

SWITCHGRASS GENOTYPE CLASSIFICATION USING HYPERSPECTRAL IMAGERY

By

HUDANYUN SHENG

A THESIS PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

UNIVERSITY OF FLORIDA

2019

© 2019 Hudanyun Sheng

To the very best of times! To the days and nights in either lab or gym! To coffee and salad!

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my dearest advisor, Dr. Alina Zare, for all her guidance and support for my studies and research. I would also like to thank my committee members, Dr. Paul Gader and Dr. Damon Woodard, for all their valuable suggestions and advice.

I would also like to thank Dr. Brandon Davis, Dr. Felix Fritschi, Dr. Thomas Juenger, Mr. Jason Bonnette, and Ms. Hannah Wilson for their dedicated work on data collection, and numerous advice and discussions with me.

Additionally, thank you my labmates! I am particularly thankful to Weihuang Xu and Guohao Yu for their patience and willingness to offering help.

I also want to express my appreciation to my boyfriend Ruochen Liu for his support.

Finally many thanks to Dr. Panos M. Pardalos, whose course first made me feel like digging into the world of Machine Learning.

TABLE OF CONTENTS

	<u>page</u>
ACKNOWLEDGMENTS	4
LIST OF FIGURES	7
ABSTRACT	9
 CHAPTER	
1 INTRODUCTION	11
1.1 Goal of Study	11
1.2 Hyperspectral Imaging and Application	11
1.3 Classification in High Dimensional Space	12
1.4 Statement of the Problem	12
2 LITERATURE REVIEW	14
2.1 Stitching and Making Orthomosaics	14
2.2 Hyperspectral Unmixing	16
2.3 Classification in High Dimensional Space	20
2.3.1 Dimensionality Reduction	21
2.3.2 Classification	23
3 MATERIALS AND METHODS	26
3.1 Dataset	26
3.2 Preprocessing	26
3.2.1 Radiometric Calibration	27
3.2.2 Noisy Band Removal	29
3.2.3 Stitching and Making Orthomosaics	29
3.3 Switchgrass Detection	30
3.3.1 Sparsity Promoting Iterated Constrained Endmember Detection (SPICE)	30
3.3.2 Switchgrass Detection Generation	31
3.4 Switchgrass Genotype Classification	31
3.4.1 Ground Truth Labeling	32
3.4.2 Dimensionality Reduction	33
3.4.2.1 Dimensionality reduction by learning an invariant mapping	33
3.4.2.2 Dimensionality reduction by principal component analysis transformation	34
3.4.2.3 Dimensionality reduction by linear discriminant analysis	34
3.4.3 K-Nearest Neighbor Classifier	34
3.5 Summary of Proposed Approach	35

4	RESULTS	50
4.1	Dimensionality Reduction using The Siamese Network	50
4.2	Dimensionality Reduction by Principal Component Analysis Transformation	51
4.3	Dimensionality Reduction by Linear Discriminant Analysis	51
4.4	Results Comparison	51
5	SUMMARY AND CONCLUSIONS	63
	REFERENCES	66
	BIOGRAPHICAL SKETCH	72

LIST OF FIGURES

<u>Figure</u>		<u>page</u>
1-1	Different materials produce different reflectance spectra.	13
2-1	Toy example of the data and the simplex shown in two-dimensional space.	25
2-2	Contrastive loss function representation	25
3-1	The planting layout of two planting techniques.	37
3-2	Calibration processing steps and white tarp used for calibration.	37
3-3	Radiometric Calibration	38
3-4	Sample spectra before and after noisy band removal.	38
3-5	Stitched images from one example scanning of the STND switchgrass field.	39
3-6	Detected endmembers using SPICE algorithm.	40
3-7	Abundance maps and the original hyperspectral image for data collected from the GWAS field.	41
3-8	Abundance maps and the original hyperspectral image for data collected from the STND field.	42
3-9	Final endmembers and the original hyperspectral image shown in RGB for data collected from the STND field.	43
3-10	Abundance maps correspond to the final set of endmembers for data collected from the STND field.	44
3-11	Selected switchgrass detection results compared to the original image for data collected from the STND field.	45
3-12	Selected switchgrass detection results compared to the original image for data collected from the GWAS field.	46
3-13	Example of manual labeling the ground truth interactively using MATLAB. Courtesy of author.	47
3-14	Abundance maps and original hyperspectral image of the STND field.	47
3-15	The Siamese network architecture and details of network layers.	48
3-16	The flowchart of the adopted approach.	49
4-1	Learning curve for Siamese network training.	54
4-2	Errorbar of classification performance versus the dimensionality of the learned manifold.	54

4-3	The cumulative explained variance ratio against the number of components of PCA.	55
4-4	Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 15.	56
4-5	Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 30.	57
4-6	Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 55.	58
4-7	Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 5.	59
4-8	Sample classification results shown in images.	60
4-9	Visualization of original spectra for 6 switchgrass genotypes, with the variances shown in the error bars.	61
4-10	Visualization of dimensionality reduction results using the Siamese network and principal component analysis.	62

Abstract of Thesis Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Master of Science

SWITCHGRASS GENOTYPE CLASSIFICATION USING HYPERSPECTRAL IMAGERY

By

Hudanyun Sheng

December 2019

Chair: Alina Zare

Major: Electrical and Computer Engineering

The adoption of remote sensing techniques in plant science enables noninvasive or minimally invasive measurement, which is also time and labor saving when compared to traditional field measurements. In this thesis, a method to distinguish switchgrass genotypes with the analysis of remotely-sensed hyperspectral imagery is proposed.

A processing protocol for hyperspectral imagery including preprocessing, ortho-mosaic generation, hyperspectral endmember detection and unmixing, and pixel-level genotype classification is presented. Machine learning algorithms for material detection, dimensionality reduction, and classification are proposed and discussed in detail. The impact of dimensionality reduction on classification is examined in depth.

The challenges in distinguishing between genotypes of the same species arise from the fact that the spectra of different genotypes from the same species are very similar with overlapping distribution. Furthermore, high between-band correlation in hyperspectral data suggests that dimensionality reduction of hyperspectral images may be useful. Whether dimensionality reduction can help in discriminating between spectra of genotypes of the same species is investigated. Three dimensionality reduction methods are described. The first method, principal component analysis, reduces dimensionality by linearly transforming the data to a space with lower dimensionality such that the maximal data variance is preserved. The other two methods reduce the dimensionality by finding a new feature space where the class separability is maximized. The second method, linear discriminant analysis,

reduces dimensionality by linear projection. The third method, the Siamese network, reduces dimensionality by a non-linear mapping from the original space.

K-nearest neighbor (KNN) is used as the multi-class classifier in this work. KNN is one of the simplest non-parametric supervised classification algorithms which assumes that similar data points exist in proximity. The significance of reducing dimensionality by non-linear methods in improving the classification performance is manifested.

CHAPTER 1

INTRODUCTION

1.1 Goal of Study

Switchgrass (*panicum virgatum*) is one of the dominant plant species native to North America that can be used as ground cover for soil conservation to control erosion, feedstock for biomass energy production, or grazing ([Commons, 2018](#)). Different genotypes of switchgrass behave differently in terms of height, tiller counts, and dry biomass weight. Plant phenotyping attempts to quantitatively measure the structural and functional properties of different genotypes of plants, and can be used to select the genotypes best suited the desired application. Yet, traditional phenotyping methods are laborious and time-consuming ([Mahlein, 2016](#)). The adoption of remote sensing in plant science is promising. However, the main drawback is the amount of data collected resulting in high complexity of processing needed to analyze the data. Hyperspectral imagery is an example of the types of data collected using remote sensing techniques. The machine learning algorithms introduced into the analysis become a key technology for the utilization of the collected remote sensing data ([Mahlein, 2016; Fiorani and Schurr, 2013](#)).

1.2 Hyperspectral Imaging and Application

Hyperspectral images capture not only spatial information but also spectral information of a scene. [Nasrabadi \(2013\)](#) explained that a hyperspectral spectrometer provides hundreds of contiguous wavelength channels over a range of the electromagnetic spectrum. [Manolakis and Shaw \(2002\)](#) pointed out that all materials, having different molecular compositions, absorb, emit, and reflect electromagnetic energy at specific wavelengths in different patterns. The variation of reflectance or emittance of a material with respect to wavelengths is known as the spectral signature. Figure 1-1 shows an example of different materials having different spectral signatures. Exploiting the difference in spectral signatures of different materials enables the detection or identification of target materials in a scene.

The potential to distinguish materials using spectra enables major applications of hyperspectral imagery such as classification and target detection. Hyperspectral classification aims at assigning class labels for every pixel automatically. Target detection, which is considered as a binary classification problem, works towards searching pixels for the presence of the desired material ([Manolakis and Shaw, 2002](#)).

The ability of uniquely characterizing materials with spectral signatures enables the application of identifying different materials using spectra. However, challenges arise from multiple scattering, the low spatial resolution of sensors, variations in material surface, atmospheric conditions, location, and microscopic material mixing ([Manolakis and Shaw, 2002](#)). The major challenge is that the measured spectra in a pixel are mixtures of spectra. In other words, pixels with pure spectra of a single material (pure pixels) rarely exist ([Bioucas-Dias et al., 2012](#)).

1.3 Classification in High Dimensional Space

As a special case of high dimensionality data, hyperspectral data suffers from the *Curse of Dimensionality* ([Schürmann, 1996](#)). As pointed out by [Jain and Waller \(1978\)](#), the size of training data limits the optimal dimension of feature vectors. In other words, a substantial number of training samples are needed to accurately classify high dimensional data. However, the number of training samples is limited in most real-life applications ([Belharbi, 2018](#)). Additionally, including large amounts of training data makes the whole process time-consuming and computationally expensive. The high dimensional space for a given hyperspectral data set is mostly empty, so it would be efficient to reduce the dimensionality of the data before classification ([Jain and Waller, 1978](#)). Furthermore, the existence of a high between band correlation also suggests the necessity of dimensionality reduction ([Green et al., 1988](#)).

1.4 Statement of the Problem

Hyperspectral imagery can be used for target material detection because different materials have different spectral signatures. However, the task becomes difficult when trying to classify materials with similar spectral signatures, different genotypes of the same plant

species, for instance. The differences between hyperspectral signatures of different genotypes of switchgrass are not as significant as ones between switchgrass and background materials.

The existence of a high between band correlation of hyperspectral data, which means values for different bands only differ by a scaling factor, indicates the need for data compression. Consequently, band selection or feature selection is often adopted to reduce the dimensionality of hyperspectral data. A comparison of three linear and non-linear dimensionality reduction algorithms are investigated in this work.

A protocol for processing and analyzing hyperspectral data using machine learning algorithms is put forward in this work. This protocol includes preprocessing, target detection, and classification.

This work manifests the necessity of dimensionality reduction to classify high dimensional data. The result of adopting the nonlinear dimensionality reduction method is compared to those with two linear dimensionality reduction methods applied to the same data.

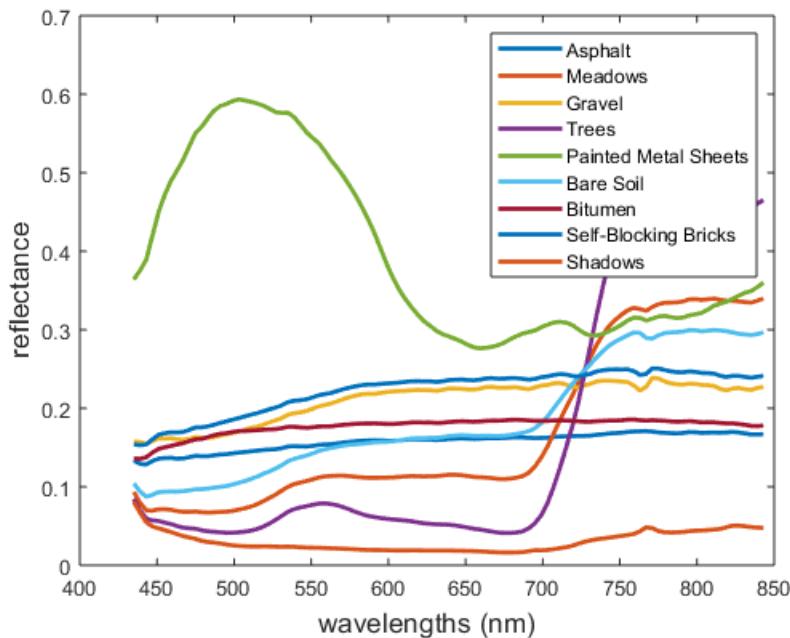


Figure 1-1. Different material produce different reflectance spectra.

CHAPTER 2

LITERATURE REVIEW

This chapter presents a review of existing methods for stitching and making ortho-mosaics, hyperspectral unmixing, and classification in high dimensional space. Stitching and making ortho-mosaics is a common preprocessing step in hyperspectral imagery analysis, which give an overall understanding of the field of interest. Hyperspectral unmixing is adopted to address the issue that each pixel is a mixture of several materials in hyperspectral images, which is caused by the low spatial resolution of hyperspectral sensors. Hyperspectral data is a subclass of high-dimensional data, and it shares similar characteristics of general high dimensional data and classification in high dimensional data. Furthermore, hyperspectral images have a high between band correlation. These stated above indicate that dimensionality reduction is necessary for classifying hyperspectral data.

2.1 Stitching and Making Orthomosaics

The utilization of unmanned aerial systems is becoming popular for remote sensing applications in both military ([DeGarmo, 2004](#); [Ma'Sum et al., 2013](#)) and civil research areas. Examples include precision agriculture ([Honkavaara et al., 2013](#); [Saari et al., 2011](#)) and biological research ([Bolland-Breen et al., 2015](#)). As a result, hyperspectral imagery has been extensively used and studied.

In most cases a large number of images are taken to cover the whole field. To be able to represent the earth's surface, it is necessary to convert multiple or hundreds of raw scanning images into a small number of images which are usable and in good quality ([Gross, 2015](#)). As stated in the review of [Szeliski et al. \(2007\)](#), after image capture, the first step in creating ortho-mosaics is to have a good understanding of the image. In other words, find a suitable mathematical model to describe the motion of the image and the alignment algorithms that can be applied afterward. One category of major alignment algorithms is the direct methods that use pixel-to-pixel matching. Examples of direct methods are hierarchical motion estimation that performs a full search over some range of shifts based on the defined

cost function, and Fourier-based alignment based on the properties of Fourier transform. Another category of major alignment algorithms is feature-based methods which use the extracted distinctive features for alignment. It includes the steps of key-point detection, feature matching, geometric registration. [Zitova and Flusser \(2003\)](#) discussed several feature-based methods in detail and argued that feature-based methods are recommended when enough amount of easily detected objects are observed.

One of the main challenges when creating composite images from aerial photographs is that the quality of scanned images is deteriorated by parallax effects caused by the variation in the apparent position of an object in view ([Mills and McLeod, 2013](#)). To solve this problem, [Zitova and Flusser \(2003\)](#) suggested that image registration is a required step for remote sensing image analysis. [Zitova and Flusser \(2003\)](#) explained that image registration is the technique of overlaying multiple images of the same scene taken at different times, from different viewpoints, or sometimes by different sensors. A number of existing algorithms can automatically align multiple images. One approach is called structure from motion in which the well-known scale-invariant feature transform (SIFT) locates the key-points. ([Gross, 2015](#)).

There is also software for stitching and making ortho-mosaics available such as Pix4Dmapper (Pix4D, S.A., Lausanne, Switzerland), Photoscan (Agisoft LLC, St. Petersburg, Russia), and Microsoft Image Composite Editor (Microsoft, Redmond, Washington, U.S.). As [Gross \(2015\)](#) stated, the choice of software is application dependent: no simple best solution suits all cases. Both Pix4Dmapper and Photoscan are commercial software, while Microsoft ICE is a freely available software designed for the application of image stitching. Both Pix4Dmapper and Photoscan have the ability to generate 3D point cloud or 3D images, and they have options of parameter tuning or including ground control point (GCP) data, while Microsoft ICE focuses solely on panoramic image stitching, with no parameters to tune, nor being able to include additional information apart from images themselves.

2.2 Hyperspectral Unmixing

Hyperspectral images are images captured by hyperspectral cameras in hundreds or thousands of spectral channels, which are used to measure electromagnetic energy scattered in the scene (Bioucas-Dias et al., 2012). Mills and McLeod (2013) explained that the signal at one pixel of a hyperspectral image is a mixture of light scattered by materials in the field of view. Endmembers are the spectral signatures of materials in the scene that are assumed to be pure. Unmixing is a common step when analyzing hyperspectral images. Generally speaking, unmixing follows the steps of determining the number of endmembers, estimating spectral signatures of endmembers, and calculating the abundance of every endmember.

Unmixing algorithms are categorized based mainly on the assumption of mixing mechanism, namely linear or nonlinear mixing (Keshava and Mustard, 2002). The linear mixing model is an adequate approximation for most of the applications in spite of its subtlety and simplicity (Bioucas-Dias et al., 2012). When a linear mixture is assumed, every pixel is believed to be a linear combination of the endmembers in the scene (Berman et al., 2004):

$$X_i = \sum_{k=1}^M p_{ik} E_k + \varepsilon_i, \quad i = 1, \dots, N \quad (2-1)$$

where N represents the number of total pixels, M represents the number of endmembers in the scene, E_k represents the k th endmember, p_{ik} represents the proportion of endmember k in pixel i , and ε_i represents an error term. The proportions satisfy the constraints

$$\sum_{k=1}^M p_{ik} = 1; \quad p_{ik} \geq 0, \quad k = 1, \dots, M \quad (2-2)$$

This model shows that data lies inside a d-dimensional simplex whose vertices are the M endmembers. In other words, all data are in the simplex constructed by endmembers. Figure 2-1 presents a toy example of simplex and data in two-dimensional space.

Geometrical and statistical frameworks are the two most used frameworks with the linear mixture model. Geometrical approaches make use of the geometric orientation of hyperspectral data in multidimensional spaces (Parente and Plaza, 2010). Two main

categories of Geometrical-based approaches are Pure Pixel-based approaches and Minimum Volume-based approaches. Examples of most used algorithms with this assumption are: pixel purity index algorithm ([Boardman, 1993](#)), N-FINDER ([Winter, 1999](#)), vertex component analysis ([Nascimento and Dias, 2005](#)), and simplex growing algorithm ([Chang et al., 2006](#)). With the dimensionality of original data being reduced by maximum noise fraction (MNF) transformation, the pixel purity index algorithm detects the endmembers by finding N pixels with the highest score of being extreme, where N is a pre-defined value by other algorithms or domain knowledge ([Boardman et al., 1995; Boardman, 1993](#)). The MNF transformation improves the signal-to-noise ratio simultaneously ([Green et al., 1988](#)). The N-FINDER algorithm detects the endmembers by finding the largest-volume simplex generated by the hyperspectral data based on two assumptions: 1) The endmembers are the vertices of a simplex; 2) The simplex determined by pure pixels is larger than any other simplexes determined by other input hyperspectral data. Relying on the geometric properties of convex sets, the vertex component analysis (VCA) detects the endmembers in an iterative way until no endmembers left. In every iteration, the algorithm projects the input data onto a subspace that is orthogonal to the subspace whose spanning set is the existing endmembers. Endmembers detected in every iteration are the extremes of the projected data ([Nascimento and Dias, 2005](#)).

Some other algorithms are proposed based on the algorithms mentioned above. For example, the N-FINDER inspired alternating volume maximization, maximizes the volume of the simplex repeatedly ([Chan et al., 2011](#)). The successive volume maximization is similar to vertex component analysis, with the only difference being the direction of the projected data ([Chan et al., 2011](#)).

A general case of geometrical-based approaches is used when the pure pixel assumption no longer holds. As discussed above, hyperspectral data defines a simplex of high dimensionality. General case geometrical-based algorithms try to find the mixing matrices that make the volume of the simplex minimum. The minimum volume transform-nonnegative matrix

factorization (MVC-NMF) algorithm limits the volume of the simplex from growing large by a regularization term (Miao and Qi, 2007). Similar to the MVC-NMF algorithm, the iterative constrained endmembers (ICE) algorithm solves the optimization problem with the objective function regularized by the sum of squared distances between all the simplex vertices (Berman et al., 2004; Bioucas-Dias et al., 2012). Adding sparsity-promoting priors to the ICE algorithm, Zare and Gader (2007) provided the means of estimating the number of endmembers automatically. The algorithm is called sparsity-promoting ICE (SPICE) algorithm.

The ICE algorithm tries to minimize the residual sum of squares (RSS) between the data and the linear model:

$$RSS = \sum_{i=1}^N (X_i - \sum_{k=1}^M p_{ik} E_k)^T (X_i - \sum_{k=1}^M p_{ik} E_k) \quad (2-3)$$

Berman et al. (2004) further argued that without any constraints, there would be more than one minimizer that satisfies. So they proposed adding a simplex size constraining term to the objective function. This term is known as the sum of squared distances (SSD) and is later proved to be equivalent to $M(M - 1)V$. In the final expression of the proposed objective function, as is shown in Equation 2-4, V was used rather than $M(M - 1)V$ because Berman et al. (2004) wanted to make the objective function approximately independent of the number of endmembers M :

$$RSS_{reg} = (1 - \mu) \frac{RSS}{N} + \mu V \quad (2-4)$$

Zare and Gader (2007) extended the ICE algorithm by adding a term that promotes sparsity to the original objective function. The sparsity-promoting term is

$$SPT = \sum_{k=1}^M \gamma_k \sum_{i=1}^N |p_{ik}| = \sum_{k=1}^M \gamma_k \sum_{i=1}^N p_{ik} \quad (2-5)$$

where

$$\gamma_k = \frac{\Gamma}{\sum_{i=1}^N p_{ik}} \quad (2-6)$$

The updated objective function is expressed as

$$RSS_{reg}^* = (1 - \mu) \frac{RSS}{N} + \mu V + SPT \quad (2-7)$$

The SPICE algorithm uses the same iterative optimization procedure as is used in ICE. Setting the initial number of endmembers to be a large one, the SPICE algorithm automatically learns the number of endmembers by discarding the endmembers whose proportion values are smaller than the threshold indicated by the sparsity-promoting term. The spectral signatures and the abundance values of every pixel corresponding to every endmember are calculated simultaneously.

In some cases where data are highly mixed, the geometrical-based approaches are unsuitable. In these cases, the statistical methods are considered as a compelling option to solve the hyperspectral unmixing problem. To be specific, the hyperspectral unmixing problem is treated as a statistical inference problem ([Bioucas-Dias et al., 2012](#)). Bayesian approaches, specifically joint *maximum a posterior* (MAP) estimators ([Bernardo and Smith, 2009](#)), which are capable of modeling statistical variability with priors to have constrained results, are widely used when formulating the unmixing problem. Above mentioned geometrical based algorithms such as MVC-NMF ([Miao and Qi, 2007](#)), ICE ([Berman et al., 2004](#)) and SPICE ([Zare and Gader, 2007](#)) can also be considered as belonging to statistical methods because they solve the unmixing problem by the joint MAP approach.

With a spectral library of pure spectral signatures available, the unmixing problem can be converted to a problem of finding a subset of signatures from the signature library such that every pixel can be modeled ([Iordache et al., 2011; Rogge et al., 2006](#)).

As argued by [Shi and Wang \(2014\)](#), the aforementioned unmixing algorithms use only spectral information, which means the pixels are treated as isolated entities instead of also including the local correlation between them ([Xu et al., 2018](#)). Methods that make use of the spectral and spatial information cooperatively are proposed. One category of methods is called spatial preprocessing methods, which adopt a preprocessing technique before

endmember extraction. Examples include Spatial Preprocessing ([Martin and Plaza, 2011](#)), and Region-based Spatial Preprocessing ([Zortea and Plaza, 2009](#)). The other category of methods that utilize both spectral and spatial information is known as integrated spatial-spectral methods. Examples of existing methods are Automatic Morphological Endmember Extraction ([Plaza et al., 2002](#)), and Regional Clustering-based Spatial Preprocessing ([Xu et al., 2018](#)).

In past decades, there has been an increasing number of applications of deep learning methods for hyperspectral unmixing. With prior knowledge of endmembers available, [Li et al. \(2016\)](#) and [Licciardi and Del Frate \(2011\)](#) proposed to use autoencoder for estimation of abundances. [Su et al. \(2017\)](#) came up with the idea of using a sparse autoencoder for endmember extraction and an autoencoder with a deep encoder for estimation of the endmember spectra and their abundances simultaneously. Later on, [Su et al. \(2019b\)](#) proposed a deep autoencoder network whose architecture is stacked autoencoders (SAEs) followed by a variational autoencoder (VAE). The SAEs are used to learn endmember signatures, and the VAE is used for un-mixing.

2.3 Classification in High Dimensional Space

Data has become increasingly large not only in the quantity of data but also in the dimension of feature vectors, especially in applications such as healthcare, remote sensing, image retrieval ([Yu and Liu, 2003](#)). Although conveying information, data of high dimensionality may cause problems when machine learning algorithms are applied to analyze it, resulting from what is known as the *Curse of Dimensionality*. To be specific, the amount of data needed to densely populate a space rises quickly with dimensionality, while the amount of data is limited. Furthermore, the hyperspectral data is known to have high between-band correlations, which indicates that the useful information is conveyed in a subset of bands. Consequently, dimensionality reduction is often applied to the hyperspectral data before other machine learning algorithms are applied to.

2.3.1 Dimensionality Reduction

Ready and Wintz (1973) and Singh and Harrison (1985) proposed the use of the well-known dimensionality reduction algorithm, Principal Component Analysis (PCA), to compress high dimensional remote sensing data. To be specific, the PCA reduces the dimensionality based on a linear transformation. After the linear transformation, the principal components have descending variances. Green et al. (1988) argues that the Maximum Noise Fraction (MNF) algorithm linearly transforms the input data and consistently outputs images in the order of decreasing quality. Both PCA and MNF are well-known algorithms for hyperspectral imagery denoising (Luo et al., 2016). Feature selection reduces dimensionality by choosing a subset of features such that original physical meaning is retained. Yu and Liu (2003) proposed a feature selection method known as faster filtering, which is capable of identifying not only relevant features but also the redundancy among relevant features. When considered in the application of hyperspectral imagery, feature selection is also referred to as band selection. Based on whether the desired object information is known, band selection can be categorized to supervised or unsupervised (Yang et al., 2010).

Focusing on the ultimate goal of classification, Jimenez et al. (1998) proposed the use of Projection Pursuit technique that reduces the dimensionality of data while preserving as much information as possible by maximizing the separability, which is believed to project the data to a highly separable manifold. Generalized from Fisher's linear discriminant (Fisher, 1936), the linear discriminant analysis (LDA) reduces the dimensionality by finding a new feature space to project the data onto, such that the class separability is maximized. To be specific, this method reduces the dimensionality from the original to $C-1$ where C is the number of classes, trying to maximize the function that represents the differences between the mean values, normalized by the within-class variability. A drawback of this method is that it assumes the dataset has a normal distribution.

Proposed by Hadsell et al. (2006), *Dimensionality Reduction by Learning an Invariant Mapping* reduces the dimensionality by transforming data from the original space to a low

dimensional manifold such that the "similar" points in the input space will be close to each other in the output space and vice versa. In other words, a nonlinear mapping function is learned with prior knowledge of neighborhood relationships between pairs of input data in the input space. Prior knowledge can come from manual labeling.

The algorithm is detailed as follows. Given the high dimensional input data, the algorithm learns a transformation to low dimensionality such that the distance measures in the output space correspond to neighborhood information in the input space. Mathematically, denote input data points as $\mathcal{I} = \{\vec{X}_1, \dots, \vec{X}_P\}$, where $\vec{X}_i \in \mathcal{R}^D$, $\forall i = 1, \dots, n$. Denote the function to be learned as $G_W : \mathcal{R}^D \rightarrow \mathcal{R}^d$ ($d \ll D$). Different from conventional network training where the loss function is a sum over losses of all training samples, the loss function defined here known as the contrastive loss function sums over all pairs of samples. Assume a pair of input vectors are \vec{X}_1 and \vec{X}_2 , and the corresponding binary label describing the similarity of this pair of data is Y . The binary label Y is defined based on the similarity of the input data pair. For a pair of input data \vec{X}_1 and \vec{X}_2 , define $Y = 0$ if they are similar in the input space, and $Y = 1$ if they are dissimilar in the input space. The distance between the transformed \vec{X}_1 and \vec{X}_2 , denoted as $D_W(\vec{X}_1, \vec{X}_2)$, and written as D_W , is the function to be learned. One choice of the distance function is the Euclidean distance:

$$D_W(\vec{X}_1, \vec{X}_2) = \|G_W(\vec{X}_1) - G_W(\vec{X}_2)\|_2 \quad (2-8)$$

The general form of contrastive loss function is defined as:

$$\mathcal{L}(W) = \sum_{i=1}^P L(W, (Y, \vec{X}_1, \vec{X}_2)^i) \quad (2-9)$$

where $(Y, \vec{X}_1, \vec{X}_2)^i$ is the i -th pair of training data; P denotes the number of training pairs.

The contrastive loss function is further expressed as:

$$\mathcal{L}(W(Y, \vec{X}_1, \vec{X}_2)^i) = (1 - Y)L_S(D_W^i) + YL_D(D_W^i) \quad (2-10)$$

where L_S represents the partial loss for similar pairs of input data; L_D denotes the partial loss for dissimilar pairs of input data. The loss function treats similar pairs and dissimilar pairs of input differently, such that the closer two similar points the lower the loss function value; whereas the closer two dissimilar points the higher the loss value:

$$L(W, Y, \vec{X}_1, \vec{X}_2) = (1 - Y) \frac{1}{2} (D_W)^2 + (Y) \frac{1}{2} \{ \max(0, m - D_W) \}^2 \quad (2-11)$$

where $m > 0$ is a user-defined margin, which a threshold on the distance such that loss value for dissimilar pairs becomes zero if their distance is greater than m . In other words, the loss of a dissimilar pair of input adds to the loss function only if the distance value between two data points is smaller than the value of user-defined margin. The contrastive loss function is illustrated in Figure 2-2.

2.3.2 Classification

Classification aims at assigning class labels based on the input feature vectors and is one of the most studied problems in machine learning. Classification methods can be categorized into parametric or non-parametric based on whether having the assumption of the underlying probability distribution of the data set. Examples for parametric classifiers are linear support vector machines (SVM) classifiers ([Cortes and Vapnik, 1995](#)), logistic regression ([Walker and Duncan, 1967](#)). Examples for non-parametric classifier include the K nearest-neighbor (KNN) classifier ([Altman, 1992](#)), and decision tree classifiers ([Breiman et al., 1984](#)).

KNN is one of the easiest and most widely used classification algorithms. KNN accomplishes the classification task by first finding the K closest points to every test pixel, and then assign labels by a majority vote of these K closest points. The closeness between two points is indeed the similarity between them, which is defined using a distance or similarity measure ([Altman, 1992](#)). A popular choice is the Euclidean distance:

$$d(x, x') = \sqrt{(x_1 - x'_1)^2 + \dots + (x_n - x'_n)^2} \quad (2-12)$$

It is interesting to note that KNN is an instance-based algorithm. In other words, instead of learning a model from training data, it predicts the class labels of new samples by using the stored training instances.

Based on the total number of classes, classification tasks can be categorized into binary and multi-class classification. The binary classifiers classify data into 2 classes, while the multi-class classifiers classify data into K classes. Multi-class classification problems are much more complicated than binary-class classification problems. Some classifiers can be extended from binary-class case to multi-class case naturally, for instance, KNN, neural networks, Naive Bayes, and decision trees ([Aly, 2005](#)). Besides the extension from binary class cases, another category of multi-class classifiers tries to transform the problem into binary. [Sejnowski and Rosenberg \(1987\)](#) proposed the method known as one-versus-all (OVA) that solves the K-class classification problem by solving K binary classification problems. The objective of each of these K problems is to classify one class from the other K-1 classes. [Hastie and Tibshirani \(1998\)](#) came up with the idea of using K(K-1) binary classifiers to discriminate between each pair of classes. This method is known as all-versus-all (AVA). Inspired by the idea of error-correcting codes, [Dietterich and Bakiri \(1994\)](#) came up with the error-correcting output-coding (ECOC) for multi-class classification. The generalized idea based on ECOC called Generalized Coding was proposed by [Allwein et al. \(2000\)](#). Last but not least, another approach addresses the multi-class classification problem by a hierarchical splitting of the output space. Examples include binary hierarchical classifier (BHC) ([Kumar et al., 2002](#)), hierarchical SVM (HSVM) ([Chen et al., 2004](#)), and divide-by-2 (DB2) ([Vural and Dy, 2004](#)).

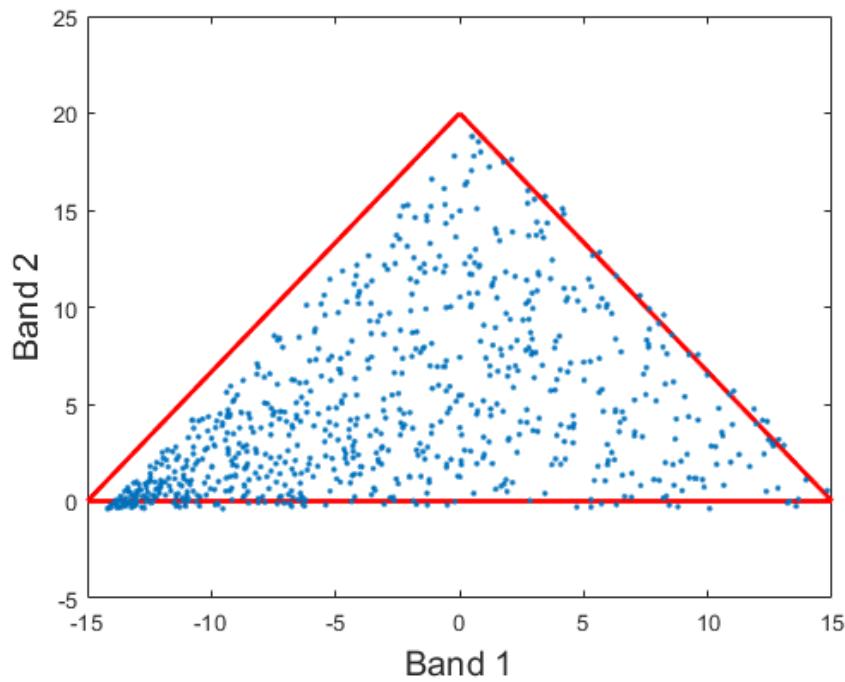


Figure 2-1. Toy example of the data and the simplex shown in two-dimensional space.

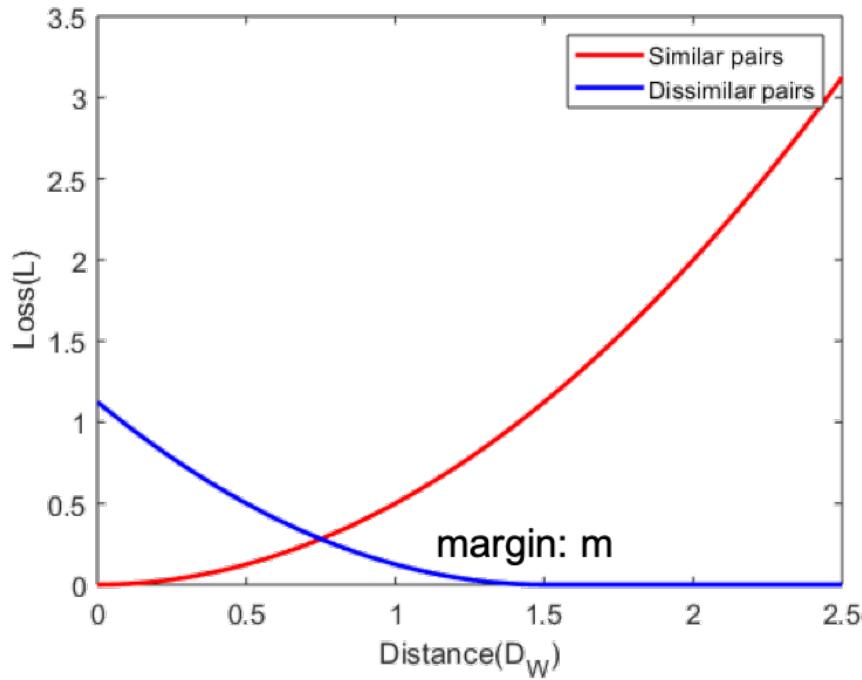


Figure 2-2. Representation of the loss function value L as a function of D_W . The red curve represents the loss function values for the similar input data pairs and the blue curve represents that for the dissimilar input data pairs.

CHAPTER 3

MATERIALS AND METHODS

In this chapter, an introduction of the dataset used as well as a review of the methodology adopted in this work is presented.

3.1 Dataset

The data used in this work are hyperspectral images collected from two fields of switchgrass in Columbia, Missouri.¹ In one field, a plot-based stand experiment design was adopted, where a group of all the same genotype of switchgrass was planted in a 6 m × 6m plot. A randomized block design was adopted with five replications in the field. This planting technique is referred to as “Stand” planting (STND). In this field, there are in total 6 genotypes of switchgrass, namely, Liberty, Blackwell, Alamo, Kanlow, Cave in rock (CIR) and Carthage. In the other field, the Genome-Wide Association Study (GWAS) is adopted. GWAS is an observational study to explore whether any variant is associated with a characteristic, by planting different individuals with genetic variants in the field. Figure 3-1 shows the layout of both STND and GWAS planting. It can be seen from Figure 3-1A that when STND planting is adopted, a group of plants with the same genotypes stand next to each other in a large plot. While when GWAS planting is adopted, as is shown in Figure 3-1B, the plants are spaced in a honey-comb pattern to ensure the maximum amount of growing space. There are 950 switchgrass genotypes present in the field.

A hyperspectral camera mounted on a UAV developed by Headwall Photonics, Inc. was used to scan the fields. The hyperspectral camera has coverage of wavelengths from 396nm to 996 nm.

3.2 Preprocessing

Before applying machine learning algorithms, necessary preprocessing including radiometric calibration, noisy band removal, and ortho-mosaic generation are implemented.

¹ Collaborators: Dr. Felix Fritchi, Dr. Thomas Juenger.

3.2.1 Radiometric Calibration

Hyperspectral cameras record the measured energy as raw digital numbers (DN) for every pixel. Those digital numbers are different from actual energy reflected from the interested earth's surface. The sun's azimuth and elevation are examples of the causes of this difference. Additionally, atmospheric conditions also influence the energy observed by the sensor. In other words, for a given pixel, the recorded DN represents the summation of radiation reflected or emitted by the material and the radiation scattered or emitted by the atmosphere ([Su et al., 2019a](#)). However, only the former radiation is of interest. Therefore, radiometric errors must be accounted for to get real radiance or reflectance values. This process is known as radiometric calibration.

A general process from raw data to reflectance data including two steps. Firstly, convert DN to radiance based on the rescaling factors. Secondly, convert radiance to reflectance. Different conversion methods are developed based on different assumptions. Figure [3-2A](#) shows a flowchart of this process.

There are two methods to convert DNs to radiance values. One of them relies on the gain and bias values included in the header files, as the formula is shown below:

$$L_\lambda = \text{gain} \times DN + \text{bias} \quad (3-1)$$

where L_λ represents the radiance value, DN represents the digital number recorded by the sensor, *gain* and *bias* represent the gain and bias values for a specific band respectively([Khalid et al., 2019](#)). The other method uses min and max spectral radiance scaling factors:

$$L_\lambda = ((L_{\text{MAX}} - L_{\text{MIN}}) / (\text{QCALMAX} - \text{QCALMIN})) \times (DN - \text{QCALMIN}) + L_{\text{MIN}} \quad (3-2)$$

where L_λ represents the radiance value, DN represents the digital number recorded by the sensor, QCALMIN and QCALMAX represent the minimum and maximum values of quantized calibrated pixel, respectively, L_{MIN} and L_{MAX} represent spectral radiance values scaled to QCALMIN and QCALMAX respectively. The reflectance values can be obtained from the

radiance values with additional information including solar zenith angle and earth-sun distance:

$$\rho_\lambda = \pi \times L_\lambda \times d^2 / ESUN_\lambda \times \cos\theta_s \quad (3-3)$$

where ρ_λ is the unitless reflectance, L_λ is the radiance, d is the Earth-Sun distance in astronomical units, $ESUN_\lambda$ is the mean solar exoatmospheric irradiances, and θ_s is the solar zenith angle ([Markham, 1986](#)).

In this work, the radiometric calibration is accomplished using the SpectralView software (Headwall Photonics, Inc. Boston, United States). To be specific, the software has a built-in function that converts DNs to reflectance values. The raw data values are first converted to radiance values based on the dark reference information, and then the radiance values are converted to reflectance values using white reference information extracted from calibration tarps placed in the field during image collection. The dark and white references are used to normalize the data. The “dark reference file” containing gain and bias information is needed to convert raw data cubes to radiance data cubes - the “zero reflectance” value. The “white reference file” containing additional information as described above is needed to convert radiance data cubes to reflectance data cubes. In this work the “white reference file” contains the 56% reflectance information. The “dark reference file” is collected by measuring sample data cube with the lens cover on. The “white reference file” is generated by the SpectralView software using the white calibration tarp placed on the field before scanning. The white calibration tarp is assumed to have 56% reflectance. An example of calibration tarp is shown in Figure [3-2B](#). The reflectance cubes have 270 reflectance bands with wavelengths ranging from 396 nm to 996 nm. Figure [3-3](#) presents an example of the data cube before and after radiometric calibration.

As shown in the example, the raw data cube consists of raw digital numbers the sensor detected. The radiance data cube and the reflectance data cube are generated by the software successively. It is worth mentioning that the scene-white reference used to generate the reflectance data cube, which is generated by the white calibration tarp in the scene, is believed

to have 56% reflectance. And as can be seen from Figure 3-3C the white tarp has the reflectance values at all bands around 56%.

3.2.2 Noisy Band Removal

After the generation of reflectance data cubes, the reflectance spectra of every pixel are available. Before analyzing the spectra, noisy spectral bands are removed. The existence of noisy bands would lead to problems such as spectra dominated by several extreme values, which would degrade the performance of machine learning algorithms. Noise reduction is also considered as a necessary preprocessing step for the analysis of hyperspectral images. Because of having a similar application to the work of ([Moghimi et al., 2019](#)), a similar rule of noisy band removal is adopted. The first and last few bands are removed owing to their high noise. Furthermore, based on ([Moghimi et al., 2018](#)), the bands near O_2 and H_2O absorption regions are also removed due to large noise. Case in point, the first 20 bands (396nm-439nm) and the last 44 bands (900nm - 996nm) are discarded. Additionally, 7 bands around wavelength 760nm and 7 bands around wavelength 820nm are discarded because they are close to the noisy absorption bands of O_2 and H_2O , respectively. In the end, 192 spectral bands out of 270 bands are kept for the upcoming analysis. Figure 3-4 shows an example of spectra before and after removing noisy bands. It is observed that with the noisy bands discarded, the spectra become smoother.

3.2.3 Stitching and Making Orthomosaics

Around 40 images were taken to capture the whole field, so making a stitched image from all images would help understand the whole field better. Additionally, the stitched image suggests whether there is enough coverage of the field. Shown in Figure 3-5 is an example of ortho-mosaics generated by scanning images using the SpectralView software.

In this example, gaps are observed. The gaps between rows represent areas where no images were taken. The existence of these gaps indicates the lack of information at these places, which suggests modifications of the flight plan. Although the upcoming analyses would

be on every single image instead of analysis of the stitched data cube, this stitched data cube is used to locate every switchgrass plot for assigning pixel-level labels to the plots.

3.3 Switchgrass Detection

Before switchgrass genotypes classification, it is necessary to segment the switchgrass pixels from the images. The switchgrass pixels are detected by hyperspectral unmixing. Specifically, the SPICE algorithm is used to detect the endmembers in the scene automatically and unmix all images with abundance maps being generated. The endmembers of switchgrass are selected, and the corresponding abundance maps generate the final detection result. The analyses described in this section are conducted using MATLAB R2018a (MathWorks, Inc., Natick, MA, USA).

3.3.1 Sparsity Promoting Iterated Constrained Endmember Detection (SPICE)

To guarantee all endmembers in the scene are detected, several images considered to be inclusive of all endmembers are used for hyperspectral endmember detection. The initial number of endmembers is set to be 15, and the initial endmembers are randomly selected from the training data. As presented and discussed previously in Chapter 2, the sparsity-promoting term added to the ICE algorithm would prune the endmembers with contribution smaller than the threshold and hence learn the set of endmembers automatically.

Several experiments are repeated to ensure that the detected endmembers are consistent. During some experiments, a great number of endmembers are detected. In those cases, the detected endmembers are used as initial endmembers, and the optimization process is done again for further pruning. The detected endmembers are then used to unmix all the remaining images and generate abundance maps. Figure 3-6 shows the detected endmembers for both fields.

For the hyperspectral data collected from the GWAS field, endmember 5 is chosen as the switchgrass endmember, and the switchgrass detection result is generated thereby. Results for one image is shown in Figure 3-7. Analyzing the endmembers presented in Figure 3-6B and the abundance maps in Figure 3-8 of the data collected from the STND field, it is evident that

further pruning is needed. The detected 10 endmembers are used as initial endmembers, and a larger pruning threshold is used to ensure pruning all endmembers that are unnecessary. The SPICE algorithm converges to 8 endmembers and the result is shown in Figure 3-10.

3.3.2 Switchgrass Detection Generation

With detected spectral signatures using SPICE, signatures representing switchgrass are selected, and the corresponding abundance maps are aggregated. The switchgrass pixels are segmented by thresholding the final abundance map images with a threshold of 0.5. Examples of switchgrass detection results are shown in Figure 3-11 and Figure 3-12.

It can be seen from the results that the SPICE algorithm can detect all the endmembers in the scene and learn the number of endmembers automatically. And by thresholding on the abundance map generated by hyperspectral unmixing, most of the switchgrass pixels can be segmented out consistently. However, some pixels have false-positive errors, and some have false-negative errors. After a close examination, it is believed that the main reason for the false-positive errors is that the background in the scene is also the vegetation. It is not surprising that the spectra for switchgrass and background grass look similar. Likewise, the fact that the switchgrass is similar to background grass causes false-negative errors. Additionally, some switchgrass pixels being in the shadow makes the spectra of those pixels a mixture of switchgrass and shadow, thus they cannot be recognized as switchgrass.

3.4 Switchgrass Genotype Classification

The spectra of detected switchgrass pixels are ready for classification. There are two steps in classification: (1) reducing dimensionality by the Siamese network and (2) assigning class labels by the k-nearest neighbor classifier. Cross-validation is designed to show the generalization of the network. Specifically, several randomly selected images are training images, and the rest of images are test images. As described above in Chapter 3, 12 images are randomly chosen for every fold to ensure the training data are not hand-selected. As an illustration, two images containing pixels of all classes are chosen. Furthermore, to ensure the class balance of training data, a down-sampling method is adopted for classes with samples

more than needed, while an over-sampling method called Synthetic Minority Over-sampling Technique (SMOTE) ([Chawla et al., 2002](#)) is applied to the training data to generate artificial samples for classes with insufficient training samples. Only spectra for switchgrass pixels are used for classification.

Except for the two mentioned tasks, another task to be done before both of them is generating precise pixel-level ground-truth. Either of the subtasks needs precise pixel-level labels, so the ground-truth is generated first. The ground truth generation is completed using MATLAB R2018a (MathWorks, Inc., Natick, MA, USA), and the subsequent dimensionality reduction, as well as classification, are completed using Python programming language (Python Software Foundation, <https://www.python.org/>).

3.4.1 Ground Truth Labeling

The training of the Siamese network and the classifier need precisely labeled pixel-level ground truth. A method that is a combination of manual labeling and automated detection is proposed to accomplish this task. As shown in Figure 3-13, the ground truth is first generated by tracing the boundaries of the switchgrass plots and assigning the name of the switchgrass genotype to them. This ground truth generation method assigns all pixels inside the traced boundary one of the switchgrass genotypes. In other words, this method assumes that all pixels inside the traced boundary are switchgrass pixels. However, this assumption does not hold because some plots are not fully covered by switchgrass - there are regions of bare soil and large shadows. The switchgrass detection result of the SPICE algorithm is used together with manual labeling to address this issue. Two ground-truth maps generated from manual labeling and the SPICE algorithm are combined to find the switchgrass pixels, and the class labels for pixels inside every plot are assigned by manual labeling, to put it another way.

As discussed above, filtering the abundance maps for data collected from the STND field with a threshold of 0.5 would lead to false-negative errors, that is to say, some switchgrass pixels would be considered to be non-switchgrass pixels. To make sure finding all switchgrass pixels, thus the ground truth can be as precise as possible, a smaller value of threshold 0.3 is

used to filter the abundance maps so that most of those pixels that cannot be switchgrass are discarded. Then the manual labeled ground truth is used to generate a ground truth map to filter out those pixels not belonging to the switchgrass plot. Finally, the combination of these two results generates the ground-truth, which is used in the training of the Siamese network and the classifier. Figure 3-14 shows an example of the processing protocol.

3.4.2 Dimensionality Reduction

The Siamese network that reduces the dimensionality by learning a nonlinear mapping is proposed to reduce the dimensionality in this work. Theoretically, it improves the performance of classification while reducing the dimensionality. Additionally, principal component analysis is the baseline method in this work. Principal component reduces dimensionality by learning a linear transformation. Furthermore, as a comparison of dimensionality reduction methods that maintain the maximum class separability, another linear dimensionality reduction method, linear discriminant analysis, is also adopted.

3.4.2.1 Dimensionality reduction by learning an invariant mapping

The dimensionality reduction method *Dimensionality Reduction by Learning an Invariant Mapping* mentioned in Chapter 2 is used to reduce dimensionality. Figure 3-15A shows the architecture of the network used in this work. Network with this architecture is also known as the Siamese network, where both data points of the pair of data points are processed by the same network independently. It should be noted that the training of the network relies on pairs of inputs: a binary label is generated based on the definition of the contrastive loss function and is compared with the ground truth label. However, when the Siamese network is trained and ready to be used for dimensionality reduction, it is unnecessary to use pairs of input.

Fully connected networks are used, and the desired output dimension d is determined by the number of neurons of the output layer. Classification performances with various values of d are compared. As shown in Figure 3-15B, different output dimension d are obtained by using different network architecture.

In this work, the neighborhood relationships are obtained from the pixel-level ground-truth. To be specific, pixel-level class labels are obtained in advance by manual labeling described above. During the network training process, labels “0” are assigned to the pairs of pixels belonging to the same class of switchgrass genotype, and labels “1” are assigned to the pairs of pixels belonging to different classes of switchgrass genotypes. The number of neurons in the output layer of the Siamese network determines output dimension. Experiments of different numbers of dimensions to be kept are done by varying the number of neurons in the output layer, and the results are compared.

3.4.2.2 Dimensionality reduction by principal component analysis transformation

In this work, Principal Component Analysis is the baseline method. The same training data for the Siamese network is used for learning the transformation of PCA. The results of PCA are compared to the results of the proposed method.

3.4.2.3 Dimensionality reduction by linear discriminant analysis

Linear Discriminant Analysis (LDA) reduces the dimensionality from the original number of features to C-1, where C is the number of classes. In this work, the number of classes is 6, so the LDA is used to reduce the dimensionality to 5. The experiments of reducing dimensionality to 5 using the other two methods are also performed to compare the performances.

3.4.3 K-Nearest Neighbor Classifier

The dimensionality reduced spectra are then classified using a multi-class K-nearest neighbor classifier. The KNN is trained using the same data as used for the Siamese network training. K is set to be the same as a common choice of five without further examinations thus far.

Because different dimensionality reduction outputs are classified using the same classifier, the classification performance evaluates the efficiency of dimensionality reduction methods. In this work, performances of classifiers are evaluated by confusion matrices and receiver operating characteristic (ROC) curves. A confusion matrix is a specific table layout allowing

visualization of the performance of classification algorithms ([Stehman, 1997](#)). Typically, each row of the matrix represents the number (or percentage) of instances predicted to belong to a class while each column represents the number (or percentage) of instances actually belonging to a class. A ROC curve is a plot demonstrating the diagnostic ability of a binary classifier system when varying the discrimination threshold ([Fawcett, 2006](#)). For multi-class classification problems with K classes, K ROC curves are displayed simultaneously, where every curve represents the performance of the binary classifier. Additionally, the classification results are also shown on original images as a comparison to the ground truth images.

3.5 Summary of Proposed Approach

A flowchart that includes the complete processing and analyzing strategy is shown in Figure [3-16](#). Because of the special training technique of the Siamese network (taking pairs of input), the training and test scheme differs a little from each other. The differences between training and test are indicated in the flowchart using different colors.

A cross-validation experiment is designed to justify that the proposed method is robust and does not rely on the training data. The pseudocode below shows the whole training process. In every fold, the training images are selected randomly, and the rest images are used as test images. The variance among different folds indicates the generalization of the proposed method. The details will be presented and discussed in Chapter [4](#).

Algorithm 3.1. *Experiment design of cross validation*

```

1: for every fold do
2:   randomly select 12 images as training images.      ▷ Ensure that there are 2 images for
   every class of the 6 classes.
3:   for every class do
4:     downsample if number of pixel greater than set; oversample if number of pixel
   smaller than set                                ▷ Ensure balanced training samples for every class
5:   end for
6:   split the training data: 80% of training data are training data:  $X_{train}^D$ ; 10% are
   validation data:  $X_{valid}^D$ ; 10 % are test data:  $X_{test}^D$ .
7:   while not converge nor not reaching maximum training epoch do
8:     Train the Siamese network with pairs of input generated from  $X_{train}^D$ , optimized by
   Adam and evaluated by contrastive loss function.
9:     Evaluate the performance using  $X_{valid}^D$  every 200 training batches.
10:    if performance of validation not improved after defined number of epochs then
11:      convergence of training reached
12:      model_S trained
13:    end if
14:  end while
15:  Evaluate the performance on  $X_{test}^D$ ;
16:  Reduce the dimensionality of training data using the trained Siamese network:  $X_{train}^d$ ;
17:  Train the KNN model using  $X_{train}^d$  and corresponding class labels.
18:  model_K trained
19:  for every test image do
20:    Identify the switchgrass pixels
21:    Reduce dimensionality using model_S
22:    Classify every pixel using model_K
23:  end for
24: end for

```

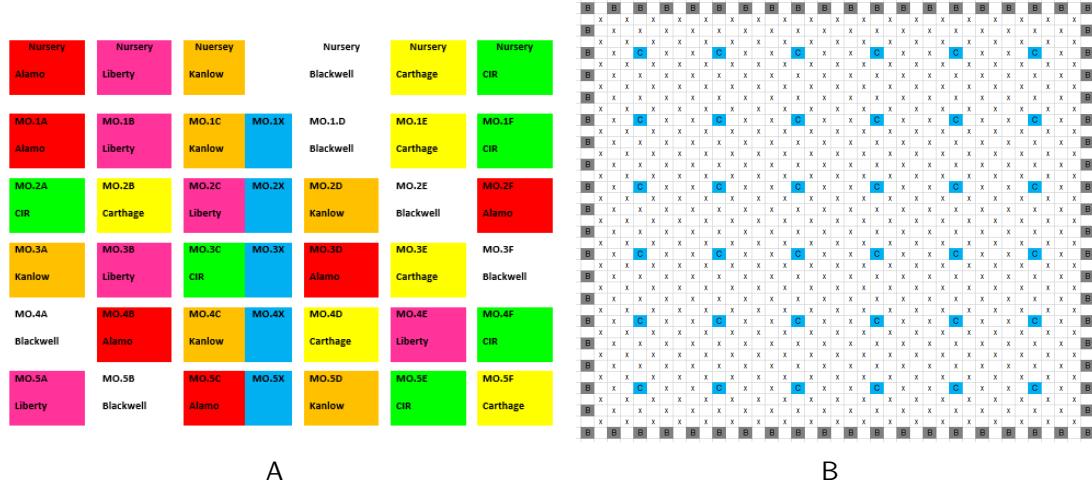


Figure 3-1. The planting layout of two planting techniques. A) The planting layout of the STND planting. Switchgrass of same class are plant next to each other in a plot. B) The planting layout of the GWAS planting. "B" represents a border plant, "X" represents an experimental plant, "C" indicates a check plant and blank spots are empty spots.

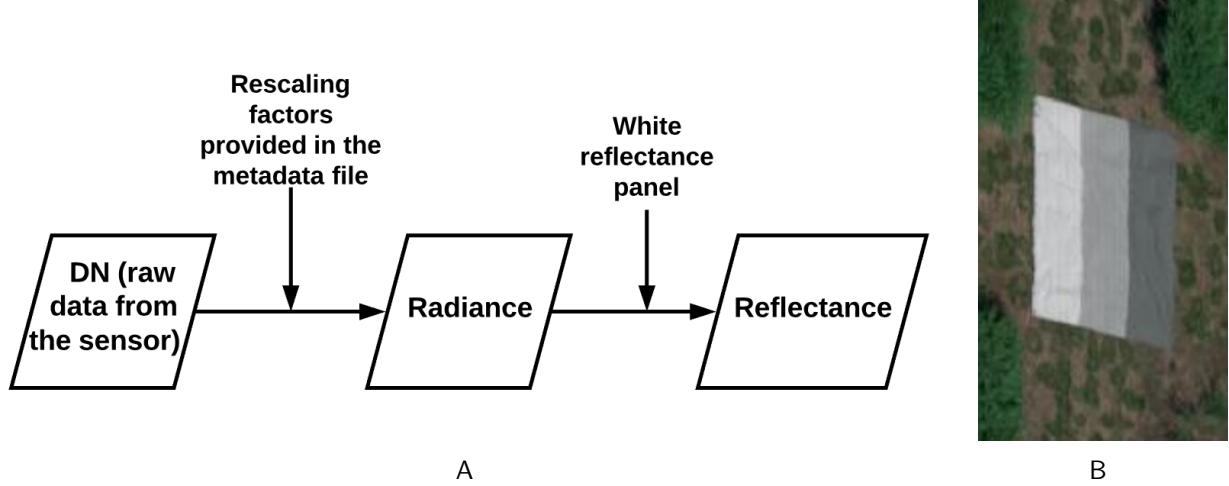


Figure 3-2. Calibration processing steps and white tarp used for calibration. A) Calibration processing steps adopted in this work. The SpectralView software is used. Courtesy of author. B) An example of calibration tarp used in this work. The brightest part of the tarp is used to generate the white reference file and is assumed to have 56% reflectance. B) Photo courtesy of Brandon Davis. June 25th, 2019. Columbia, MO.

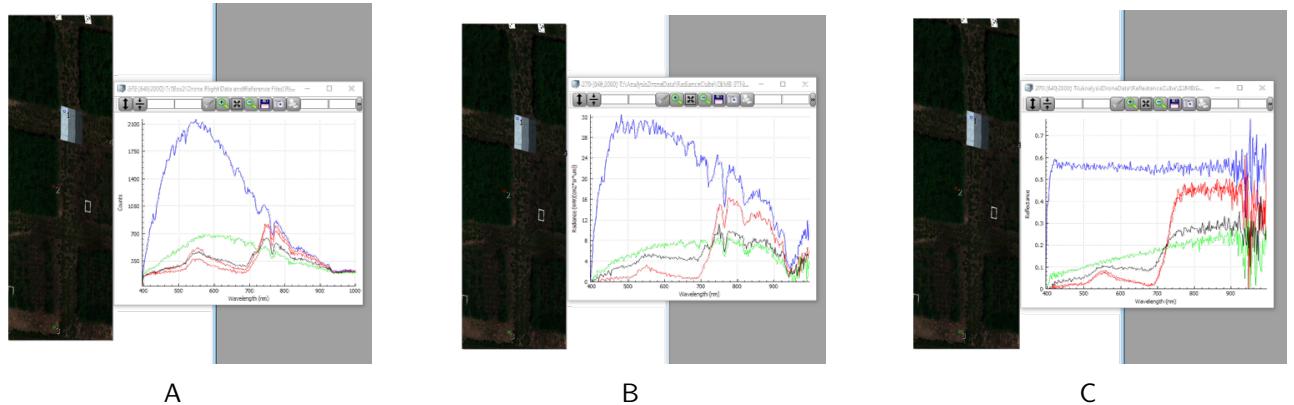


Figure 3-3. Radiometric Calibration. A) Raw data cube opened in the SpectralView software, with the raw digital numbers for 272 wavelengths shown for selected points; B) Radiance date cube opened in the SpectralView software, with calibrated radiance values for 270 bands shown for selected points; C) Reflectance data cube opened in the SpectralView software, with selected spectra shown for selected points. The calibration tarp is also shown in this figure.

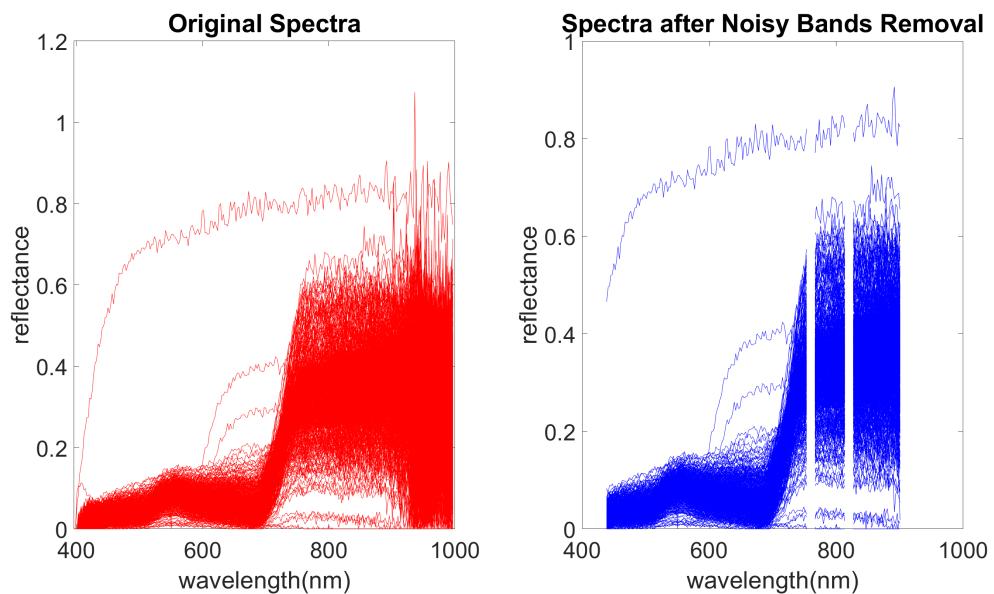


Figure 3-4. Sample spectra before and after noisy band removal.

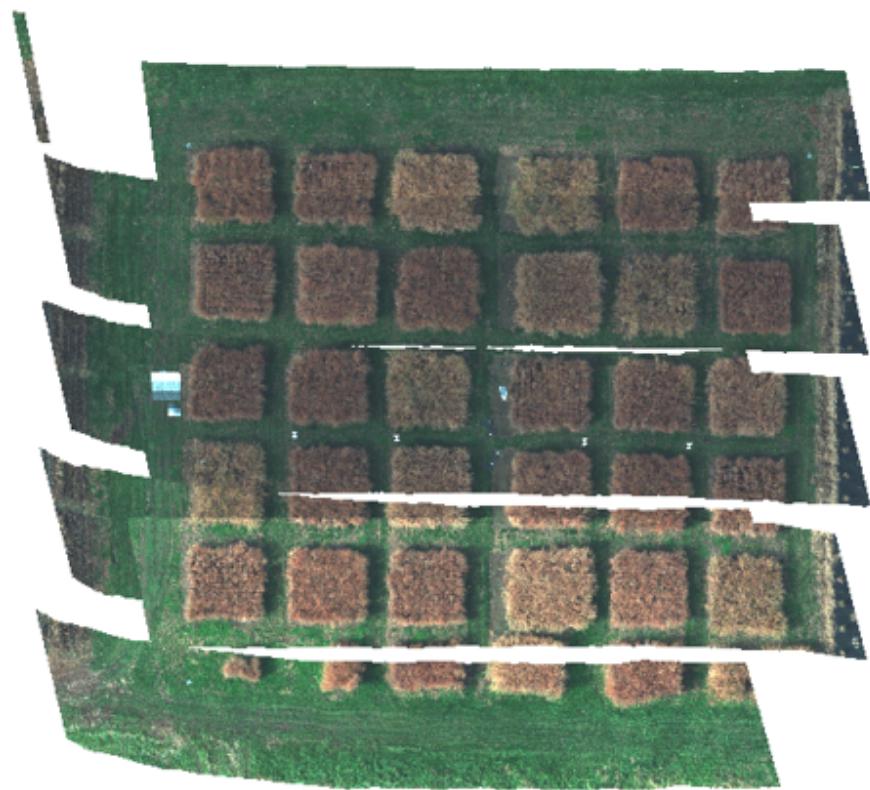


Figure 3-5. Stitched images from one example scanning of the STND switchgrass field. As can be seen, the white gaps between two rows are areas without images. The flight plan should be modified to make sure the images cover the whole field.

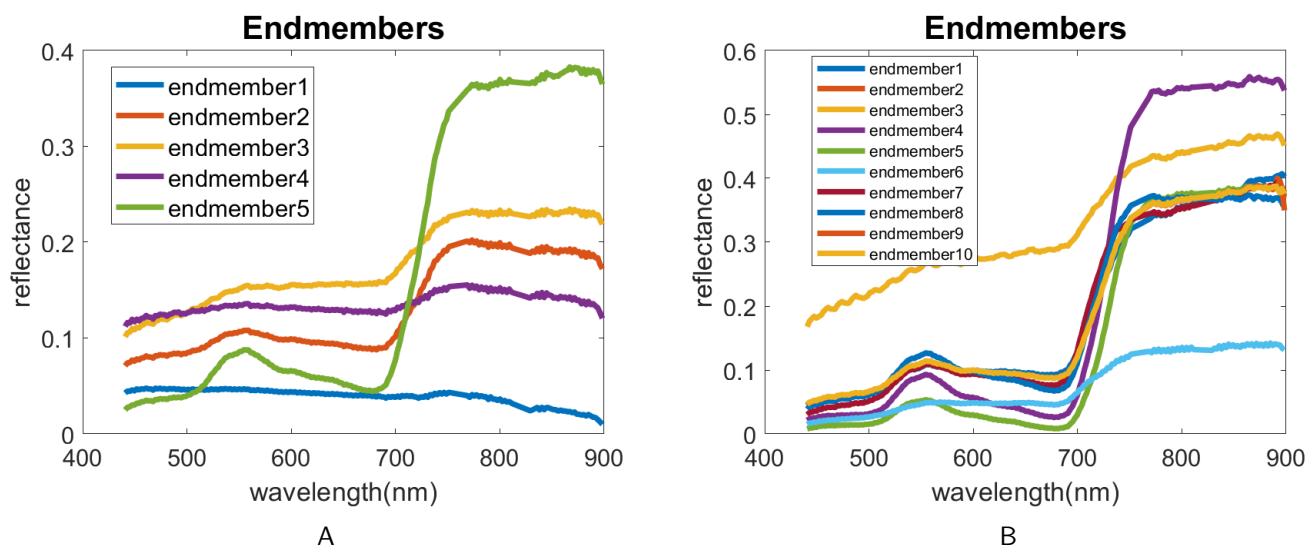


Figure 3-6. Detected endmembers using SPICE algorithm. A) Detected endmembers using SPICE algorithm for the GWAS field; B) Detected endmembers using SPICE algorithm for the STND field.

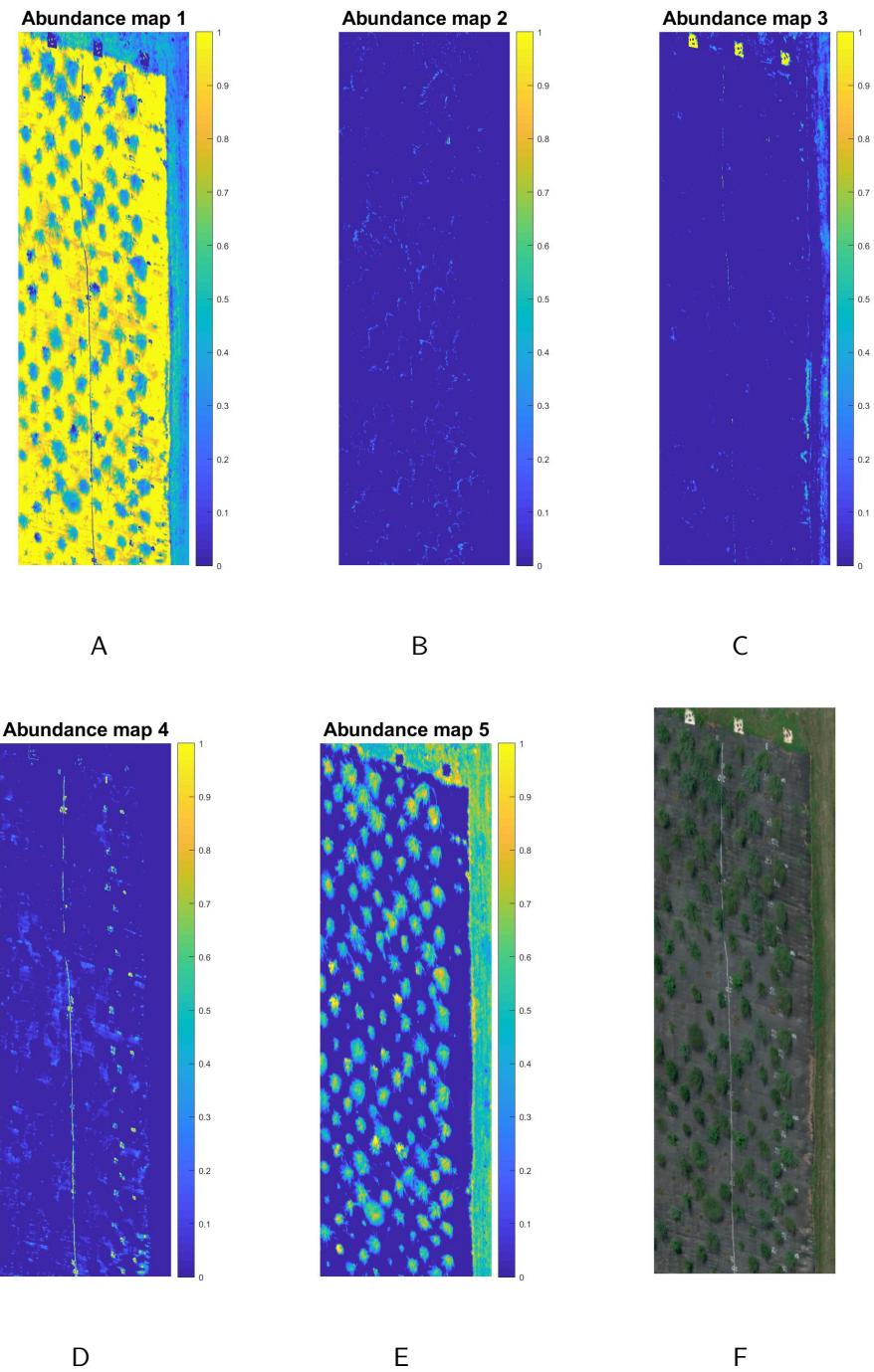


Figure 3-7. Abundance maps and the original hyperspectral image for data collected from the GWAS field. A)-E): Abundance maps correspond to detected endmembers for GWAS field; F) Original hyperspectral image visualized with red, green and blue bands. The abundance map corresponds to endmember 5 shown in E) is considered as the abundance map corresponding to the endmember representing switchgrass. F) Photo courtesy of Brandon Davis. June 25th, 2019. Columbia, MO.

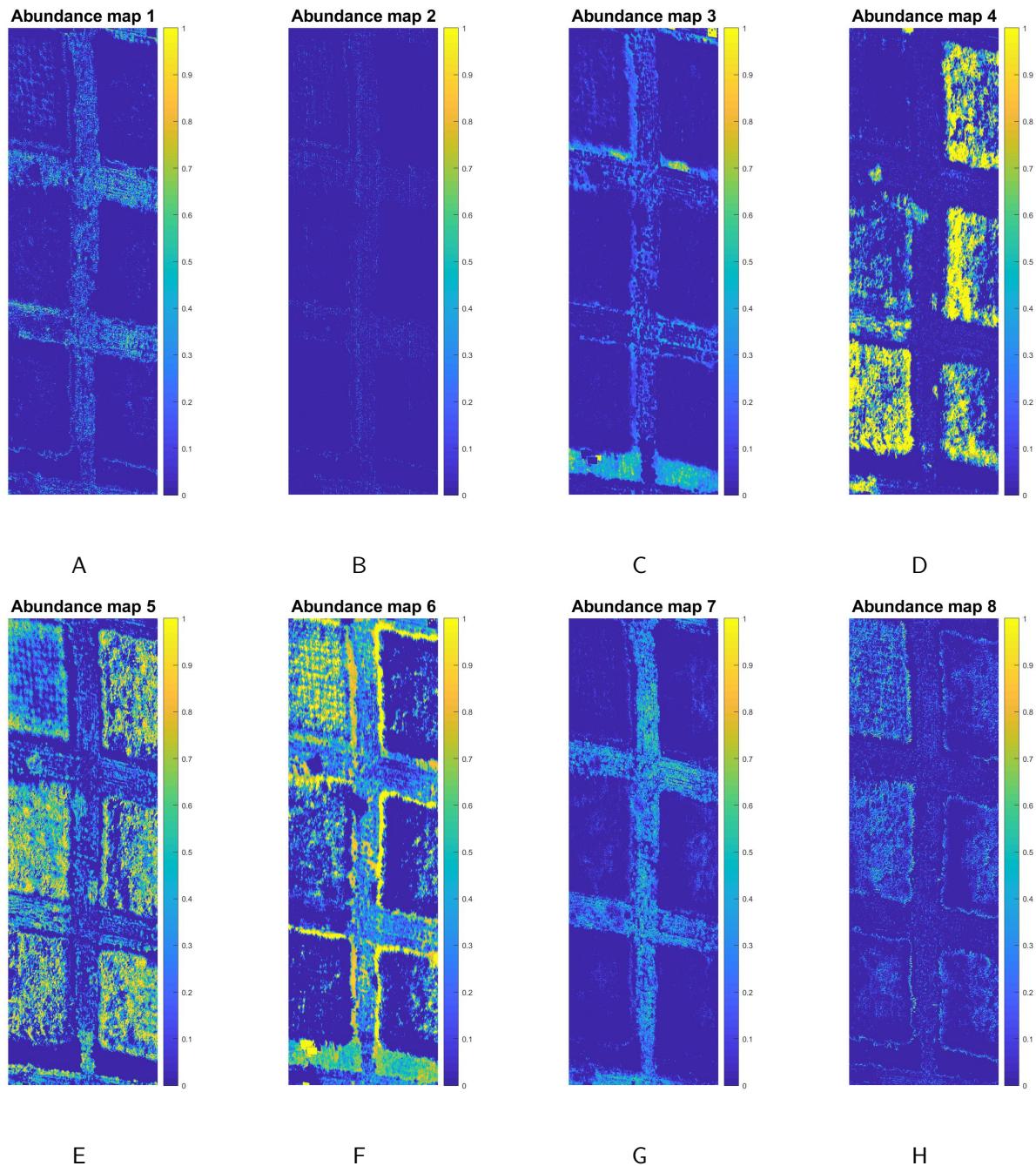


Figure 3-8. Abundance maps and the original hyperspectral image for data collected from the STND field. A)-J) Abundance maps correspond to 10 detected endmembers; K) Original hyperspectral image visualized with red, green and blue bands. K) Photo courtesy of Brandon Davis. June 25th, 2019. Columbia, MO.

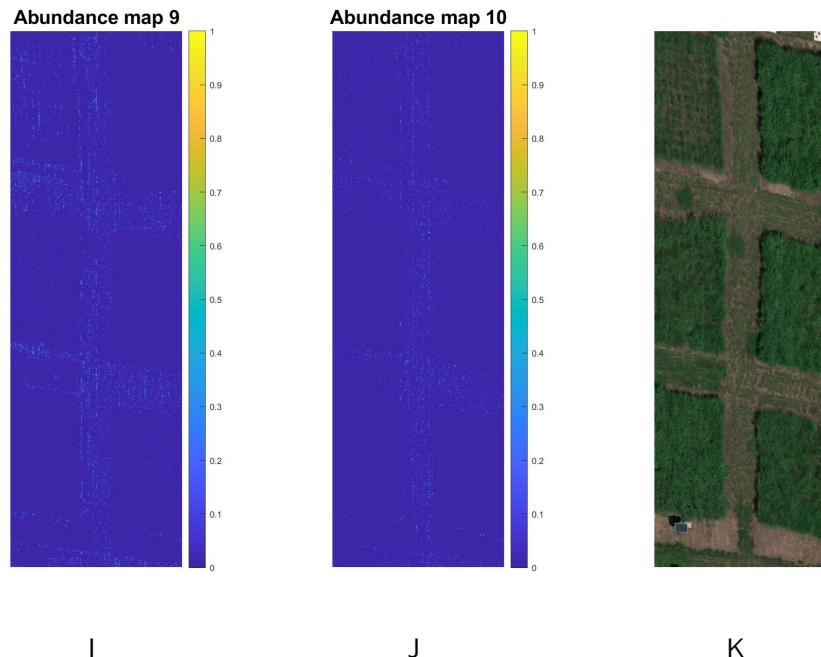


Figure 3-8. *cont.*

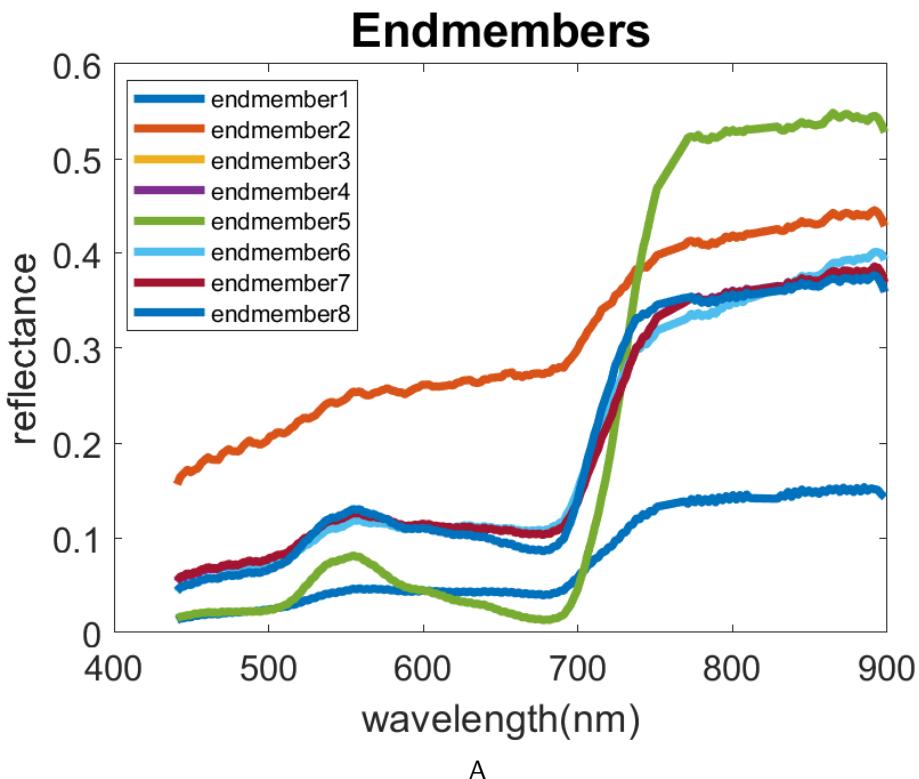


Figure 3-9. Final endmembers and original hyperspectral image shown in RGB for data collected from the STND field.

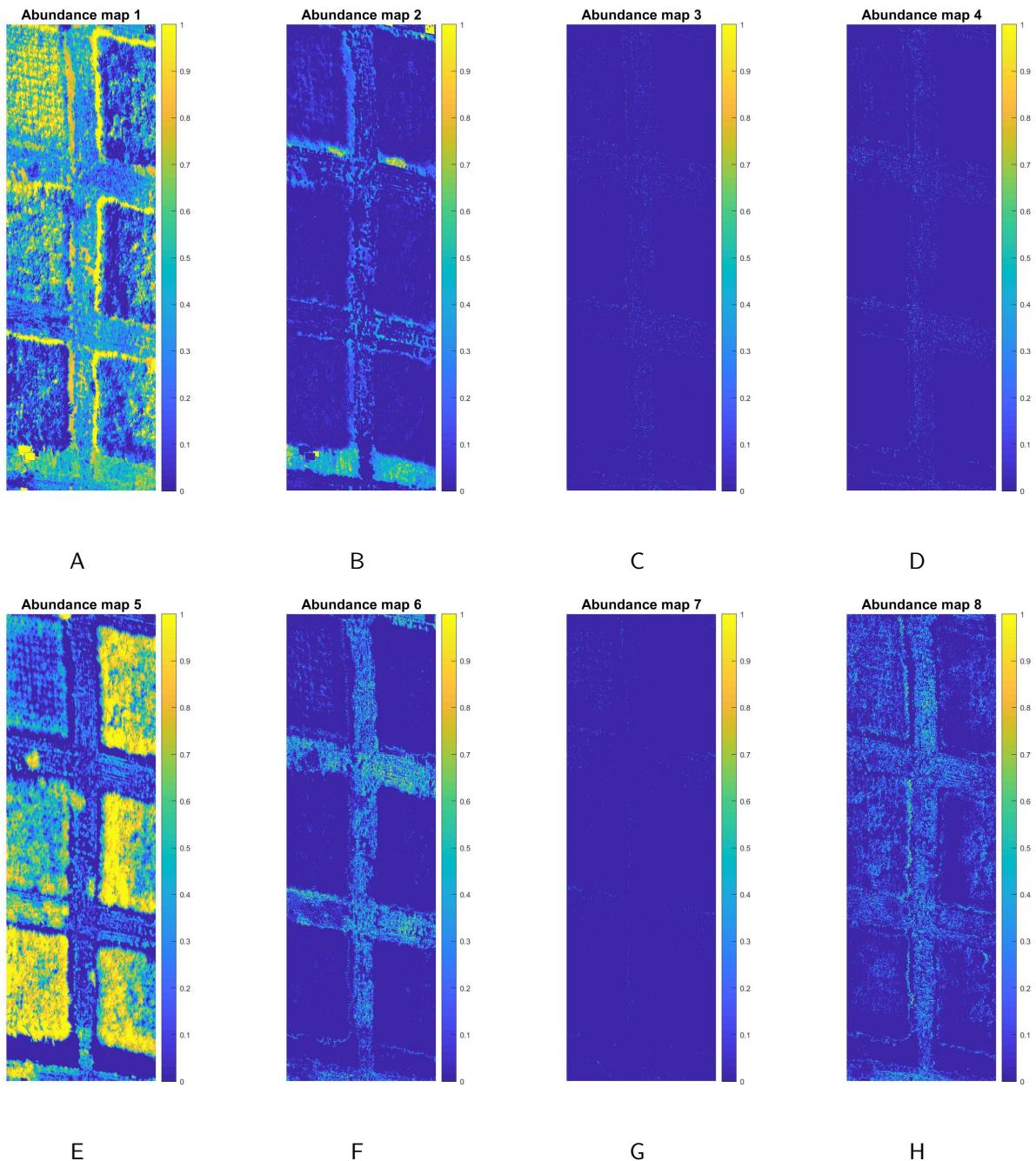


Figure 3-10. Abundance maps correspond to the final set of endmembers for data collected from the STND field.

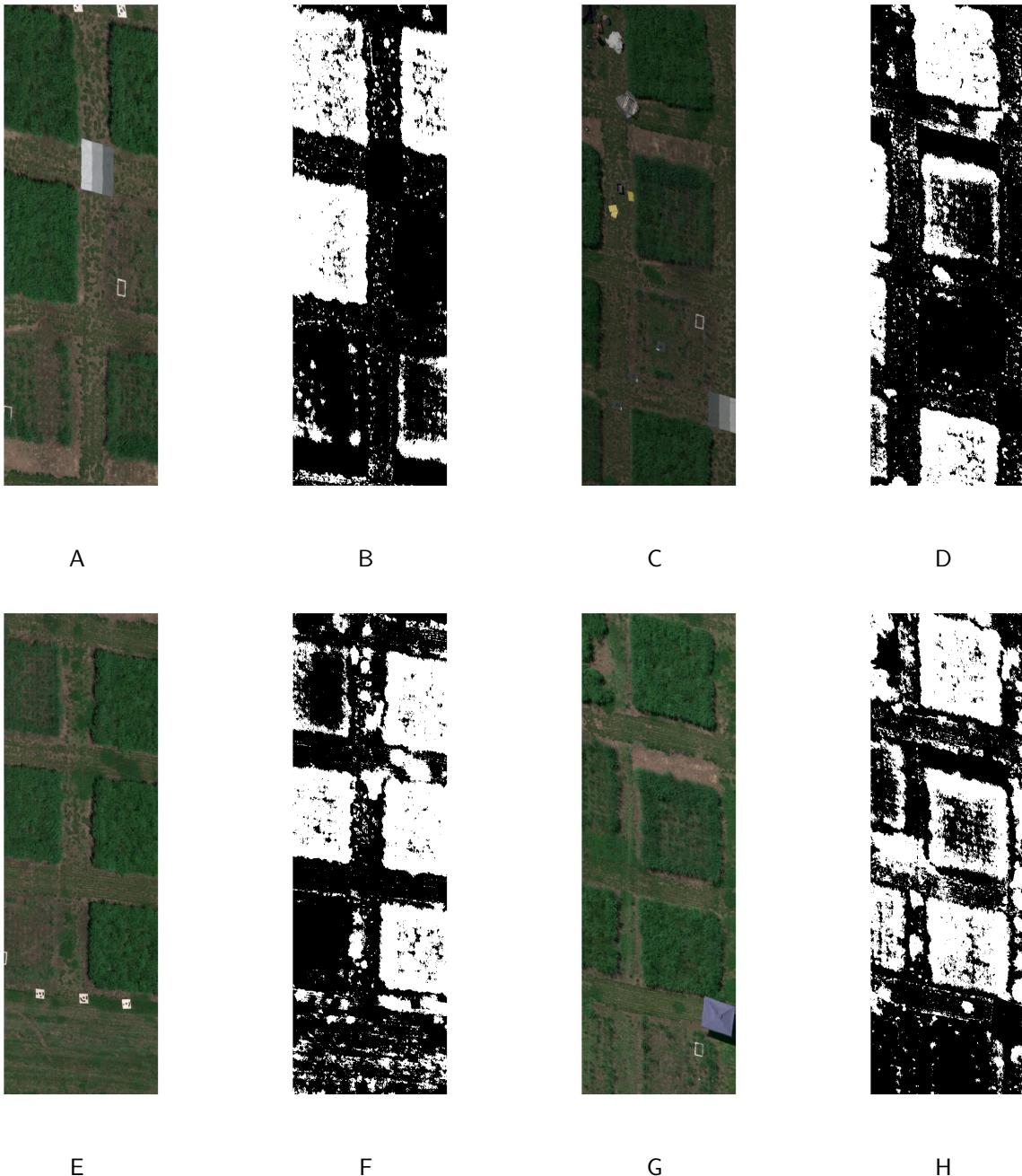


Figure 3-11. Selected switchgrass detection results compared to the original image for data collected from the STND field. A), C), E) and G) photo courtesy of Brandon Davis. June 25th, 2019. Columbia, MO.

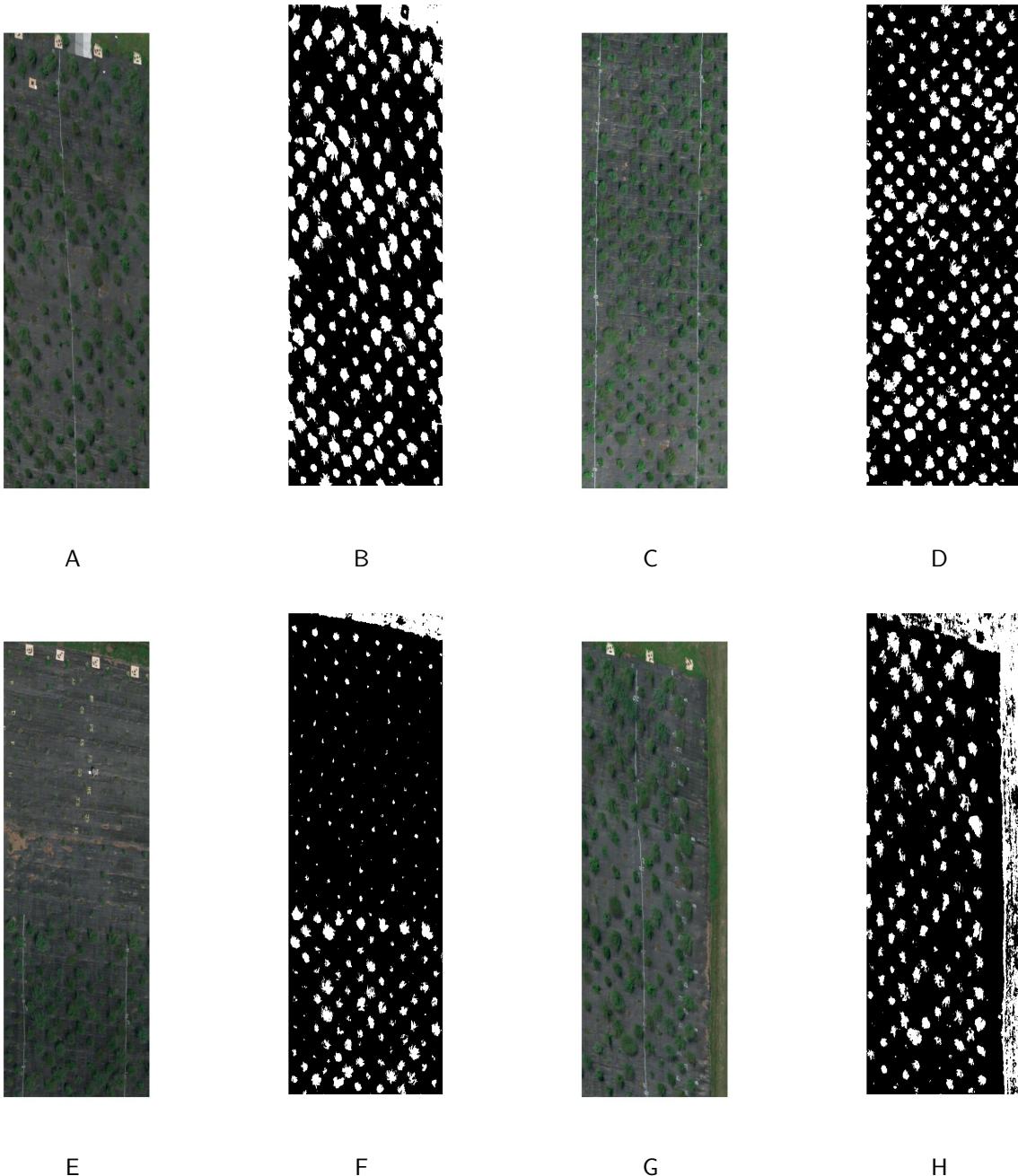


Figure 3-12. Selected switchgrass detection results compared to the original image for data collected from the GWAS field. A), C), E) and G) photo courtesy of Brandon Davis. June 25th, 2019. Columbia, MO.

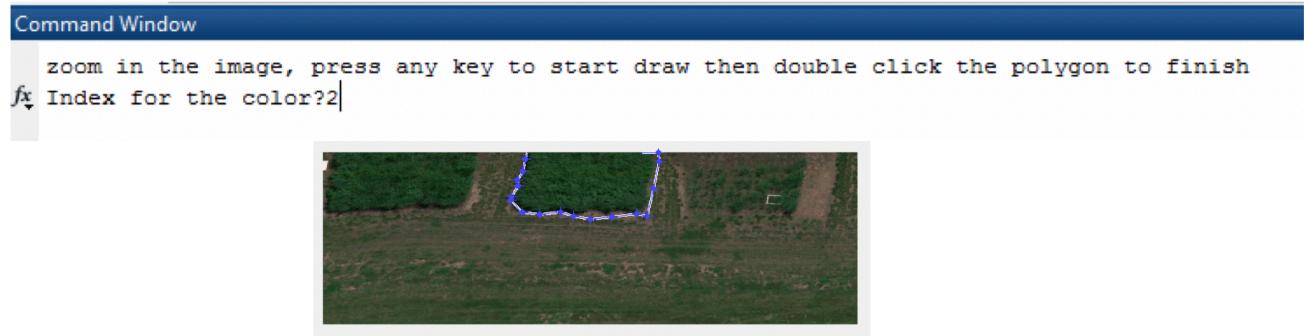


Figure 3-13. Example of manual labeling the ground truth interactively using MATLAB.
Courtesy of author.

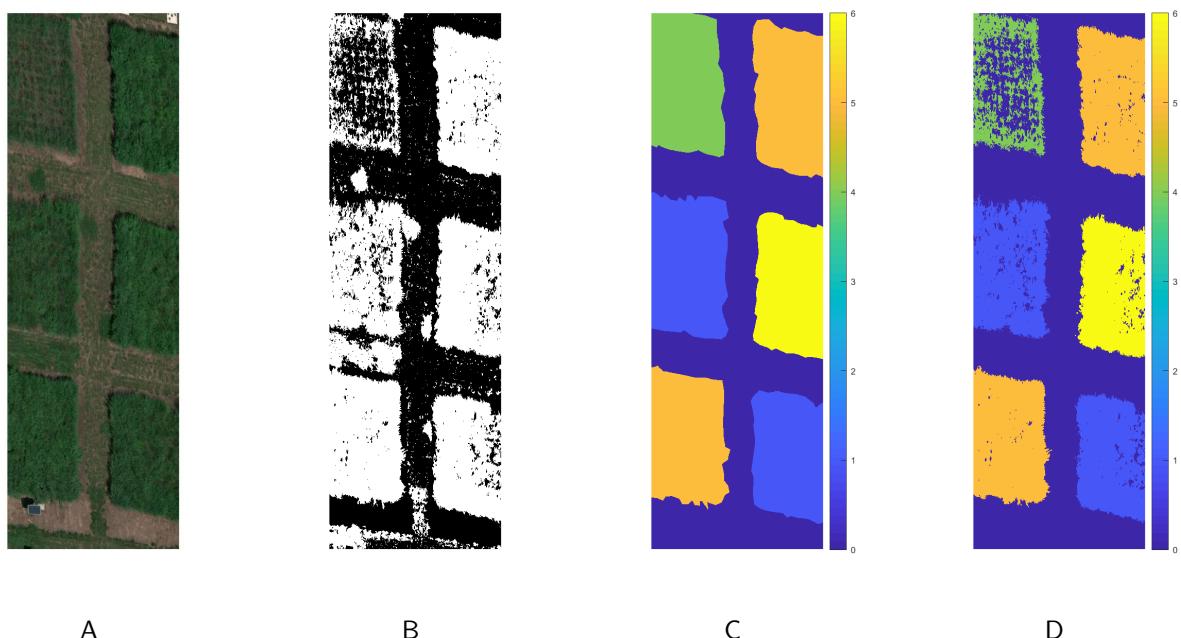
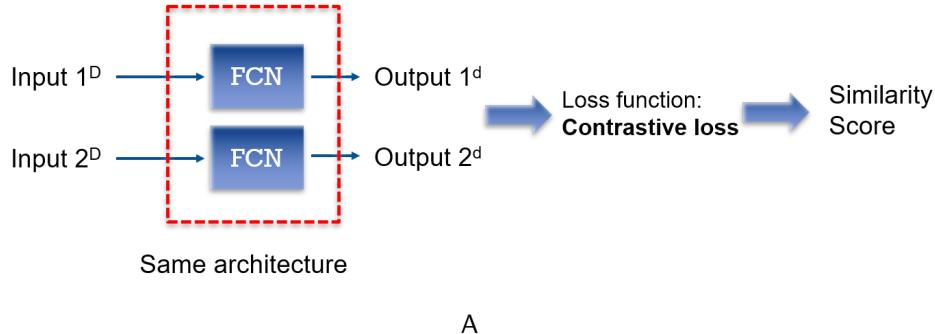
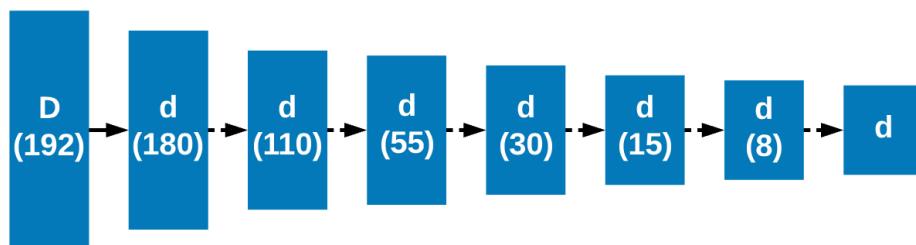


Figure 3-14. Abundance maps and original hyperspectral image of the STND field. A) Original hyperspectral image visualized in RGB space; B) Switchgrass detection result by thresholding the aggregated abundance maps from SPICE unmixing; C) Ground truth generated by manual labeling; D) Final switchgrass detection result by combining results of B) and C). A) photo courtesy of Brandon Davis. June 25th, 2019. Columbia, MO.



A



B

Figure 3-15. The Siamese network architecture and details of network layers. A) The Siamese network architecture. It should be noticed that the two networks shown in the figure are actually one; it is shown this way to indicate how the network deal with a pair of inputs separately. B) The details of the neurons in every layer. Fully connected neurons are used. D represents input dimension and d represents output dimension. The last d can be any number smaller than 8. The network architecture becomes much complicated when a lower output dimension is desired.

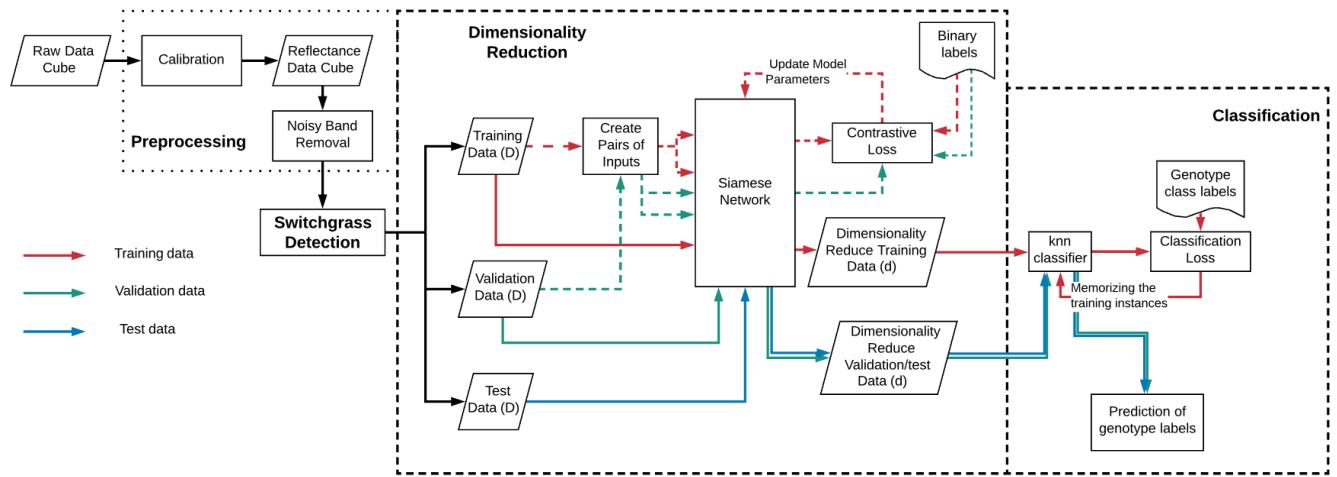


Figure 3-16. The flowchart of the adopted approach. The blue arrows illustrate the training steps of the Siamese network; the red arrows illustrate the training steps of the KNN classifier; the green arrows illustrate the test/validation steps of the Siamese network followed by the KNN classifier. During training of the Siamese network, the input data is always a pair of data points with label either 0 or 1; during test/validation of the Siamese network, the input data is always single data point, and the output is dimensionality reduction result.

CHAPTER 4 RESULTS

In this chapter, the experiment results of dimensionality reduction and classification are presented and discussed. As stated in Chapter 3, the dimensionality of the acquired data is reduced by three methods before classification. The reduced dimension spectra are then used to train the multi-class k-nearest neighbor (knn) classifier. Receiver operating characteristic (ROC) curves and confusion matrices are metrics to evaluate the final classification performance in this work. The classification results are also presented as visualizations on top of the original images. The effectiveness of dimensionality reduction is evaluated by the classification performance. The results are then compared and discussed.

4.1 Dimensionality Reduction using The Siamese Network

The design of cross-validation has been stated in Chapter 3, where for every fold, 12 images are randomly selected to form training data. The switchgrass pixels are then detected, and the spectra for switchgrass pixels make up to the set of training spectra. The set of training spectra is then divided into three parts: the training set, the validation set, and the test set. The validation set is used for the parameter tuning of the Siamese network model, and the test set is used for model performance evaluation. Then the trained Siamese model reduces the dimensionality of the test images.

Figure 4-1 shows an example of the learning curve when training the Siamese network. It is seen that the loss values of both the training data and the validation data are initially large, and decrease afterward. The variation of the loss values of the validation data is greater than that of the loss values of the training data. An early-stopping criterion is utilized to determine whether the training process can be stopped earlier.

Different numbers of dimensionality to be kept are also experimented on to find an optimal number of dimensionality. The classification accuracy is used to evaluate the efficiency of dimensionality reduction. Figure 4-2 shows that reducing the dimensionality to 15, 30, or 55 by the Siamese network yields comparatively good classification results with a small variance.

For this reason, the comparison experiments that reduce dimensionality using PCA only reduce the dimensionality to these values of d .

4.2 Dimensionality Reduction by Principal Component Analysis Transformation

As stated in Chapter 3, the baseline dimensionality reduction method adopted in this work is principal component analysis (PCA). The PCA models are trained using the same data in each fold, like ones used to train the Siamese Network.

Denote the dimensionality of the output space as d , and when using PCA to reduce dimensionality, d is the number of principal components kept. As explained above, only cases with $d = 15$, $d = 30$ and $d = 55$ are experimented on. When examining the relationship of the explained variance and the number of components kept, these values of the output dimension are proven to be reasonable. As can be observed from Figure 4-3, the first 15 components contain over 97% of the variance, and the first 55 components contain over 99% of the variance of the original data.

4.3 Dimensionality Reduction by Linear Discriminant Analysis

As stated in Chapter 3, another experiment is to reduce dimensionality to C-1 using linear discriminant analysis (LDA). There are 6 classes in this problem, so the dimensionality is reduced to 5. Experiments to reduce dimensionality to 5 using principal component analysis transformation, as well as the Siamese network, are also done to compare the result.

4.4 Results Comparison

Final classification results when adopting different dimensionality reduction methods are presented in forms of confusion matrices, ROC curves. Figure 4-8 shows examples of results in the form of images.

Figure 4-4, Figure 4-5 and 4-6 show the differences in classification performance when adopting different dimensionality reduction algorithms, with the dimensionality of the output space being 15, 30 and 55 respectively, or without dimensionality reduction. Comparing the results with dimensionality reduced by the Siamese network and PCA, it is found that reducing the dimensionality of original data by the Siamese network guarantees better classification

performances evaluated by either confusion matrices or ROC curves. Indeed, the Siamese network enhances the ability to be classified correctly for every switchgrass genotype. Although the classification result with input reduced dimensionality by the Siamese network does not outperform that of using the original full spectra, it is remarkable that they have comparable results. And for some switchgrass genotypes Alamo and Kanlow, the Siamese network improves the ability to be classified correctly of them.

Figure 4-7 shows the differences in classification performance when adopting three different dimensionality reduction algorithms to reduce the dimensionality to 5. It is found that dimensionality reduction with supervised algorithms, the Siamese network and LDA, have better performances than the unsupervised method, PCA. When comparing two supervised methods, dimensionality with either a linear or non-linear mapping, has comparable results.

Some shared characteristics of the classification performances with all three of the dimensionality reduction methods are observed. It is found that switchgrass genotypes Alamo and Kanlow have better classification performance compared to other genotypes regardless of the dimensionality reduction method being used. The switchgrass genotype having the third-best classification performance is Liberty. The other three genotypes, Carthage, CIR, and Blackwell, do not have not as good results - they tend to be misclassified to any of the three with almost equal ratios.

To explain the result, the original spectra for all genotypes and the dimensionality reduction results using both of the methods are checked and the mean and variances are shown for every genotype as errorbars. As can be seen from Figure 4-9, the spectra of Alamo, Kanlow and Liberty are more separable especially at the near-infrared region, while the spectra of Carthage, CIR and Blackwell appear close to each other and with a large variance. Although the mean value for Liberty is different from others, the separability of it is believed to be degraded. As can be observed from the figure, the variances of the Carthage, CIR and Blackwell's spectra are so large that the spectral of Liberty are included inside their distribution. The outputs of dimensionality reduction shown in Figure 4-10A and Figure

[4-10B](#) explain the final classification performance: the Siamese network learns a mapping to a manifold where the representations for Blackwell, CIR and Carthage are similar to each other, while the representations for Kanlow and Alamo show some separability. This observation is consistent with the classification results. Figure [4-10D](#), Figure [4-10E](#) and Figure [4-10F](#) show the visualization of dimensionality reduction results using PCA. And it appears that only first several principal components are significant to show the difference.

It is interesting to find that all three switchgrass genotypes having better performance in classification are lowland switchgrass genotypes. The other three genotypes belong to upland switchgrass genotypes. One possible explanation for Alamo and Kanlow's better classification performance than Liberty is that both Alamo and Kanlow die during the winter season and green up later than other genotypes in the spring. This is also the reason the original spectra of Alamo and Kanlow are different from other genotypes'.

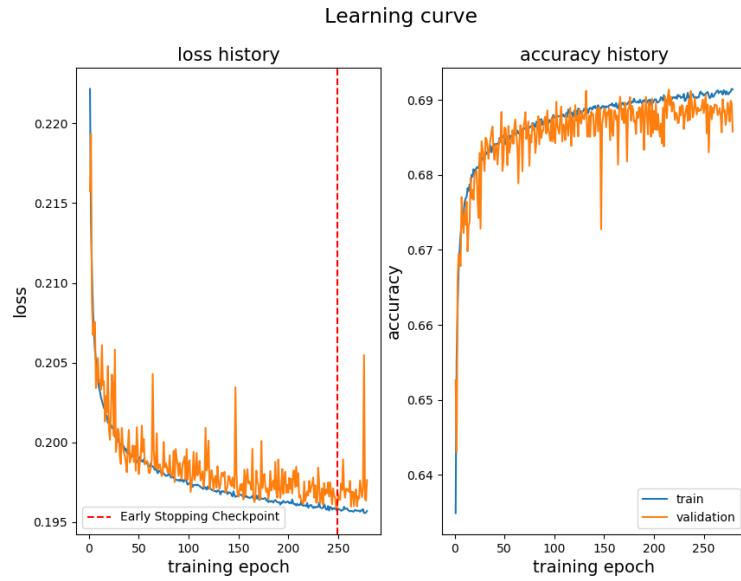


Figure 4-1. Learning curve for Siamese network training. An early stop check based on the validation loss is adopted to monitor the training process and stop training when there is no improvement of the network performance.

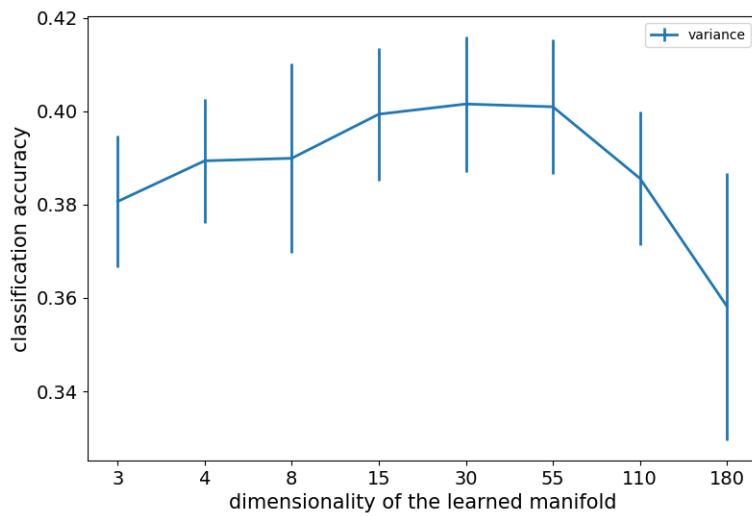


Figure 4-2. Errorbar of classification performance versus the dimensionality of the learned manifold. The Siamese network is used for dimensionality reduction.

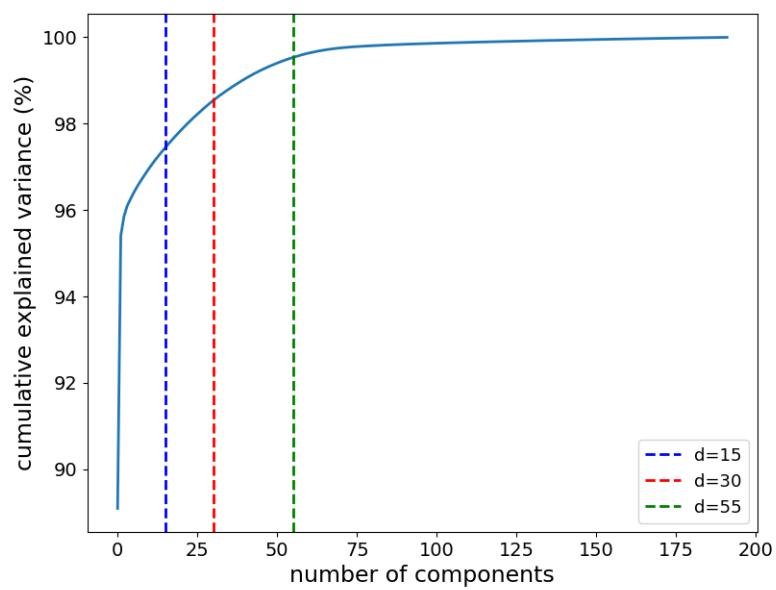
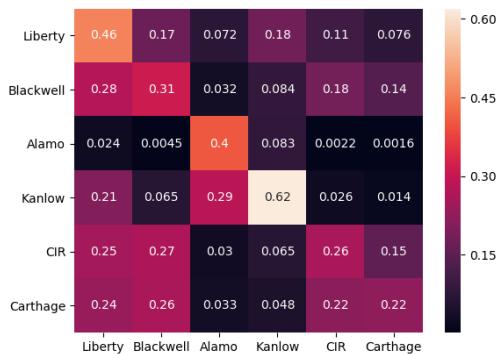
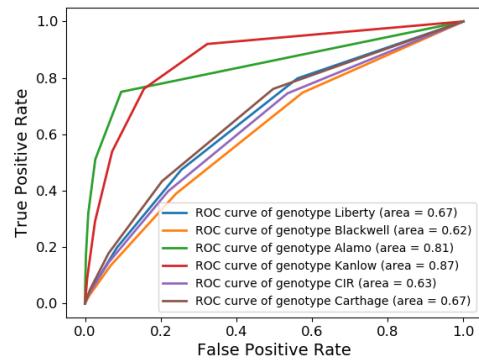


Figure 4-3. The cumulative explained variance ratio against the number of components of PCA. As can be seen from the figure, first 15 components contain over 97% of the variance, and first 55 components contain over 99% of the variance of the original data.



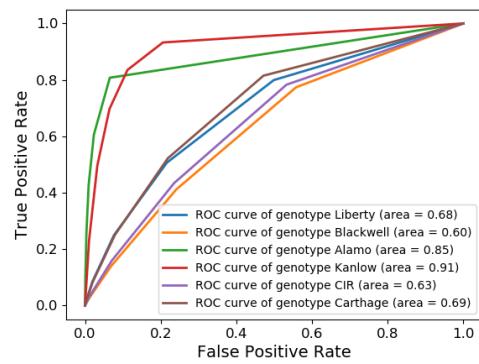
A



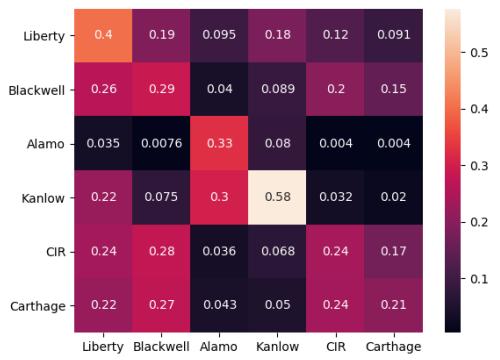
B



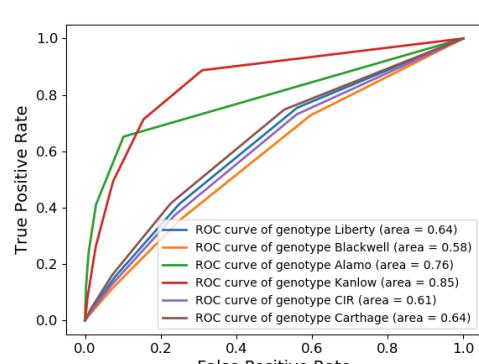
C



D



E

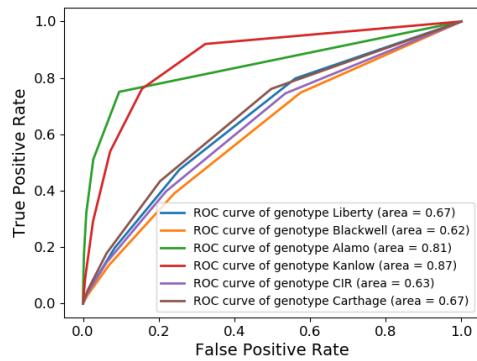


F

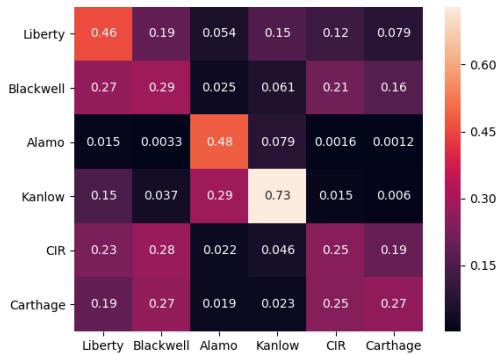
Figure 4-4. Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 15. A) and B): Classification performance when using original spectra, with an average standard deviation of 0.0141; C) and D): Classification performance with data reduced dimensionality by the Siamese network, with an average standard deviation of 0.0108; E) and F): Classification performance with data reduced dimensionality by principal component analysis transform, with an average standard deviation of 0.0105.



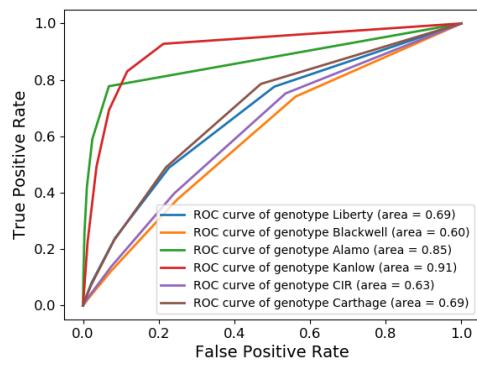
A



B



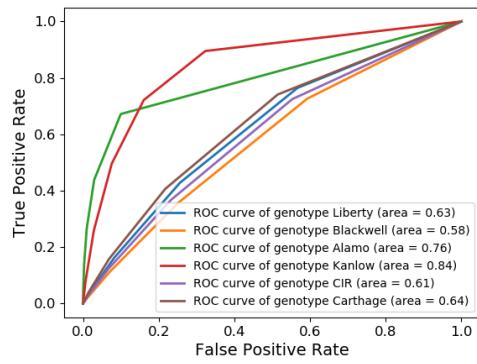
C



D



E

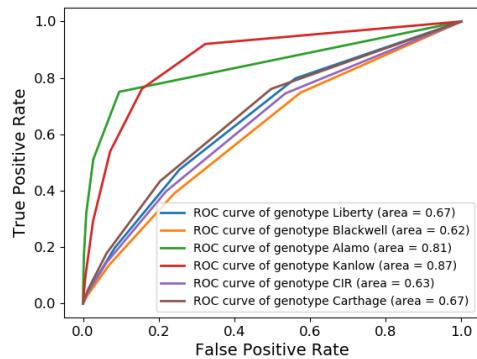


F

Figure 4-5. Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 30. A) and B): Classification performance when using original spectra, with an average standard deviation of 0.0141; C) and D): Classification performance with data reduced dimensionality by the Siamese network, with an average standard deviation of 0.0109; E) and F): Classification performance with data reduced dimensionality by principal component analysis transform, with an average standard deviation of 0.0119.



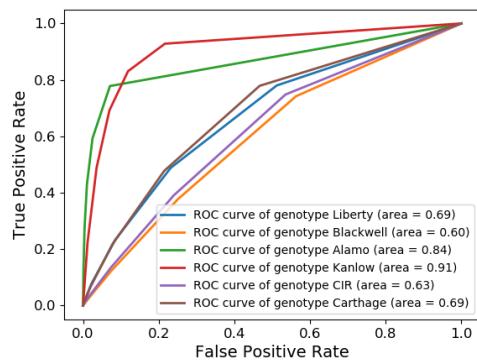
A



B



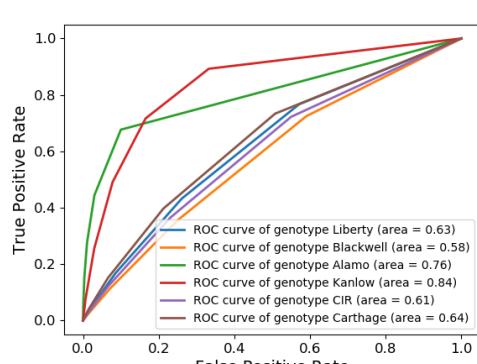
C



D



E

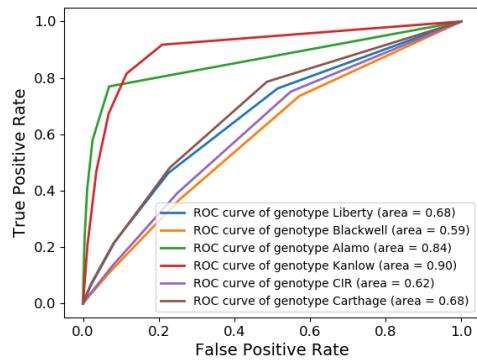


F

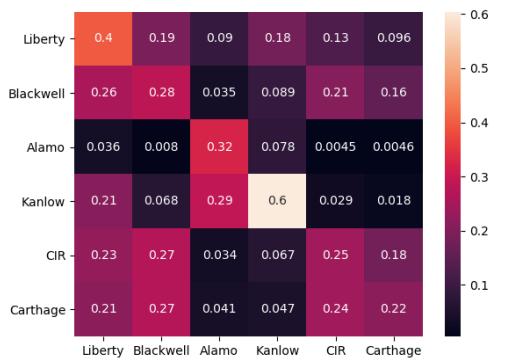
Figure 4-6. Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 55. A) and B): Classification performance when using original spectra, with an average standard deviation of 0.0141; C) and D): Classification performance with data reduced dimensionality by the Siamese network, with an average standard deviation of 0.0108; E) and F): Classification performance with data reduced dimensionality by principal component analysis transform, with an average standard deviation of 0.0143.



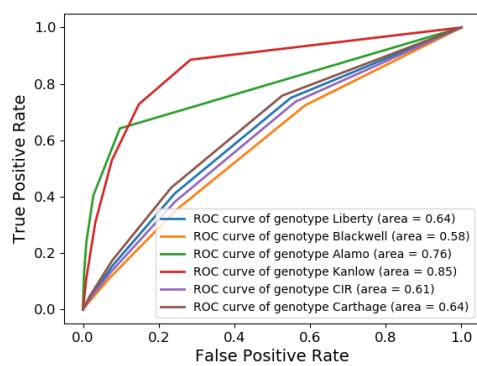
A



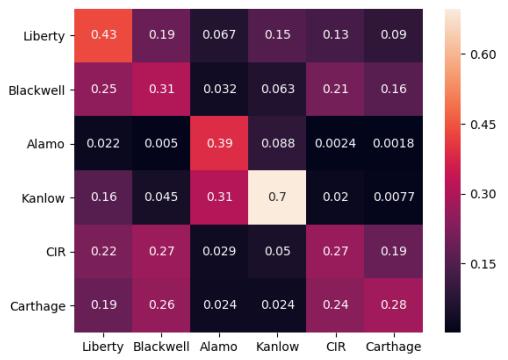
B



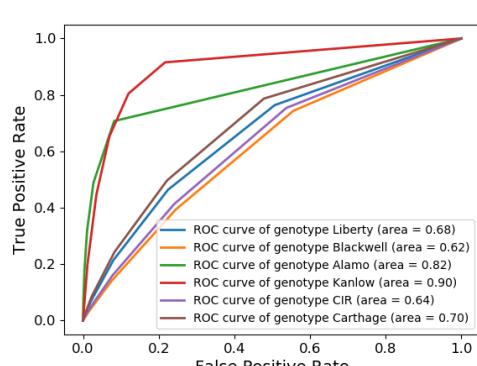
C



D



E



F

Figure 4-7. Comparison of influence of dimensionality reduction methods on classification performance when the dimensionality is reduced to 5. A) and B): Classification performance with data reduced dimensionality by the Siamese network, with an average standard deviation of 0.0107; C) and D): Classification performance with data reduced dimensionality by principal component analysis transform, with an average standard deviation of 0.0100; E) and F): Classification performance with data reduced dimensionality by linear discriminant analysis transform, with an average standard deviation of 0.0110.

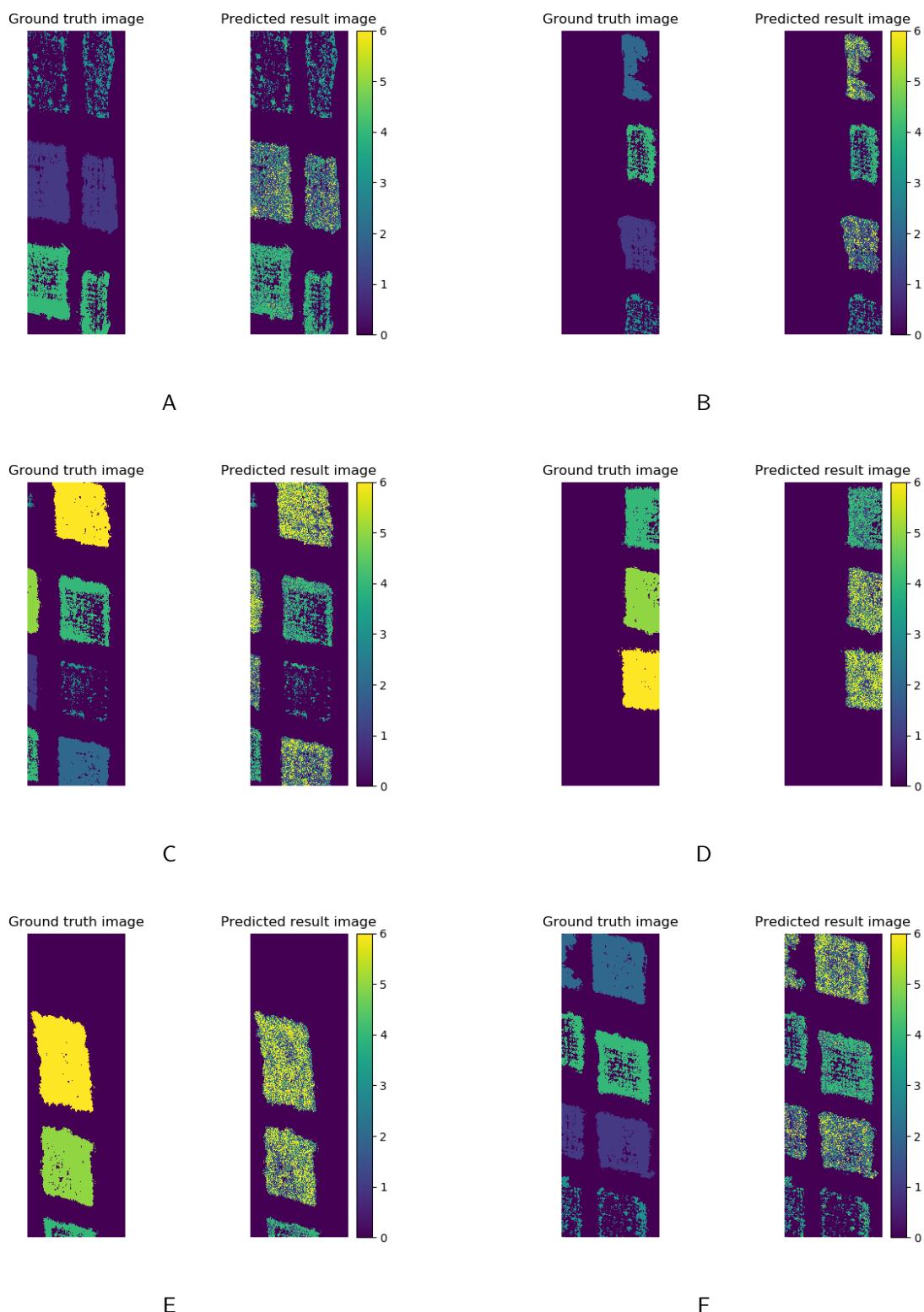


Figure 4-8. Sample classification results shown in images. Different colors represent different genotypes of switchgrass. 1: Liberty; 2: Blackwell; 3: Alamo; 4: Kanlow; 5: Cave in rock (CIR); 6: Carthage.

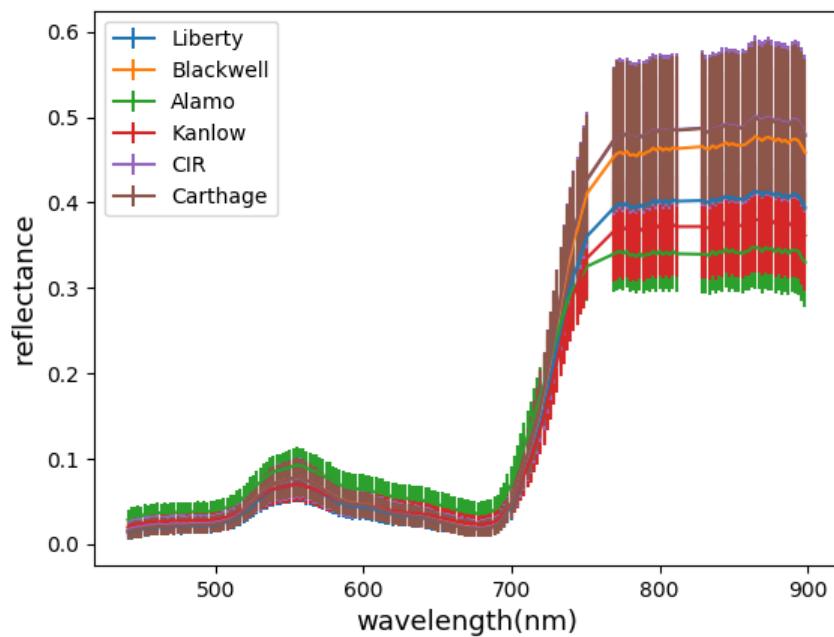


Figure 4-9. Visualization of original spectra for 6 switchgrass genotypes, with the variances shown in the error bars.

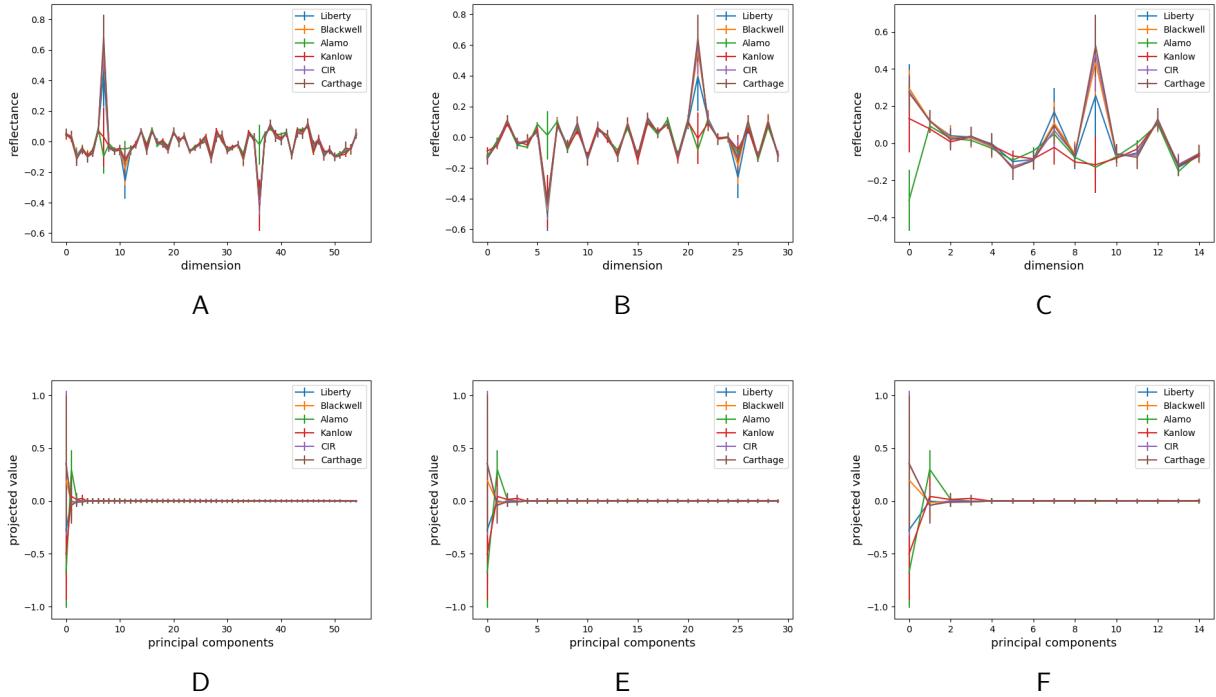


Figure 4-10. Visualization of dimensionality reduction results using the Siamese network and principal component analysis. A) Visualization of dimensionality reduction result using the Siamese network with 55 neurons in last layer; B) Visualization of dimensionality reduction result using the Siamese network with 30 neurons in last layer; C) Visualization of dimensionality reduction result using the Siamese network with 15 principal components kept; D) Visualization of dimensionality reduction result using principal component analysis with 55 principal components kept; E) Visualization of dimensionality reduction result using principal component analysis with 30 principal components kept; F) Visualization of dimensionality reduction result using principal component analysis with 55 principal components kept.

CHAPTER 5

SUMMARY AND CONCLUSIONS

The application of remote sensing in plant science is promising. However, the amount of data leads to a need for machine learning algorithms to analyze tons of data automatically.

An automated hyperspectral imagery processing protocol is created in the first place. With necessary preprocessing, dimensionality reduction and classification are performed to the data successively. The main algorithms involved are Sparsity Promoting Iterated Constrained Endmember (SPICE) algorithm and Dimensionality Reduction by Learning an Invariant Mapping (the Siamese network).

The SPICE algorithm is proved to be a powerful automated hyperspectral endmember detection method for the switchgrass detection problem presented in this work. Unmixing with the detected endmembers enables the switchgrass pixel detection task.

Owing to the fact that the differences between genotypes of switchgrass are not as noticeable as the difference between switchgrass and the background, the SPICE algorithm is not capable of classifying genotypes. Additionally, hyperspectral data is known to have high correlations among bands. Dimensionality reduction is often considered as a necessary step for hyperspectral image analysis. Indeed, classifying data of high dimensionality is time-consuming, which justifies the necessity of dimensionality reduction. Comparing the classification results where the supervised and unsupervised methods are used for dimensionality reduction, it is manifested that the supervised methods perform better in improving the classification performance at the same time of dimensionality reduction. The Siamese network nonlinearly transforms the data to a lower manifold where the data from different classes are more separable, while LDA linearly projects the data on a manifold of dimension C-1 such that as much of the class discrimination information is preserved as possible.

The Siamese network takes pairs of inputs with their neighbor information to learn a mapping from the high dimensional space to a low dimensional one such that with low dimensional representations, same class data points are close to each other, and vice versa.

Ideally, data points from different classes should have good separation in the output space. With the fact that multi-class classification being a challenging task, this method presents a reasonable result. There are several reasons for not having a good separation for all switchgrass genotypes after dimensionality reduction:

The first reason is not having enough training pairs from different classes when training the Siamese network. During the training, the Siamese network takes a pair of input data every time, either from the same class or from different classes. For instance, to make the maximum separable of class "1" and all the other classes, pairs with both data points from class "1" are considered to be a pair of same class points, and pairs with one data from class "1" and the other from any other classes are considered to be a pair of different classes points. This situation results in the need for much more pairs of samples to be able to distinguish any class from the other five classes. As also be observed from the dimensionality reduction outputs, which do not show the desired separability.

The second reason is regarding not having enough training samples for the K-nearest neighbor classifier. KNN is an instance-based algorithm, in other words, more training samples would provide much knowledge for the model to make a better prediction for the test data.

Additionally, original images taken in different lighting conditions lead to the issue that spectra belonging to the same class from different images differ from each other. This also explains the large variance observed in the original spectra.

Some potential future works are proposed based on the analyses of the results.

Firstly, classify the points in a one-versus-all way, instead of the multi-class K-nearest neighbor adopted in this work. One-versus-all classification predicts every data point with a binary classifier every time, and the results are combined to generate final results.

Secondly, include more training samples for the KNN classifier training. Alternatively, classifiers that are not instance-based can be adopted.

Last but not least, if employing other classifiers, it is recommended to jointly train the Siamese network and the classifier to guarantee the Siamese network learn the desired

mapping, which departs data points from different classes and gathers together data points from the same class.

REFERENCES

- Allwein, Erin L, Schapire, Robert E, and Singer, Yoram. "Reducing multiclass to binary: A unifying approach for margin classifiers." *Journal of machine learning research* 1 (2000).Dec: 113–141.
- Altman, Naomi S. "An introduction to kernel and nearest-neighbor nonparametric regression." *The American Statistician* 46 (1992).3: 175–185.
- Aly, Mohamed. "Survey on multiclass classification methods." *Neural Netw* 19 (2005): 1–9.
- Belharbi, Soufiane. "Neural networks regularization through representation learning." *arXiv preprint arXiv:1807.05292* (2018).
- Berman, Mark, Kiiveri, Harri, Lagerstrom, Ryan, Ernst, Andreas, Dunne, Rob, and Huntington, Jonathan F. "ICE: A statistical approach to identifying endmembers in hyperspectral images." *IEEE transactions on Geoscience and Remote Sensing* 42 (2004).10: 2085–2095.
- Bernardo, José M and Smith, Adrian FM. *Bayesian theory*, vol. 405. John Wiley & Sons, 2009.
- Bioucas-Dias, José M, Plaza, Antonio, Dobigeon, Nicolas, Parente, Mario, Du, Qian, Gader, Paul, and Chanussot, Jocelyn. "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches." *IEEE journal of selected topics in applied earth observations and remote sensing* 5 (2012).2: 354–379.
- Boardman, Joseph W. "Automating spectral unmixing of AVIRIS data using convex geometry concepts." (1993).
- Boardman, Joseph W, Kruse, Fred A, and Green, Robert O. "Mapping target signatures via partial unmixing of AVIRIS data." (1995).
- Bollard-Breen, Barbara, Brooks, John D, Jones, Matthew RL, Robertson, John, Betschart, Sonja, Kung, Olivier, Cary, S Craig, Lee, Charles K, and Pointing, Stephen B. "Application of an unmanned aerial vehicle in spatial mapping of terrestrial biology and human disturbance in the McMurdo Dry Valleys, East Antarctica." *Polar biology* 38 (2015).4: 573–578.
- Breiman, Leo, Friedman, Jerome H, Olshen, Richard A, and Stone, Charles J. "Classification and regression trees. Belmont, CA: Wadsworth." *International Group* 432 (1984): 151–166.
- Chan, Tsung-Han, Ma, Wing-Kin, Ambikapathi, ArulMurugan, and Chi, Chong-Yung. "A simplex volume maximization framework for hyperspectral endmember extraction." *IEEE Transactions on Geoscience and Remote Sensing* 49 (2011).11: 4177–4193.
- Chang, C-I, Wu, C-C, Liu, Weimin, and Ouyang, Y-C. "A new growing method for simplex-based endmember extraction algorithm." *IEEE Transactions on Geoscience and Remote Sensing* 44 (2006).10: 2804–2819.

- Chawla, Nitesh V, Bowyer, Kevin W, Hall, Lawrence O, and Kegelmeyer, W Philip. "SMOTE: synthetic minority over-sampling technique." *Journal of artificial intelligence research* 16 (2002): 321–357.
- Chen, Yangchi, Crawford, Melba M, and Ghosh, Joydeep. "Integrating support vector machines in a hierarchical output space decomposition framework." *IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium*. vol. 2. IEEE, 2004, 949–952.
- Commons, Wikimedia. "File:HyperspectralCube.jpg — Wikimedia Commons, the free media repository." 2018. [Online; accessed 19-September-2019].
URL <https://commons.wikimedia.org/w/index.php?title=File:HyperspectralCube.jpg&oldid=299520270>
- Cortes, Corinna and Vapnik, Vladimir. "Support-vector networks." *Machine learning* 20 (1995).3: 273–297.
- DeGarmo, Matthew T. "Issues concerning integration of unmanned aerial vehicles in civil airspace." *Center for Advanced Aviation System Development* 4 (2004).
- Dietterich, Thomas G and Bakiri, Ghulum. "Solving multiclass learning problems via error-correcting output codes." *Journal of artificial intelligence research* 2 (1994): 263–286.
- Fawcett, Tom. "An introduction to ROC analysis." *Pattern recognition letters* 27 (2006).8: 861–874.
- Fiorani, Fabio and Schurr, Ulrich. "Future scenarios for plant phenotyping." *Annual review of plant biology* 64 (2013): 267–291.
- Fisher, Ronald A. "The use of multiple measurements in taxonomic problems." *Annals of eugenics* 7 (1936).2: 179–188.
- Green, Andrew A, Berman, Mark, Switzer, Paul, and Craig, Maurice D. "A transformation for ordering multispectral data in terms of image quality with implications for noise removal." *IEEE Transactions on geoscience and remote sensing* 26 (1988).1: 65–74.
- Gross, John W. "A comparison of orthomosaic software for use with ultra high resolution imagery of a wetland environment." *Center for Geographic Information Science and Geography Department, Central Michigan University, Mt. Pleasant, MI, USA. Available from: http://www. imagin. org/awards/sppc/2015/papers/john_gross_paper. pdf* (2015).
- Hadsell, Raia, Chopra, Sumit, and LeCun, Yann. "Dimensionality reduction by learning an invariant mapping." *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. vol. 2. IEEE, 2006, 1735–1742.
- Hastie, Trevor and Tibshirani, Robert. "Classification by pairwise coupling." *Advances in neural information processing systems*. 1998, 507–513.

- Honkavaara, Eija, Saari, Heikki, Kaivosoja, Jere, Pölönen, Ilkka, Hakala, Teemu, Litkey, Paula, Mäkinen, Jussi, and Pesonen, Liisa. "Processing and assessment of spectrometric, stereoscopic imagery collected using a lightweight UAV spectral camera for precision agriculture." *Remote Sensing* 5 (2013).10: 5006–5039.
- Iordache, Marian-Daniel, Bioucas-Dias, José M, and Plaza, Antonio. "Sparse unmixing of hyperspectral data." *IEEE Transactions on Geoscience and Remote Sensing* 49 (2011).6: 2014–2039.
- Jain, Anil K and Waller, William G. "On the optimal number of features in the classification of multivariate Gaussian data." *Pattern recognition* 10 (1978).5-6: 365–374.
- Jimenez, Luis O, Landgrebe, David A, et al. "Supervised classification in high-dimensional space: geometrical, statistical, and asymptotical properties of multivariate data." *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 28 (1998).1: 39–54.
- Keshava, Nirmal and Mustard, John F. "Spectral unmixing." *IEEE signal processing magazine* 19 (2002).1: 44–57.
- Khalid, Noora, Ullah, Saleem, Ahmad, Sheikh Saeed, Ali, Asad, and Chishtie, Farrukh. "A remotely sensed tracking of forest cover and associated temperature change in Margalla hills." *International Journal of Digital Earth* 12 (2019).10: 1133–1150.
- Kumar, Shailesh, Ghosh, Joydeep, and Crawford, Melba M. "Hierarchical fusion of multiple classifiers for hyperspectral data analysis." *Pattern Analysis & Applications* 5 (2002).2: 210–220.
- Li, Jing, Li, Xiaorun, Huang, Bormin, and Zhao, Liaoing. "Hopfield neural network approach for supervised nonlinear spectral unmixing." *IEEE Geoscience and Remote Sensing Letters* 13 (2016).7: 1002–1006.
- Licciardi, Giorgio A and Del Frate, Fabio. "Pixel unmixing in hyperspectral data by means of neural networks." *IEEE transactions on Geoscience and remote sensing* 49 (2011).11: 4163–4172.
- Luo, Guangchun, Chen, Guangyi, Tian, Ling, Qin, Ke, and Qian, Shen-En. "Minimum noise fraction versus principal component analysis as a preprocessing step for hyperspectral imagery denoising." *Canadian Journal of Remote Sensing* 42 (2016).2: 106–116.
- Mahlein, Anne-Katrin. "Plant disease detection by imaging sensors—parallels and specific demands for precision agriculture and plant phenotyping." *Plant disease* 100 (2016).2: 241–251.
- Manolakis, Dimitris and Shaw, Gary. "Detection algorithms for hyperspectral imaging applications." *IEEE signal processing magazine* 19 (2002).1: 29–43.
- Markham, Brian L. "Landsat MSS and TM post-calibration dynamic ranges, exoatmospheric reflectances and at-satellite temperatures." *Landsat Technical Notes* 1 (1986): 3–8.

- Martin, Gabriel and Plaza, Antonio. "Region-based spatial preprocessing for endmember extraction and spectral unmixing." *IEEE Geoscience and Remote Sensing Letters* 8 (2011).4: 745–749.
- Ma'Sum, M Anwar, Arrofi, M Kholid, Jati, Grafika, Arifin, Futuhal, Kurniawan, M Nanda, Mursanto, Petrus, and Jatmiko, Wisnu. "Simulation of intelligent unmanned aerial vehicle (uav) for military surveillance." *2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. IEEE, 2013, 161–166.
- Miao, Lidan and Qi, Hairong. "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization." *IEEE Transactions on Geoscience and Remote Sensing* 45 (2007).3: 765–777.
- Mills, Steven and McLeod, Philip. "Global seamline networks for orthomosaic generation via local search." *ISPRS journal of photogrammetry and remote sensing* 75 (2013): 101–111.
- Moghimi, Ali, Yang, Ce, and Anderson, James A. "Aerial hyperspectral imagery and deep neural networks for high-throughput yield phenotyping in wheat." *arXiv preprint arXiv:1906.09666* (2019).
- Moghimi, Ali, Yang, Ce, Miller, Marisa E, Kianian, Shahryar, and Marchetto, Peter. "A novel approach to assess salt stress tolerance in wheat using hyperspectral imaging." *Frontiers in plant science* 9 (2018): 1182.
- Nascimento, José MP and Dias, José MB. "Vertex component analysis: A fast algorithm to unmix hyperspectral data." *IEEE transactions on Geoscience and Remote Sensing* 43 (2005).4: 898–910.
- Nasrabadi, Nasser M. "Hyperspectral target detection: An overview of current and future challenges." *IEEE Signal Processing Magazine* 31 (2013).1: 34–44.
- Parente, Mario and Plaza, Antonio. "Survey of geometric and statistical unmixing algorithms for hyperspectral images." *2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*. IEEE, 2010, 1–4.
- Plaza, Antonio, Martínez, Pablo, Pérez, Rosa, and Plaza, Javier. "Spatial/spectral endmember extraction by multidimensional morphological operations." *IEEE transactions on geoscience and remote sensing* 40 (2002).9: 2025–2041.
- Ready, P and Wintz, P. "Information extraction, SNR improvement, and data compression in multispectral imagery." *IEEE Transactions on communications* 21 (1973).10: 1123–1131.
- Rogge, Derek M, Rivard, Benoit, Zhang, Jinkai, and Feng, Jilu. "Iterative spectral unmixing for optimizing per-pixel endmember sets." *IEEE Transactions on Geoscience and Remote Sensing* 44 (2006).12: 3725–3736.
- Saari, Heikki, Pellikka, Ismo, Pesonen, Liisa, Tuominen, Sakari, Heikkilä, Jan, Holmlund, Christer, Mäkinen, Jussi, Ojala, Kai, and Antila, Tapani. "Unmanned Aerial Vehicle (UAV) operated spectral camera system for forest and agriculture applications." *Remote Sensing for*

- Agriculture, Ecosystems, and Hydrology XIII*. vol. 8174. International Society for Optics and Photonics, 2011, 81740H.
- Schürmann, Jürgen. "Pattern classification: a unified view of statistical and neural approaches." (1996).
- Sejnowski, Terrence J and Rosenberg, Charles R. "Parallel networks that learn to pronounce English text." *Complex systems* 1 (1987).1: 145–168.
- Shi, Chen and Wang, Le. "Incorporating spatial information in spectral unmixing: A review." *Remote Sensing of Environment* 149 (2014): 70–87.
- Singh, Ashbindu and Harrison, Andrew. "Standardized principal components." *International journal of remote sensing* 6 (1985).6: 883–896.
- Stehman, Stephen V. "Selecting and interpreting measures of thematic classification accuracy." *Remote sensing of Environment* 62 (1997).1: 77–89.
- Su, Jason G, Dadvand, Payam, Nieuwenhuijsen, Mark J, Bartoll, Xavier, and Jerrett, Michael. "Associations of green space metrics with health and behavior outcomes at different buffer sizes and remote sensing sensor resolutions." *Environment international* 126 (2019a): 162–170.
- Su, Yuanchao, Li, Jun, Plaza, Antonio, Marinoni, Andrea, Gamba, Paolo, and Chakravortty, Somdatta. "Daen: Deep autoencoder networks for hyperspectral unmixing." *IEEE Transactions on Geoscience and Remote Sensing* (2019b).
- Su, Yuanchao, Marinoni, Andrea, Li, Jun, Plaza, Antonio, and Gamba, Paolo. "Nonnegative sparse autoencoder for robust endmember extraction from remotely sensed hyperspectral images." *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE, 2017, 205–208.
- Szeliski, Richard et al. "Image alignment and stitching: A tutorial." *Foundations and Trends® in Computer Graphics and Vision* 2 (2007).1: 1–104.
- Vural, Volkan and Dy, Jennifer G. "A hierarchical method for multi-class support vector machines." *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, 105.
- Walker, Strother H and Duncan, David B. "Estimation of the probability of an event as a function of several independent variables." *Biometrika* 54 (1967).1-2: 167–179.
- Winter, Michael E. "N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data." *Imaging Spectrometry V*. vol. 3753. International Society for Optics and Photonics, 1999, 266–275.
- Xu, Xiang, Li, Jun, Wu, Changshan, and Plaza, Antonio. "Regional clustering-based spatial preprocessing for hyperspectral unmixing." *Remote Sensing of Environment* 204 (2018): 333–346.

- Yang, He, Du, Qian, Su, Hongjun, and Sheng, Yehua. "An efficient method for supervised hyperspectral band selection." *IEEE Geoscience and Remote Sensing Letters* 8 (2010).1: 138–142.
- Yu, Lei and Liu, Huan. "Feature selection for high-dimensional data: A fast correlation-based filter solution." *Proceedings of the 20th international conference on machine learning (ICML-03)*. 2003, 856–863.
- Zare, Alina and Gader, Paul. "Sparsity promoting iterated constrained endmember detection in hyperspectral imagery." *IEEE Geoscience and Remote Sensing Letters* 4 (2007).3: 446–450.
- Zitova, Barbara and Flusser, Jan. "Image registration methods: a survey." *Image and vision computing* 21 (2003).11: 977–1000.
- Zortea, Maciel and Plaza, Antonio. "Spatial preprocessing for endmember extraction." *IEEE Transactions on Geoscience and Remote Sensing* 47 (2009).8: 2679–2693.

BIOGRAPHICAL SKETCH

Hudanyun Sheng received her Bachelor of Science degree in physics from Tongji University in 2015. She entered the graduate school in University of Florida to graduate with her Master of Science degree in industrial and systems engineering and also electrical and computer engineering, in 2017 and 2019, respectively.