | **2.31** | **2.99** | **2.99** | **2.99** | **1.30** |
| 10th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 103 * 1 | 103 | 10609 | 10609 | 10609 | $103d$ |
| | Saving factors | - | **2.49** | **1.54** | **1.54** | **1.54** | 0.62 |

Table 6.10 Computational complexity (content consumption) for the different stages of the F-LDA when applied separately on the full and selected GA features using the rice seed datasets of 20 species ($N_j$ and $d$ represent the number of samples per species and the number of features extracted by F-LDA respectively)

| Dataset (ratio of training to testing sample size used is 80:20) | | Best configuration based on the classifier's accuracy ($G \times B$) | Within-class variance matrix | Between-class variance matrix | Transformation matrix | Eigen problem | Data projection |
|---|---|---|---|---|---|---|---|
| Random Subsets | Computational complexity (formulated for F-LDA in Chapter 5) | - | $o(cN_jG^2B)$ | $o(cG^2B)$ | $o(G^3)$ | $o(G^3)$ | $o(sGd)$ |
| 1st | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 31 * 5 | o(96100$N_j$) | o(96100) | o(29791) | o(29791) | o(31$sd$) |
| | Saving factors | - | **6.82** | **6.82** | **70.40** | **70.40** | **4.13** |
| 2nd | F-LDA (full features) | 64 * 4 | o(327680$N_j$) | o(327680) | o(262144) | o(262144) | o(64$sd$) |
| | F-LDA (GA features) | 41 * 3 | o(100860$N_j$) | o(100860) | o(68921) | o(68921) | o(41$sd$) |
| | Saving factors | - | **3.25** | **3.25** | **3.80** | **3.80** | **1.56** |
| 3rd | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 183 * 1 | o(669780$N_j$) | o(669780) | o(6128487) | o(6128487) | o(183$sd$) |
| | Saving factors | - | 0.98 | 0.98 | 0.34 | 0.34 | 0.70 |
| 4th | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 75 * 3 | o(337500$N_j$) | o(337500) | o(421875) | o(421875) | o(75$sd$) |
| | Saving factors | - | **1.94** | **1.94** | **4.97** | **4.97** | **1.71** |
| 5th | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 106 * 2 | o(449440$N_j$) | o(449440) | o(1191016) | o(1191016) | o(106$sd$) |
| | Saving factors | - | **1.46** | **1.46** | **1.76** | **1.76** | **1.21** |
| 6th | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 157 * 1 | o(492980$N_j$) | o(492980) | o(3869893) | o(3869893) | o(157$sd$) |
| | Saving factors | - | **1.33** | **1.33** | 0.54 | 0.54 | 0.82 |
| 7th | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 137 * 1 | o(375380$N_j$) | o(375380) | o(2571353) | o(2571353) | o(137$sd$) |
| | Saving factors | - | **1.75** | **1.75** | 0.82 | 0.82 | 0.93 |
| 8th | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 94 * 1 | o(176720$N_j$) | o(176720) | o(830584) | o(830584) | o(94$sd$) |
| | Saving factors | - | **3.71** | **3.71** | **2.52** | **2.52** | **1.36** |
| 9th | F-LDA (full features) | 128 * 2 | o(655360$N_j$) | o(655360) | o(2097152) | o(2097152) | o(128$sd$) |
| | F-LDA (GA features) | 70 * 2 | o(196000$N_j$) | o(196000) | o(343000) | o(343000) | o(70$sd$) |
| | Saving factors | - | **3.34** | **3.34** | **6.11** | **6.11** | **1.83** |
| 10th | F-LDA (full features) | 64 * 4 | o(327680$N_j$) | o(327680) | o(262144) | o(262144) | o(64$sd$) |
| | F-LDA (GA features) | 128 * 1 | o(327680$N_j$) | o(327680) | o(2097152) | o(2097152) | o(128$sd$) |
| | Saving factors | - | **1.00** | **1.00** | 0.13 | 0.13 | 0.50 |

Table 6.11 Memory requirement (content consumption) at the different stages of the F-LDA when applied separately on the full and selected GA features using the rice seed datasets of 20 species ($N_j$ and $d$ represent the number of samples per species and the number of features extracted by F-LDA respectively).

| Dataset (ratio of training to testing sample size used is 80:20) | | Best configuration based on the classifier's accuracy ($G \times B$) | Data matrix size | Within-class variance matrix size | Between-class variance matrix size | Transformation matrix size | Projection matrix size |
|---|---|---|---|---|---|---|---|
| Random Subsets | Memory requirement (formulated for F-LDA in Chapter 5) | - | $G \times B$ | $G \times G$ | $G \times G$ | $G \times G$ | $G \times {}^{d}/_{B}$ |
| 1st | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 31 * 5 | 155 | 155 | 155 | 155 | $6.20d$ |
| | Saving factors | - | **1.65** | **105.70** | **105.70** | **105.70** | **10.32** |
| 2nd | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | $16d$ |
| | F-LDA (GA features) | 41 * 3 | 123 | 1681 | 1681 | 1681 | $13.67d$ |
| | Saving factors | - | **2.08** | **2.44** | **2.44** | **2.44** | **1.17** |
| 3rd | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 183 * 1 | 183 | 33489 | 33489 | 33489 | $183d$ |
| | Saving factors | - | **1.40** | 0.49 | 0.49 | 0.49 | 0.35 |
| 4th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 75 * 3 | 225 | 5625 | 5625 | 5625 | $25d$ |
| | Saving factors | - | **1.14** | **2.91** | **2.91** | **2.91** | **2.56** |
| 5th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 106 * 2 | 212 | 11236 | 11236 | 11236 | $53d$ |
| | Saving factors | - | **1.21** | **1.46** | **1.46** | **1.46** | **1.21** |
| 6th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 157 * 1 | 157 | 24649 | 24649 | 24649 | $157d$ |
| | Saving factors | - | **1.63** | 0.66 | 0.66 | 0.66 | 0.41 |
| 7th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 137 * 1 | 137 | 18769 | 18769 | 18769 | $137d$ |
| | Saving factors | - | **1.87** | 0.87 | 0.87 | 0.87 | 0.47 |
| 8th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 94 * 1 | 94 | 8836 | 8836 | 8836 | $94d$ |
| | Saving factors | - | **2.72** | **1.85** | **1.85** | **1.85** | 0.68 |
| 9th | F-LDA (full features) | 128 * 2 | 256 | 16384 | 16384 | 16384 | $64d$ |
| | F-LDA (GA features) | 70 * 2 | 140 | 4900 | 4900 | 4900 | $35d$ |
| | Saving factors | - | **1.83** | **3.34** | **3.34** | **3.34** | **1.83** |
| 10th | F-LDA (full features) | 64 * 4 | 256 | 4096 | 4096 | 4096 | $16d$ |
| | F-LDA (GA features) | 128 * 1 | 128 | 16384 | 16384 | 16384 | $128d$ |
| | Saving factors | - | **2.00** | 0.25 | 0.25 | 0.25 | 0.13 |

Table 6.12 Computational complexity (content consumption) for the different stages of the F-LDA when applied separately on the full and selected GA features using the sugar dataset ($N_j$ and $d$ represent the number of samples per species and the number of features extracted by F-LDA respectively).

| Dataset (ratio of training to testing sample size used is 80:20) | | Best configuration based on the classifier's accuracy ($G \times B$) | Within-class variance matrix | Between-class variance matrix | Transformation matrix | Eigen problem | Data projection |
|---|---|---|---|---|---|---|---|
| | Computational complexity (formulated for F-LDA in Chapter 5) | - | $o(cN_jG^2B)$ | $o(cG^2B)$ | $o(G^3)$ | $o(G^3)$ | $o(sGd)$ |
| Sugar data | F-LDA (using the full features) | 32×5 | $o(46080N_j)$ | $o(46080)$ | $o(32768)$ | $o(32768)$ | $o(32sd)$ |
| | F-LDA (using the optimal features) | 16×4 | $o(9216N_j)$ | $o(9216)$ | $o(4096)$ | $o(4096)$ | $o(16sd)$ |
| | Saving factors | - | **5** | **5** | **8** | **8** | **2** |

Table 6.13 Memory requirement (content consumption) at the different stages of the F-LDA when applied separately on the full and selected GA features using the sugar dataset ($N_j$ and $d$ represent the number of samples per species and the number of features extracted by F-LDA respectively).

| Dataset (ratio of training to testing sample size used is 80:20) | | Best configuration based on the classifier's accuracy ($G \times B$) | Data matrix size | Within-class variance matrix size | Between-class variance matrix size | Transformation matrix size | Projection matrix size |
|---|---|---|---|---|---|---|---|
| | Memory requirement (formulated for F-LDA in Chapter 5) | - | $G \times B$ | $G \times G$ | $G \times G$ | $G \times G$ | $G \times {}^{d}/_{B}$ |
| Sugar data | F-LDA (using the full features) | 32×5 | 160 | 1024 | 1024 | 1024 | $6.40d$ |
| | F-LDA (using the optimal features) | 16×4 | 64 | 256 | 256 | 256 | $4d$ |
| | Saving factors | - | **2.50** | **4** | **4** | **4** | **1.60** |

Table 6.14 Feature extraction time (s) of F-LDA when applied separately on the full and selected GA features (the ratio of training to testing sample size used is 80:20)

| F-LDA Approach | Rice seed datasets (10 species) | Rice seed datasets (20 species) | Sugar dataset |
|---|---|---|---|
| F-LDA (using the full features) | 0.0739 ± 0.0314 | 0.2055 ± 0.0393 | 0.0510 |
| F-LDA (using the GA features) | **0.0581 ± 0.0319** | **0.1798 ± 0.0844** | **0.0260** |

### 6.2.4. Comparing performance of the proposed approach on the selected data subsets of 6 species which were utilised in Section 4.8.3 of Chapter 4 with those reported for the said data subsets of 6 species using the approach proposed in Chapter 4.

In this section, performance of F-LDA and GA+F-LDA when applied on the data subsets of 6 species which were utilised in Section 4.8.3 of Chapter 4 are compared

with those reported for the same data subsets of 6 species in Chapter 4 (classifiers' performance on the data subsets of 6 species using the approach presented in Chapter 4 are shown in Table 4.5 of that same chapter, i.e. Chapter 4). A summary of all the subsets utilised in this section can be found in Table 4.4 in Chapter 4. As in Chapter 4 and for fair comparison, the ratio of training to testing samples used in this section is 80:20.

Firstly, F-LDA and GA + F-LDA are separately applied on the spectral data of each subset to extract features which are combined with corresponding spatial features. The RF classifier is trained separately using the combined spectral and spatial features obtained from the 6 subsets and the classification results obtained are presented in Table 6.15 and Table 6.16. From Table 6.15 and Table 6.16, it can be observed that the classification results were significantly improved for F-LDA and GA+F-LDA when compared to the classification results on the data subsets of 6 species which are presented in Table 4.5 of Chapter 4. These experimental results further demonstrate the potential of F-LDA and GA+F-LDA in feature extraction and dimensionality reduction of hyperspectral data.

Table 6.15 $F_1$ scores using F-LDA and GA+F-LDA on the selected data subsets of 6 species utilised in Section 4.8.3 of Chapter 4.

| Subsets | F-LDA | BEST CONFIG | GA + F-LDA | BEST CONFIG |
|---|---|---|---|---|
| 1 | 100.00 | 64 × 4 | 100.00 | 70 × 2 |
| 2 | 100.00 | 16 × 16 | 100.00 | 76 × 2 |
| 3 | 100.00 | 32 × 8 | 99.33 | 23 × 6 |
| 4 | 99.12 | 128 × 2 | 99.07 | 74 × 2 |
| 5 | 98.44 | 128 × 2 | 96.51 | 49 × 3 |
| 6 | 69.62 | 128 × 2 | 61.94 | 23 × 5 |

Table 6.16 Classification accuracy using F-LDA and GA+F-LDA on the selected data subsets of 6 species utilised in Section 4.8.3 of Chapter 4.

| Subsets | F-LDA | BEST CONFIG | GA + F-LDA | BEST CONFIG |
|---|---|---|---|---|
| 1 | 100.00 | 64 × 4 | 100.00 | 70 × 2 |
| 2 | 100.00 | 16 × 16 | 100.00 | 38 × 4 |
| 3 | 100.00 | 32 × 8 | 99.26 | 23 × 6 |
| 4 | 99.17 | 128 × 2 | 99.17 | 74 × 2 |
| 5 | 98.26 | 128 × 2 | 96.52 | 21 × 7 |
| 6 | 70.49 | 128 × 2 | 62.30 | 23 × 5 |

## 6.3. Summary

In this chapter, a novel hybridized GA and F-LDA (GA+F-LDA) has been introduced and its effectiveness evaluated for dimensionality reduction of hyperspectral imaging data. Rice seed spectral datasets are used for performance evaluation of the proposed approach. The experimental results obtained by the hybridized dimensionality reduction scheme (GA+F-LDA) are promising and demonstrate the potential of applying F-LDA on hyperspectral datasets reduced by GA (reduction in computational complexity and memory requirement can be achieved). The application of the proposed approach can also be extended to the classification and quality evaluation of other Agri-Tech products and similar hyperspectral data related classification tasks in other areas too.

# 7. Conclusion and Future Work

Challenges which limit the potential of hyperspectral imaging data in classification applications have been identified in this thesis. One of these is the problem of curse of dimensionality (Hughes phenomenon) which arises due to the presence of very high number of features (usually in hundreds) in the hyperspectral data. Another challenge which was identified is the reduced fidelity in the appearance-based features from hyperspectral images when compared with those extracted from RGB images.

This thesis therefore focuses on proposing novel solutions to address the identified challenges. Specifically, new dimensionality reduction approaches were proposed for hyperspectral imaging data. Also, the effectiveness of a new way of combining spatial and spectral features was evaluated for hyperspectral data classification. The following subsections present a summary of the contributions of this thesis, highlight their limitations and provide future research directions.

## 7.1. Hyperspectral Imaging Data Classification: Evaluating the Effectiveness of Combining Spectral Features from Hyperspectral Images and Spatial Features from RGB Images

Chapter 4 evaluates the effectiveness of combining spatial features from RGB images (which offer high spatial resolution) and spectral features from hyperspectral images (which offer high spectral resolution) for hyperspectral data classification. Chapter 4 applied LDA as an alternative dimensionality reduction approach to the commonly applied PCA for dimensionality reduction of spectral data. The performances of LDA and PCA for dimensionality reduction of the hyperspectral imaging data were presented and compared. A dataset containing spatial and spectral features (which

were extracted from the acquired RGB images and hyperspectral image data cubes of rice seeds respectively) and a large number of species (90 rice seed varieties with 96 seeds per variety) was used to evaluate performance of the proposed approach.

Experimental results showed that LDA can perform better than PCA when applied to reduce dimensionality of spectral data. The results obtained also demonstrate the ability of the proposed approach (combining spectral features, extracted from hyperspectral images, and spatial features, extracted from high resolution RGB) to achieve good classification results. The large hyperspectral data of rice seeds was made publicly available [34],[35] to the community to assist in benchmarking of the proposed approach.

### 7.2. F-LDA for Feature Extraction of Hyperspectral Imaging Data

Based on the superior performance of LDA to PCA as a dimensionality reduction technique for hyperspectral imaging data in Chapter 4 and the increased accuracy and reduced computational complexity achieved in a previous work that extended PCA [37], Chapter 5 proposed a new Folded-LDA (F-LDA). The proposed F-LDA is an extended and improved version of the LDA transform, for feature extraction of hyperspectral imaging data. The proposed F-LDA is based on a mathematical 'trick' (folding the pixels) which was inspired by the paper in [37].

Performance evaluation of the proposed technique was carried out using five publicly available hyperspectral datasets which were acquired using different sensors (AVIRIS, ROSIS, Hyperion). The proposed F-LDA produced more informative features and achieved higher classification accuracy than the original feature space, conventional LDA, 2D LDA [38], and other state-of-the-art methods namely GDA [39], NWFE

[40], KPCA [41] and F-PCA [37]. Experimental results showed that the proposed technique is superior to the other approaches when applied in Small Sample Size (SSS) scenarios. The proposed F-LDA also achieved reduced contiguous memory requirement and reduced complexity when compared with the conventional LDA. Though the proposed F-LDA and 2D LDA [38] share some concepts, the proposed F-LDA improved the classification performance of 2D LDA, according to the experimental results.

### 7.3. Hybridizing GA and F-LDA for Dimensionality Reduction of Hyperspectral Data

Chapter 6 explored the effectiveness of hybridizing GA and F-LDA (GA+F-LDA) for dimensionality reduction of hyperspectral imaging data. This is based on:

A. The superior performance of LDA to PCA as a dimensionality reduction technique when applied on hyperspectral imaging data in Chapter 4.

B. The improved classification performance achieved by F-LDA in Chapter 5 when applied on hyperspectral imaging data.

C. The improved classification performance achieved when GA was used for the selection of optimal feature subset from the original feature set prior to feature extraction using PCA [28],[29] or LDA [30],[31] in other work.

In the proposed approach, GA was applied on the datasets to select optimal spectral features. This was followed by the application of F-LDA on the data with the selected optimal features. Performance of the proposed approach (GA + F-LDA) was evaluated on two spectral datasets of 10 and 20 rice seed species and another spectral dataset of sugar containing 9 species. Experimental results obtained show that by applying F-

LDA on the selected optimal spectral features, its computational complexity and memory requirement can be reduced further. Though the proposed GA+F-LDA has some benefits (reduction in computational complexity and memory requirement), it is slightly outperformed by the standard F-LDA in terms of classification accuracy and $F_1$ score.

### 7.4. Future Work

This thesis focused on proposing novel approaches to address the problem of high data dimensionality and reduced fidelity in appearance-based features which limit the potential of hyperspectral imaging data in classification applications. While the various approaches presented in this thesis performed very well, they have some limitations which can translate into gaps and opportunities for future research investigations. Future work will be focused on addressing these limitations which are summarized below:

1) The suboptimal classification results reported for some rice seed species in Chapter 4 were linked to the use of samples with similarities among the species in the data. It is therefore necessary to assess the similarity among species of different rice seed species and explore ways in which the degrading effects on classifiers' performance due to seed similarity can be mitigated.
2) Also, the proposed approach in Chapter 4 was applied on a large set of 90 rice seeds which were provided by the National Center of Protection of New Varieties and Goods of Plants (NCPNVGGP) in Vietnam. This approach can be extended to datasets with larger number of species which are available at other agricultural institutes.

3) In the F-LDA proposed in Chapter 5, each pixel in the hyperspectral data was folded from vector to matrix. Different configurations ($G \times B$) of the matrices were exploited, and the optimal configuration was chosen as one that gave the best classification results. New techniques can be developed to automatically determine the F-LDA configuration that gives the best classification results.
4) In Chapter 6, hybridized GA and F-LDA were introduced and their effectiveness evaluated for dimensionality reduction of hyperspectral imaging using rice seed datasets. As demonstrated with the sugar data, the application of the proposed approaches can also be extended to the classification and quality evaluation of other Agri-tech products, and similar HSI related classification tasks in other areas too.

## Publications by the Author

Listed below are published papers which have resulted from the work presented in this thesis.

1) Journal papers:
    - S. D. Fabiyi, H. Vu, C. Tachtatzis, P. Murray, D. Harle, T. K. Dao, I. Andonovic, J. Ren and S. Marshall, "Varietal Classification of Rice Seeds Using RGB and Hyperspectral Images," IEEE Access, vol. 8, pp. 22493–22505, 2020, doi: 10.1109/ACCESS.2020.2969847.
    - S. D. Fabiyi, P. Murray, J. Zabalza and J. Ren, "Folded LDA: Extending LDA Algorithm for Feature Extraction and Data Reduction in Hyperspectral Remote Sensing," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 12312-12331, 2021, doi: 10.1109/JSTARS.2021.3129818.
2) Conference papers:

    S. D. Fabiyi, H. Vu, C. Tachtatzis, P. Murray, D. Harle, T. K. Dao, I. Andonovic, J. Ren and S. Marshall, "Comparative Study of PCA and LDA for Rice Seeds Quality Inspection," in 2019 IEEE Africon, 2019, pp. 1–4 , doi: 10.1109/AFRICON46755.2019.9134059.

## References

[1] H. Fu, G. Sun, J. Zabalza, A. Zhang, J. Ren, and X. Jia, "A Novel Spectral-Spatial Singular Spectrum Analysis Technique for near Real-Time in Situ Feature Extraction in Hyperspectral Imaging," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 13, pp. 2214–2225, 2020.

[2] P. Ghamisi et al., "Advances in Hyperspectral Image and Signal Processing: A Comprehensive Overview of the State of the Art," IEEE Geoscience and Remote Sensing Magazine, vol. 5, no. 4. Institute of Electrical and Electronics Engineers Inc., pp. 37–78, 2017.

[3] Q. Yang, Y. Xu, Z. Wu, and Z. Wei, "Hyperspectral and Multispectral Image Fusion Based on Deep Attention Network," in Workshop on Hyperspectral Image and Signal Processing, Evolution in Remote Sensing, 2019, pp. 1-5, doi: 10.1109/WHISPERS.2019.8920825.

[4] X. Zhang, Y. Sun, K. Shang, L. Zhang, and S. Wang, "Crop Classification Based on Feature Band Set Construction and Object-Oriented Approach Using Hyperspectral Images," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 9, no. 9, pp. 4117–4128, Sept. 2016, doi: 10.1109/JSTARS.2016.2577339.

[5] K. Karalas, G. Tsagkatakis, M. Zervakis, and P. Tsakalides, "Land Classification Using Remotely Sensed Data: Going Multilabel," IEEE Trans. Geosci. Remote Sens., vol. 54, no. 6, pp. 3548–3563, Jun. 2016.

[6] A. Ferreira et al., "Eyes in the skies: A data-driven fusion approach to identifying drug crops from remote sensing images," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 12, no. 12, pp. 4773–4786, Dec. 2019.

[7] Z. Qiu, J. Chen, Y. Zhao, S. Zhu, Y. He, and C. Zhang, “Variety Identification of Single Rice Seed Using Hyperspectral Imaging Combined with Convolutional Neural Network,” Appl. Sci., vol. 8, no. 2, p. 212, Jan. 2018.

[8] W. Kong, C. Zhang, F. Liu, P. Nie, and Y. He, “Rice seed cultivar identification using near-infrared hyperspectral imaging and multivariate data analysis,” Sensors (Basel)., vol. 13, no. 7, pp. 8916–8927, 2013.

[9] H. Vu et al., “Spatial and spectral features utilization on a Hyperspectral imaging system for rice seed varietal purity inspection,” in 2016 IEEE RIVF International Conference on Computing and Communication Technologies: Research, Innovation, and Vision for the Future, RIVF 2016 - Proceedings, 2016, pp. 169–174.

[10] J. Sun, X. Lu, H. Mao, X. Jin, and X. Wu, “A Method for Rapid Identification of Rice Origin by Hyperspectral Imaging Technology,” J. Food Process Eng., vol. 40, no. 1, Feb. 2017.

[11] X. He et al., “Rapid and Nondestructive Measurement of Rice Seed Vitality of Different Years Using Near-Infrared Hyperspectral Imaging,” Molecules, vol. 24, no. 12, p. 2227, Jun. 2019.

[12] T. Qiao, J. Ren, C. Craigie, J. Zabalza, C. Maltin, and S. Marshall, “Quantitative Prediction of Beef Quality Using Visible and NIR Spectroscopy with Large Data Samples Under Industry Conditions,” J. Appl. Spectrosc., vol. 82, no. 1, pp. 137–144, Mar. 2015.

[13] M. Kamruzzaman, G. Elmasry, D. W. Sun, and P. Allen, “Application of NIR hyperspectral imaging for discrimination of lamb muscles,” J. Food Eng., vol. 104, no. 3, pp. 332–340, Jun. 2011.

[14] G. ElMasry, N. Wang, A. ElSayed, and M. Ngadi, “Hyperspectral imaging for nondestructive determination of some quality attributes for strawberry,” J. Food Eng., vol. 81, no. 1, pp. 98–107, Jul. 2007.

[15] L. Wang, D. Liu, H. Pu, D. W. Sun, W. Gao, and Z. Xiong, “Use of Hyperspectral Imaging to Discriminate the Variety and Quality of Rice,” Food Anal. Methods, vol. 8, no. 2, pp. 515–523, 2015.

[16] Shwetank, K. Jain, and K. Bhatia, “Hyperspectral Data Compression Model Using SPCA (Segmented Principal Component Analysis) and Classification of Rice Crop Varieties,” Springer, Berlin, Heidelberg, 2010, pp. 360–372.

[17] M. Pal and G. M. Foody, “Feature selection for classification of hyperspectral data by SVM,” IEEE Trans. Geosci. Remote Sens., vol. 48, no. 5, pp. 2297–2307, May 2010.

[18] M. Kamandar and H. Ghassemian, “Linear feature extraction for hyperspectral images based on information theoretic learning,” IEEE Geosci. Remote Sens. Lett., vol. 10, no. 4, pp. 702–706, 2013.

[19] S. Paulus and A. K. Mahlein, “Technical workflows for hyperspectral plant image assessment and processing on the greenhouse and laboratory scale,” Gigascience, vol. 9, no. 8, pp. 1–10, Aug. 2020.

[20] Z. W. Pan, H. L. Shen, C. Li, S. J. Chen, and J. H. Xin, “Fast Multispectral Imaging by Spatial Pixel-Binning and Spectral Unmixing,” IEEE Trans. Image Process., vol. 25, no. 8, pp. 3612–3625, Aug. 2016.

[21] T. Kelman, J. Ren, and S. Marshall, “Effective classification of Chinese tea samples in hyperspectral imaging,” Artif. Intell. Res., vol. 2, no. 4, p. 87, Oct. 2013.

[22] K. Kiratiratanapruk et al., “Development of Paddy Rice Seed Classification Process using Machine Learning Techniques for Automatic Grading Machine,” J. Sensors, vol. 2020, pp. 1–14, Jul. 2020.

[23] C. Liu, W. Liu, X. Lu, W. Chen, J. Yang, and L. Zheng, “Nondestructive determination of transgenic Bacillus thuringiensis rice seeds (Oryza sativa L.) using multispectral imaging and chemometric methods,” Food Chem., vol. 153, pp. 87–93, Jun. 2014.

[24] X. Yang, Y. Ye, X. Li, R. Y. K. Lau, X. Zhang, and X. Huang, “Hyperspectral image classification with deep learning models,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 9, pp. 5408–5423, Sept. 2018.

[25] J. Xia, P. Du, X. He, and J. Chanussot, “Hyperspectral remote sensing image classification based on rotation forest,” IEEE Geosci. Remote Sens. Lett., vol. 11, no. 1, pp. 239–243, 2014.

[26] W. Liao, A. Pižurica, P. Scheunders, W. Philips, and Y. Pi, “Semisupervised local discriminant analysis for feature extraction in hyperspectral images,” IEEE Trans. Geosci. Remote Sens., vol. 51, no. 1, pp. 184–198, 2013.

[27] L. He, H. Yang, and L. Zhao, “Tensor subspace learning and classification: Tensor local discriminant embedding for hyperspectral image,” in Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019, 2019, pp. 589–598.

[28] L. Ghoualmi, A. Draa, and S. Chikhi, “An efficient feature selection scheme based on genetic algorithm for ear biometrics authentication,” in 2015 12th International Symposium on Programming and Systems (ISPS), 2015, pp. 1–5.

[29] A. M. Abo El-Maaty and A. G. Wassal, “Hybrid GA-PCA Feature Selection Approach for Inertial Human Activity Recognition,” in 2018 IEEE Symposium Series on Computational Intelligence (SSCI), 2018, pp. 1027–1032.

[30] K. Belattar, S. Mostefai, and A. Draa, “A Hybrid GA-LDA Scheme for Feature Selection in Content-Based Image Retrieval,” Int. J. Appl. Metaheuristic Comput., vol. 9, no. 2, pp. 48–71, Feb. 2018.

[31] M. Cui, S. Prasad, M. Mahrooghy, L. M. Bruce, and J. Aanstoos, “Genetic algorithms and linear discriminant analysis based dimensionality reduction for remotely sensed image analysis,” in International Geoscience and Remote Sensing Symposium (IGARSS), 2011, pp. 2373–2376.

[32] T. Y. Kuo, C. L. Chung, S. Y. Chen, H. A. Lin, and Y. F. Kuo, “Identifying rice grains using image analysis and sparse-representation-based classification,” Comput. Electron. Agric., vol. 127, pp. 716–725, Sept. 2016.

[33] C. N. M. Peralta, J. P. Pabico, and V. Y. Mariano, “Modeling shapes using uniform cubic B-splines for rice seed image analysis,” in 2016 IEEE 6th International Conference on Communications and Electronics, IEEE ICCE 2016, 2016, pp. 326–331.

[34] S. D. Fabiyi et al., “Varietal Classification of Rice Seeds Using RGB and Hyperspectral Images,” IEEE Access, vol. 8, pp. 22493–22505, 2020.

[35] H. Vu et al., “RGB and VIS/NIR Hyperspectral Imaging Data for 90 Rice Seed Varieties,” Dec. 2019.

[36] M. Calzolari, “manuel-calzolari/sklearn-genetic: sklearn-genetic 0.2 | Zenodo.” 2019.

[37] J. Zabalza et al., “Novel Folded-PCA for improved feature extraction and data reduction with hyperspectral imaging and SAR in remote sensing,” ISPRS J. Photogramm. Remote Sens., vol. 93, pp. 112–122, 2014.

[38] M. Imani and H. Ghassemian, “Two dimensional linear discriminant analyses for hyperspectral data,” Photogramm. Eng. Remote Sensing, vol. 81, no. 10, pp. 777–786, Oct. 2015.

[39] G. Yang, X. Yu, and X. Zhou, “Hyperspectral Image Feature Extraction Based On Generalized Discriminant Analysis,” in XXIst ISPRS Congress, 2008, pp. 285–290.

[40] B. C. Kuo and D. A. Landgrebe, “Nonparametric weighted feature extraction for classification,” IEEE Trans. Geosci. Remote Sens., vol. 42, no. 5, pp. 1096–1105, May 2004.

[41] M. Fauvel, J. Chanussot, and on Atli Benediktsson, “Kernel Principal Component Analysis for the Classification of Hyperspectral Remote Sensing Data over Urban Areas,” EURASIP J. Adv. Signal Process., vol. 783194, p. 14, 2009.

[42] D. B. Gillis and J. H. Bowles, “An introduction to hyperspectral image data modeling,” in Applied and Numerical Harmonic Analysis, no. 9780817683757, Springer International Publishing, 2013, pp. 173–194.

[43] D. W. Sun, Hyperspectral Imaging for Food Quality Analysis and Control. Elsevier Inc., 2010.

[44] Y. Yang, Y. Xie, X. Chen, and Y. Sun, “Hyperspectral Snapshot Compressive Imaging with Non-Local Spatial-Spectral Residual Network,” Remote Sens. 2021, Vol. 13, Page 1812, vol. 13, no. 9, p. 1812, May 2021.

[45] Q. Li, X. He, Y. Wang, H. Liu, D. Xu, and F. Guo, “Review of spectral imaging technology in biomedical engineering: achievements and challenges,” https://doi.org/10.1117/1.JBO.18.10.100901, vol. 18, no. 10, p. 100901, Oct. 2013.

[46] M. S. Shaikh, K. Jaferzadeh, B. Thörnberg, and J. Casselgren, “Calibration of a Hyper-Spectral Imaging System Using a Low-Cost Reference,” Sensors 2021, Vol. 21, Page 3738, vol. 21, no. 11, p. 3738, May 2021.

[47] N. Gupta, “Development of staring hyperspectral imagers,” in Proceedings - Applied Imagery Pattern Recognition Workshop, 2011, pp. 1–8, doi: 10.1109/AIPR.2011.6176379.

[48] G. Bonifazi, G. Capobianco, R. Gasbarrone, and S. Serranti, “Hazelnuts classification by hyperspectral imaging coupled with variable selection methods,” SPIE, vol. 11754, p. 117540Q, Apr. 2021.

[49] H. Yao and D. Lewis, “Spectral Preprocessing and Calibration Techniques,” Hyperspectral Imaging Food Qual. Anal. Control, pp. 45–78, Jan. 2010.

[50] “Hyperspectral Remote Sensing Scenes - Grupo de Inteligencia Computacional (GIC).” [Online]. Available:

http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes#Pavia_Centre_scene. [Accessed: 27-Nov-2020].

[51] D. Manea and M. A. Calin, “Hyperspectral imaging in different light conditions,” http://dx.doi.org/10.1179/1743131X15Y.0000000001, vol. 63, no. 4, pp. 214–219, 2015.

[52] S. W. Kim and R. P. W. Duin, “On using a pre-clustering technique to optimize LDA-based classifiers for appearance-based face recognition,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2007, vol. 4756 LNCS, pp. 466–476.

[53] X. Song, S. Jiang, S. Wang, J. Tang, and Q. Huang, “Cross concept local fisher discriminant analysis for image classification,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2013, vol. 7733 LNCS, no. PART 2, pp. 407–416.

[54] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, “Linear discriminant analysis: A detailed tutorial,” vol. 30, no. 2, pp. 169–190, 2017.

[55] A. Y. Nikravesh, S. A. Ajila, and C.-H. Lung, “Using genetic algorithms to find optimal solution in a search space for a cloud predictive cost-driven decision maker,” J. Cloud Comput. 2018 71, vol. 7, no. 1, pp. 1–21, Nov. 2018.

[56] K. Drachal and M. Pawłowski, “A Review of the Applications of Genetic Algorithms to Forecasting Prices of Commodities,” Econ. 2021, Vol. 9, Page 6, vol. 9, no. 1, p. 6, Jan. 2021.

[57] Harwati and A. Sudiya, “Application of Decision Tree Approach to Student Selection Model- A Case Study,” in IOP Conference Series: Materials Science and Engineering, 2016, vol. 105, no. 1.

[58] I. Jenhani, N. Ben Amor, and Z. Elouedi, “Decision trees as possibilistic classifiers,” Int. J. Approx. Reason., vol. 48, no. 3, pp. 784–807, Aug. 2008.

[59] P. T. T. Hong, T. T. T. Hai, L. T. Lan, V. T. Hoang, V. Hai, and T. T. Nguyen, “Comparative Study on Vision Based Rice Seed Varieties Identification,” in Proceedings - 2015 IEEE International Conference on Knowledge and Systems Engineering, KSE 2015, 2016, pp. 377–382.

[60] B. Tu, C. Zhou, D. He, S. Huang, and A. Plaza, “Hyperspectral Classification with Noisy Label Detection via Superpixel-to-Pixel Weighting Distance,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 6, pp. 4116–4131, Jun. 2020.

[61] C. Mu, J. Liu, Y. Liu, and Y. Liu, “Hyperspectral Image Classification Based on Active Learning and Spectral-Spatial Feature Fusion Using Spatial Coordinates,” IEEE Access, vol. 8, pp. 6768–6781, 2020.

[62] R. Delgado and X.-A. Tibau, “Why Cohen’s Kappa should be avoided as performance measure in classification,” PLoS One, vol. 14, no. 9, Sept. 2019.

[63] X. Wei, W. Zhu, B. Liao, and L. Cai, “Matrix-based margin-maximization band selection with data-driven diversity for hyperspectral image classification,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 12, pp. 7294–7309, Dec. 2018.

[64] S. D. Fabiyi et al., “Comparative Study of PCA and LDA for Rice Seeds Quality Inspection,” in 2019 IEEE AFRICON, 2019, pp. 1–4.

[65] A. Zhang et al., “Hyperspectral band selection using crossover-based gravitational search algorithm,” IET Image Process., vol. 13, no. 2, pp. 280–286, Feb. 2019.

[66] J. Zabalza et al., “Novel two-dimensional singular spectrum analysis for effective feature extraction and data classification in hyperspectral imaging,” IEEE Trans. Geosci. Remote Sens., vol. 53, no. 8, pp. 4418–4433, Aug. 2015.

[67] Y. Zhang, G. Cao, A. Shafique, and P. Fu, “Label Propagation Ensemble for Hyperspectral Image Classification,” IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 12, no. 9, pp. 3623–3636, Sept. 2019.

[68] X. Dai and W. Xue, “Hyperspectral Remote Sensing Image Classification Based on Convolutional Neural Network,” in Chinese Control Conference, CCC, 2018, pp. 10373–10377.

[69] J. Leng, T. Li, G. Bai, Q. Dong, and H. Dong, “Cube-CNN-SVM: A Novel Hyperspectral Image Classification Method,” in 2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), 2016, pp. 1027–1034.

[70] H. Yu, L. Gao, J. Li, S. Li, B. Zhang, and J. Benediktsson, “Spectral-Spatial Hyperspectral Image Classification Using Subspace-Based Support Vector Machines and Adaptive Markov Random Fields,” Remote Sens., vol. 8, no. 4, p. 355, Apr. 2016.

[71] X. Yin, R. Wang, X. Liu, and Y. Cai, “Deep Forest-Based Classification of Hyperspectral Images,” in Chinese Control Conference, CCC, 2018, pp. 10367–10372.

[72] S. M and G. Sadashivappa, “Supervised Hyperspectral Image Classification using SVM and Linear Discriminant Analysis,” Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 10, pp. 403–409, 2020.

[73] B. Tu, J. Wang, G. Zhang, X. Zhang, and W. He, “Texture Pattern Separation for Hyperspectral Image Classification,” IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 12, no. 9, pp. 3602–3614, Sept. 2019.

[74] J. Zabalza, J. Ren, Z. Wang, S. Marshall, and J. Wang, “Singular spectrum analysis for effective feature extraction in hyperspectral imaging,” IEEE Geosci. Remote Sens. Lett., vol. 11, no. 11, pp. 1886–1890, 2014.

[75] J. Zabalza, J. Ren, Z. Wang, H. Zhao, J. Wang, and S. Marshall, “Fast Implementation of Singular Spectrum Analysis for Effective Feature Extraction in Hyperspectral Imaging,” IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens., vol. 8, no. 6, pp. 2845–2853, Jun. 2015.

[76] N. Elhadji, I. Gado, E. Grall-Maës, and M. Kharouf, “Linear Discriminant Analysis based on Fast Approximate SVD,” in Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM, 2017, pp. 359–365.

[77] D. Cai, X. He, and J. Han, “SRDA: An efficient algorithm for large scale discriminant analysis,” IEEE Trans. Knowl. Data Eng., vol. 20, no. 1, Jan. 2008.

[78] C. E. Thomaz, E. C. Kitani, and D. F. Gillies, “A maximum uncertainty LDA-based approach for limited sample size problems - with application to face recognition,” Journal of the Brazilian Computer Society, vol. 12, no. 2. Springer-Verlag London Ltd, 01-Jun-2006.

[79] J. Ye, R. Janardan, and Q. Li, “Two-Dimensional Linear Discriminant Analysis,” in Advances in Neural Information Processing Systems 17 - Proceedings of the 2004 Conference, NIPS 2004 (Advances in Neural Information Processing Systems), 2004.

[80] H. Kong, E. K. Teoh, J. G. Wang, and R. Venkateswarlu, “Two dimensional fisher discriminant analysis: Forget about small sample size problem,” in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 2005, pp. ii/761-ii/764 vol. 2, doi: 10.1109/ICASSP.2005.1415516.

[81] M. Li and B. Yuan, “2D-LDA: A statistical linear discriminant analysis for image matrix,” Pattern Recognit. Lett., vol. 26, no. 5, pp. 527–532, Apr. 2005.

[82] D.-W. Sun, Computer vision technology for food quality evaluation. Elsevier/Academic Press, 2008.

[83] A. Polak, F. K. Coutts, P. Murray, and S. Marshall, “Use of hyperspectral imaging for cake moisture and hardness prediction,” IET Image Process., vol. 13, no. 7, pp. 1152–1160, May 2019.

[84] Y. Ogawa, “Quality Evaluation of Rice,” Comput. Vis. Technol. Food Qual. Eval., pp. 377–400, Jan. 2008.

[85] K. Y. Huang and M. C. Chien, “A novel method of identifying paddy seed varieties,” Sensors (Switzerland), vol. 17, no. 4, pp. 1–8, Apr. 2017.

[86] Z. Liu, F. Cheng, Y. Ying, and X. Rao, “Identification of rice seed varieties using neural network,” J. Zhejiang Univ. Sci., vol. 6B, no. 11, pp. 1095–1100, Oct. 2005.

[87] A. G. OuYang, R. J. Gao, Y. De Liu, X. D. Sun, Y. Y. Pan, and X. L. Dong, “An automatic method for identifying different variety of rice seeds using machine vision technology,” in Proceedings - 2010 6th International Conference on Natural Computation, ICNC 2010, 2010, vol. 1, pp. 84–88.

[88] A. A. Aznan, I. H. Rukunudin, A. Y. M. Shakaff, R. Ruslan, A. Zakaria, and F. S. A. Saad, “The use of machine vision technique to classify cultivated rice seed variety and weedy rice seed variants for the seed industry,” Int. Food Res. J., vol. 23, pp. S31 – S35, 2016.

[89] A. R. Pazoki, F. Farokhi, and Z. Pazoki, “Classification of rice grain varieties using two artificial neural networks (mlp and neuro-fuzzy),” J. Anim. Plant Sci., vol. 24, no. 1, pp. 336–343, 2014.

[90] K. R. Singh and S. Chaudhury, “Efficient technique for rice grain classification using back-propagation neural network and wavelet decomposition,” IET Comput. Vis., vol. 10, no. 8, pp. 780–787, May 2016.

[91] Z. Liu et al., “Comparison of Spectral Indices and Principal Component Analysis for Differentiating Lodged Rice Crop from Normal Ones.” in 5th Computer and Computing Technologies in Agriculture (CCTA), 2011, pp. 84–92.

[92] G. Gilanie, N. Nasir, U. I. Bajwa, and H. Ullah, “RiceNet: convolutional neural networks-based model to classify Pakistani grown rice seed types,” Multimed. Syst., pp. 1–9, Feb. 2021.

[93] D. Joshi et al., “Label-free non-invasive classification of rice seeds using optical coherence tomography assisted with deep neural network,” Opt. Laser Technol., vol. 137, p. 106861, May 2021.

[94] J. Sun, L. Zhang, X. Zhou, K. Yao, Y. Tian, and A. Nirere, “A method of information fusion for identification of rice seed varieties based on hyperspectral imaging technology,” J. Food Process Eng., vol. 44, no. 9, 2021.

[95] Y. Yang, J. Chen, Y. He, F. Liu, X. Feng, and J. Zhang, “Assessment of the vigor of rice seeds by near-infrared hyperspectral imaging combined with transfer learning,” RSC Adv., vol. 10, no. 72, pp. 44149–44158, Nov. 2020.

[96] F. S. Lai, I. Zayas, and Y. Pomeranz, “Application of pattern recognition techniques in the analysis of cereal grains,” Cereal Chem., vol. 63, no. 2, pp. 168–172, 1986.

[97] N. Sakai, S. Yonekawa, A. Matsuzaki, and H. Morishima, “Two-dimensional image analysis of the shape of rice and its application to separating varieties,” J. Food Eng., vol. 27, no. 4, pp. 397–407, Jan. 1996.

[98] Y. Shao, C. Zhao, Y. Bao, and Y. He, “Quantification of Nitrogen Status in Rice by Least Squares Support Vector Machines and Reflectance Spectroscopy,” Food Bioprocess Technol., vol. 5, no. 1, pp. 100–107, Jan. 2012.

[99] U. Knauer, A. Matros, T. Petrovic, T. Zanker, E. S. Scott, and U. Seiffert, “Improved classification accuracy of powdery mildew infection levels of wine grapes by spatial-spectral analysis of hyperspectral images,” Plant Methods 2017 131, vol. 13, no. 1, pp. 1–15, Jun. 2017.

[100] H. Faris et al., “An intelligent system for spam detection and identification of the most relevant features based on evolutionary Random Weight Networks,” Inf. Fusion, vol. 48, pp. 67–83, Aug. 2019.

[101] Y. Korkmaz, A. Boyacı, and T. Tuncer, “Turkish vowel classification based on acoustical and decompositional features optimized by Genetic Algorithm,” Appl. Acoust., vol. 154, pp. 28–35, Nov. 2019.

[102] M. Jansi Rani and D. Devaraj, “Two-Stage Hybrid Gene Selection Using Mutual Information and Genetic Algorithm for Cancer Data Classification,” J. Med. Syst., vol. 43, no. 8, p. 235, Aug. 2019.

[103] CHENG Fang, LIU Zhao-yan, and YING Yi-bin, “Machine Vision Analysis of Characteristics and Image Information Base Construction for Hybrid Rice Seed,” Rice Sci., vol. 12, no. 1, pp. 13–18, 2005.

[104] W. Liu, C. Liu, F. Ma, X. Lu, J. Yang, and L. Zheng, “Online Variety Discrimination of Rice Seeds Using Multispectral Imaging and Chemometric Methods,” J. Appl. Spectrosc., vol. 82, no. 6, pp. 993–999, Jan. 2016.

[105] Y. Kimori, “Morphological image processing for quantitative shape analysis of biomedical structures: effective contrast enhancement,” urn:issn:0909-0495, vol. 20, no. 6, pp. 848–853, Sept. 2013.

[106] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms,” IEEE Trans. Syst. Man. Cybern., vol. 9, no. 1, pp. 62–66, Jan. 1979.

[107] X. Bai, F. Zhou, and Y. Xie, “New class of top-hat transformation to enhance infrared small targets,” J. Electron. Imaging, vol. 17, no. 3, p. 030501, Jul. 2008.

[108] P. Geladi, J. Burger, and T. Lestander, “Hyperspectral imaging: calibration problems and solutions,” vol. 72, no. 2, pp. 209–217, 2004.

[109] W. Li, F. Feng, H. Li, and Q. Du, “Discriminant Analysis-Based Dimension Reduction for Hyperspectral Image Classification: A Survey of the Most Recent Advances and an Experimental Comparison of Different Techniques,” IEEE Geosci. Remote Sens. Mag., vol. 6, no. 1, pp. 15–34, Mar. 2018.

[110] H. R. Shahdoosti and F. Mirzapour, “Spectral-spatial feature extraction using orthogonal linear discriminant analysis for classification of hyperspectral data,” Eur. J. Remote Sens., vol. 50, 2017.

[111] H. Yuan, Y. Lu, L. Yang, H. Luo, and Y. Y. Tang, “Spectral-spatial linear discriminant analysis for hyperspectral image classification,” in 2013 IEEE International Conference on Cybernetics, CYBCONF 2013, 2013, pp. 144–149.

[112] M. Imani and H. Ghassemian, “Feature reduction of hyperspectral images: Discriminant analysis and the first principal component,” J. Artif. Intell. Data Min., vol. 3, no. 1, pp. 1–9, Jan. 2015.

[113] A. Santara et al., “Bass net: Band-adaptive spectral-spatial feature learning neural network for hyperspectral image classification,” IEEE Trans. Geosci. Remote Sens., vol. 55, no. 9, pp. 5293–5301, 2017.

[114] N. Symeonidis, C. Nalmpantis, and D. Vrakas, “A Benchmark Framework to Evaluate Energy Disaggregation Solutions,” Commun. Comput. Inf. Sci., vol. 1000, pp. 19–30, 2019.

[115] F. (Friedrich) Melchert, A. (Andrea) Matros, M. (Michael) Biehl, and U. (Udo) Seiffert, “The sugar dataset - A multimodal hyperspectral dataset for classification and research.” University of Groningen, 2016.

[116] F. Melchert, A. Matros, M. Biehl, and U. Seiffert, “The sugar dataset - A multimodal hyperspectral dataset for classification and research,” in Mittweida Workshop on Computational Intelligence 2016, 2016, pp. 15–18.

[117] J.-Y. Kim and S.-B. Cho, “Exploiting deep convolutional neural networks for a neural-based learning classifier system,” Neurocomputing, vol. 354, pp. 61–70, Aug. 2019.

[118] R. Zarkami, Z. Darizin, R. Sadeghi Pasvisheh, A. Bani, and A. Ghane, “Use of data-driven model to analyse the occurrence patterns of an indicator fish species in river: A case study for Alburnoides eichwaldii (De Filippi, 1863) in Shafaroud River, north of Iran,” Ecol. Eng., vol. 133, pp. 10–19, Aug. 2019.

[119] A. B. Eldin, “Near Infra Red Spectroscopy,” Isin Akyar, Ed. InTech, 2011, pp. 237–248.

[120] M. Calzolari, “manuel-calzolari/sklearn-genetic: sklearn-genetic 0.2 | Zenodo.” 2019.

[121] Eugenio Bertolini (2022). Precision, Specificity, Sensitivity, Accuracy & F1-score (https://www.mathworks.com/matlabcentral/fileexchange/86158-precision-specificity-sensitivity-accuracy-f1-score), MATLAB Central File Exchange. Retrieved February 18, 2022.

[122] S. A. Hosseini and H. Ghassemian, “Rational Function Approximation for Feature Reduction in Hyperspectral Data,” Remote Sens. Lett., vol. 7, no. 2, pp. 101–110, Feb. 2016.

[123] M. Haghighat, S. Zonouz, and M. Abdel-Mottaleb, “CloudID: Trustworthy Cloud-Based and Cross-Enterprise Biometric Identification,” Expert Syst. Appl., vol. 42, no. 21, pp. 7905–7916, Jul. 2015.

[124] S. D. Fabiyi et al., “Folded LDA: Extending LDA Algorithm for Feature Extraction and Data Reduction in Hyperspectral Remote Sensing,” IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 12312-12331, 2021, doi: 10.1109/JSTARS.2021.3129818.

# Appendix

## Appendix A. Comparison of Classification Results Obtained Using Rice Seed Datasets Before and After the Removal of Values Below 400nm.

The hyperspectral data of rice seeds used in this thesis were collected at a Visible - Near Infrared (VIS/NIR) range of ~ (385 – 1000) nm. Some extra work was therefore carried out to investigate whether the values below 400nm affect the results, and this is presented in this section. Firstly, the values below 400 nm were removed from the rice seed datasets of 20 and 10 species. Secondly, the original datasets (containing 256 spectral features) and resulting datasets were separately presented to a Random Forest model for classification. The classification results are presented in Table 1 and Table 2 below. As can be seen in Table 1 and Table 2, the classification accuracies and $F_1$ scores achieved using the resulting datasets are lower than those achieved using the original datasets. This is because those values which were removed carry some significance i.e. they contain useful information.

Table 17 Classification results using the rice seed datasets of 20 species before and after the removal of values below 400 nm

| Number of bands discarded | Number of bands used | Wavelength range (nm) | Accuracy | $F_1$ Score |
|---|---|---|---|---|
| - | **256** | **383.22 - 1006.5** | **58.33** | **56.87** |
| 7 | 249 | 400.25 - 1006.5 | 57.03 | 55.77 |

Table 18 Classification results using the rice seed datasets of 10 species before and after the removal of values below 400 nm

| Number of bands discarded | Number of bands used | Wavelength range (nm) | Accuracy | $F_1$ Score |
|---|---|---|---|---|
| - | **256** | **383.22 - 1006.5** | **70.31** | **68.72** |
| 7 | 249 | 400.25 - 1006.5 | 68.75 | 66.77 |

The average spectral profiles of the datasets before and after the removal of values below 400 nm are also illustrated in Figure 1 - Figure 4 below.

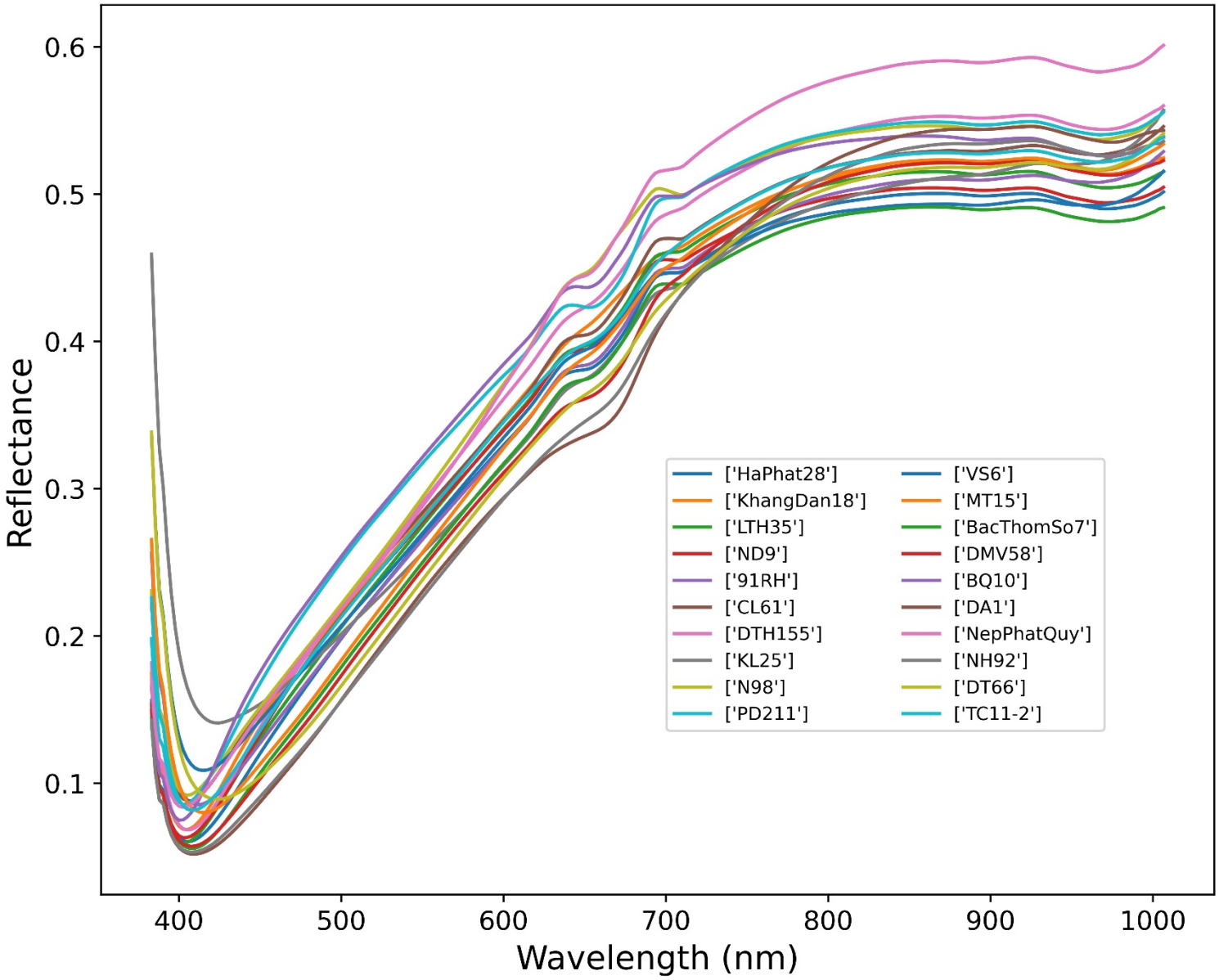



Figure 26 The average spectral profiles of 20 rice seed species before the removal of values below 400 nm

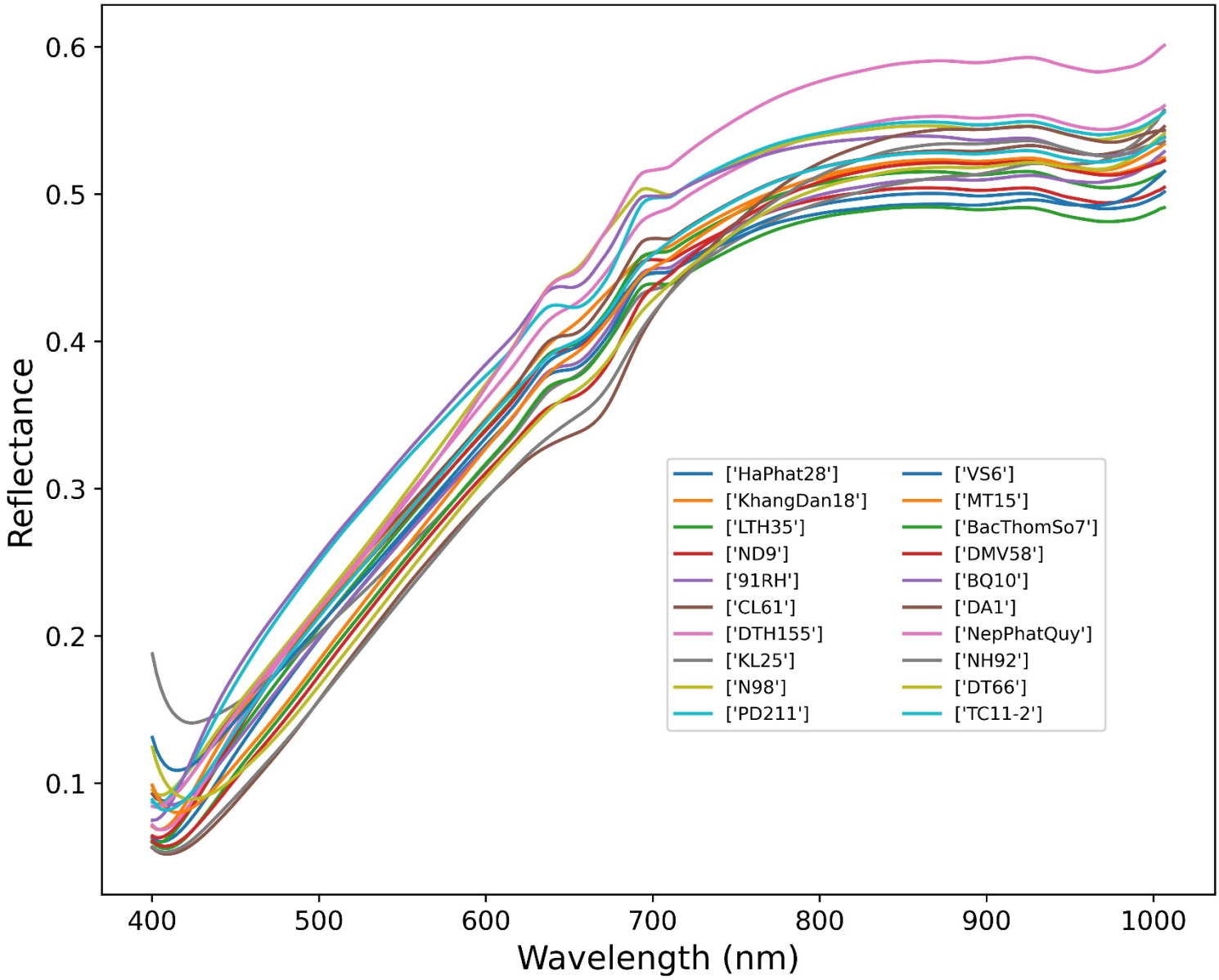



Figure 27 The average spectral profiles of 20 rice seed species after the removal of values below 400 nm

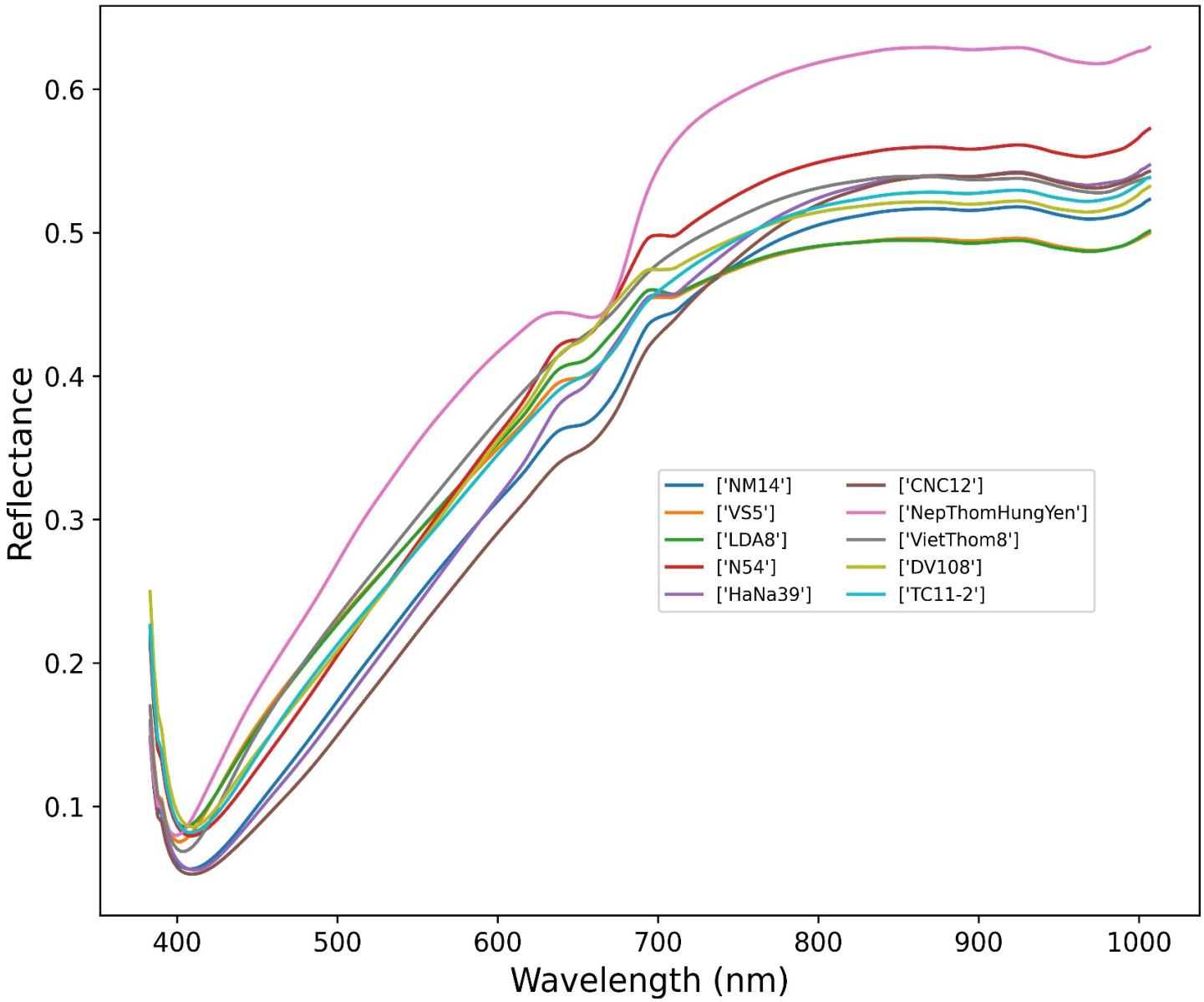



Figure 28 The average spectral profiles of 10 rice seed species before the removal of values below 400 nm

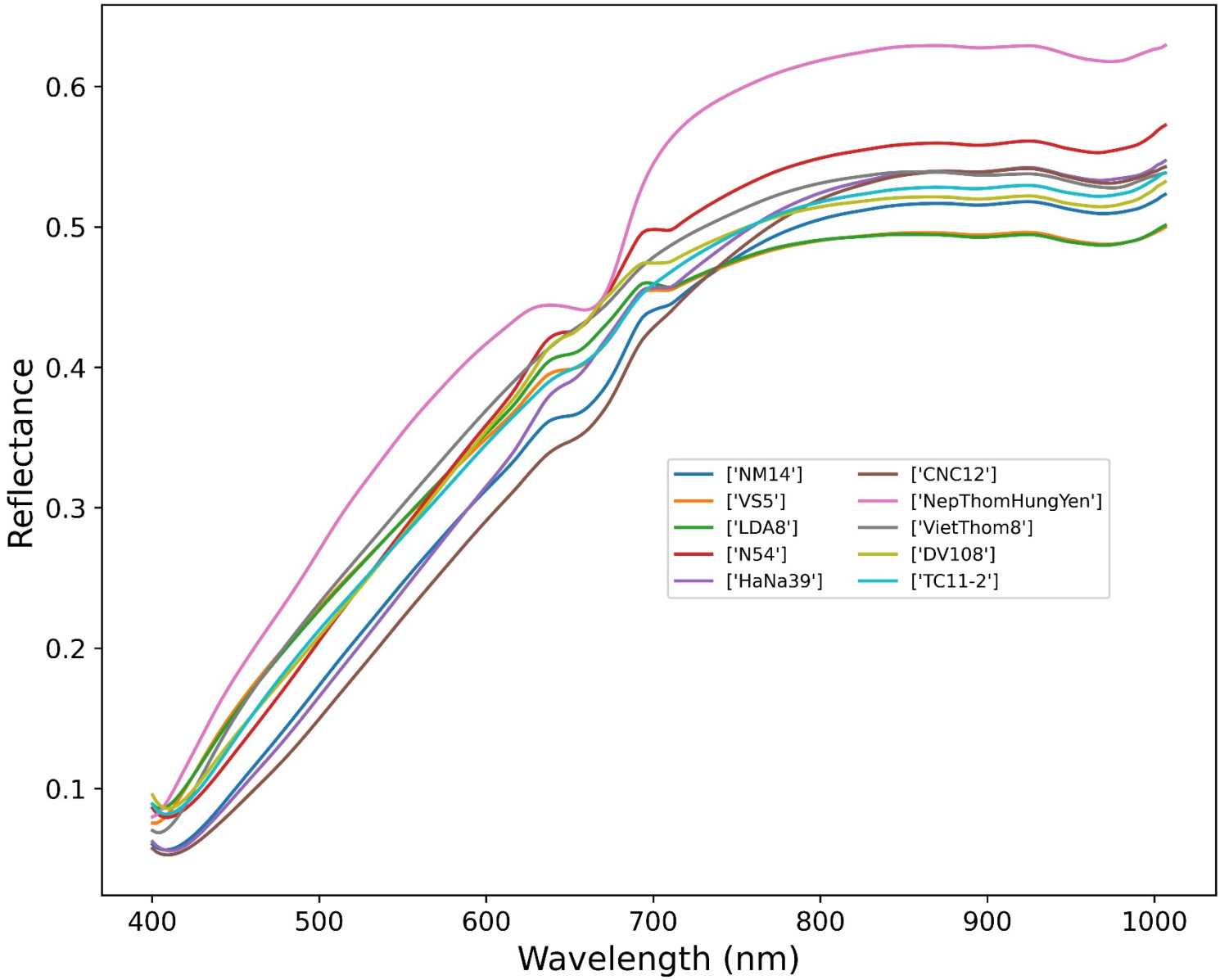



Figure 29 The average spectral profiles of 10 rice seed species after the removal of values below 400 nm