Latent Dirichlet Variational Autoencoder: a novel approach for Hyperspectral Image Analysis and Pixel Unmixing Exploring Deep Learning Architectures

by

Kiran Mantripragada

A thesis submitted to the
School of Graduate and Postdoctoral Studies in partial
fulfillment of the requirements for the degree of

Doctor of Philosophy in **Computer Science**

Faculty of Science
University of Ontario Institute of Technology (Ontario Tech University)
Oshawa, Ontario, Canada
May 2025

THESIS EXAMINATION INFORMATION

Submitted by: **Kiran Mantripragada**

Doctor of Philosophy in **Computer Science**

Thesis title: Latent Dirichlet Variational Autoencoder: a novel approach for Hyperspectral Image Analysis and Pixel Unmixing Exploring Deep Learning Architectures

An oral defense of this thesis took place on **May 21st, 2025** in front of the following examining committee:

Examining Committee:

Chair of Examining Committe	Dr. Bill Kapralos
Research Supervisor	Dr Faisal Z. Qureshi
Examining Committee Member	Dr. Ken Pu
University Examiner	Dr. Gregory Lewis
External Examiner	Dr. Alireza Sadeghian, Toronto Metropolitan University

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

Abstract

This thesis investigates deep learning-based methods for hyperspectral image analysis, focusing on pixel unmixing and classification tasks. Recognizing the challenges of high data dimensionality and limited labeled data availability, this research proposes innovative techniques to improve both the accuracy and efficiency of hyperspectral image interpretation. Initially, the impact of spectral band normalization and outlier removal on image segmentation scale selection is explored, leading to a robust method for Object-Based Image Analysis (OBIA). Subsequently, the research delves into the application of autoencoders for spectral dimensionality reduction, culminating in a comparative analysis demonstrating their efficacy in preserving crucial information for classification while achieving significant data compression. Building upon these findings, this thesis introduces the Latent Dirichlet Variational Autoencoder (LDVAE), a novel architecture specifically designed for hyperspectral pixel unmixing.

The LDVAE model introduces an approach to hyperspectral pixel unmixing by incorporating a Dirichlet distribution within its latent space. This design enables LDVAE to effectively model abundance vectors, satisfying the inherent sum-to-one and non-negativity constraints, while simultaneously learning a low-dimensional representation of endmember spectra. The generative nature of LDVAE further allows for the synthesis of new hyperspectral pixels by reconstructing spectra from the learned Dirichlet distributions. Evaluations on benchmark datasets demonstrate that LDVAE achieves state-of-the-art performance in both endmember extraction and abundance estimation tasks.

This thesis also introduces additional contributions to hyperspectral unmixing, addressing the challenges posed by limited labeled data and the potential for exploiting spatial information. Specifically, we extend the Latent Dirichlet Variational Autoencoder (LDVAE) framework in two key directions. First, recognizing the scarcity of labeled data and the inherent spatial coherence within hyperspectral imagery, we develop an

iterative analysis-synthesis approach using the LDVAE (iLDVAE). This novel framework facilitates automatic endmember extraction and refines the unmixing process iteratively. Second, acknowledging the importance of spatial context, we propose SpACNN-LDVAE, which integrates the LDVAE with Convolutional Neural Networks (CNNs) and spatial attention mechanisms. This architecture effectively captures local spatial relationships between pixels, yielding a more informative latent representation for improved unmixing performance. The SpACNN-LDVAE enhances both endmember extraction and abundance estimation accuracy, particularly in scenes exhibiting complex spatial structures. These contributions provide robust and efficient tools for hyperspectral image analysis, offering potential benefits across various application domains, including agriculture, forestry, mineralogy, and environmental monitoring.

Keywords: Inverse Noise Weighting, Outlier Detection, Optimal Scale Selection, Image Segmentation, Object-Based Image Analysis, Hyperspectral Images (HSI), Hsi Analysis, Hsi Dimensionality Reduction, Hsi Segmentation, Hsi Classification, Hsi Unmixing, Variational Autoencoder, Latent Dirichlet Variational Autoencoder, Generative Neural Networks, Probabilistic Graphical Models, Deep Learning

Author's Declaration

I hereby declare that this thesis consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize the University of Ontario Institute of Technology (Ontario Tech University) to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize the University of Ontario Institute of Technology (Ontario Tech University) to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public.

Kiran Mantripragada

Statement of Contributions

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published. I have used standard referencing practices to acknowledge ideas, research techniques, or other materials that belong to others. Furthermore, I hereby certify that I am sole source of the creative works and/or inventive knowledge described in this thesis.

Acknowledgements

It is with profound gratitude that I acknowledge the invaluable support and contributions of numerous individuals and institutions who have made the completion of this doctoral thesis possible. This extensive journey represents not merely the culmination of rigorous research but also the embodiment of collective encouragement, guidance, and personal sacrifice.

My deepest appreciation is extended to my supervisor, **Dr. Faisal Z. Qureshi**, from the Ontario Tech University. Your mentorship, intellectual insights, encouragement, and patience have been the bedrock of my research journey. Your guidance has not only helped to shape the trajectory of this thesis but has also influenced my development as a researcher and critical thinker. I am grateful for the invaluable discussions, the constructive feedback, and the freedom you provided, which allowed me to explore and innovate within my chosen field.

I am also indebted to the distinguished faculty members of the Ontario Tech University. Their insightful lectures, academic discussions, and commitment to excellence have significantly enriched my understanding and fostered a stimulating academic environment. Special thanks are extended to the members of my supervisory committee, **Dr. Ken Pu**, **Dr. Gregory Lewis**, **Dr. Bill Kapralos**, and **Alireza Sadeghian** from Toronto Metropolitan University, for their expertise, constructive criticism, and to upholding the academic rigor and quality of this work.

This research has significantly benefited from the collaborative support and exceptional resources provided by external organizations. I would like to express my gratitude to **Dr. Phuong D. Dao** and **Dr. Yuhong He** from the University of Toronto, also to **Dr. Paul R. Adler** from the U.S. Department of Agriculture (USDA) and **Dr. Peder A. Olsen** from Microsoft Research for their invaluable collaboration, specifically for providing access to essential data, expertise, and research infrastructure, which were crucial for the empirical components of this study. These partnerships were instrumental

in bridging academic inquiry with real-world applications and innovation.

Finally, and most importantly, I wish to express my heartfelt gratitude to my family. To my incredible wife, **Monica Moreto**, your love, steadfast support, patience, and understanding throughout the demanding and often challenging years of this doctoral journey have been my constant source of strength and solace. Your sacrifices, selfless encouragement, and belief in my abilities have been immeasurable, making every obstacle surmountable. This academic achievement is as much yours as it is mine, and I dedicate it to your enduring love.

To my precious daughter, **Karina Mantripragada**, your boundless energy, infectious joy, and curiosity have been a continuous reminder of the essential balance in life beyond academia. You provided the necessary perspective, cherished moments of distraction, and the ultimate motivation to persevere, always bringing light and laughter into the most challenging moments. Your presence filled our home with warmth and purpose.

This thesis stands as a testament to the collective efforts, intellectual contributions, and steadfast support of all those mentioned and many others who contributed in various capacities, both directly and indirectly. Thank you for being an integral part of this transformative journey.

Dedication

I dedicate this work to my parents, who instilled in me the value of education since my early childhood.

I also dedicate this work to my father-in-law, Mr. Maximiliano Moreto. His unwavering support and belief in my potential were certainly instrumental in my accomplishments.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter provides an overview of the research presented in this thesis, outlining its progression, key contributions, and the underlying context of hyperspectral image analysis. The research journey began with an exploration of pre-processing techniques for hyperspectral images, focusing on their impact on segmentation performance. This foundational work led to the investigation of dimensionality reduction methods and ultimately culminated in the development of novel deep learning architectures for hyperspectral pixel unmixing.

The primary contributions of this research are the introduction of two innovative deep learning models: the Latent Dirichlet Variational Autoencoder (LDVAE) and its iterative variant, iLDVAE. LDVAE offers a novel approach to pixel unmixing by incorporating a Dirichlet distribution within its latent space, effectively capturing the inherent constraints of abundance vectors. iLDVAE further refines this approach by addressing scenarios with limited labeled data. Beyond these core contributions, the thesis also explores the integration of spatial context into the LDVAE framework through convolutional neural networks and attention mechanisms, yielding enhanced performance. Additionally, the impact of spectral band normalization and outlier removal on image segmentation scale selection is investigated, providing a robust methodology for Object-Based Image Analysis.

The backdrop for this research is the challenging domain of hyperspectral image analysis. Hyperspectral images, unlike traditional RGB images, capture information across hundreds of narrow, contiguous spectral bands, providing a rich spectral signature for each pixel. This high dimensionality, while offering valuable insights, presents significant challenges in terms of data storage, processing, and interpretation. One crucial task in hyperspectral image analysis is pixel unmixing, which aims to decompose mixed pixels into their constituent materials (endmembers) and their corresponding proportions (abundances). This process is essential for understanding the composition of the observed scene and has widespread applications in diverse fields. The inherent complexity of pixel unmixing, coupled with the high dimensionality of hyperspectral data, motivates the development of efficient and accurate methods for this task, which forms the central focus of this thesis.

1.1 Research Overview

This research journey delves into the realm of hyperspectral image (HSI) analysis, specifically focusing on pixel-level classification and unmixing through the lens of machine learning and dimensionality reduction techniques. Although the primary contribution of this research lies in the development of Latent Dirichlet Variational Autoencoder (LDVAE), the initial stages involved exploring established methods for HSI classification and unmixing, utilizing both benchmark datasets and real-world HSI data acquired through a collaborative partnership with the Remote Sensing Labs at the University of Toronto.

Early investigations, in conjunction with Dr. Phuong Dao and Dr. Yuhong He, both from the University of Toronto, explored the efficacy of Inverse Noise Weighting for spectral band normalization and its impact on subsequent segmentation algorithms. This research (Dao et al., 2021), published in the ISPRS Journal of Photogrammetry and Remote Sensing, highlighted the challenges associated with the high dimensionality

of HSI data and the inherent redundancy within spectral bands.

Consequently, the research focus shifted towards dimensionality reduction techniques, with a particular emphasis on autoencoders as a deep learning approach. A comparative analysis of various methods demonstrated the autoencoder's ability to generate compressed representations of HSI spectra while preserving essential information for classification tasks, culminating in a publication in PLOS One (Mantripragada et al., 2022).

Building upon these insights, the research progressed deeper into the potential of variational autoencoders (VAEs) for joint dimensionality reduction and pixel unmixing. This exploration led to the development of the Latent Dirichlet Variational Autoencoder (LDVAE), an innovative architecture integrating a Dirichlet distribution within the latent space to represent abundance vectors, thereby inherently satisfying the abundance sum-to-one and non-negativity constraints. This work, which is the most significant contribution of this research, addresses the inherent challenges of hyperspectral image unmixing and has demonstrated promising performance in various HSI analysis tasks. It has garnered significant attention within the remote sensing community and was ultimately published in the IEEE Transactions on Geoscience and Remote Sensing (Mantripragada and Qureshi, 2024).

Further collaborations with the US Department of Agriculture (USDA) and Microsoft Research (MSR) provided access to additional HSI datasets, including the "Cover Crop" dataset, which presented unique challenges due to the lack of per-pixel ground truth labels. To address this, an iterative approach was devised, leading to the development of the iLDVAE (Iterative Latent Dirichlet Variational Autoencoder) and subsequent publication in the IEEE International Geoscience and Remote Sensing Symposium (Mantripragada et al., 2023).

To improve classification accuracy, we initiated a research collaboration exploring the integration of spatial-spectral features within a Latent Dirichlet Variational Autoencoder (LDVAE) framework. This work leverages a spatial soft attention mechanism in the

encoder and has culminated in a paper that was presented at the 2024 IGARSS - IEEE International Geoscience and Remote Sensing Symposium (Chitnis et al., 2024).

Furthermore, this research has made valuable contributions to several key areas of HSI analysis. Firstly, a comprehensive evaluation of segmentation and classification algorithms was conducted on pixel-level data with single labels. This analysis provided crucial insights into the strengths and weaknesses of various methods, particularly in the context of dimensionality reduction techniques. By exploring the impact of different dimensionality reduction approaches, this research enabled the identification of reduced representations that effectively preserve essential information for subsequent classification tasks.

Finally, a synthetic hyperspectral image generator was developed to facilitate further research and algorithm development in the HSI domain (Mantripragada et al., 2021). This generator enables the creation of customizable synthetic HSI datasets, with spatial coherence and with known ground truth information, providing a valuable resource for training and evaluating new algorithms without the limitations associated with real-world data acquisition and labeling.

In summary, this research trajectory illustrates a progressive exploration of HSI analysis (Figure 1.1), transitioning from established methodologies to the development of deep learning architectures that leverage dimensionality reduction and variational inference for improved pixel-level classification and unmixing. The ongoing pursuit of incorporating spatial information further underscores the commitment to advancing the state-of-the-art in HSI analysis techniques.

1.2 Motivations

Hyperspectral imaging (HSI) has emerged as a transformative technology across diverse fields, including remote sensing, agriculture, environmental monitoring, and mineralogy,

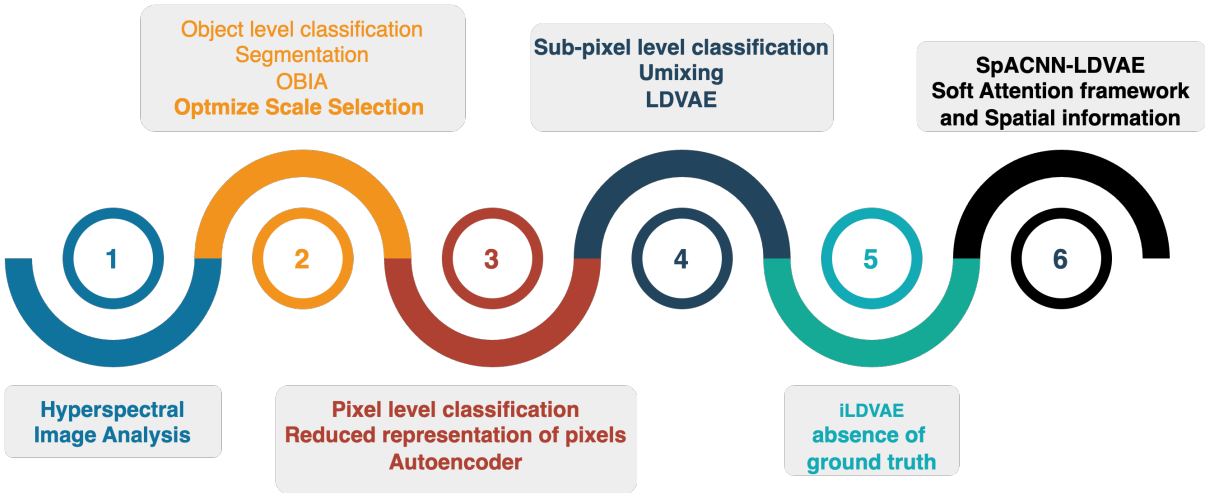

Figure 1.1: Overview and evolution of this research.

owing to its ability to capture rich spectral signatures across hundreds of narrow, contiguous bands. The rich spectral information details provide a unique "fingerprint" for identifying and characterizing materials within a scene, far surpassing the capabilities of traditional RGB or multispectral imagery. However, despite its immense potential, the inherent characteristics of HSI data present several formidable challenges that currently hinder its full utilization and widespread adoption in practical applications.

The primary motivations for the research presented in this thesis stem directly from the following challenges:

The **Curse of High Dimensionality**: HSI data typically consists of hundreds of spectral bands, leading to extremely high-dimensional feature spaces. This "curse of dimensionality" exacerbates computational complexity for storage, processing, and analysis. Traditional statistical and machine learning algorithms often struggle with this high dimensionality, leading to increased computational costs, memory requirements, and a higher risk of overfitting when labeled training data is limited. Consequently, there is a compelling need for robust dimensionality reduction techniques that can compress HSI data effectively while preserving crucial discriminative information for downstream tasks.

The **Pervasive Problem of Mixed Pixels**: In real-world HSI acquisitions, spatial resolution limitations often mean that individual pixels capture reflected light from

multiple distinct materials. This phenomenon, known as "mixed pixels", fundamentally complicates material identification and quantification. Accurate hyperspectral pixel unmixing—the process of decomposing a mixed pixel's spectrum into its constituent pure material spectra (endmembers) and their corresponding proportions (abundances)—is paramount for precise scene interpretation. Existing unmixing methods often rely on restrictive assumptions (e.g., purely linear mixing), struggle with inherent spectral variability, or do not naturally enforce physical constraints like the Abundance Sum-to-One Constraint (ASC) and Abundance Non-negativity Constraint (ANC) in their formulation. This highlights a critical gap for unmixing models that are both physically consistent and capable of handling non-linear mixing effects.

Scarcity of Labeled Ground Truth Data: A significant practical barrier to developing and deploying supervised machine learning models for HSI analysis, particularly for pixel unmixing, is the prohibitive cost and labor involved in acquiring high-quality, pixel-level ground truth abundance maps. Manual labeling is tedious, time-consuming, and often imprecise. This scarcity of labeled data necessitates the development of novel approaches that can learn effectively from limited annotations, leverage unsupervised learning paradigms, or utilize synthetic data to bridge the data gap.

Neglecting Spatial Context in Pixel-Level Analysis: While HSI provides rich spectral information, traditional pixel-wise analysis often disregards the spatial relationships between neighboring pixels. Spatial coherence is a fundamental characteristic of natural scenes, where adjacent pixels are likely to belong to the same or related land cover types. Incorporating this spatial context can significantly enhance the robustness and accuracy of HSI analysis tasks, including unmixing and classification, by mitigating noise and resolving spectral ambiguities. There is a clear motivation to develop integrated spectral-spatial models that capture these relationships.

The Need for Generative Capabilities: Beyond analysis, the ability to synthesize realistic hyperspectral data with known ground truth offers immense value for algorithm

development, benchmarking, and augmentation of limited real-world datasets. Current methods often lack the inherent generative capacity that would allow for the creation of new, diverse HSI scenes with controllable properties, which could greatly accelerate research and model training.

This thesis directly addresses these motivations by proposing novel deep learning-based solutions. It begins by examining the impact of preprocessing on image segmentation and investigating various dimensionality reduction techniques to overcome the curse of dimensionality. Crucially, it introduces the Latent Dirichlet Variational Autoencoder (LDVAE), a novel deep learning architecture designed to perform hyperspectral pixel unmixing by inherently modeling abundance vectors within a Dirichlet latent space, thereby satisfying the fundamental ASC and ANC constraints. Furthermore, to overcome the challenge of limited labeled data, the thesis develops an iterative LDVAE (iLDVAE), enabling effective unmixing even in the absence of explicit ground truth. Finally, recognizing the importance of spatial context, the SpACNN-LDVAE is proposed, integrating convolutional neural networks and spatial attention mechanisms to leverage local spatial relationships for enhanced unmixing performance. Collectively, this research aims to significantly advance the accuracy, efficiency, and practical applicability of hyperspectral image analysis, moving towards more robust and intelligent interpretation of complex HSI data.

1.3 Contributions

This dissertation presents a series of contributions to the field of hyperspectral image analysis, focusing on pixel unmixing, classification, segmentation, and dimensionality reduction. These contributions are listed below in descending order of importance:

- **LDVAE**: The core contribution of this dissertation is the development of the Latent Dirichlet Variational Autoencoder (LDVAE) for hyperspectral pixel unmixing.

LDVAE leverages the power of deep learning to effectively model the underlying statistical properties of hyperspectral data, achieving state-of-the-art performance on several benchmark datasets (Mantripragada and Qureshi, 2024).

- **Iterative LDVAE**: Building upon the LDVAE framework, this research also culminated in the development of an iterative approach (iLDVAE) to address the challenge of limited labeled data. iLDVAE demonstrates the ability to perform accurate pixel unmixing without requiring extensive ground truth information, making it highly valuable for real-world applications (Mantripragada et al., 2023).

- **SpACNN-LDVAE**: To further enhance the performance of LDVAE, the incorporation of a spatial attention mechanism within the LDVAE framework resulted in SpACNN-LDVAE, which effectively leverages spatial context to improve both endmember extraction and abundance estimation accuracy. This work resulted from a collaboration with a student researcher (Chitnis et al., 2024).

- **Segmentation and Dimensionality Reduction**: This research also contributed significantly to research in hyperspectral image segmentation and dimensionality reduction. The contributions include the development of novel methods for optimal scale selection in segmentation (Dao et al., 2021), the implementation and evaluation of various dimensionality reduction techniques, and the investigation of the impact of compression on classification accuracy (Mantripragada et al., 2022).

- **Synthetic Hyperspectral Data Generator**: This work also produced a synthetic hyperspectral image generator to facilitate further research and algorithm development in the HSI domain. This generator enables the creation of customizable hyperspectral datasets with known ground truth information, providing a valuable resource for training and evaluating new algorithms without the limitations associated with real-world data acquisition and labeling (Mantripragada et al., 2021).

These contributions advance the state-of-the-art in hyperspectral image analysis, providing valuable tools for researchers and practitioners in remote sensing, agriculture, and other domains.

Chapter 2

Research Background

This section provides essential background information for the research presented in this thesis. It begins by introducing hyperspectral images (HSI), highlighting their unique characteristics and the wealth of information they contain. The discussion then delves into the spectral and spatial features that characterize HSIs, drawing connections to the fields of spectroscopy and computer vision, respectively. The inherent challenges associated with the high dimensionality of hyperspectral data are also addressed, emphasizing the need for efficient processing and analysis techniques. Finally, the concept of HSI pixel unmixing is introduced, explaining its significance and the difficulties encountered in accurately decomposing mixed pixels. The following subsections elaborate on each of these aspects: Hyperspectral Images, Spectral Features, Spatial Features, High-Dimensionality, and HSI Pixel Unmixing. These combined characteristics present specific challenges in machine learning and computer vision, necessitating specialized approaches for effective HSI analysis, which will be explored in the subsequent chapters.

2.1 Hyperspectral Images

A hyperspectral image is a type of image that contains data across a wide range of wavelengths (or frequencies), including those beyond the visible spectrum. Unlike standard

color images, that capture information only across the three color channels (red, green, and blue), hyperspectral images capture data across hundreds or even thousands of narrow and contiguous spectral bands. Each band represents a slightly different wavelength of light. This allows hyperspectral images to provide a much more detailed and comprehensive view of the object or scene being imaged, making them useful for a wide range of applications such as remote sensing, agriculture, mineral exploration, and medical imaging (Amigo et al., 2015; Crane et al., 2004; Khan et al., 2018; Landgrebe, 2002). Hyperspectral remote sensing combines two sensing modalities familiar to most scientists and engineers: imaging and spectrometry (Eismann, 2012). Figure 2.1 shows a representation of an HSI datacube, the spectral curve of one pixel, and an example of image acquisition system embedded in a low-altitude aircraft. Post-processing of remotely sensed imagery acquired by surveys or drones is crucial for transforming raw data into valuable information. These tasks can be broadly categorized into geometric, radiometric, and atmospheric corrections, followed by further processing steps depending on the specific application.

- **Geometric Corrections:** These address distortions introduced by platform instability, sensor geometry, and terrain relief. Common techniques include: 1) georeferencing which is assigning geographic coordinates to the image using ground control points (GCPs) and/or sensor position and orientation data; and 2) orthorectification which consists of removing geometric distortions and projecting the image onto a flat surface, compensating for terrain relief effects using a Digital Elevation Model (DEM).

- **Radiometric Corrections:** These account for variations in sensor response and illumination conditions: 1) radiometric calibration, converting digital numbers to at-sensor radiance or reflectance using sensor calibration parameters; and 2) atmospheric correction, which removes the effects of atmospheric scattering and absorp-

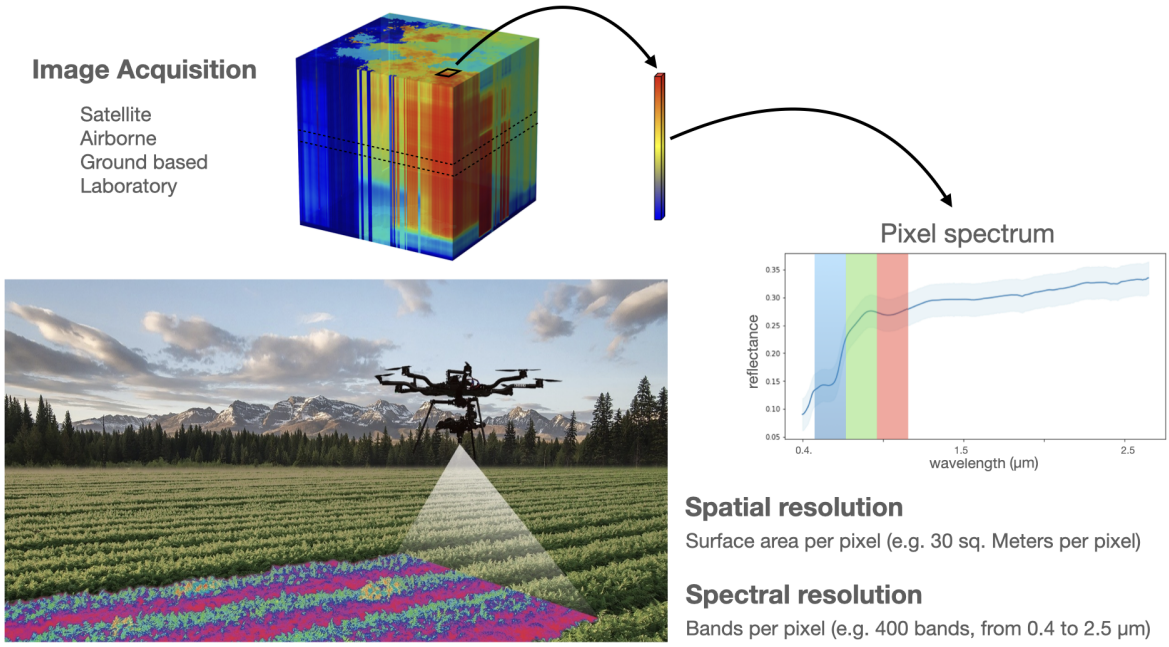

Figure 2.1: HSI Datacube, a single pixel, and an example of data acquisition process. This figure was created by the author of this thesis. The image with aircraft and crop was adapted from (Hyperspectral Imaging Solutions, 2023).

tion to retrieve surface reflectance.

- **Atmospheric Corrections:** Crucial for quantitative analysis, these remove the influence of the atmosphere: 1) atmospheric scattering, which involves compensating for scattering effects that depend on wavelength and atmospheric conditions; and 2) atmospheric absorption, which corrects for the absorption by atmospheric gases like water vapor, ozone, and carbon dioxide.

2.2 Spectral Features: Spectroscopy

Spectroscopy is a scientific discipline that aims to study the interaction between materials and electromagnetic radiation. It refers to the analysis of the absorption, emission, or scattering of electromagnetic radiation by a sample of material in any form, be it gas, liquid, or solid. Spectroscopy is used to identify and quantify the chemical and

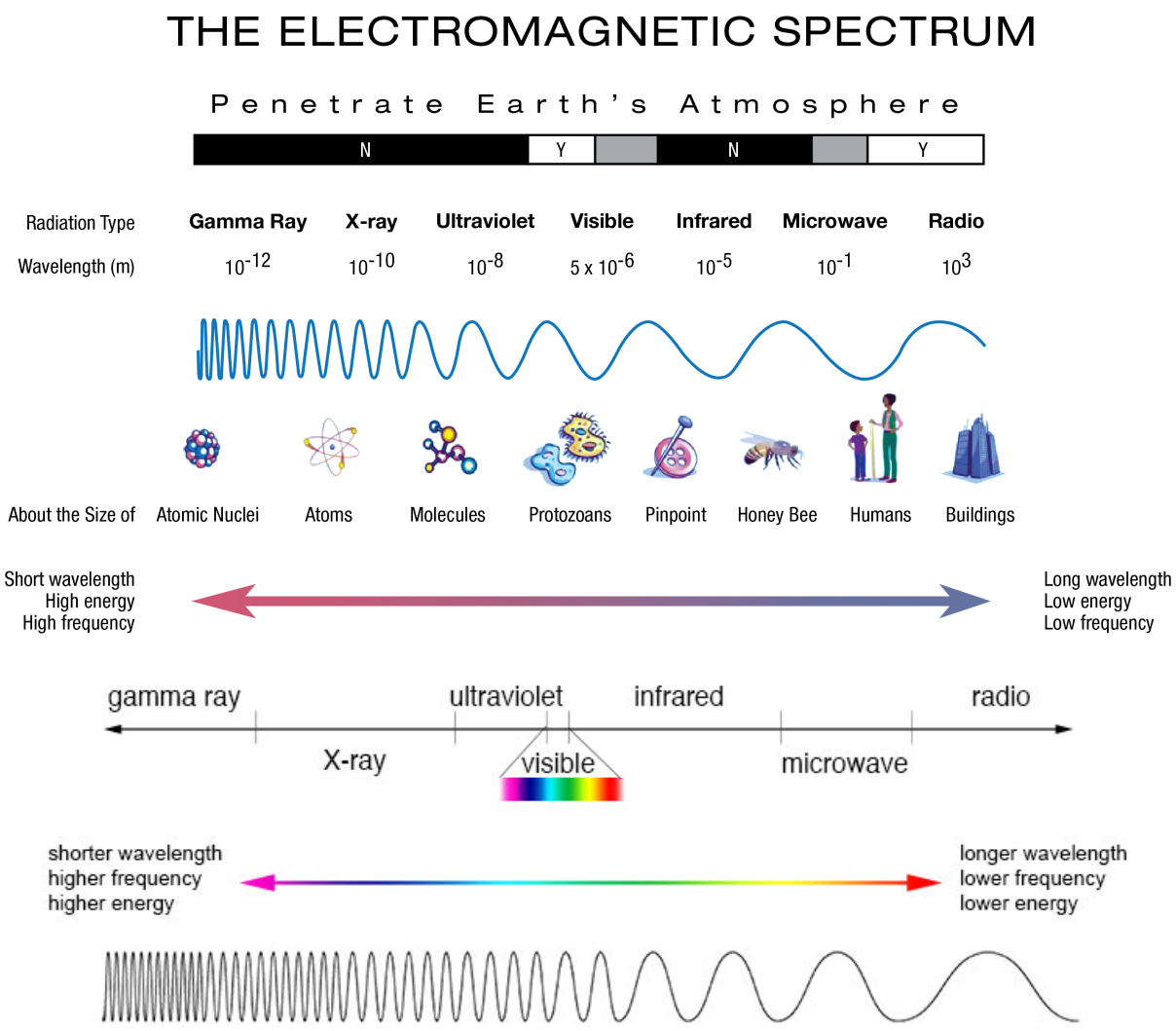

Figure 2.2: Electromagnetic Spectrum Diagram. Source: My NASA Data (Data, 2019; NASA, 2023)

physical properties of a material or substance, such as composition, concentration, and temperature. There are many different types of spectroscopy, each utilizing a specific range of the electromagnetic spectrum, such as infrared, ultraviolet, visible, microwave, or radio waves. Spectroscopy has applications in a wide range of fields, including chemistry, physics, astronomy, biology, medicine, and environmental science (Eismann, 2012).

Figure 2.2 illustrates the electromagnetic spectrum. Hyperspectral sensors frequently capture images from $400nm$ to $2500nm$ in wavelength, which spans from ultraviolet to far infrared, distributed across 100 to 300 bands. This range also includes the visible

part of the spectrum (red, green, and blue: channels between $400nm$ to $780nm$).

In remote sensing studies, hyperspectral sensors on satellites are usually limited to a few ranges of the spectrum, due to the inability of some wavelengths to penetrate the Earth's atmosphere. Most satellite-based missions, such as Landsat, are equipped with multispectral sensors, which collect images, usually 5 to 8 channels, from $430nm$ to $670nm$ (Benediktsson et al., 2012; Hossain and Chen, 2019; Kussul et al., 2017; Ma et al., 2015; Richards and Jia, 2006).

2.3 Spatial Features: Computer Vision

While **spectral features** represent reflectance values of pixels at different wavelengths, **spatial features** refer to the shape, size, texture, and location of objects in an image. In computer vision, the visible part of the spectrum, *i.e.*, the red, green, and blue channels, is also used to represent the pixel or object. Combining spatial and spectral information helps not only in exploring features that describe the different materials present in a scene or a pixel but also in representing the neighbourhood of objects or materials. Therefore, exploring the fusion of spatial and spectral features can help to overcome the limitations of each type of feature alone and improve the accuracy and robustness of hyperspectral image processing.

2.4 High-Dimensionality in the Feature Space

High dimensionality in the feature space refers to the situation where there is a very large number of features or attributes used to describe a pixel. High dimensionality in the feature space refers to a large number of features or attributes used to describe each pixel. High dimensionality can pose several challenges in image processing and machine learning. One of the most significant challenges is the "curse of dimensionality", where the number of data points required to obtain accurate models grows exponentially with the

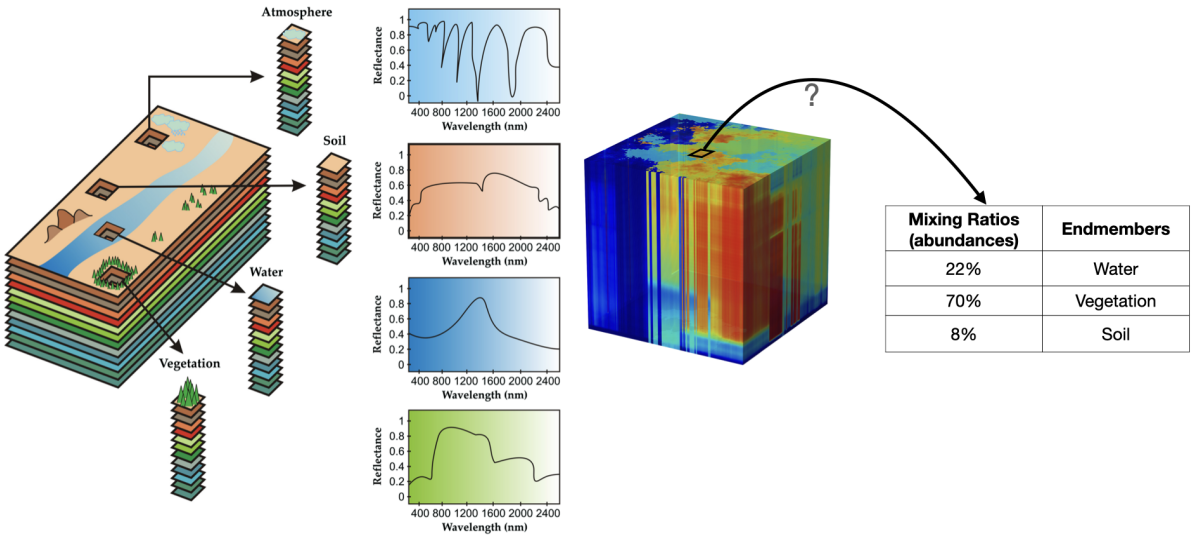

Figure 2.3: Overview of HSI umixing: endmembers, abudances and mixel pixels. Figure adapted from Bioucas-Dias *et al.* (Bioucas-Dias et al., 2012).

number of features. High dimensionality can also lead to overfitting and sparsity issues, making it harder to extract meaningful patterns from the data. Therefore, dimensionality reduction techniques are often used to overcome these issues and obtain more manageable feature spaces. Various dimensionality reduction techniques, including feature selection and feature extraction, are employed to address these challenges.

2.5 HSI Pixel Unmixing

Hyperspectral sensors capture light across numerous wavelengths, offering a detailed fingerprint of materials within a scene. Often, individual pixels capture light from multiple materials simultaneously, resulting in mixed pixels. This poses a significant challenge for accurately identifying and quantifying the constituent materials. Pixel unmixing aims to solve this problem by decomposing the mixed pixel spectrum into a set of pure material signatures called "endmembers", and their corresponding proportions within the pixel, known as "abundances" (see Figure 2.3). Such methods essentially unmix the spectral information, allowing us to understand the individual components present within a single

pixel. Hyperspectral pixel unmixing finds applications in a variety of fields such as remote sensing for analyzing land cover, environmental monitoring for tracking pollution, and medical imaging for diagnosing diseases. However, challenges remain in accurately estimating endmembers, handling variations in illumination, and addressing non-linear mixing effects. Despite these complexities, hyperspectral pixel unmixing remains a crucial tool for extracting valuable information from hyperspectral data, enabling us to see beyond the limitations of spatial resolution.

In summary, this chapter has laid the groundwork for understanding the context of this research by introducing hyperspectral images, their unique spectral and spatial characteristics, and the challenges associated with their high dimensionality. The concept of pixel unmixing has been presented as a key technique for extracting valuable information from mixed pixels, paving the way for a detailed exploration of the novel deep learning approaches proposed in subsequent chapters to address the complexities and limitations of current hyperspectral image analysis methods.

Chapter 3

Bibliographic Review

This chapter provides a comprehensive review of the literature relevant to the research presented in this thesis. It covers three key areas: hyperspectral pixel unmixing, spectral dimensionality reduction, and image segmentation. While the research chronologically unfolded as described in Chapter 1 (Introduction), this chapter presents the literature in a sequence aligned with the main contributions of the thesis. This allows for a more focused and coherent discussion of the relevant background and state-of-the-art in each area, leading into the novel approaches proposed in the subsequent chapters. Specifically, it begins with a discussion of hyperspectral unmixing, the core focus of the thesis and the area where the primary contribution lies, before transitioning to dimensionality reduction and segmentation, which play supporting roles in the overall framework. This ordering facilitates a clearer understanding of the motivations and context for the developed methods.

3.1 Hyperspectral Pixel Unmixing

Hyperspectral imaging, with its rich spectral information, offers immense potential for detailed analysis of the Earth's surface. However, the mixed nature of hyperspectral pixels, where multiple materials contribute to the observed spectral signature, poses a

significant challenge for accurate interpretation. Pixel unmixing addresses this challenge by decomposing mixed pixel spectra into their constituent materials (endmembers) and their corresponding proportions (abundances). This process is crucial for various applications, including material identification, change detection, and quantitative analysis of surface composition. This section provides an overview of different pixel unmixing techniques, categorized based on their underlying principles and methodologies. These categories span from physically-grounded models to purely data-driven approaches, each with its own set of advantages and limitations.

The field of pixel unmixing can be broadly divided into two categories: (a) physics-based methods (Heylen et al., 2014) and (b) data-driven methods.

3.1.1 Physics-based Models

Physics-based schemes use models of light reflection, scattering, transmission and absorption, e.g., Hapke's Bidirectional Reflectance Model (BRDF) (Hapke, 2012; Drumetz et al., 2020; Sun and Lucey, 2021) and the Atmospheric Dispersion Model (Janiczek et al., 2020), for hyperspectral pixel unmixing. Physics-based models are laborious to use in practice, as these require radiance models that are situation-specific. Hapke's reflectance model is expressed as:

$$r(i,e,g) = K\frac{\omega}{4\pi}\frac{\mu_o}{\mu_o+\mu}\big\{p(g)\,[1+B(g)] + [H(\mu_o/K)H(\mu/K)-1]\,\big\},$$

where r is the reflectance factor, which depends on the incidence, emission, and phase angles (respectively i, e, and g). The values for incidence and emission used in the equation are actually their cosines: $\mu_0 = \cos(i)$, $\mu = \cos(e)$; K is the porosity coefficient which describes how closely the particles are embedded within the medium and affects reflectance through the opposition effects $B(g)$, including shadow hiding and backscatter. ω is the albedo of a mixture and $H(x)$ is the Ambartsumian-Chandrasekhar H function (Azzolini

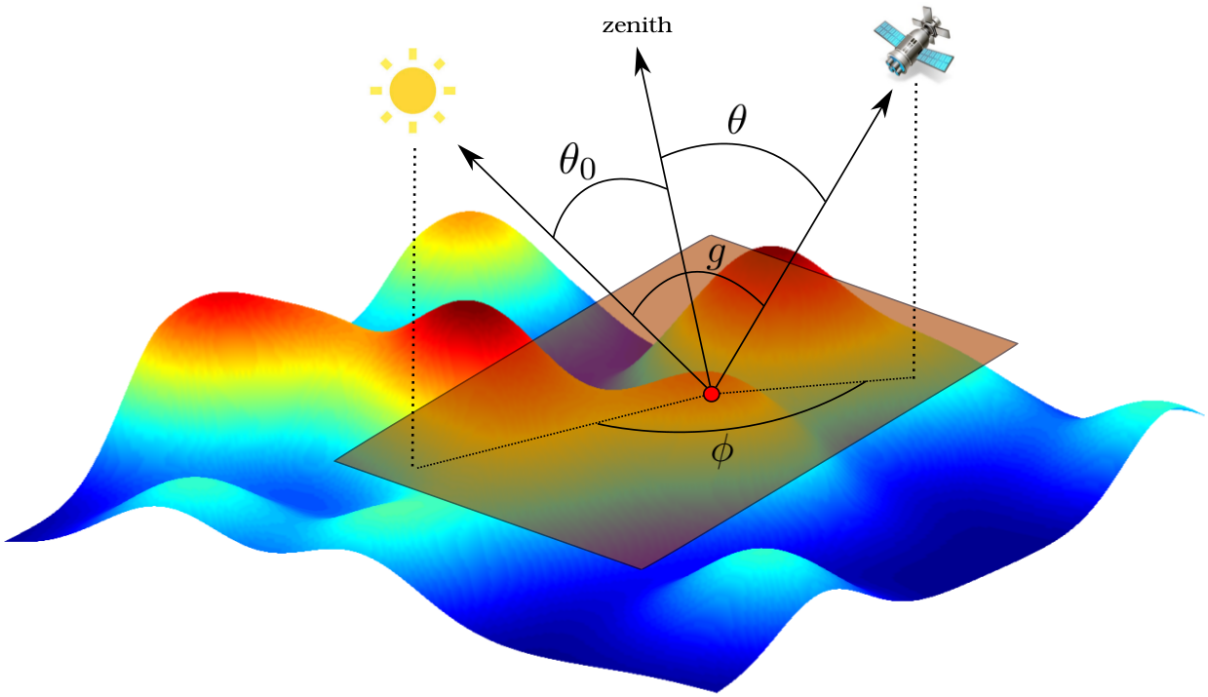

Figure 3.1: The Drumetz model (Drumetz et al., 2020) uses the acquisition angles for a given spatial location (red dot) to derive an Extended Linear Mixing Model (ELMM). Θ_0 and Θ are respectively i and e in the Hapke model. Image from Drumetz et al. (2020).

et al., 2020; Nagirner and Ivanov, 2020), which is widely used in analytical radiative transfer theory. The difficulty in obtaining accurate results with Hapke's model stems from the complexity of the model, the difficulty in accurately determining its parameters, the non-uniqueness of solutions, the underlying assumptions and simplifications, and the computational demands. Drumetz et al. (2020) also combine Hapke's model with linear mixing models, sitting at the intersection of physics-based and data-driven approaches. Figure 3.1 shows the angles used in the BRDF model.

3.1.2 Data-Driven methods

The complexities and limitations of physics-based approaches have motivated the development of data-driven pixel unmixing techniques. These methods offer several advantages, including streamlined implementation, reduced reliance on specific physical models, and the potential for improved generalization across diverse datasets and environmen-

tal conditions. However, they also introduce a dependence on the quantity and quality of training data. This section explores the various categories of data-driven methods. Data-driven methods can be further categorized into: i) endmember extraction methods, ii) Nonnegative Matrix Factorization (NMF), and iii) machine learning-based methods.

Methods for Endmember Extraction (Blind Source Separation)

The LDVAE method proposed in this thesis belongs to the class of data-driven approaches which, in contrast to the physics-based models, are simpler to apply and to use in practice. Consequently, a majority of pixel unmixing methods fall into this category. Although, data-driven approaches require large amounts of data for training, these methods are more generalizable than tailored custom physics that must be defined for each material and environmental conditions. The following paragraphs provide a brief overview of data-driven methods for pixel unmixing.

Blind Source Separation (BSS) type methods, such as N-FINDR, PPI, and VCA, divide the problem of unmixing into two steps: 1) endmember extraction and 2) abundance estimation. Abundance estimation (step 2) often requires *a priori* knowledge of the endmembers; therefore, it is sensitive to the accuracy of the estimated endmembers from step 1 (Winter, 1999; Drumetz et al., 2020, 2016a). N-FINDR, for example, is an iterative algorithm for endmember extraction that seeks to find the vertices, which represent the endmembers, of the n-simplex containing the pixel spectra (Winter, 1999).

Pure Pixel Index (PPI) is another commonly used scheme for endmember extraction that can address atmospheric, solar, and instrument-induced artifacts. PPI achieves the endmember extraction task by compressing (via dimensionality reduction) and denoising (via noise whitening) the input spectra before projecting it onto an n-simplex hyperplane. The pixels closest to the vertices (of the n-simplex) are used to identify endmembers present in the pixel, *i.e.* , points A, B, and C in Figure 3.2) are the theoretical endmembers. These methods use Abundance Sum-to-One Constraint (ASC) and

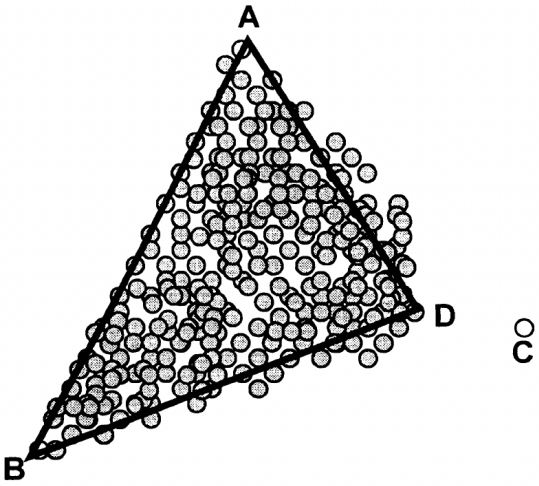

Figure 3.2: In an unmixing problem with three materials, the points A, B, and C would represent pure end-members. However due to the lack of any pure pixels in several datasets, D was selected as an endmember, given its proximity to the true theoretical vertex C Image from (Winter, 1999).

Abundance Non-negative Constraint (ANC) constraints to set up a fully constrained least square optimization problem for abundance estimation (Ibarrola-Ulzurrun et al., 2019; Mahabir et al., 2018; Bioucas-Dias et al., 2012; Nascimento and Dias, 2005a; Winter, 1999).

NMF-based Methods

Using a matrix notation, a hyperspectral dataset can be expressed as:

$$\mathbf{Y} = \mathbf{MS} + \mathbf{N},$$

where $\mathbf{Y}$ is the HSI dataset, $\mathbf{M}$ is the endmember matrix, $\mathbf{S}$ is the abundance matrix, and $\mathbf{N}$ is the noise matrix.

Methods—such as Spectral-Spatial Weighted Sparse Non-Negative Matrix Factorization (SSWNMF) (Zhang et al., 2022), Spatial Group Sparsity Regularized Nonnegative Matrix Factorization (SGSNMF) (Wang et al., 2017), Total Variation Regularized Reweighted Sparse Nonnegative Matrix Factorization (TV-RSNMF) (He et al., 2017),

and Graph-Regularized $L_{1/2}$-NMF (GLNMF) (Lu et al., 2013)—rely upon non-negative matrix factorization to estimate abundances.

The key idea is to express the hyperspectral image as a product of two matrices representing endmembers and abundances. SSWNMF and SGSNMF use spatial information when performing pixel unmixing.

The SSWNMF method separates the unmixing problem into two main steps: i) an endmember estimation step and ii) an abundance estimation step. The first step seeks to minimize the objective function:

$$J(\mathbf{M}) = \arg\min_{M} \frac{1}{2}\|\mathbf{Y} - \mathbf{MS}\|_F^2 + Trace(\Psi\mathbf{M}),$$

where Ψ is the Lagrange multiplier in the matrix format. The second step seeks to minimize the objective function:

$$J(\mathbf{S}) = \arg\min_{S} \frac{1}{2}\|\mathbf{Y_f} - \mathbf{M_f S}\|_F^2 + \lambda\|\mathbf{H_{spe}H_{spa}} \odot \mathbf{S}\|_{1,1} + Trace(\Phi\mathbf{S}),$$

where $\mathbf{Y_f}$ is the augmented $\mathbf{Y}$ matrix, $\mathbf{M_f}$ is augmented $\mathbf{M}$ matrix, $\mathbf{H_{spe}}$ is the spectral weighting factor matrix, $\mathbf{H_{spa}}$ is the spatial weighting factor matrix, and Ψ is the Lagrange multiplier. For more details on this derivation and the update algorithm for $J(\mathbf{M})$ and $J(\mathbf{S})$, refer to the article by (Zhang et al., 2022). While SSWNMF defines a neighbourhood using a weighted-window around the pixel of interest, SGSNMF defines the neighbourhood as a super-pixel and TV-RSNMF iteratively updates endmembers' matrix and abundance maps. It can be considered as an abundance map *denoising* procedure.

GLNMF extends TV-RSNMF and builds a graph that defines the local neighbourhood around the pixel of interest. Both TV-RSNMF and GLNMF methods make sparsity assumptions when solving for hyperspectral pixel unmixing. Non-negative matrix factorization-based methods achieve compelling results on hyperspectral unmixing benchmarks. Specifically, SSWNMF achieves the state-of-the-art results on hyperspectral pixel

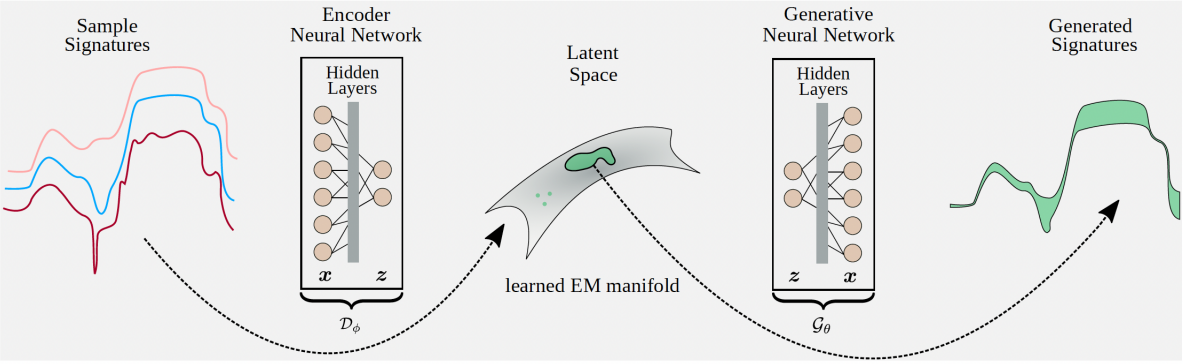

Figure 3.3: Deep Generative method proposed by (Borsoi et al., 2020).

unmixing benchmarks. Therefore, we have followed the evaluation scheme proposed by SSWNMF, and we use the same benchmark datasets and metrics to evaluate our methods as those used in (Zhang et al., 2022).

Machine Learning-based Methods

More recently, researchers have been exploring deep Learning-based approaches for HSI pixel unmixing. DeepGUn (Borsoi et al., 2020) is a deep learning method for pixel unmixing that explores regularization techniques to learn latent representations that are amenable to Vertex Component Analysis (VCA) for endmember extraction. The extracted endmembers are subsequently used to train deep learning models to reconstruct pure pixels. Borsoi et al. (2020) define the optimization problem as:

$$J(\mathbf{A}, \mathbb{Z}) = \frac{1}{2} \sum_{n=1}^{N} \|\mathbf{y_n} - \tilde{\mathcal{G}}(\mathbf{Z}_n)\mathbf{a}_n\|_F^2 + \mathcal{R}(\mathbf{A}) + \mathcal{R}(\mathbb{Z}),$$

where A is the abundance matrix, $\mathbb{Z}$ is a tensor with three dimensions obtained by stacking all pixel-dependent latent endmember representations Z_n; $\mathcal{R}(A)$ and $\mathcal{R}(\mathbb{Z})$ are regularization matrices. Figure 3.3 illustrates the DeepGUn architecture.

Palsson et al. (2018) employ an autoencoder architecture where the encoder stage learns to represent abundances by enforcing pixel reconstruction at the decoder stage. This approach assumes a linear mixing of endmembers within a pixel and the authors

employed three objective functions as follows:

$$J_{MSE} = \frac{1}{P}\sum_{p=1}^{P} \|\mathbf{x_p} - \hat{\mathbf{x_p}}\|_2^2,$$
$$J_{SAD} = \frac{1}{P}\sum_{p=1}^{P} \arccos\left(\frac{\langle \mathbf{x_p}, \hat{\mathbf{x_p}}\rangle}{\|\mathbf{x_p}\|_2 \|\hat{\mathbf{x_p}}\|_2}\right), \text{ and}$$
$$J_{SID} = \frac{1}{P}\sum_{p=1}^{P}\sum_{n=1}^{B} p_n \log\left(\frac{p_n}{q_n}\right) + \sum_{n=1}^{B} q_n \log\left(\frac{q_n}{p_n}\right),$$

where MSE is the mean squared error, SAD is the spectral angle distance, and SID (spectral information divergence) estimates the divergence between the probability mass functions of the target and estimated spectra. The vector $\mathbf{x}$ is the input HSI pixel, and the vector $\hat{\mathbf{x}}$ is the reconstructed pixel. The values p_n and q_n are computed as follows:

$$p_n = \frac{\mathbf{x}_{i,n}}{\sum_{k=1}^{M} \mathbf{x}_{i,k}}, \text{ and}$$
$$q_n = \frac{\hat{\mathbf{x}}_{i,n}}{\sum_{k=1}^{M} \hat{\mathbf{x}}_{i,k}}.$$

While MSE measures the direct difference between target and estimated spectra, the SID and SAD offer a scale-invariant alternative, and SID focuses on the shapes of the probability mass functions rather than absolute magnitudes. This distinction is crucial for unmixing, as MSE's sensitivity to scale can lead to discrimination between identical endmembers based solely on their brightness. Although the scale invariance of SID and SAD might introduce uncertainty in the absolute scale of estimated endmembers, enforcing the ASC within the neural network effectively mitigates this issue by preserving the essential relative scales of the endmember spectra. Figure 3.4 illustrates the achitecture proposed by Palsson et al. (2018). Following from this research, the same authors explored the use of a variational autoencoder to generate synthetic data in other iterations of this same approach (Palsson et al., 2022).

The DAEN (Deep Autoencoder Networks) architecture, presented by Su et al. (2019),

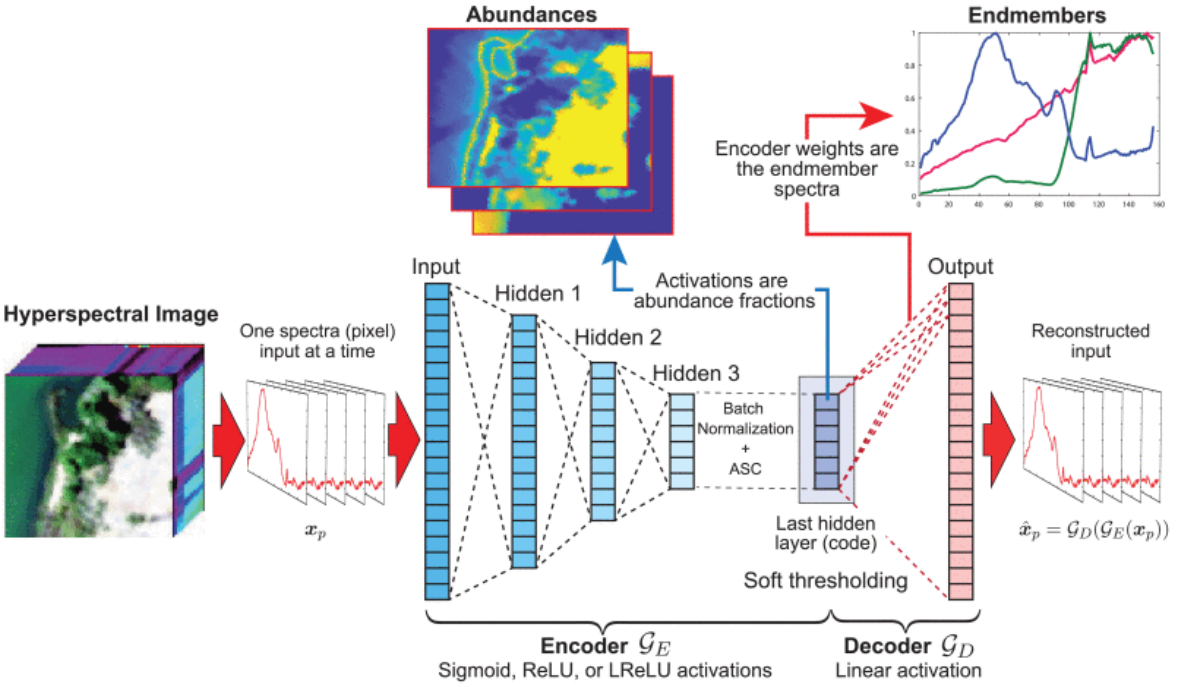

Figure 3.4: Architecture proposed by (Palsson et al., 2018). The autoencoder is trained on all the spectra in the HSI for a number of epochs. After training, abundance maps can be extracted as the activations of the last hidden layer for each input spectra, and the weights of the decoder are the endmember spectra. Image from (Palsson et al., 2018).

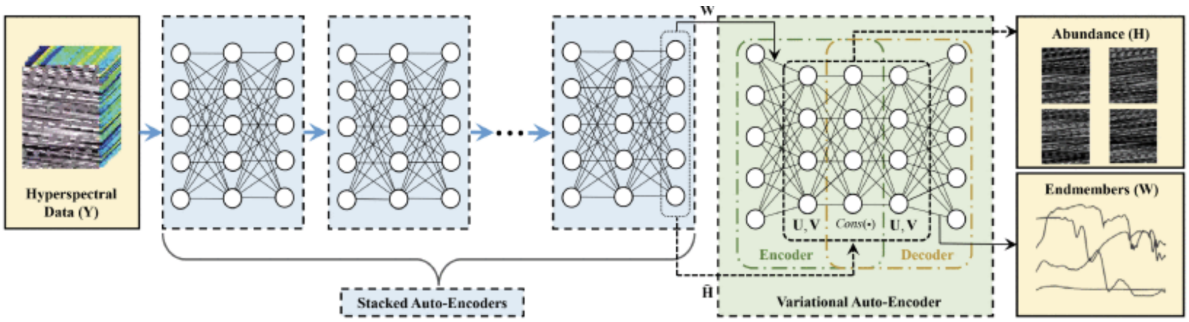

Figure 3.5: DAEN Architecture proposed by Su et al. (2019).

also explored autoencoders for pixel unmixing. Here, first, a stacked autoencoder uses VCA to identify candidate pixels based on their purity-index. Next, a variational autoencoder is used to solve the underlying non-negative matrix factorization problem. The quality of the candidate pixels identified by the stacked autoencoder influences the overall unmixing results (see Figure 3.5) for an overview of the DAEN architecture).

Another method by Shahid and Schizas (2021) uses an autoencoder architecture for hyperspectral pixel unmixing. This method requires an initial abundances computations, which is provided either via K-means clustering or via Radial Basis Functions. The decoder stage incorporates one of the following mixing models: Fan, bilinear, or Postpolynomial.

Palsson et al. (2020) propose the Convolutional Neural Network Autoencoder Unmixing (CNNAEU) model for the problem of hyperspectral pixel unmixing. Similar to the non-negative matrix factorization schemes discussed above, this method uses both spatial and spectral information when performing pixel unmixing. Unlike our model, CNNAEU assumes a linear mixing model and does not provide a generative decoder; therefore, it cannot cover higher spectral variability and is not capable of generating unseen pixels.

Our method differs from the existing schemes in an important way—given an input pixel, our method learns to construct its latent representation that models a Dirichlet Distribution, which perfectly captures the ASC and ANC constraints that arise in abundance estimation. Due to the generative nature of our architecture, the proposed

model is able to synthesize new pixel spectra given known abundances or endmembers. CNNAEU (Palsson et al., 2020), similar to our model SpACNN-LDVAE also explores spatial information. It is, however, feasible to extend our model to use a CNN-based encoder that would incorporate the spatial neighbourhood information of a pixel when constructing the latent representation. This is a potential avenue for future research. The proposed model requires training data in the form of pixel-level abundances. We show in Section 6.4.1 that it is possible to train the proposed model using only synthetic data when the "real" data is missing pixel-level abundances for training purposes.

Others, such as Li et al. (2020) have explored the use of latent dirichlet VAEs in other domains, and introduced the mathematical framework of Dirichlet Graph Variational Autoencoder (DGVAE). Their goal was to replace the Gaussian latent space with the Dirichlet latent space and their work deals with cluster membership. Kim and Kim (2023) propose a method for anomaly detection in high-dimensional data using a Dirichlet Variational Autoencoder. Xu et al. (2023) proposed a Variational Autoencoder with Dirichlet priors for feature disentanglement. This work explores the reparameterization trick using the Laplace approximation. Our method uses the reparameterization trick as presented in (Joo et al., 2020).

In summary, others have explored latent Dirichlet VAEs; however, ours is the first approach that applies this architecture to the problem of hyperspectral unmixing. Similarly, others have explored autoencoders and variational autoencoders for hyperspectral pixel unmixing, none have employed a Latent Dirichlet Variational Autoencoder for the task of pixel unmixing. Consequently, our work represents an important contribution to the field of hyperspectral pixel unmixing.

3.2 Spectral Dimensionality Reduction

Hyperspectral images (HSIs) offer a wealth of spectral information, capturing hundreds of narrow, contiguous bands across the electromagnetic spectrum. This high dimensionality, while beneficial for detailed material identification, presents significant challenges in terms of storage requirements, processing time, and computational complexity. Spectral dimensionality reduction techniques aim to alleviate these challenges by transforming the data into a lower-dimensional representation while preserving the essential spectral information necessary for downstream tasks like classification and segmentation. Crucially, effective dimensionality reduction must balance minimizing information loss with maximizing computational efficiency. This section presents a taxonomy of spectral dimensionality reduction methods, reviewing their strengths, limitations, and applicability to HSI analysis, with a particular focus on their impact on subsequent classification performance.

3.2.1 Taxonomy of Spectral Dimensionality Reduction Methods

Dimensionality reduction methods for HSIs can be broadly categorized based on their underlying principles:

1. **Feature Transformation:** These methods project the original high-dimensional spectral data into a lower-dimensional space by transforming the feature space.
    - Linear Transformations: These methods utilize linear combinations of the original bands to create new, uncorrelated features.
    - Nonlinear Transformations: These methods capture nonlinear relationships in the spectral data, often by mapping it to a higher-dimensional feature space before projection.

2. **Band Selection:** These methods select a subset of the original spectral bands deemed most informative for the target application, discarding the rest.
    - Supervised Methods: Leverage labeled data to select bands optimizing classification accuracy or other performance metrics.
    - Unsupervised Methods: Employ statistical measures or information-theoretic criteria (e.g., mutual information) to rank and select bands.
3. **Deep Learning-based Methods:** These methods utilize deep neural networks to learn complex nonlinear mappings for dimensionality reduction.
    - Autoencoders (AE): Learn a compressed representation by reconstructing the input data, often with architectural constraints for dimensionality reduction.
    - Variational Autoencoders (VAE): Learn a probabilistic latent representation suitable for generative tasks and dimensionality reduction.

3.2.2 Review and Evaluation of Methods

The increasing availability of high-resolution HSIs (Ghamisi, 2017; Lu et al., 2020) has driven the demand for efficient processing techniques. The high dimensionality, often exceeding 300 spectral bands per pixel, poses challenges for storage and computation. Spectral redundancy (Guo et al., 2006) motivates dimensionality reduction as a crucial preprocessing step.

Feature Transformation: Linear methods like PCA (Cao et al., 2003; Du et al., 2003; Rasti et al., 2018) and ICA (Du et al., 2003) have been widely used. However, the inherent nonlinearity of spectral relationships due to factors like reflection, refraction, absorption, and atmospheric effects limits their effectiveness. Nonlinear methods like KPCA, manifold learning, NLPCA (Licciardi and Chanussot, 2018), and wavelet transforms (Moser and Zerubia, 2018; Aroma and Raimond, 2020) attempt to address these nonlinearities.

Band Selection: Band selection techniques (Sun and Du, 2019) offer a direct approach to dimensionality reduction by identifying the most informative bands. While effective, the optimal subset of bands can be application-specific.

Deep Learning-based Methods: Autoencoders (AEs) (Zhou et al., 2019; Belwalkar et al., 2018; Ball and Wei, 2018) and VAEs provide powerful tools for learning nonlinear, low-dimensional representations.

A common focus in dimensionality reduction is minimizing reconstruction loss. However, for tasks like classification, preserving discriminative information is paramount. Therefore, evaluating the impact of dimensionality reduction on classification performance is crucial (Guo et al., 2006; Maggiori et al., 2018; Drumetz et al., 2019). Studies have shown the potential for accurate classification using reduced spectral information (Vidal and Amigo, 2012), and the choice of dimensionality reduction method can significantly influence classification results (Cheriyadat and Bruce, 2003; Du et al., 2003).

This work advocates for evaluating compression methods based on their impact on the final classification task rather than solely on reconstruction error, aligning with the philosophy of lossy compression techniques like JPEG that balance perceptual quality with compression ratio. Further research should explore the interplay between compression rate and classification accuracy across different dimensionality reduction methods to identify optimal strategies for specific HSI analysis tasks.

3.3 Segmentation

Image segmentation is a crucial step in Object-Based Image Analysis (OBIA) for remote sensing. OBIA approaches, which outperform traditional pixel-based schemes, have gained significant attention in recent years (Myint et al., 2011; Yu et al., 2006). OBIA utilizes objects within an image for subsequent tasks such as image classification. Its key advantage lies in the fact that objects possess richer spectral and spatial information

Figure 3.6: Example of edge-based segmentation. The algorithms search for borders, lines, and corners to identify different objects. Image from He et al. (2020).

compared to individual pixels (Blaschke, 2010, 2008; Dao et al., 2019b). The initial step in an OBIA pipeline often involves image segmentation to create these objects. Numerous image segmentation techniques have been explored over the past few decades, and they can be broadly categorized into four main groups (Hossain and Chen, 2019):

1. Edge-based methods (Martin et al., 2004; Vincent and Soille, 1991; Wang et al., 2005). An example of edge-based segmentation is depicted in Figure 3.6.

2. Region-based methods (Baatz and Schäpe, 2000; Bellens et al., 2008; Epshtein et al., 2010; Karl and Maurer, 2010; Zhang et al., 2013). Figure 3.7 shows an example of region-based segmentation.

3. Hybrid methods (Kruse et al., 1993; Yin et al., 2015). Figure 3.8 shows an example of hybrid-based methods, where initial edges are detected, followed by merging segments based on pixel similarities.

4. More recent Machine Learning (ML) techniques based on Support Vector Machines (SVMs) (Mitra et al., 2004) or Neural Networks (Kurnaz et al., 2005; Long et al., 2015).

With the increasing availability of high spatial and spectral resolution imagery from various sources, including close-range, airborne, and spaceborne platforms, there is a

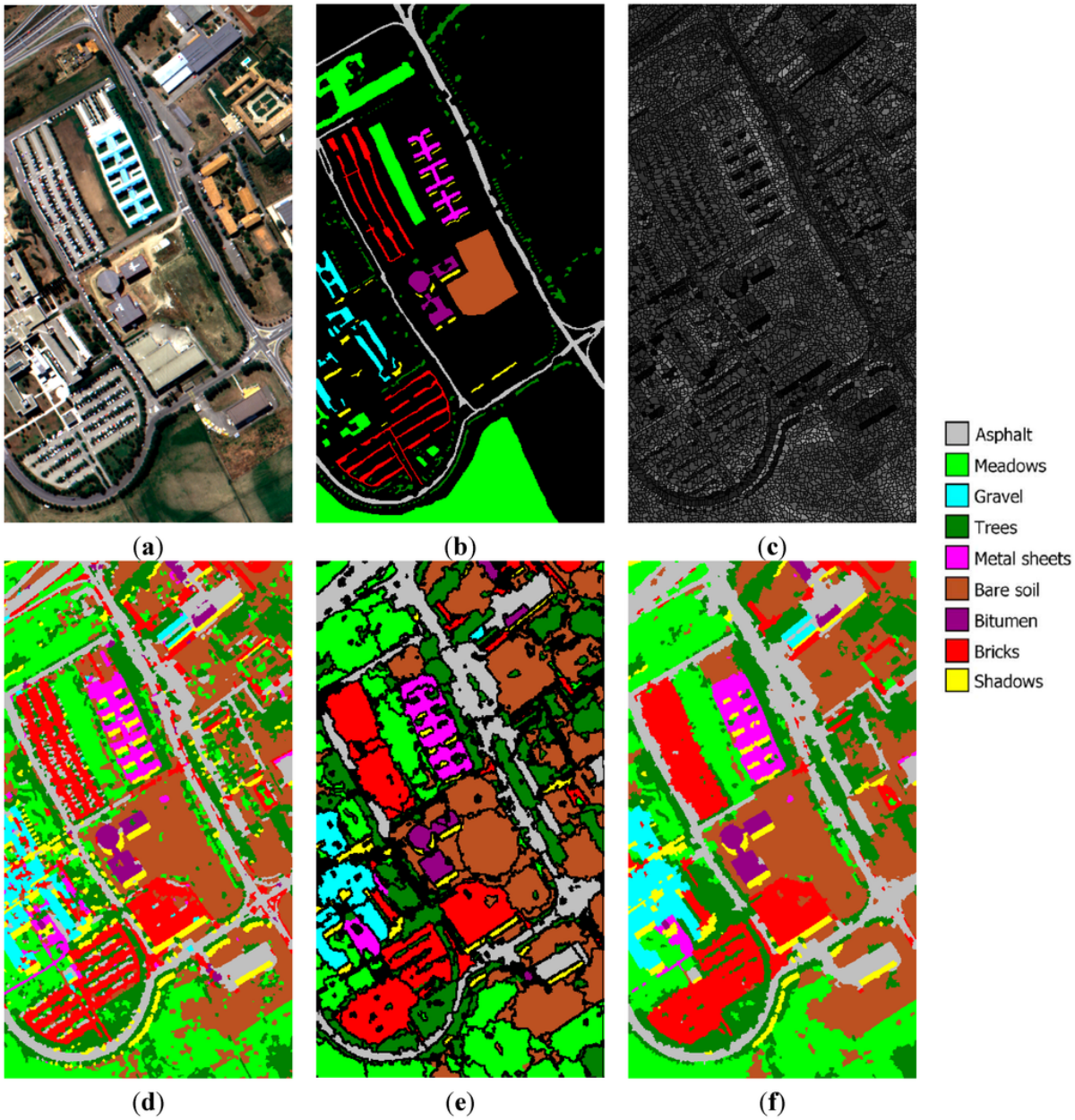

Figure 3.7: Region-based segmentation. The algorithm groups similar pixels. The criteria for similarity may vary for each algorithm. Image from Mylonas et al. (2015). University of Pavia: (a) three-band false color composite, (b) reference sites, (c) watershed segmentation map, (d) initial segmentation map after CC labeling, (e) segmentation map after GeneSIS, and (f) classification map after FMV-fusion.

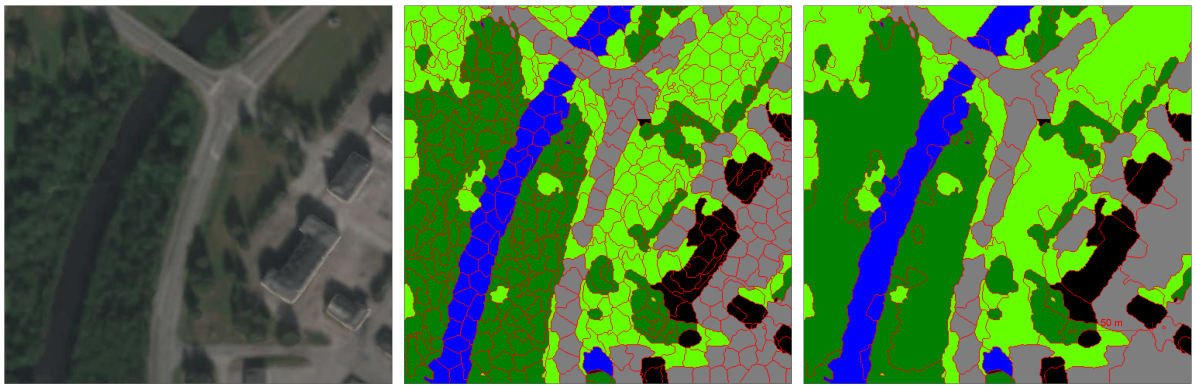

Figure 3.8: Hybrid based segmentation. Closed borders are initially detected followed by a merge of against region-based segmentation. Image from Längkvist et al. (2016).

growing demand for improved image segmentation approaches in the remote sensing community. This is particularly true for hyperspectral images. Deep learning (DL), a subfield of ML, has become one of the most potent and widely utilized segmentation methods (Gao et al., 2018; Mou et al., 2017; Nalepa et al., 2019; Zhong et al., 2017). Despite recent advancements in DL-based segmentation techniques, hyperspectral image segmentation remains an ongoing challenge. This challenge is partly attributed to the wide variations in sizes, shapes, and spectral properties of objects encountered in such imagery.

Supervised DL segmentation algorithms, often referred to as semantic segmentation, require a large amount of representative ground truth data, which comes at a high cost. This is particularly true when segmenting high-resolution hyperspectral (HrHS) images with a large number of spectral bands (Nalepa et al., 2019). On the other hand, unsupervised DL algorithms, such as the Fully Convolution-Deconvolution Network (Mou et al., 2018) for spectral-spatial feature learning, deep clustering with Convolutional Autoencoders (Guo et al., 2017), and 3-D Convolutional Autoencoders (Nalepa et al., 2019), have been proposed for segmenting images without the need for labeled data. However, the high computational expense associated with these unsupervised DL algorithms limits their practicality for hyperspectral image segmentation scale selection, which necessitates

empirical evaluation of segmentation results across a wide range of scales. Consequently, there is a need to develop robust algorithms that can meet technical requirements while ensuring computational efficiency when processing large hyperspectral datasets.

Chapter 4

Materials

This chapter details the datasets employed in this research, encompassing both real-world hyperspectral imagery and synthetically generated data. The real-world datasets provide practical scenarios for evaluating the performance of the proposed methodologies, while the synthetic data allows for controlled experiments and comparisons with existing techniques under specific conditions. Each dataset's characteristics, including spatial resolution, spectral range, and available ground truth information, are thoroughly described. The chapter also outlines the process of generating synthetic hyperspectral data, emphasizing the importance of creating realistic and diverse datasets for robust algorithm development and evaluation.

4.1 Open HSI datasets

4.1.1 Cuprite

Cuprite HSI dataset covers a region around Las Vegas, Nevada, US and comprises a 512×614, 188-channel hyperspectral image. The area under observation contains twelve minerals (or, for our purposes, endmembers): *Alunite*, *Andradite*, *Buddingtonite*, *Dumortierite*, *Kaolinite1*, *Kaolinite2*, *Muscovite*, *Montmorillonite*, *Nontronite*, *Pyrope*,

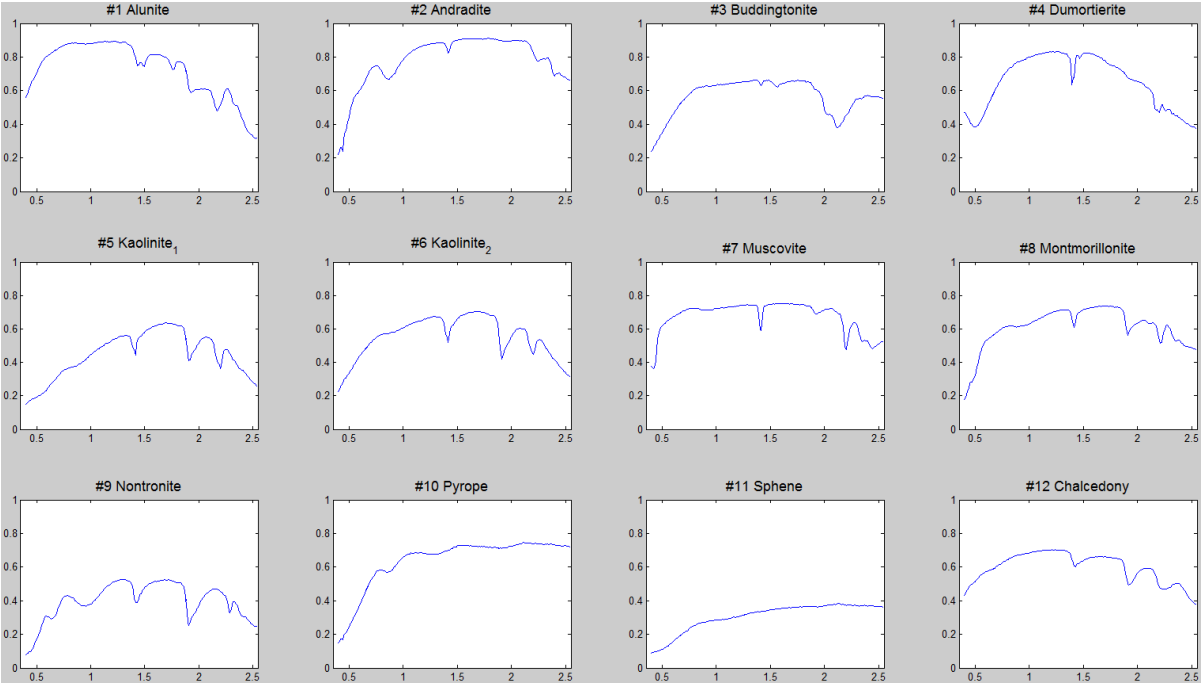

Figure 4.1: Cuprite dataset. Spectra of 12 endmembers. Image from Cuprite (2024).

Sphene, and *Chalcedony*. This dataset does not include ground truth abundances. Therefore, we generated a synthetic dataset, called *Cuprite-Synthetic*, that uses the same minerals as those present in the Cuprite dataset. The spectra of these minerals were taken from USGS spectral library. Various endmembers are randomly mixed to generate Cuprite-Synthetic pixels. We trained the model on the Cuprite-Synthetic dataset and used this model to analyze the Cuprite dataset. The results, presented in Sections 6.3 and 6.4, showcase the applicability and usefulness of using a model trained on synthetic data to analyze real data (Cuprite, 2024). Figure 4.1 shows the endmembers of the Cuprite dataset, while Figure 4.2 shows its 3D representation (datacube).

4.1.2 HYDICE Urban Dataset

HYDICE Urban dataset is a widely-used hyperspectral pixel unmixing benchmark. It comprises a 307×307, 162-channel hyperspectral image covering a $2 \times 2\text{m}^2$ region. This dataset is available in three versions, containing four, five, and six endmembers, respec-

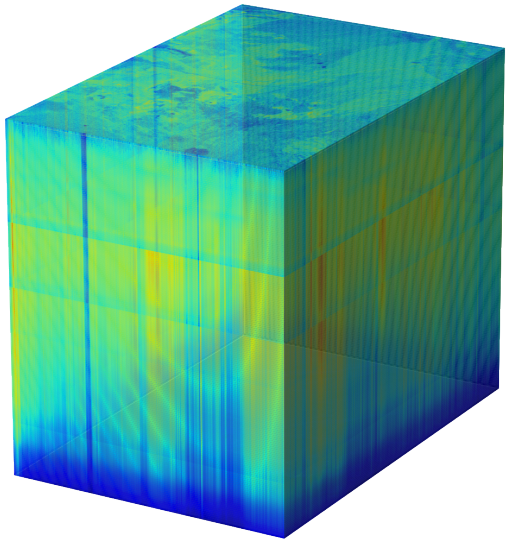

Figure 4.2: Datacube representation of the Cuprite dataset. Image generated by the author of this thesis.

tively. In this work, we use the version containing six endmembers. Further information about this dataset is available in (Zhang et al., 2022; HYDICE, 2024). Figure 4.3 shows the 3D representation of the HYDICE Urban dataset and Figure 4.4 shows the abudance maps, in which each image depicts the percentage mix of each material in a heatmap format, *i.e.* , each red color represent high percentage of the material in that (x,y) pixel.

4.1.3 Samson Dataset

Samson dataset comprises a 95×95, 156-channel hyperspectral image and contains three endmembers: soil, tree, and water (Samson, 2024). Figure 4.5 shows the 3D representation of the Samson dataset and Figure 4.6 shows the abudance maps used for training.

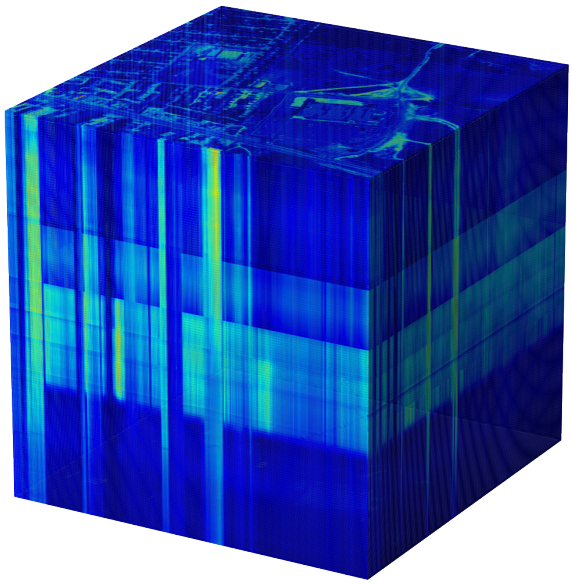

Figure 4.3: HYDICE Urban dataset.

4.2 UT-HSI301 High-Resolution

The segmentation and dimensionality reduction studies presented in this research were conducted on High-Resolution Hyperspectral Images (HRHSI) collected by researchers at the Remote Sensing Laboratory of the University of Toronto (Figure 4.7). The images and annotations were used for the first time by Dao *et al.* (Dao et al., 2021) in our collaborative investigation about hyperspectral image segmentation. These datasets were captured using the Micro-HyperSpec III sensor (from Headwall Photonics Inc., USA) mounted at the bottom of a helicopter, during the daytime at 10:30 am on August 20, 2017. The original images with 325 bands were resampled to obtain 301 bands from 400 nm to 1000 nm with an interval of 2 nm. Raw images were converted to at-sensor radiance using HyperSpec III software. The images were also atmospherically corrected to surface reflectance using the empirical line calibration method (Dao et al., 2019a) with field spectral reflectance measured by FieldSpec 3 spectroradiometer from Malvern Panalytical, Malvern, United Kingdom. These images represent 1) urban, 2) transitional

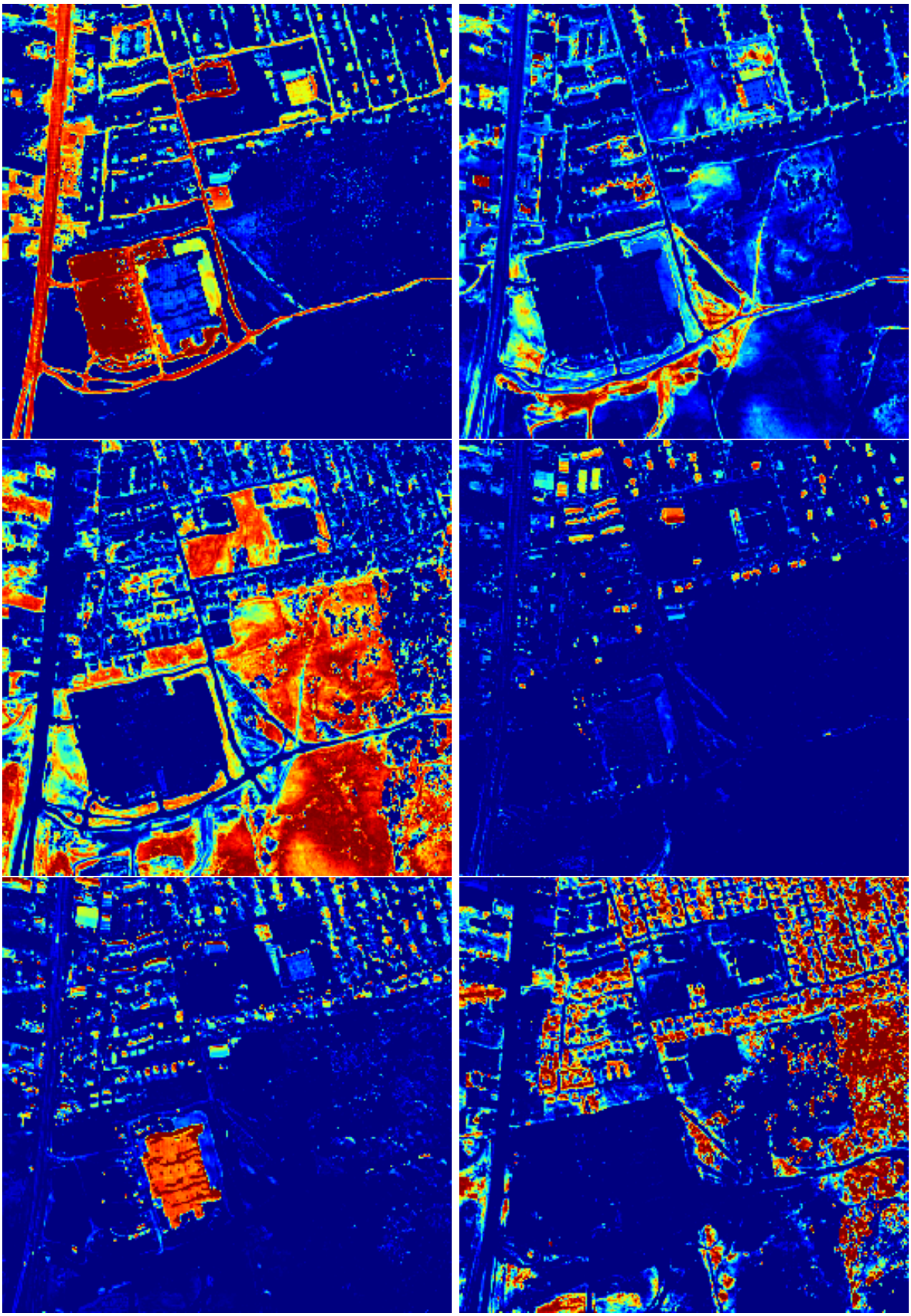

Figure 4.4: Abundance maps of the HYDICE Urban dataset; From top to bottom and left to right: asphalt, dirt, grass, metal, roof, tree

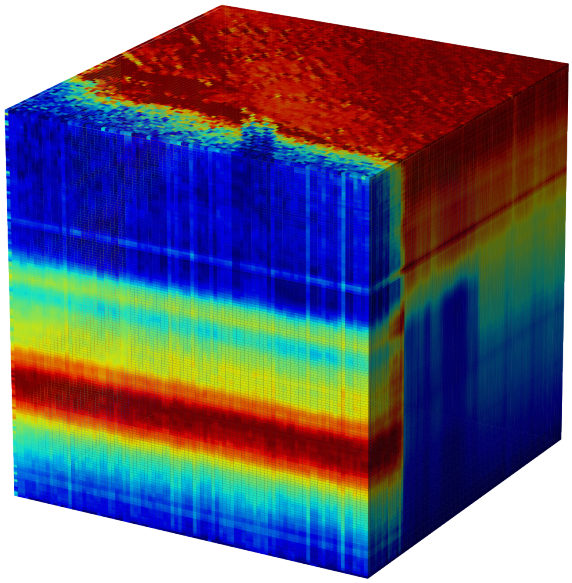

Figure 4.5: Samson dataset.

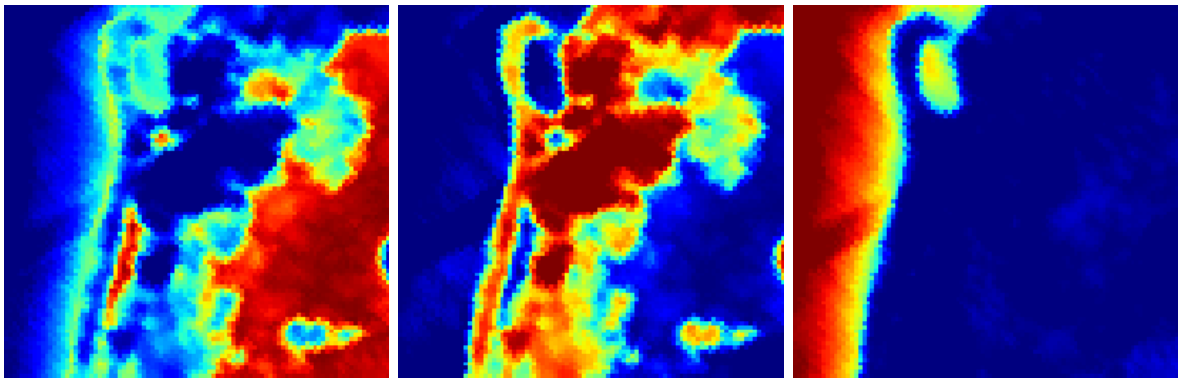

Figure 4.6: Abundances maps of the Samson dataset. From left to right respectively: Soil, Tree, and Water.

suburban, and 3) forests landcover types. These three landcover types cover a large fraction of use cases for hyperspectral imagery; urban and sub-urban images are often used for city planning and land use analysis and forest images are typically used for forest management, ecological monitoring, and vegetation analysis. The overlaid polygons in Figure 4.7 depict the annotated regions for which ground-truth pixel labels are available.

4.2.1 Suburban

Figure 4.7 (second row, left) shows the hyperspectral image collected in an urban-rural transitional area. We refer to this image as the "Suburban" dataset. It was captured around the Bolton area in southern Ontario and covers an area between 43°52′32" and 43°53′04" in latitude and −79°44′15" and −79°43′34" in longitude. This region consists of various land cover types, such as rooftops, asphalt roads, swimming pools, ponds, grassland, shrubs, urban forest, etc. The image also contains regions that are in shadows. The image resolution is 0.3 square meters and the covered area is around 41,182 square meters.

Table 4.1: splits of train,test, and validation samples for **Suburban** dataset

label	train	validation	test
Asphalt	9155	4578	4578
Rooftop	7910	3955	3955
Shadow	10385	5192	5193
Vegetation	15147	7573	7574

4.2.2 Urban

Figure 4.7 (second row, middle) shows the hyperspectral image collected in a residential urban area, also around Bolton region in southern Ontario. We refer to this image as "Urban" dataset. It contains rooftops, under-construction residences, roads, and lawn types of land cover. The dataset also exhibits regions that are in shadows. This image covers the area between 43°45′30" and 43°45′43" in latitude and −79°50′06" and

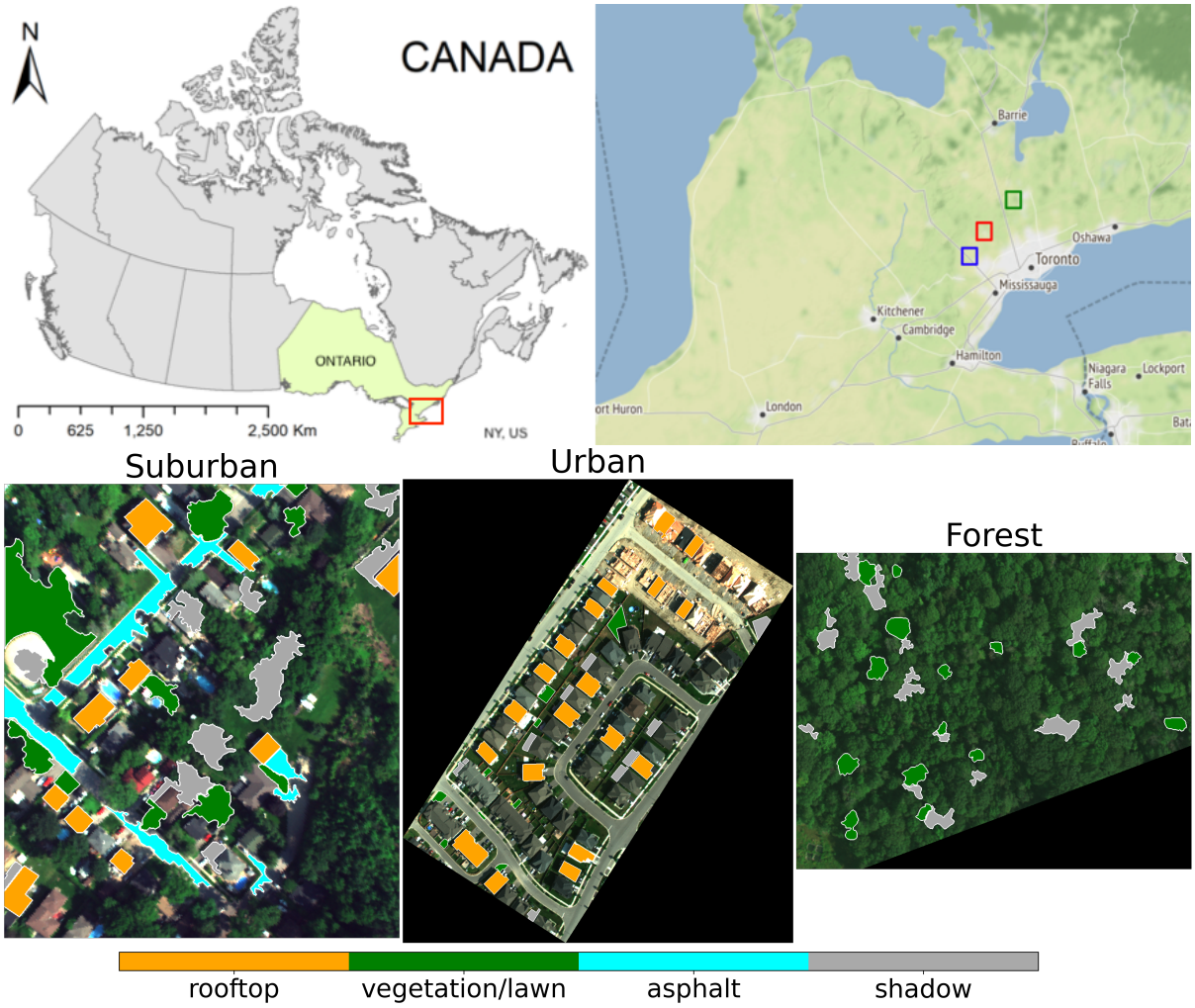

Figure 4.7: Hyperspectral datasets were collected by Remote Sensing and Spatial Ecosystem Modeling (RSSEM) Laboratory Department of Geography, Geomatics and Environment - University of Toronto using an airborne sensor over an area around Toronto, Ontario, Canada. **Top left:** Study area; **Top right:** the red, blue, and green areas represent Suburban, Urban, and Forest images, respectively (the rectangles are not to scale). **Bottom row**: shows the three datasets in pseudo color (RGB images). This visualization was constructed using the 670 nm (red), 540 nm (green), and 470 nm (blue) bands from the original HSI data. The yellow, green, blue, and gray polygons overlaid on the hyperspectral images are the areas for which ground-truth pixel labels are available.

$-79°49'51"$ in longitude. The image resolution is 0.3 square meters and the area after removing background pixels is around 59,834 square meters.

Table 4.2: splits of train,test, and validation samples for **Urban** dataset

label	train	validation	test
Lawn	3432	1716	1716
Rooftop	22323	11162	11162
Shadow	4384	2192	2192

4.2.3 Forest

[H] Figure 4.7 (second row, right) shows the hyperspectral dataset collected in a natural forest located at a biological site of the University of Toronto in the King City region in southern Ontario. We refer to this dataset as the "Forest" dataset. It covers the area between $44°01'58"$ and $44°02'04"$ in latitude and $-79°32'06"$ and $-79°31'55"$ in longitude. The image resolution is 0.3 square meters and the area after removing background pixels is around 43,084 square meters.

Table 4.3: splits of train,test, and validation samples for **Forest** dataset

label	train	validation	test
Shadow	9200	4600	4600
Tree	7343	3672	3672

4.3 Cover Crop —USDA

Cover Crop - USDA is a Hyperspectral imagery dataset collected from field plots using a DJI Matrice 600 Pro (DJI Technology Co. Ltd., Shenzhen, China) equipped with a Headwall Nano-Hyperspec camera (Headwall Photonics, Inc., Bolton, MA, USA), which collects data from 270 spectral bands over the range of $400nm$ to $1000nm$ with a spectral resolution of $2.2nm$. The imagery was collected by the United States Department of

Agriculture (USDA) over 4 species of vegetation: canola, clover, triticale, and vetch. Additionally the class "soil" is also part of the categories to represent all the materials present in a scene. There are 120 datacubes, 40 of them categorized as monocultures, *i.e.* , datacubes with only one of the species plus soil. Figure 4.8 is one example of Cover Crop USDA datacube and Figure 4.9 shows the RGB images taken from regular iPhone cameras.

4.4 HSI Synthetic Data Generator

Supervised machine learning algorithms have demonstrated remarkable success in hyperspectral image (HSI) analysis, particularly in tasks like pixel unmixing. However, the efficacy of these algorithms hinges critically on the availability of large, accurately labeled datasets. Creating such datasets for HSI analysis is a significant challenge, as generating ground truth abundance maps is often a laborious and time-consuming manual process. To address this bottleneck and facilitate the development and evaluation of HSI processing techniques, a synthetic HSI data generator was developed. This generator provides a flexible and efficient means to create realistic synthetic HSI datasets with corresponding ground truth abundance maps, eliminating the burden of manual labeling and enabling the generation of diverse training data tailored to specific research needs. We developed a synthetic hyperspectral image generator to create datasets with known ground truth abundances. The generator operates in the following steps:

1. Endmember Selection: A specified number of endmembers are manually selected from the USGS Spectral Library (U. S. Geological Survey et al., 2017). These endmembers represent the pure spectral signatures of different materials.

2. Abundance Map Generation: Synthetic abundance maps are generated by Gaussian fields method (Kozintsev, 1999; ICSynthesis, 2024). This method ensures spatial

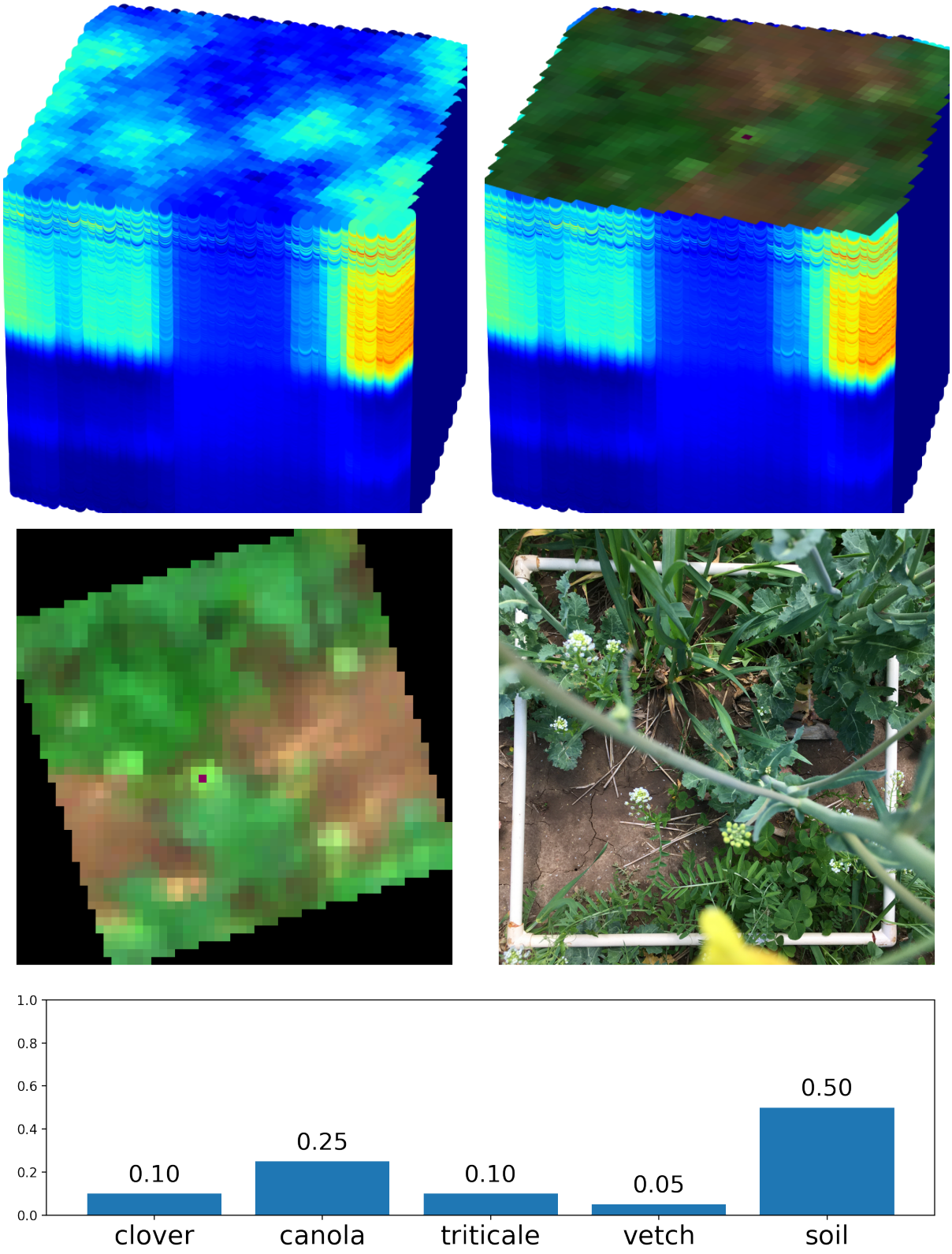

Figure 4.8: Examples of a quadrat: Datacube, Datacube with RGB projection on top, RGB composite image, iPhone High-Resolution image, and abundances (note the abundances are provided for the entire quadrat).

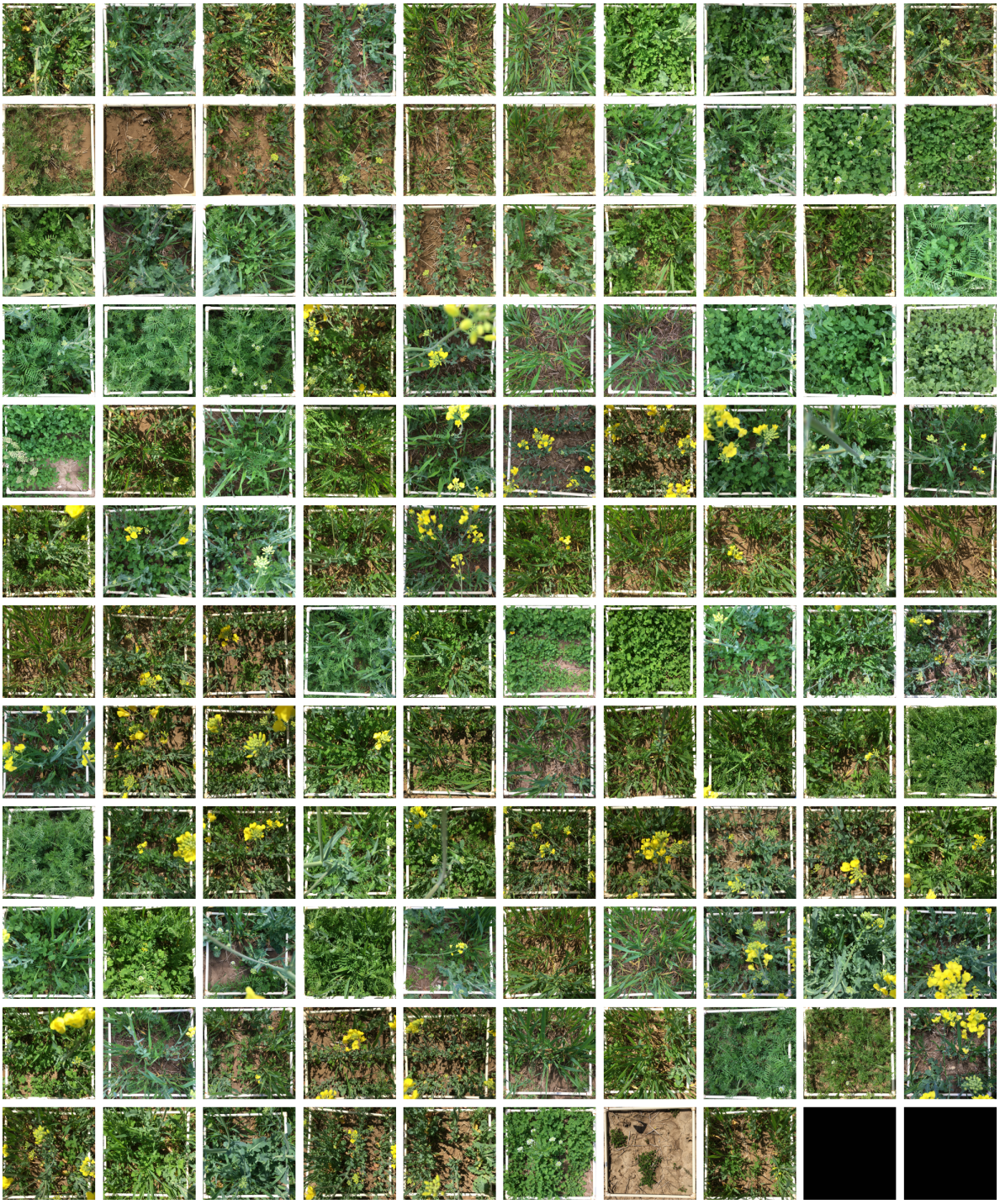

Figure 4.9: Cover Crop USDA RGB images taken with a regular iPhone camera. Each image has a correspondent datacube with 270 spectral bands.

coherence in the distribution of materials within the image while simultaneously enforcing the two fundamental constraints of abundance vectors.

- Abundance Sum-to-One Constraint (ASC): The sum of abundances for each pixel must equal one, reflecting the fact that the pixel's spectrum is a linear combination of the selected endmembers.
- Abundance Non-Negativity Constraint (ANC): All abundance values must be non-negative.

3. A matrix multiplication operation is performed between the generated abundance maps and the selected endmembers. This produces a synthetic hyperspectral image where each pixel's spectrum is a linear combination of the chosen endmembers weighted by their corresponding abundances. The synthetic dataset I is computed by:

$$I_{\text{synthetic}} = A_{(M,N)} \cdot E_{(N,B)},$$

where A is the the tensor representing abundance maps, E is tensor of endmembers, M is the number of pixels (image rows * image columns), N is the number of endmembers, and B is the number of spectral bands from the endmebers vectors .

4. Generate a synthetic datacube $I_{\text{synthetic}}$ that contains that same endmembers as the original image. Each pixel i in the generated image is

$$\sum_{j=1}^{n} a_j^i \mathbf{e}_j,$$

where n denotes the number of endmembers, $a_j^i \in [0, 1]$ and $\sum_j a_j^i = 1$.

5. Ground Truth Generation: A ground truth image is simultaneously created, containing the true abundance values for each pixel. This ground truth information is essential for training and evaluating pixel unmixing algorithms.

The generator allows for customization of parameters such as the number of channels, spatial dimensions, number of endmembers, and the spatial correlation of the abundance maps, enabling the creation of diverse and realistic synthetic datasets for a wide range of research and development tasks in hyperspectral image analysis (Mantripragada et al., 2021). The generator is available at `https://github.com/kiranmantri/hyperspectral-data-loader`.

4.4.1 OnTech-HSI-Syn-21 Synthetic Dataset

Previous authors (Li et al., 2021) and (Zhang et al., 2022) used synthetic HSI data to evaluate the performance of their proposed schemes. Specifically, these models use synthetic data with 224-channel pixels containing the following nine endmembers: *Adularia GDS57*, *Jarosite GDS99*, *Jarosite GDS101*, *Anorthite HS349.1B*, *Calcite WS272*, *Alunite GDS83*, *Howlite GDS155*, *Corrensite CorWa-1*, and *Fassaite HS118.3B*. However, the datasets used by these authors are not freely available.

Therefore, we generated two 128×128, 224-channel hyperspectral images containing the nine endmembers listed above. The spectra for these endmembers were taken from USGS spectral library. Figure 4.10 shows the endmembers spectra used in this dataset. We refer to this dataset as OnTech-HSI-Syn-21 dataset. One of the images is used for training while the second image is used for evaluation. Thus, we compare our model to methods developed by (Li et al., 2021) and (Zhang et al., 2022). This dataset is available to the research community, currently upon request, however it will be made publicly available soon at Ontario Tech University VCLab's website. Figure 4.11 shows the 3D representation of the two synthetic datasets.

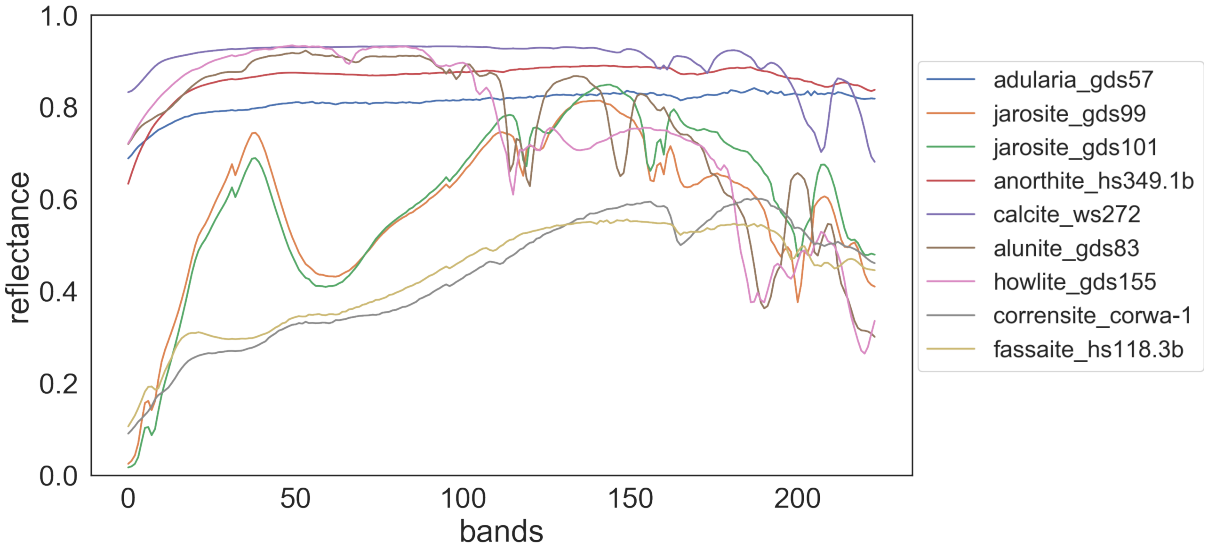

Figure 4.10: Endmember spectra (taken from USGS spectral library) used to generate OnTech-HSI-Syn-21 dataset. Each pixel represents a linear combination of these spectra where mixing coefficients are randomly drawn non-negative numbers that sum to one.

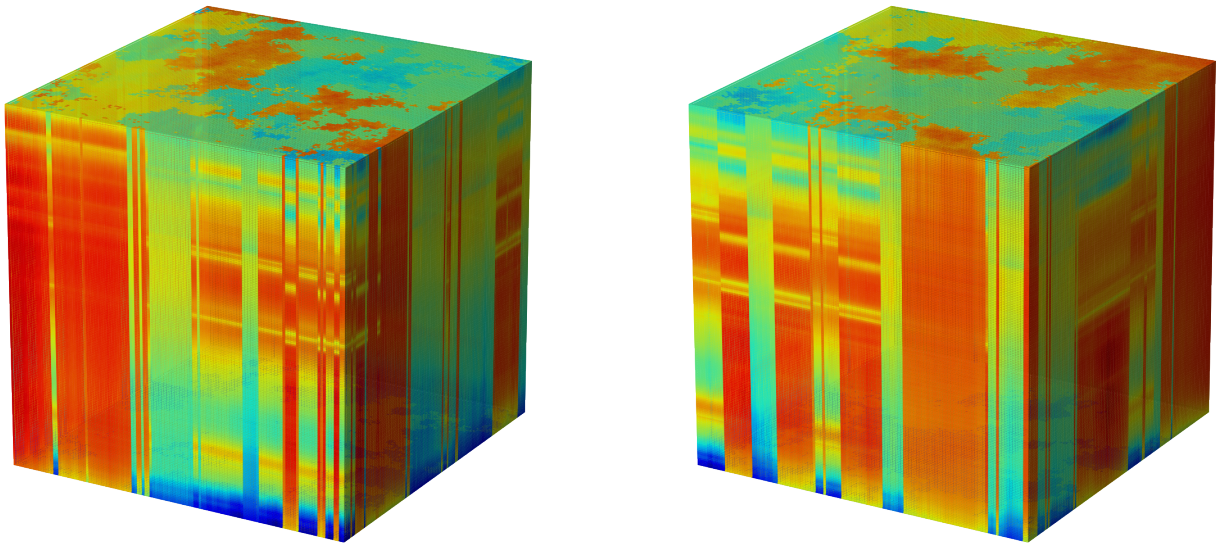

Figure 4.11: OnTech-HSI-Syn-21 Synthetic dataset. Left: Training data; Right: Validation and test data. Both were created using the USGS Spectral Library (U. S. Geological Survey et al., 2017).

Chapter 5

Methods

This chapter details the methodologies developed and employed in this research to address the challenges of hyperspectral image analysis. We begin by investigating the impact of noise and outliers on hyperspectral image segmentation and propose a robust method for optimal scale selection. Next, we explore dimensionality reduction techniques, evaluating the efficacy of both linear and non-linear approaches for compressing hyperspectral data while preserving essential information for classification. The core contribution of this chapter is the introduction of the Latent Dirichlet Variational Autoencoder (LDVAE), a novel deep learning architecture specifically designed for hyperspectral pixel unmixing. We provide a comprehensive description of LDVAE, including its architecture, training process, and the derivation of the Evidence Lower Bound (ELBO) for Dirichlet distributions. Furthermore, we present two extensions of LDVAE: an iterative approach (iLDVAE) for unmixing scenarios with limited labeled data and the integration of spatial attention mechanisms (SpACNN-LDVAE) to enhance unmixing accuracy.

5.1 Segment-level Classification

The optimal scale selection procedure proposed by Drăguţ et al. (2010), based on the Local Variance (LV) graphs and Rate of Change (RoC), is one of the most widely used scale selection procedures in remote sensing image segmentation in recent years. The scale parameter in image

segmentation governs the size and homogeneity of the resulting image objects. Selecting too small a scale can lead to over-segmentation, where meaningful features are fragmented into many small, insignificant objects. Conversely, choosing too large a scale results in under-segmentation, where distinct features are merged together, obscuring important details. Therefore, identifying the appropriate scale is crucial for capturing the desired level of detail and ensuring that the segmented objects correspond to real-world features of interest. This is particularly important in high-resolution imagery where variations in spectral and spatial properties can occur at multiple scales.

Optimal scale parameters are defined using the highest peak point of dramatic change in LV (in the form of absolute standard deviation) and RoC graphs (Dao and Liou, 2015). This method is easily implemented and reproducible that has been confirmed by an automatic scale selection tool that was created by (Drăguţ et al., 2014) using this procedure. However, the standard deviation is only an absolute measure of clustering (Sorensen, 2000) of individual segments; therefore it cannot be used for the comparison of homogeneity among segments, especially the homogeneity of neighboring segments which is needed during region merging and region splitting steps in the segmentation process.

Furthermore, the use of LV and RoC graphs without removing outliers may result in biases in the assessment of segmentation quality, leading to an inaccurate estimation of segmentation scales. To avoid these issues, we propose an improved method based on the non-outlier statistics and the curves of the Coefficient of Variation (CV), also called the relative standard deviation, (Abdi, 2010; Weber et al., 2004) and its RoC. This method is expected to be a more reliable measure of dispersion for the selection of optimal scales for HrHS images. The workflow for determining the optimal parameters and scales for image segmentation is depicted in Figure 5.1.

An image can be segmented at various levels, and the target levels of interest depend on the spatial and spectral resolutions, the complexity of the scene, and the target application of the segmented images. The selection of a suitable segmentation scale is thus essential not only for creating meaningful image objects and improving the classification performance but also for avoiding unexpected over- or under-segmentation.

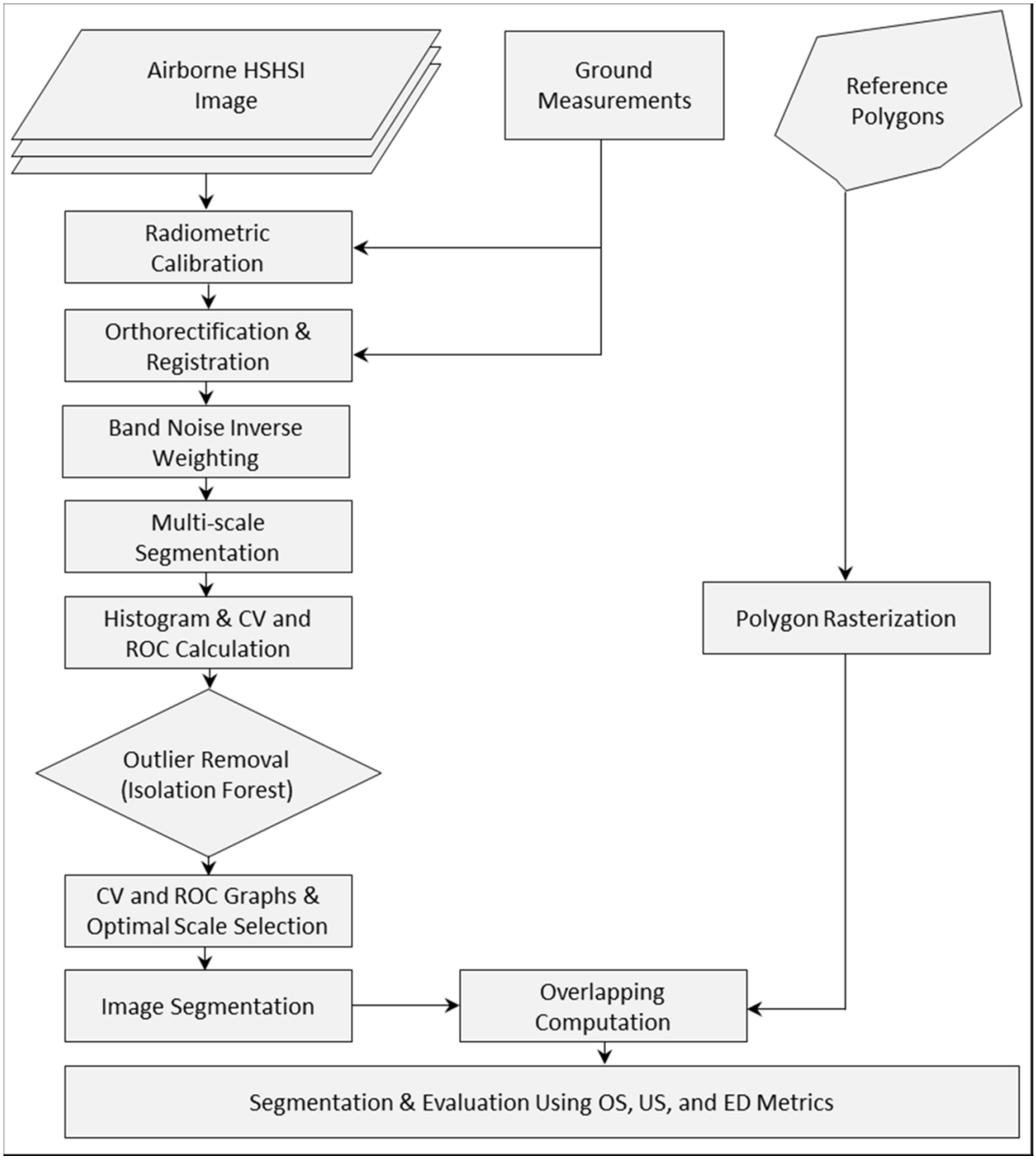

Figure 5.1: The workflow for determining the optimal parameters and scales for image segmentation.

In this study, the optimal scale selection for HrHS image segmentation proceeds as follows: 1. segmenting the HrHS image at different scales and calculating CV; 2. detecting and removing outliers using IF algorithm; 3. calculating and constructing NN-nCV and NN-nRoC graphs; 4. inspecting NN-nRoC graphs and selecting an optimal scale.

The CV of a segment in a one-band image is calculated as follows:

$$CV_p = \frac{\sqrt{\frac{\sum_{i=1}^{N}(x_i-\mu_p)^2}{N}}}{\mu_p},$$

where CV_p is the CV of segment p, N is the number of pixels within this segment, x_i is the gray value of ith pixel within the segment p, and μ_p is the mean value of all the pixels within the segment p. The averaged CV of a segment across all hyperspectral bands is expressed in the following equation.

$$CV_B = \frac{\sum_{b=1}^{B} CV_{p,b}}{B},$$

where B is the number of spectral bands, and $CV_{p,b}$ is the coefficient of variation of the segment p in band b. The final averaged CV of all segments (P) in the image is calculated as follows:

$$CV_{avg} = \frac{\sum_{P=1}^{P} CV_{B,P}}{P}.$$

After calculating CV_{avg}, segments with extreme values are removed from the calculation of the mean CV and the construction of CV and ROC graphs at all segmentation scales using the IF algorithm (Liu et al., 2012). The method is less computationally expensive and requires smaller memory. The method recursively builds an ensemble of isolation trees for a given dataset. Due to the susceptibility to isolation, abnormal observations are more likely to be isolated at a shorter distance to the root of an isolation tree compared to normal observations. IF randomly selects a feature and a splitting value between the minima and maxima. The splitting is processed if the observation is greater or smaller than the splitting value. The splitting process is stopped if the depth limit is reached. The algorithm returns an anomaly score for each observation, expressed as follows:

$$S(x,y) = 2^{\frac{-E(h(x))}{c(y)}},$$

where, y is the subsampling size, $h(x)$ is the path length of observation x, $c(y)$ is the average of $h(x)$ given y, and $E(h(x))$ is the average of $h(x)$ from a set of isolation trees. A threshold score of 0.5 is usually used in literature. A score greater than 0.5 indicates anomalies, a score smaller than 0.5 indicates normal observations, while a score equal to 0.5 indicates there are no distinct anomalies in the dataset. In this study, a score of 0.5 was used as the threshold for outlier detection. After removing the outliers, the NN-nRoC is calculated from the final NN-nCV using the following equation:

$$\text{NN-nRoC} = \frac{\text{NN-nCV}_n - \text{NN-nCV}_{n-1}}{\text{NN-nCV}_{n-1}},$$

where NN-nCV_n and NN-nCV_{n-1} are the averaged NN-nCV of all segments at scale n and $n-1$, respectively.

The final NN-nCV and NN-nRoC graphs were constructed for each of the three segmentation methods that we applied, and the change on the NN-nRoC graphs was used for optimal segmentation scale selection. Specifically, the peaks on the NN-nRoC graph where the NN-nCV changes abruptly indicate dramatic changes in intra-segment homogeneity and the segmentation scale. In this study, the first highest peak (Dao et al., 2021; Dao and Liou, 2015; Dao et al., 2019b; Drăguţ et al., 2010) on the NN-nRoC graph was selected as the optimal scale. In the following sections describe the segmentation methods used to conduct this research: 1) Multiresolution Segmentation (MRS), 2) K-means, 3) Mean-Shift, and 4) Compact Watershed segmentation.

5.1.1 Multiresolution Segmentation (MRS)

The multiresolution segmentation (MRS) algorithm (Drăguţ et al., 2010) is a bottom-up region-merging technique. It starts with single-pixel objects and iteratively merges smaller objects into larger ones. The decision to merge is based on minimizing the heterogeneity of the resulting object.

The heterogeneity h of an image object is defined as the weighted sum of color (spectral) and shape heterogeneity: $h = w_{\text{color}} \cdot h_{\text{color}} + w_{\text{shape}} \cdot h_{\text{shape}}$, where w_{color} and w_{shape} are the

weights assigned to color and shape, respectively. These weights can be adjusted by the user.

The color heterogeneity h_{color} is calculated as the standard deviation of the spectral values within the object. The shape heterogeneity h_{shape} is determined by a combination of smoothness and compactness.

The scale parameter, a crucial input to MRS, controls the degree of heterogeneity allowed within an object. A higher scale parameter allows for greater heterogeneity, leading to larger objects. The process continues until the heterogeneity of all objects is below the threshold defined by the scale parameter. Importantly, there is no direct relationship between the scale parameter and the resulting object size.

5.1.2 K-means Segmentation

K-means clustering, a simple yet powerful segmentation technique, was implemented using the FAISS library developed by Facebook AI Research Douze et al. (2024); Johnson et al. (2019). Despite its simplicity, k-means often outperforms more complex methods while maintaining computational and memory efficiency. Given an image $X \subset \mathbb{R}^d$, the algorithm seeks to partition the image pixels into k clusters C such that the sum of squared distances between each pixel and its closest cluster center is minimized. This objective can be expressed as: $d_{min} = \sum_{\mathbf{x_i} \in X} \min_{\mu_j \in C} \|\mathbf{x_i} - \mu_j\|^2$, where x_i denotes the value of the i-th pixel and μ_j represents the mean value of the j-th cluster.

The k-means algorithm employs an iterative refinement procedure to achieve this minimization. This process involves four key steps: (1) initialization of k centroids $C = \{c_1, c_2, \ldots, c_k\}$; (2) assignment of each pixel to its nearest centroid; (3) recalculation of centroids based on the mean value of the assigned pixels; and (4) repetition of steps 2 and 3 until the change in centroids falls below a predefined threshold or the centroids stabilize.

Since k-means can converge to local minima depending on the initial centroid placement, multiple runs with different initializations are often performed. The FAISS implementation addresses this by utilizing the k-means++ initialization strategy (Arthur and Vassilvitskii, 2006), which strategically chooses initial centroids to reduce the likelihood of converging to suboptimal solutions. In our experiments, the number of clusters k served as a crucial tuning

parameter for controlling the segmentation scale, allowing for exploration of different levels of detail. The method's computational efficiency, coupled with GPU compatibility, makes it particularly advantageous for high-dimensional hyperspectral data.

5.1.3 Mean-Shift Segmentation

The mean-shift algorithm, an adaptive clustering method introduced by Fukunaga and Hostetler (Fukunaga and Hostetler, 1975), has found widespread use in various applications, including remote sensing image segmentation. This technique leverages non-parametric density estimation to identify the maxima of the density function in feature space (Ming et al., 2015).

Given N data points x_i in a d-dimensional space $\mathbb{R}^d$, the kernel density estimate at a location x is given by:

$$\hat{f}_{h,K}(\mathbf{x}) = \frac{c_{k,d}}{Nh^d} \sum_{i=1}^{N} k\left(\left\| \frac{\mathbf{x} - \mathbf{x_i}}{h} \right\|^2 \right), \tag{5.1}$$

where $h > 0$ is the bandwidth parameter and $k(x)$ is the radially symmetric kernel profile, defined as:

$$K(x) = c_{k,d} k\left(\|x\|^2\right), \quad \|\mathbf{x}\| \leq 1.$$

Here, $c_{k,d}$ serves as a normalization constant. To estimate the density gradient, assuming the derivative of the kernel profile $k(x)$ exists, we introduce a profile $g(x) = -k'(x)$ and define a kernel $G(x) = c_{g,d} g(\|x\|^2)$. The mean-shift vector is then calculated as:

$$m_G(x) = C \frac{\hat{\nabla} f_K(\mathbf{x})}{\hat{f}_G(\mathbf{x})},$$

where C is a positive constant. At location x, the mean-shift vector computed with kernel G is proportional to the normalized density gradient computed with kernel K. This leads to the following expression for the mean-shift vector:

$$m_{h,G}(\mathbf{x}) = \frac{\sum_{i=1}^{N} \mathbf{x_i} g\left(\left\| \frac{\mathbf{x} - \mathbf{x_i}}{h} \right\|^2 \right)}{\sum_{i=1}^{N} g\left(\left\| \frac{\mathbf{x} - \mathbf{x_i}}{h} \right\|^2 \right)} - \mathbf{x}$$

As demonstrated by Comaniciu and Meer (2002), this vector always points towards the direction of maximum density increase. The mean-shift algorithm iteratively computes the mean-shift vector $m_{h,G}(x)$ and translates the kernel $G(x)$ accordingly. This iterative process is guaranteed to converge to a nearby point where the gradient of the density estimate is zero, effectively making mean-shift an adaptive gradient ascent method (Comaniciu and Meer, 2002; Ming et al., 2015). The sequence of successive kernel locations is given by:

$$\mathbf{y_{j+1}} = \frac{\sum_{i=1}^{N} \mathbf{x_i} g\left(\left\|\frac{\mathbf{y_j}-\mathbf{x_i}}{h}\right\|^2\right)}{\sum_{i=1}^{N} g\left(\left\|\frac{\mathbf{y_j}-\mathbf{x_i}}{h}\right\|^2\right)}, \quad j = 1, 2, \ldots$$

Extending this to a multivariate kernel with spatial bandwidth h_s and spectral bandwidth h_r, the kernel is redefined as: $K_{h_s,h_r}(x) = \frac{c}{h_s^2 h_r^p} k\left(\left\|\frac{\mathbf{x_s}}{h_s}\right\|^2\right) k\left(\left\|\frac{\mathbf{x_r}}{h_r}\right\|^2\right)$, where x_s represents the spatial component, x_r the spectral component, and p the number of spectral bands. The minimum spatial size (number of pixels M) of patches within a class can also be considered. Thus, the key parameters influencing mean-shift segmentation scale are the spatial bandwidth h_s, the spectral bandwidth h_r, and the minimum spatial size M. In this study, we introduce a parameter k representing the number of seeds used for initialization, effectively controlling the bandwidth of the density estimator. This parameter was tuned to determine the optimal segmentation scale. The mean-shift segmentation was implemented using the Scikit-learn library in Python 3.

5.1.4 Compact Watershed Segmentation

Since its introduction in 1979 (Beucher, 1979), the watershed algorithm has become a powerful image segmentation technique, widely applied in computer vision, pattern recognition, and image processing. The algorithm treats a grayscale image as a topographic surface, simulating flooding to delineate catchment basins around minima (Tarabalka et al., 2010). Each distinct basin represents an image segment. Compact watershed extends this traditional seeded approach by incorporating a compactness constraint, mitigating over-segmentation and yielding more regularly shaped superpixels (Neubert and Protzel, 2014).

The multiband compact watershed segmentation process comprises four steps: (1) per-band

gradient calculation for feature extraction, (2) combination of gradient images, (3) compact watershed segmentation, and (4) region merging.

Rather than operating directly on color images, the watershed algorithm utilizes grayscale gradient images. These images highlight transitions between regions, exhibiting high values at segment borders and minima in homogeneous areas. In this study, the Sobel operator (Chen et al., 2016; Gupta and Mazumdar, 2013) was employed to compute the 2D spatial gradient image for each band of the hyperspectral images. Because the watershed algorithm requires grayscale input, the gradient images of all bands must be combined. Three common combination methods exist: vectorial gradient, multidimensional gradient, and segmentation map combination (Tarabalka et al., 2010). Here, we employed the multidimensional gradient approach. For an N-band hyperspectral image, the combined gradient $\nabla^+ E(x)$ is calculated as: $\nabla_E^+(x) = \sum_{\lambda=1}^{N} w_\lambda \rho_E(x_\lambda)$, where $\rho_E(x_\lambda)$ represents the gradient image of band λ and w_λ denotes the corresponding weight, computed as the inverse of the band's noise. This operation yields a single two-dimensional gradient image.

Compact watershed segmentation (Neubert and Protzel, 2014) was then performed at different scales by varying the number of markers, while maintaining default values for other parameters (compactness = 10^{-5}). Gradients were computed using a 3×3 Sobel filter within the Scikit-image library, which was also used for the implementation of the compact watershed algorithm in Python 3.

5.2 Pixel-level Classification

We used the following five methods to compress pixel spectral signal: 1) PCA, 2) KPCA, 3) ICA, 4) AE, and 5) DAE. We also trained a gradient boosted tree model to classify the hyperspectral image pixels given their compressed signal. In addition, we measured the reconstruction errors by recovering the original pixel spectra from its compressed signal. Mathematically, $\mathbf{x}_i$ represents the hyperspectral pixel i. Here, $\mathbf{x}_i \in \mathbb{R}^D$ is a D-dimensional vector of real numbers. The three datasets used in this paper, namely Suburban, Urban, and Forest (Sections 4.2.1, 4.2.2, and 4.2.3), have dimensionality $D = 301$, $D = 301$, and $D = 251$, respectively.

We used a compression method $\mathcal{E}$ to construct the compressed signal $\mathbf{z}_i = \mathcal{E}(\mathbf{x}_i)$, where $\mathbf{z}_i \in \mathbb{R}^d$ and $\mathcal{E}$ is one of the following: PCA, KPCA, ICA, AE, or DAE. Here $1 \leq d < 301$ is a controllable parameter and lower values of d means higher compression rates. We computed classification labels $\mathcal{C}(\mathbf{z}_i)$ for pixel i using its compressed signal, where $\mathcal{C}$ is the gradient boosted tree classifier. We were able to recover the original signal $\hat{\mathbf{x}}_i$ from $\mathbf{z}_i$ and computed the reconstruction error as $\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|^2$, *i.e.* the Euclidean distance between the original pixel $\mathbf{x}_i$ and the reconstructed pixel $\hat{\mathbf{x}}_i$.

5.2.1 Reduced representation of Pixel Spectra

Below, we discuss the compression methods used in this paper–PCA, KPCA, and ICA– which have been widely used as dimensionality reduction methods. The two autoencoder models (AE and DAE) used in this study are discussed later in the section.

Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a dimensionality reduction technique that aims to transform a dataset with potentially correlated variables into a new set of uncorrelated variables called principal components (PCs). These PCs are ordered such that the first few PCs capture the majority of the variance present in the original data. This allows for simplification of the data while retaining most of the important information.

Let $\mathbf{X}$ be an $n \times p$ data matrix where n represents the number of observations and p represents the number of variables. We assume that the data is centered, meaning each column has a mean of zero. This can be achieved by subtracting the mean of each column from each element in that column.

The goal of PCA is to find a set of orthogonal vectors $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_p$ that maximize the variance of the projected data. The first principal component $\mathbf{w}_1$ is the direction that maximizes the variance of the projected data $\mathbf{w}_1 = \underset{\|\mathbf{w}\|=1}{\arg\max}\ \mathbf{w}^T\mathbf{S}\mathbf{w}$, where $\mathbf{S}$ is the sample covariance matrix of $\mathbf{X}$, given by:

$$\mathbf{S} = \frac{1}{n-1}\mathbf{X}^T\mathbf{X}$$

. The subsequent principal components $\mathbf{w}_2, \mathbf{w}_3, \ldots, \mathbf{w}_p$ are found iteratively, with the constraint that they are orthogonal to the previously found principal components and maximize the remaining variance. The principal components can be found by performing an eigenvalue decomposition of the covariance matrix $\mathbf{S}$:

$$\mathbf{S} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^T,$$

where $\mathbf{W}$ is a $p \times p$ matrix, whose columns are the eigenvectors (principal components) $\mathbf{w}_i$, and $\mathbf{\Lambda}$ is a diagonal matrix containing the corresponding eigenvalues λ_i. The eigenvalues represent the variance explained by each principal component, and are ordered in descending magnitude ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$).

The dimensionality of the data can be reduced by selecting the first k principal components, where $k < p$. These k components capture the most significant variance in the data. The reduced data matrix $\mathbf{Z}$ is then given by: $\mathbf{Z} = \mathbf{X}\mathbf{W}_k$, where $\mathbf{W}_k$ is a $p \times k$ matrix containing the first k eigenvectors.

In summary, PCA projects the data onto a feature space that consists of the eigenvectors of the data covariance matrix. Dimensionality reduction is achieved by discarding data dimensions with low variance. The intuition being that data dimensions that exhibit low variance contains little useful information. We refer the reader to Jolliffe (2011) and Pearson (1901) for more information on PCA.

Kernel Principal Component Analysis (KPCA)

Kernel Principal Component Analysis (KPCA) extends the capabilities of standard PCA to handle non-linear data. It achieves this by using kernel functions to implicitly map the data into a higher-dimensional feature space where linear PCA can be applied. This allows KPCA to capture non-linear relationships in the original data space.

A kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$ computes the inner product of two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ in the feature space without explicitly calculating the mapping. Commonly used kernel functions include:

- **Linear Kernel:** $k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$;

- **Polynomial Kernel:** $k(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i^T \mathbf{x}_j)^d$;

- **Gaussian (RBF) Kernel:** $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$.

Let $\mathbf{X}$ be an $n \times p$ data matrix, and Φ be a mapping function that projects the data into a higher-dimensional feature space: $\Phi : \mathbb{R}^p \to \mathcal{F}$. The kernel matrix $\mathbf{K}$ is an $n \times n$ matrix where each element is defined as: $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}_j)$. Similar to PCA, we center the data in the feature space. The centered kernel matrix is an $n \times n$ matrix with all entries equal to $1/n$.

We then perform eigenvalue decomposition on the centered kernel matrix: $\mathbf{K}_c = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$, where $\mathbf{V}$ contains the eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix with the corresponding eigenvalues. The eigenvectors are normalized such that $\mathbf{v}_i^T \mathbf{v}_i = \lambda_i$. To project a new data point $\mathbf{x}_{new}$ onto the principal components, we first compute the kernel vector between $\mathbf{x}_{new}$ and the training data points using the following equation to update k:

$$\mathbf{k}_{new} = [k(\mathbf{x}_{new}, \mathbf{x}_1), k(\mathbf{x}_{new}, \mathbf{x}_2), \ldots, k(\mathbf{x}_{new}, \mathbf{x}_n)]^T.$$

Finally, the projection of the new data point onto the i-th principal component is given by: $z_i = \mathbf{v}_i^T \mathbf{k}_{c,new}$.

In summary, Kernel PCA is an extension of the PCA in which input data are mapped to a higher dimensional space using a kernel. As per *Vapnik-Chervonenkis* theory, data mapped to a higher dimensional space provide better separability. Popular kernel choices are Gaussian, Polynomial, Radial Basis Functions, and Hyperbolic Tangent. In this work, we used a polynomial kernel, which is well-suited to capture any non-linearities present in the data. More information about KPCA is available in (Cao et al., 2003; Datta et al., 2018; Liao et al., 2010; Günter et al., 2007).

Independent Component Analysis (ICA)

Independent Component Analysis (ICA) is a statistical method used to separate a set of mixed signals into their statistically independent source signals. Unlike PCA, which focuses on maximizing variance, ICA aims to maximize the statistical independence of the recovered signals. This makes ICA particularly useful for applications like blind source separation, where the original source signals and the mixing process are unknown.

ICA relies on several key assumptions:

- **Statistical Independence:** The source signals are statistically independent. This is the core assumption of ICA.
- **Non-Gaussianity:** At most one source signal can be Gaussian. This is crucial because Gaussian signals are rotationally invariant, making it impossible to separate them based solely on second-order statistics.
- **Linearity:** The observed signals are linear mixtures of the source signals.

Let $\mathbf{s} = [s_1, s_2, \ldots, s_n]^T$ be the vector of n independent source signals, and $\mathbf{x} = [x_1, x_2, \ldots, x_m]^T$ be the vector of m observed mixed signals. The mixing process can be represented as: $\mathbf{x} = \mathbf{As}$, where $\mathbf{A}$ is an $m \times n$ mixing matrix. The goal of ICA is to find a demixing matrix $\mathbf{W}$ such that: $\mathbf{y} = \mathbf{Wx}$, where $\mathbf{y}$ is an estimate of the source signals. Ideally, $\mathbf{W}$ is the inverse of $\mathbf{A}$ (or a scaled and permuted version thereof). Various measures can be used to quantify the independence of the recovered signals. Common approaches include:

- **Kurtosis:** Measures the "peakedness" of a distribution. Super-Gaussian (leptokurtic) distributions have positive kurtosis, while sub-Gaussian (platykurtic) distributions have negative kurtosis.
- **Negentropy:** A measure of non-Gaussianity. Maximizing negentropy leads to more independent components.
- **Mutual Information:** Quantifies the dependence between random variables. Minimizing mutual information leads to more independent components.

ICA decomposes the input signal into additive subcomponents under the non-Gaussian and statistical independence assumptions. It is then possible to represent the original signal using a subset of the independent components returned by the ICA method, thereby performing data compression. We refer the reader to Du et al. (2003); Comon (1994); Hyvärinen and Oja (2000); Stone (2004); Goodfellow et al. (2016) for further details on ICA.

Autoencoders (AE)

An autoencoder is an unsupervised neural network architecture that learns efficient data codings in an unsupervised manner. It aims to learn a lower-dimensional representation (encoding) of the input data, and then reconstruct the original input from this representation (decoding). This process forces the network to capture the most salient features of the input data. An autoencoder consists of three main components:

1. **Encoder:** This component maps the input data ($\mathbf{x} \in \mathbb{R}^d$) to a lower-dimensional latent space representation ($\mathbf{z} \in \mathbb{R}^p$), where ($p < d$). The encoder can be represented by a function f_θ: $\mathbf{z} = f_\theta(\mathbf{x})$, where θ represents the encoder's parameters (weights and biases). The encoder often comprises multiple layers of neurons with non-linear activation functions, such as sigmoid, ReLU, or tanh. A typical encoder layer can be mathematically represented as: $\mathbf{h}_{l+1} = \sigma(\mathbf{W}_l\mathbf{h}_l + \mathbf{b}_l)$, where $\mathbf{h}_l$ is the output of layer l, $\mathbf{W}_l$ is the weight matrix, $\mathbf{b}_l$ is the bias vector, and σ is the activation function. The final layer's output is the latent representation $\mathbf{z}$.

2. **Latent Space (or Encoded) Representation:** This is the lower-dimensional representation of the input, also known as the bottleneck or code. It captures the essential features of the input data. The dimensionality p of the latent space controls the compression rate and the amount of information retained.

3. **Decoder:** This component maps the latent representation $\mathbf{z}$ back to the original input space, producing a reconstruction $\hat{\mathbf{x}} \in \mathbb{R}^d$. The decoder can be represented by a function g_ϕ: $\hat{\mathbf{x}} = g_\phi(\mathbf{z})$,where ϕ represents the decoder's parameters. Similar to the encoder, the decoder typically consists of multiple layers of neurons with non-linear activation

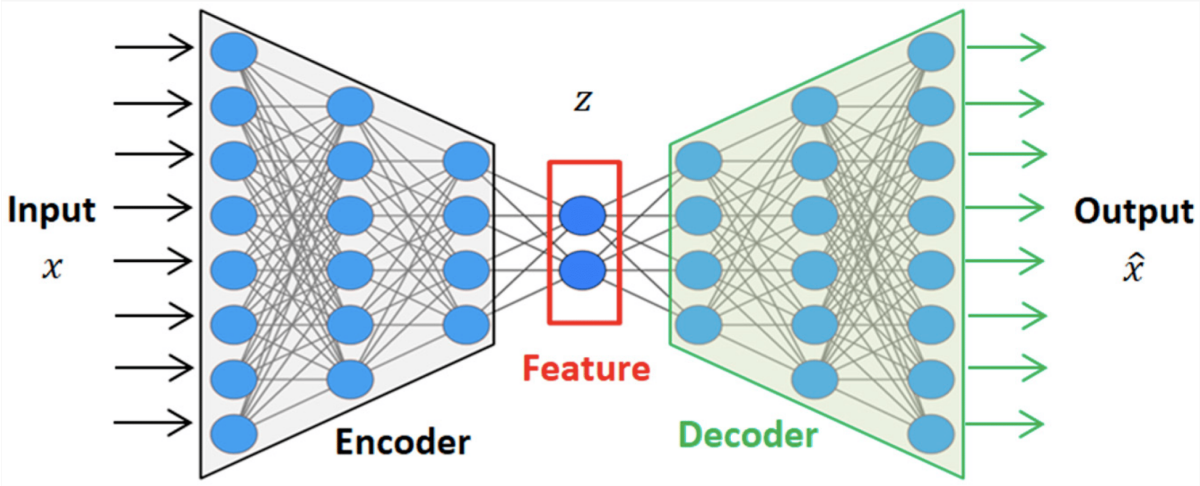

Figure 5.2: General architecture of autoencoders. Image from Alaghbari et al. (2023)

functions. The architecture of the decoder is often symmetrical to the encoder, but not necessarily identical.

The autoencoder model (Figure 5.2) is trained to minimize the reconstruction error, which is the difference between the input $\mathbf{x}$ and the reconstruction $\hat{\mathbf{x}}$. Common loss functions include:

- Mean Squared Error (MSE): $\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\mathbf{x}}_i)^2$;
- Binary Cross-Entropy (BCE): $\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}}) = -\frac{1}{n}\sum_{i=1}^{n}[\mathbf{x}_i \log(\hat{\mathbf{x}}_i) + (1 - \mathbf{x}_i)\log(1 - \hat{\mathbf{x}}_i)]$.

The parameters θ and ϕ of the encoder and decoder are optimized using backpropagation and gradient descent algorithms. There are also variations of the basic autoencoder architecture exist, including:

- **Undercomplete Autoencoders:** These force the latent space to be smaller than the input dimension, encouraging the network to learn a compressed representation.
- **Overcomplete Autoencoders:** These have a latent space larger than the input dimension. Regularization techniques are necessary to prevent the network from simply copying the input.
- **Denoising Autoencoders:** These are trained to reconstruct the original input from a corrupted version, learning robust features and improving generalization.

- **Variational Autoencoders (VAEs):** These learn a probabilistic distribution over the latent space, enabling generative capabilities.

We used the AE model proposed by Hinton and Salakhutdinov (2006). It consists of two parts: 1) an encoder, which transforms the input signal $\mathbf{x}$ into a lower-dimensional signal $\mathbf{z}$; and 2) a decoder, which reconstructs the original signal $\hat{\mathbf{x}}$ from the latent representation $\mathbf{z}$. Specifically, the encoder contains of a single hidden layer, and it transforms 301 dimensional pixel spectra into a d dimensional vector. The decoder also consists of a single hidden layer, and it reconstructs the 301 dimensional signal from a d dimensional vector. Both encoder and decoder use ReLU (Rectified Linear Unit) activation functions for the hidden layers. The output layer of the decoder uses the Sigmoid activation function as the expected values of the reconstructed signal are restricted to the values of reflectance, i.e., between 0 and 1. We refer the reader to (Wang et al., 2016) for technical details about our autoencoder model. The number of elements (i.e., neurons) in the hidden layer is a hyperparameter. We used the grid search approach to estimate a "good" value for this hyperparameter. During hyperparameter selection we set the compression rate equal to 99% (i.e., d was set to 4).

Denoising Autoencoders (DAE)

It is well-known that hyperspectral images exhibit a higher degree of noise as compared to the noise present in ordinary RGB images. Furthermore, the level of noise present in different bands of a hyperspectral image varies between bands. Atmospheric water vapor, for example, affects near-infrared bands more than higher frequency bands. If left untreated, noise will place an adverse effect on the subsequent processing and analysis tasks, such as compression, segmentation, or classification. We implemented a denoising autoencoder, which accounts for the noise present in the signal, for compressing the input spectral signal (Ball and Wei, 2018; Gondara and Wang, 2018). Denoising autoencoder also consists of an encoder and a decoder. The encoder consists of two hidden layers. The first hidden layer contains 400 neurons and the second hidden layer contains 500 neurons. The decoder also consists of two hidden layers. The first hidden layer contains 500 and the second hidden layer contains 400 neurons. All hidden

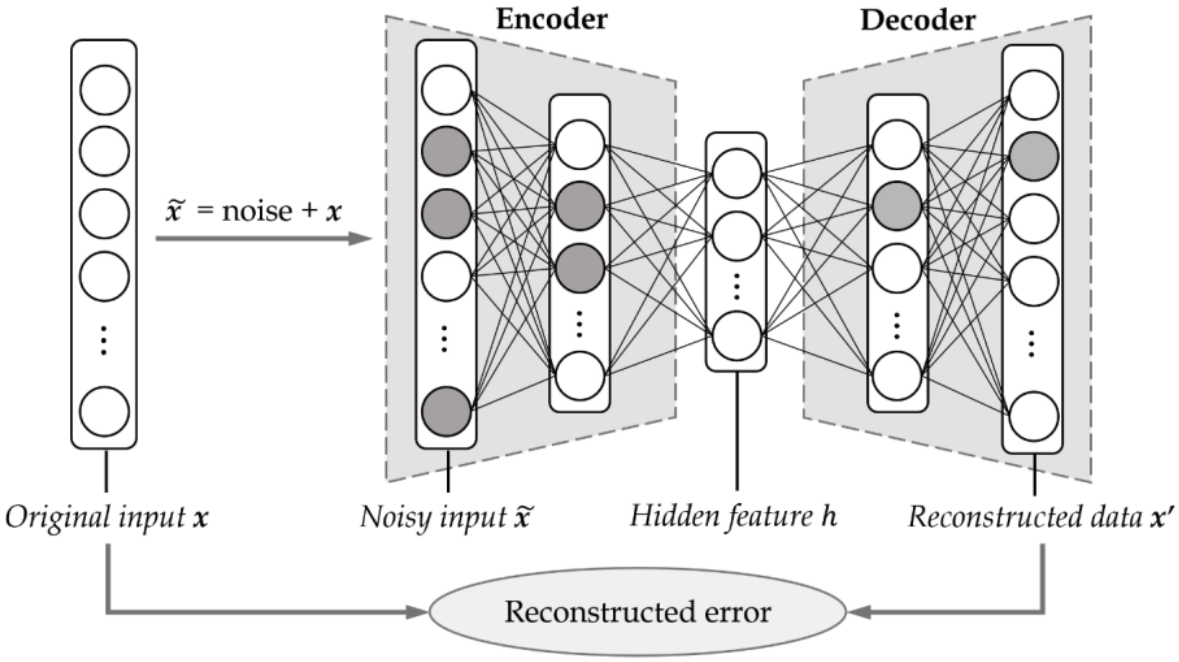

Figure 5.3: General architecture of denoising autoencoders. Note part of input are artificially corrupted to simulate random noise on the input data. Image from Park et al. (2019)

layers use ReLU activation function. Decoder's output layer uses Sigmoid activation function. Figure **??** shows the architecture of a DAE.

We selected Saviszky-Golay (SG) algorithm to understand the effect of noise on hyperspectral images. It is a widely used noise filtering method for hyperspectral images, to construct clean spectral signals (Vaiphasa, 2006; Ruffin and King, 1999). Figure 6.5 shows spectral curves for a randomly selected pixel in the three datasets. Figure 6.5 (middle) suggests that HSI+SG+DAE model did poorly in signal reconstruction, especially in the 500–800 nm range. We observe a similar trend for other pixels in the dataset. It appears that SG+DAE strongly attenuates the signal in this range. This confirms that it is unnecessary and perhaps counter-productive to use a denoising preprocessing step when using a denoising autoencoder to compress hyperspectral signal. The second row of Figure 6.5 shows the SNR (Signal-to-Noise ratio) of each compression method compared to the original HSI image. This result demonstrates that AE and DAE methods can improve the SNR of the signal, suggesting that a pre-processing step for denoising is not necessary when AE or DAE is used as a compression algorithms.

Training regime for AE and DAE

Both autoencoder and denoising autoencoder were trained using reconstruction loss, which is defined as $\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|^2$. In our experiments, both autoencoders were able to achieve low reconstruction errors even for high compression rates. Tables 4.1, 4.2, and 4.3 list the number of training and testing samples for the suburban, urban, and forest datasets, respectively. Each model was trained for 30 epochs using Adam optimizer. We trained each model ten times to capture the model variance. Figure 6.6 shows reconstruction errors for the three datasets for ten different runs for AE and DAE models. As expected, the reconstruction errors for DAE models exhibit a larger variance than those for AE models. For each image we selected the model with the lowest reconstruction error to be used as the compression method in the final classification pipeline.

5.2.2 Gradient Boosted Tree Classifier

We employed a Gradient Boosted Tree (XGBClassifier) classifier for pixel classification (Géron, 2019; Vasilev et al., 2019). XGBClassifier is a widely used ensemble model and similar to other ensemble methods, it avoids overfitting and offers good generalization properties (Breiman, 2001). It is also easy to construct intuitive interpretations of how this model arrives at a particular classification decision. We used the XGBoost library to setup our classification model. In our model, the number of trees was set to 10 and the maximum depth per tree was also set to 10.

5.2.3 Classification Metrics

We used three metrics to evaluate the accuracy of classifications. *Precision* is defined as

$$Precision = \frac{t_p}{t_p + f_p},$$

Recall is defined as

$$Recall = \frac{t_p}{t_p + f_n},$$

and *f1-score* is defined as the harmonic mean of precision and recall as follows:

$$\textit{f1-score} = \frac{t_p}{t_p + \left(\frac{f_p + f_n}{2}\right)},$$

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

5.2.4 Limitations and Scope

The dimensionality reduction methods are (1) data-driven and (2) unsupervised. Consequently, there is a very good chance that it will be able to deal with imaging artifacts (presence of clouds, low-light conditions, atmospheric noise, etc.). The classification method is supervised and requires a set of labelled pixels for training. This suggests that if a classifier that is trained using artifact-free pixels is used as is to analyze pixels with artifacts, the performance of this classifier may suffer. On the other hand, it is perhaps possible to improve the performance of this classifier by training it on pixels exhibiting artifacts. We plan to investigate the interplay of hyperspectral pixel compression and classification on data collected under different environmental and lightning setting in the future.

5.3 Pixel Unmixing — Classification at the subpixel level

The problem of pixel unmixing is similar to the topic modeling problem that aims to discover the topics in a collection of documents and how these topics are related specifically to each individual documents in this collection (Blei et al., 2003, 2001). We can extend this idea to the problem of pixel unmixing as follows: (1) the hyperspectral image is analog to the collection of documents; (2) each pixel is analog to a document: and (3) each endmember is analog to topic. The endmembers are unknown *a priori*. Additionally, for any given pixel, the mixing ratios of these endmembers (abundances) are unknown. Within this setting, we can leverage

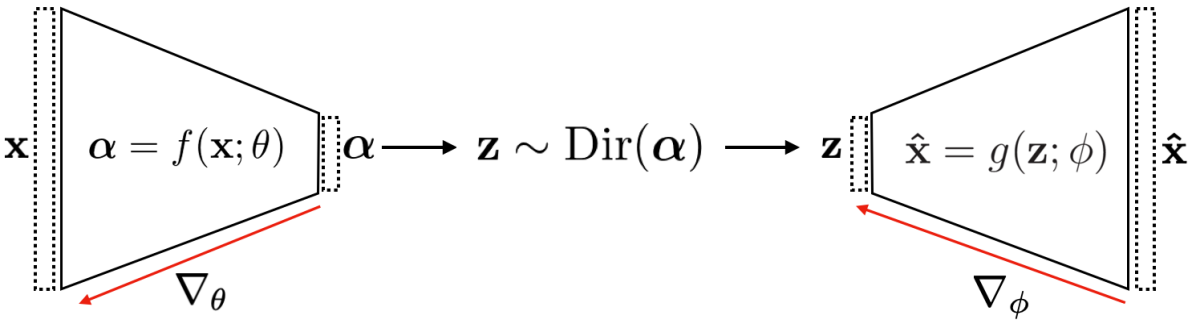

Figure 5.4: Latent Dirichlet Variational Autoencoder.

techniques available in the topic modeling literature for the problem of hyperspectral pixel unmixing. Latent Dirichlet Allocation (LDA) is a popular technique for topic modeling that, given a corpus, aims to (1) discover these latent topics and (2) estimate to what degree each topic contributes to a particular document. Inspired by LDA, we represent the abundances as a Dirichlet Distribution. Thus, hyperspectral pixel unmixing becomes the problem of constructing the latent representation that follows a dirichlet distribution. We also seek a method that reconstructs the spectra given a set of endmembers and their mixing ratios. We propose that both of these tasks can be accomplished with LDVAE, which we describe in the following section.

5.3.1 Latent Dirichlet Variational Autoencoder (LDVAE)

We implemented our model using the VAE architecture as presented in Figure 5.4. The encoder function, parameterized by θ, outputs the parameters α of a dirichlet distribution. The abundances $\mathbf{z}$ are sampled from the dirichlet distribution and fed to the decoder, which reconstructs the spectral signal $\hat{\mathbf{x}}$. The decoder is paramterized by ϕ. The input $\mathbf{x}$ represents the pixel spectra and $\mathbf{z}$ is a sample from the dirichlet distribution in the n-simplex form.

The forward pass includes generating a sample from a dirichlet distribution. However, sampling is not differentiable, which prevents the backpropagation of gradients ∇_θ and ∇_ϕ. Therefore, we need to apply the reparameterization trick, similarly explored by Kingma *et al.* on Multivariate Normal Distribution (Kingma and Welling, 2014). Specifically, for Dirichlet Distribution, we follow the method proposed in (Joo et al., 2020) and apply a reparameterization

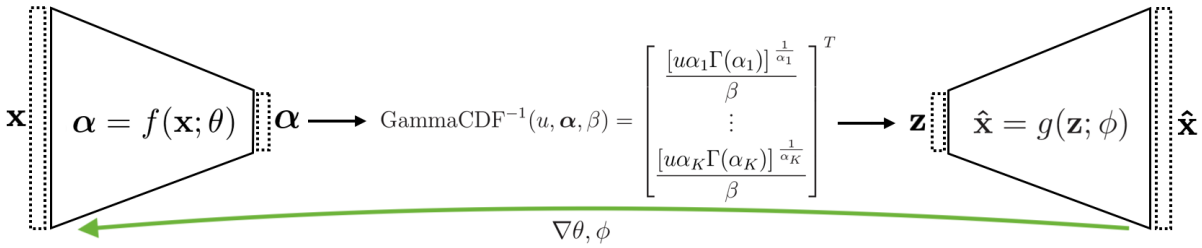

Figure 5.5: Inverse Gamma Cumulative Distribution Function as a replacement for the sampling function of a Dirichlet probability distribution.

as follows:

$$\mathbf{z} \sim \text{GammaCDF}^{-1}(u,\boldsymbol{\alpha},\beta) = \left[\frac{[u\alpha_1\Gamma(\alpha_1)]^{\frac{1}{\alpha_1}}}{\beta},\cdots,\frac{[u\alpha_K\Gamma(\alpha_K)]^{\frac{1}{\alpha_K}}}{\beta}\right]$$

The Dirichlet Probability Density function can be recasted as a Multivariate Gamma, so it becames possible to sample from the dirichlet distribution using the Inverse Gamma Cumulative Distribution Function (Equation 5.3.1). Here, $\Gamma(.)$ is the Gamma function, $\boldsymbol{\alpha}$ is the concentration parameter, a vector with size K, β is a normalization factor to ensure that the vector $\mathbf{z}$ is in the n-simplex form, and $u \sim U(0,1)$. Figure 5.5 illustrates how the reparameterization of the dirichlet function into the Inverse Gamma Function allows the network to be differentiable.

The variational autoencoder is trained using a reconstruction loss and an Evidence Lower Bound (ELBO) loss. The decoder reconstructs the input spectra given $\mathbf{z}$, i.e., the abundances. This serves two purposes: 1) the decoder is able to construct spectra given previously unseen combination of abundances and 2) the decoder is able to perform endmember extraction. Pragmatically, the decoder generates the endmembers; however, we refer to this process as "endmember extraction" to align it with the prevalent terminology in the hyperspectral pixel unmixing community. The intended purpose (1) further implies that the proposed model is capabble of generating synthetic data that mimics the characteristics of the "real" data used to train the model. The model assumes that spectra follows a multivariate Normal distribution as seen below:

$$
\begin{aligned}
&\mathbf{x} \sim \text{Normal}(\mathbf{x}; \mu, \Sigma) \text{ where} \\
&\mathbf{x} = \{x_1, x_2, x_3, \ldots, x_k\}, \\
&\mu = \{\mu_1, \mu_2\, \mu_3, \ldots, \mu_k\}, \text{ and} \\
&\Sigma = \text{diag}(\sigma_1, \sigma_2, \sigma_3, \ldots \sigma_k).
\end{aligned} \tag{5.2}
$$

Here k denotes the number of spectral bands. Note that in the current setup, each individual band are not correlated, *i.e.* Σ is a diagonal matrix.

For variational autoencoders, in addition to minimizing the reconstruction loss during training, the Kullback-Leibler (KL) divergance between the distribution induced by the latent representation and the desired distribution is also minimized during training. In our setup the latent representation α parameterizes a dirichlet distribution that leads us to the ELBO loss:

$$
\mathcal{L}(\mathbf{x}; \theta, \phi) = \mathbb{E}_{q_\theta} \left[\log p_\phi(\mathbf{x}|\mathbf{z})\right] - \text{KL}\left[q_\theta(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})\right]. \tag{5.3}
$$

For more details on the derivation of the ELBO loss, please refer to the Appendix A.1 and the works of (Pinheiro Cinelli et al., 2021; Blei et al., 2017; Kingma and Welling, 2014; Fox and Roberts, 2012). In Equation 5.3 the first term on the right-hand-side represents the reconstruction loss. The second term on the right-hand side of Equation 5.3 is a tractable KL-divergence term and it represents the divergence between the prior distributions $p(\mathbf{z})$ and the estimated $q_\theta(\mathbf{z}|\mathbf{x})$. Following (Joo et al., 2020), we re-write the KL term to account for the dirichlet distribution as follows:

$$
\begin{aligned}
\text{KL}\left[q(\mathbf{z}|\mathbf{x}; \hat{\alpha}) \| p(\mathbf{z}; \alpha)\right] = &\sum \log \Gamma(\alpha_k) - \sum \log \Gamma(\hat{\alpha_k}) \\
&+ \sum (\hat{\alpha_k} - \alpha_k) \frac{d}{dx} \ln \Gamma(\hat{\alpha_k}),
\end{aligned} \tag{5.4}
$$

where $\Gamma(.)$ is the Gamma function, α is the concentration parameter of the Dirichlet prior, and $\hat{\alpha}$ is concentration parameter of the estimated Dirichlet distribution. For more details on the derivation of the KL divergence for Dirichlet distributions, please refer to the Appendix A.2

5.4 Model training in the absence of ground truth abundances

The proposed model leverages pixel-level abundance data for training. However, such ground truth is often scarce in real-world applications and prohibitively expensive and time-consuming to acquire. In certain scenarios, while the overall material composition may be known, the precise mixing proportions remain unavailable. However, spectral signatures of endmembers are generally accessible. Consequently, these signatures can be employed to generate synthetic hyperspectral images (HSIs) with known abundances. Such synthetic HSIs can then be utilized for pre-training models within a transfer learning paradigm.

In this research we propose the use of transfer learning to train models, as follows. Say, we are given a hyperspectral image $\mathbf{I}$ alongwith the list of endmembers $\mathbf{e}$ present in this image. In this scenario there is no ground truth abundances of $\mathbf{I}$. Because pixel-level abundance information is missing, we cannot use $\mathbf{I}$ for model training. Instead, we generate a synthetic datacube as described in Section 4.4. The model is trained on $\mathbf{I}_{\text{synthetic}}$, then the trained model is subsequently used to analyze the original image $\mathbf{I}$. We show that the proposed model is able to exploit this approach to analyze Cuprite dataset where pixel-level abundances are not available.

5.5 Model training in the absence of ground truth abundances and endmembers

In many real-world scenarios, obtaining ground truth data for hyperspectral unmixing can be prohibitively expensive or even impossible. This lack of labeled data poses significant challenges for training traditional supervised algorithms. Without knowledge of the true endmembers or their corresponding abundances, the learning process becomes inherently more complex. To resolve this problem, we introduce iLDVAE, an iterative algorithm designed to address this challenge by estimating endmembers and abundances without relying on any form of ground

truth. The core idea behind iLDVAE is to leverage the concept of pixel purity as a proxy for endmember presence and iteratively refine these estimates through an analysis-synthesis loop (see Figure 5.6).

iLDVAE is an iterative algorithm that uses LDVAE (Mantripragada and Qureshi, 2024) within an analysis-synthesis loop to estimate endmembers and their abundances over successive iterations without using any labelled data (See Algorithm 1). iLDVAE implements the idea that pixels with high purity-index can serve as proxy for endmembers. Given a target image $\mathbf{I}$ that needs to be *unmixed*, at each iteration k, N randomly selected pixels with purity-index above a certain threshold are chosen as endmembers $\{\mathbf{e}_i^{(k)}\}_{i=1}^N$ that are used to synthesize a hyperspectral image $\mathbf{I}^{(k)}$. Each pixel in $\mathbf{I}^{(k)}$ contains $\{\mathbf{e}_i^{(k)}\}_{i=1}^N$ in random proprotions. LDVAE is trained on $\mathbf{I}^{(k)}$ and subsequently used to estimate endmembers $\{\mathbf{e}_i^{(k+1)}\}_{i=1}^N$ and per-pixel abundances $\mathbf{a}_{(x,y)}$ in $\mathbf{I}$, where subscript (x, y) denote pixel at that location and $\mathbf{a}_{(x,y)} \in \mathbb{R}^N$. The estimated endmembers along with their per-pixel abundances serve to compute the purity-index for pixels in $\mathbf{I}$ for the next step. The process stops when loop termination conditions are met. We discuss loop termination in the following section.

Algorithm 1 iLDVAE algorithm for hyperspectral pixel unmixing

Require: Target image $\mathbf{I}$
Require: The number of endmembers N present in the target image
Ensure: Estimated endmembers $\{\mathbf{e}_i\}_{i=1}^N$
Ensure: Estimated per-pixel abundances $\{a_i\}_{i=1}^N$ where $a_i \geq 0$ and $\sum_i a_i = 1$

1: $k = 0$ ▷ iteration count
2: Randomly select N pixels from $\mathbf{I}$ ▷ initial endmembers
3: Use $\{\mathbf{e}_i^{(k)}\}_{i=1}^N$ to synthesize $\mathbf{I}^{(k)}$ hyperspectral image where each pixel has random, but known, abundances
4: Train LDVAE on $\mathbf{k}^{(t)}$
5: Increment k
6: Use the LDVAE trained in the Step to estimate endmembers $\{\mathbf{e}_i^{(k)}\}_{i=1}^N$ and per-pixel abundances in $\mathbf{I}$
7: **if** err $\leq \epsilon$ **then** ▷ see Sec. 5.5.1 for details about err
8: Terminate loop
9: Collect pixels for which $a_i >$ pure-pixel-threshold ▷ a_i denotes the abundance value for endmember i
10: Randomly select N pixels from the set of pixels in the previous step
11: **if** Maximum numbers of iterations reached **then**
12: Terminate loop
13: Go to Step 3

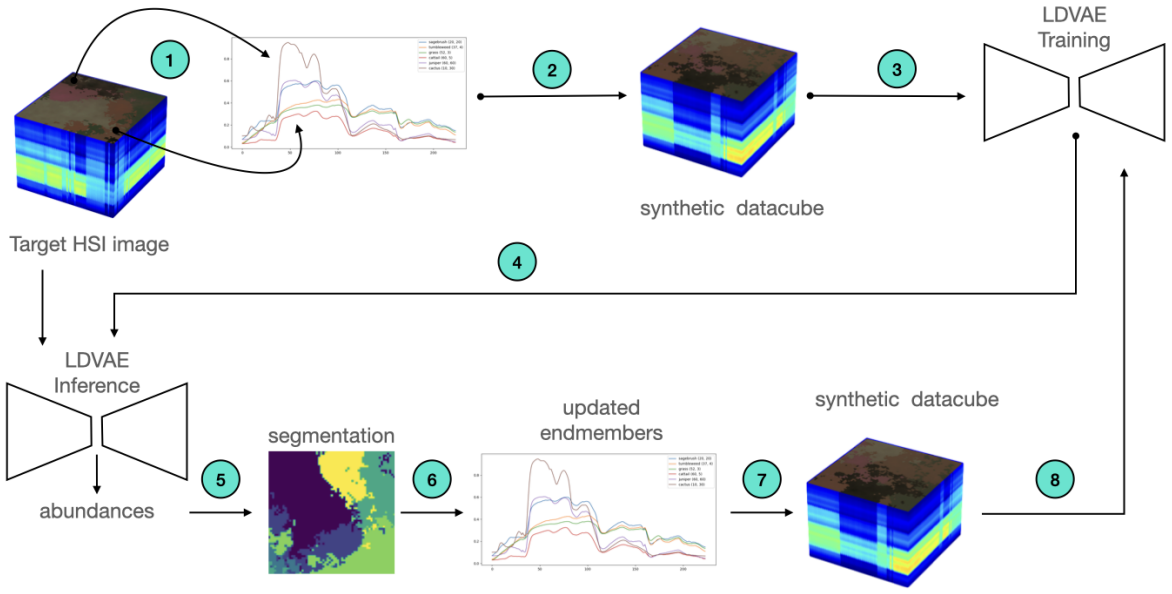

Figure 5.6: Iterative LDVAE Hyperspectral pixel unmixing and overview of the dataset

5.5.1 Loop Termination

iLDVAE loop is terminated when one of the two following conditions are met: 1) the number of allowed iterations is reached and 2) the disagreement (err) between endmembers estimations for two consecutive iterations is less than or equal to a pre-defined threshold ϵ, where

$$\text{err} = \frac{1}{\rho N} \sum_{i=1}^{N} \sqrt{\frac{\|\mathbf{e}_i^{(k+1)} - \mathbf{e}_i^{(k)}\|^2}{L_i}}. \tag{5.5}$$

We define L_i and ρ as follows. Let $\mathbf{S}$ denote the segmentation image such that $\mathbf{S}_{(x,y)} = \arg\max_i \mathbf{a}_{(x,y)}$. $\mathbf{S}$ is a single-channel image with the same spatial dimensions as $\mathbf{I}$. Each pixel in $\mathbf{S}$ contains the index of the most abundant endmember at that pixel. Then $L_i = \sum_{(x,y)} \mathbb{1}_{\mathbf{S}_{(x,y)=i}}$, where $\mathbb{1}$ is the indicator function. Let

$$p_i = \max_{\forall \mathbf{S}_{(x,y)}=i} \mathbf{a}_{(x,y)}$$

denotes the maximum abundance value for endmember i then $\rho = \min\{p_i\}_{i=1}^{N}$.

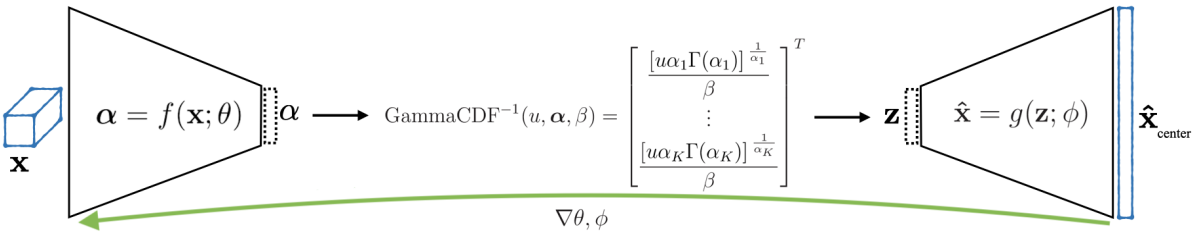

Figure 5.7: CNN Latent Dirichlet Variational Autoencoder. Encoder f takes an HSI patch $\mathbf{x}$ and constructs its latent representation (abundances). The decoder stage is able to reconstruct the pixel spectrum given abundances. Note that at training time the reconstruction loss is computed between the center pixel $\mathbf{x}_{\text{center}}$ and its reconstruction $\hat{\mathbf{x}}_{\text{center}}$.

5.6 Using spatial features

SpACNN model uses the VAE architecture depicted by Figure 5.7. LDVAE is a Variational Autoencoder where the latent representation follows a Dirichlet distribution. The encoder is parameterized by θ, which outputs the Dirichlet distribution parameter α. LDVAE takes a single signal of the pixel. We now try to leverage spatial information in the encoder using a CNN.

5.6.1 Spatial Attention Convolutional Neural Network Encoder

Our model uses a CNN encoder (Figure 5.8), which receives a rectangular patch as input and returns the Dirichlet distribution parameter α corresponding to the center pixel. The abundances $\mathbf{z}$ are sampled from the Dirichlet distribution and fed into the decoder, which reconstructs the spectral signal of the center pixel $\hat{\mathbf{x}}_{\text{center}}$. The decoder follows the model used in (Mantripragada and Qureshi, 2024).

The encoder employs an isotropic CNN model; therefore, the spatial resolution is maintained. The CNN encoder comprises of three modules: (1) stem, (2) body and (3) the spatial attention branch. The stem consists of a 2D convolution layer (kernel size: 3 and padding: 1) with Batch Normalization (BN) and Rectified Linear Unit (ReLU) activation. The body consists of six blocks of convolution layer followed by BN and ReLU. The spatial attention block follows the model introduced in (Woo et al., 2018). The output features are aggregated

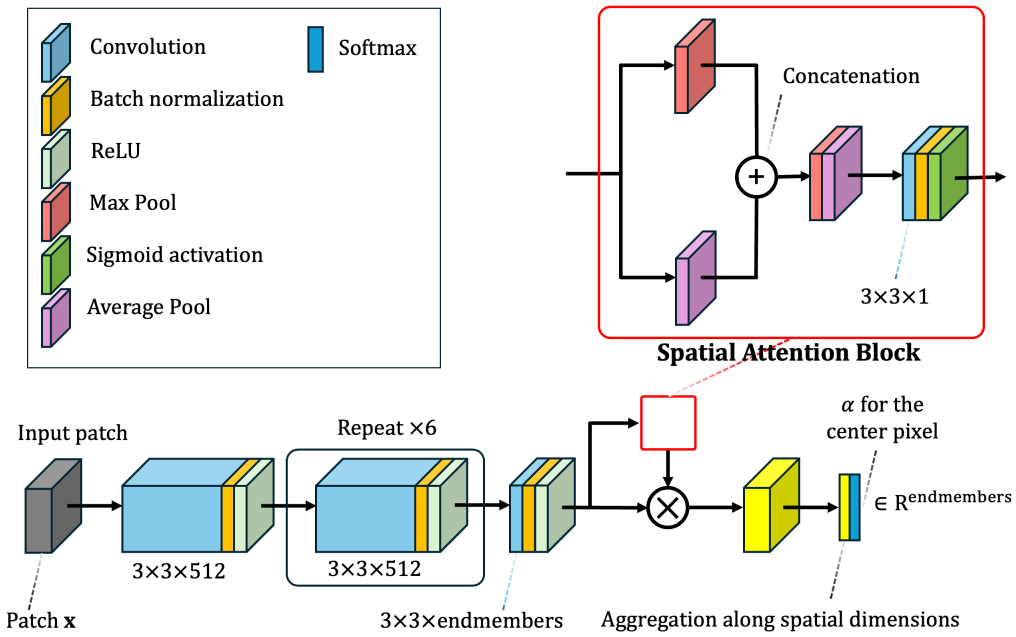

Figure 5.8: Spatial Attention Convolutional Neural Network Encoder. The network takes an HSI patch $\mathbf{x}$ and returns abundances vector α for the center pixel $\mathbf{x}_{\text{center}}$.

to generate the Dirichlet parameter α.

Given the intermediate feature map $\mathbf{F} \in \mathbb{R}^{H \times W \times C}$, apply average and max pooling along the channel dimension. Concatenate the results and perform 2D convolution with kernel size of 3 and apply sigmoid activation to obtain the 2D spatial attention map

$$\mathbf{A} = \sigma\big(f^{3\times 3}\left(\text{AvgPool}_C(\mathbf{F}) \oplus \text{MaxPool}_C(\mathbf{F})\right)\big), \tag{5.6}$$

where σ denotes the sigmoid function, $f^{3\times 3}$ denotes the convolution operation with a 3×3 filter, $\oplus$ denotes the concatenation operation, and $\mathbf{A} \in \mathbb{R}^{H \times W}$. The latent representation $\mathbf{z}' \in \mathbb{R}^C$ is computed as follows:

$$\mathbf{z}' = \sum_{i=1}^{H}\sum_{j=1}^{W} \mathbf{A}_{i,j}\mathbf{F}_{i,j}. \tag{5.7}$$

Softmax activation is applied to obtain α in the final layer of encoder as follows:

$$\alpha_k = \frac{e^{z_k}}{\sum_{k=1}^{C} e^{z_k}}. \tag{5.8}$$

This ensures that the model satisfies ASC and ANC.

5.6.2 Spectral Reconstruction With Multivariate Normal Distribution

As stated previously, we employ the decoder used in MLP-LDVAE, which uses an MLP to reconstructs the spectrum given abundances $\mathbf{z}$. The decoder serves two roles: (1) it is able to construct spectrum for previously unseen abundances and (2) it is able to perform endmember extraction by setting up the abundances appropriately. The model assumes that spectra follow a multivariate Normal Distribution as below:

$$\begin{aligned} \mathbf{x} &\sim Normal(\mathbf{x}; \mu, \Sigma) \text{ where} \\ \mathbf{x} &= \{x_1, x_2, x_3, ..., x_k\} \\ \mu &= \{\mu_1, \mu_2, \mu_3, ..., \mu_k\} \\ \Sigma &= \text{diag}(\sigma_1, \sigma_2, \sigma_3, ...\sigma_k) \end{aligned} \tag{5.9}$$

5.6.3 Loss function

The loss function has two components: (1) the reconstruction loss

$$\mathcal{L}(\mathbf{z}, \hat{\mathbf{z}}) = (\mathbf{z} - \hat{\mathbf{z}})^2, \tag{5.10}$$

where $\mathbf{z}$ and $\hat{\mathbf{z}}$ are ground truth and estimated abundances, respectively, and (2) the ELBO Loss

$$\mathcal{L}(\mathbf{x}; \theta, \phi) = \mathbb{E}_{q_\theta}[\log p_\phi] - \text{KL}\left[q_\theta(\mathbf{z}|\mathbf{x})||p(\mathbf{z})\right]. \tag{5.11}$$

The first term on the left side is the reconstruction loss for the spectrum. The second term represents KL divergence that forces the latent representation towards a Dirichlet distribution. For further details on ELBO loss, reparameterization tricks, variational autoencoders and dirichlet distributions on the latent space, please refer to the research of Mantripragada and Qureshi (2024); Pinheiro Cinelli et al. (2021); Joo et al. (2019); Blei et al. (2017); Fox and Roberts

(2012); Kingma and Welling (2014).

Chapter 6

Experiments and Results

6.1 Classification of Segments

Image segmentation result from an optimal scale is often evaluated through visual inspection by comparing it to the manually digitized reference polygons over different land cover types. However, qualitative visual inspection is labor-intensive and subjective, and the results vary when conducted by different technicians. In contrast, quantitative methods that measure the arithmetic and geographic discrepancy between segments (Yang et al., 2015) and reference polygons are more effective and can be repeated and automated easily in the process of segmentation quality assessment. Among those, the ED index (Yang et al., 2014), computed from OS and US, shows to be a more robust measure as it embeds both US and OS metrics. The ED index is calculated as follows:

$$OS_{ij} = \left(1 - \frac{area(r_i \cap s_j)}{area(r_i)}\right)^2, r_j \in R, s_j \in S \tag{6.1}$$

$$US_{ij} = \left(1 - \frac{area(r_i \cap s_j)}{area(s_j)}\right)^2, r_j \in R, s_j \in S \tag{6.2}$$

$$ED_{ij} = \frac{US_{ij}^2 + OS_{ij}^2}{2} \tag{6.3}$$

Table 6.1: The optimal scales of k-means, mean-shift, and watershed of the three images using RoC and NN-nRoC graphs.

	Using RoC Graphs			Using NN-nRoC Graphs		
	KM	MS	WS	KM	MS	WS
Suburban Image	11	15	500	11	7	1100
Urban Image	4	8	1,500	6	26	1,300
Forest Image	19	15	500	11	10	1,500

where r_i is the ith polygon in the set R of reference polygons, and s_i is the corresponding segment from the set S of evaluated segments. Both OS and US are normalized between 0 and 1, ED, therefore, indicates both the geometric and arithmetic relationships. The smaller index values indicate good segmentation results and vice versa.

To evaluate the image segmentation results, we manually digitized 50 polygons with different shapes and sizes over different land cover types, such as rooftops, vegetation, and shadows. The US, OS, and ED indices were evaluated over these land cover types to confirm the robustness of our approach for different datasets and applications. For some of the scales, the segmentation algorithms can produce tiny objects (tens of pixels), which in turn affects the evaluation of the metrics. These small objects are generated due to the high variability of hyperspectral data and can lead to misinterpretation of the results since we are evaluating a small number of cover types. Therefore, we applied a threshold where only the pairs of segments-polygons with 10% overlap and segments equal to or greater than 9 pixels were selected for evaluation.

Final image segmentation results using NN-nRoC-based optimal scale selection, in Table 6.1, are illustrated in Figures 6.1, 6.2, and 6.3. The top figures are the results from RoC-based method, and the bottom figures are the results from the NN-nRoC method. Yellow polygons are manually digitized reference polygons for validation.

For the suburban image (Figure 6.1), both RoC and NN-nRoC methods worked well in all three algorithms that successfully delineated the boundaries between various land cover types. Some of the rooftops with very similar colours to the shadows in the true-colour image were also accurately distinguished in the segmented image. In this image, mean-shift seemed to perform the best since most of the ground features were accurately delineated with the least over- and under-segmentation in comparison with the other two algorithms. Watershed

performed well with most features separated, and only slight miss-segmentation was observed in some places. K-means performed intermediately and successfully separated various land cover types with only slight over-segmentation is found in some areas. Between the two methods, NN-nRoC performed better than RoC methods in all three algorithms. In the RoC method, the image was over-segmented in k-means and mean-shift, especially in vegetated regions, while under-segmentation was observed in watershed segmentation results. More interestingly, the NN-nRoC method outperformed that the RoC method, even when the segmentation scales were the same ($k = 11$). This finding demonstrates the impact of noise in image segmentation results even it did not significantly influence the scale selection in this particular case. This finding also indicates the improvement of our proposed segmentation method.

For the urban image (Figure 6.2), overall, mean-shift performed the best, followed by k-means and watershed. The mean-shift algorithm successfully delineated rooftops from lawns and shadows, while some shadows were segmented to roof in k-means. However, over-segmentation was observed in spectrally varied areas, which made the segmentation results noisy, especially in the areas with under-construction buildings and bare soil on the north side of the image. Watershed seemed to perform well in these areas with less over-segmentation. Overall, the results from our proposed NN-nRoC method were better than the results from the RoC method in all three segmentation algorithms. Particularly in the watershed, the NN-nRoC outperformed the RoC method, and the image was segmented at a higher level of detail even the scale parameter was smaller (markers = 1300 vs. 1500). It should be noted that in the watershed, the higher the markers, the more details of the segmentation. This finding also indicates the improvement of our proposed method.

For the forest image (Figure 6.3), the performance of these segmentation algorithms was not comparable to their performance in the suburban and urban images. The k-means and mean-shift algorithms performed well in separating tree canopies from the shadows with minimal over- and under-segmentation. The watershed performance was less efficient, especially in segmenting gaps and canopies with irregular shapes. From visual interpretation, it is clear that the NN-nRoC performed better than the RoC in all three segmentation algorithms. In the RoC method, over-segmentation in k-means and mean-shift, and under-segmentation in the watershed were

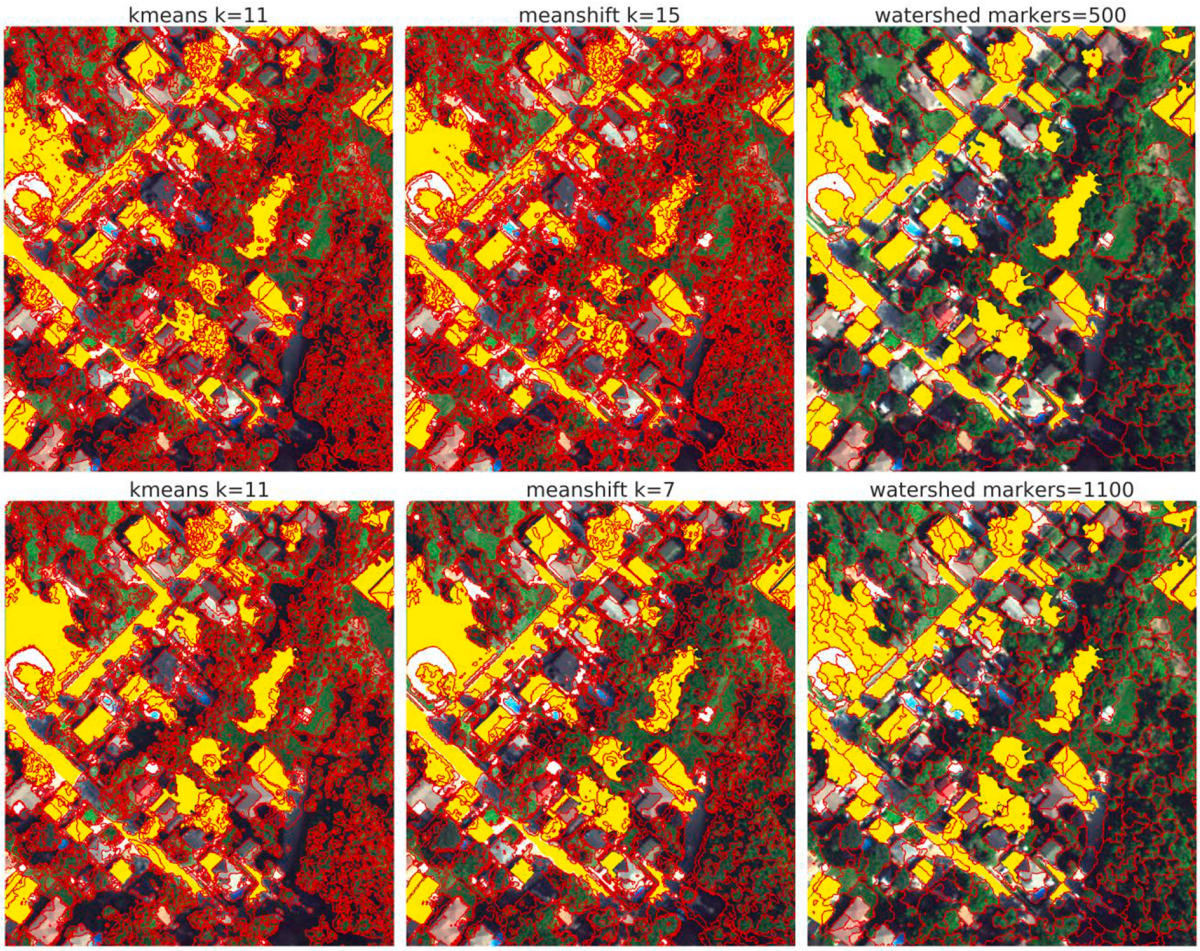

Figure 6.1: Segmentation results of the suburban image using optimal scales selected in Table 6.1. From top to bottom are the results of RoC and NN-nRoC methods, respectively. From left to right, the original image with reference polygon, k-means, mean-shift, and watershed segmentation results. Yellow polygons are manually digitized reference polygons for validation.

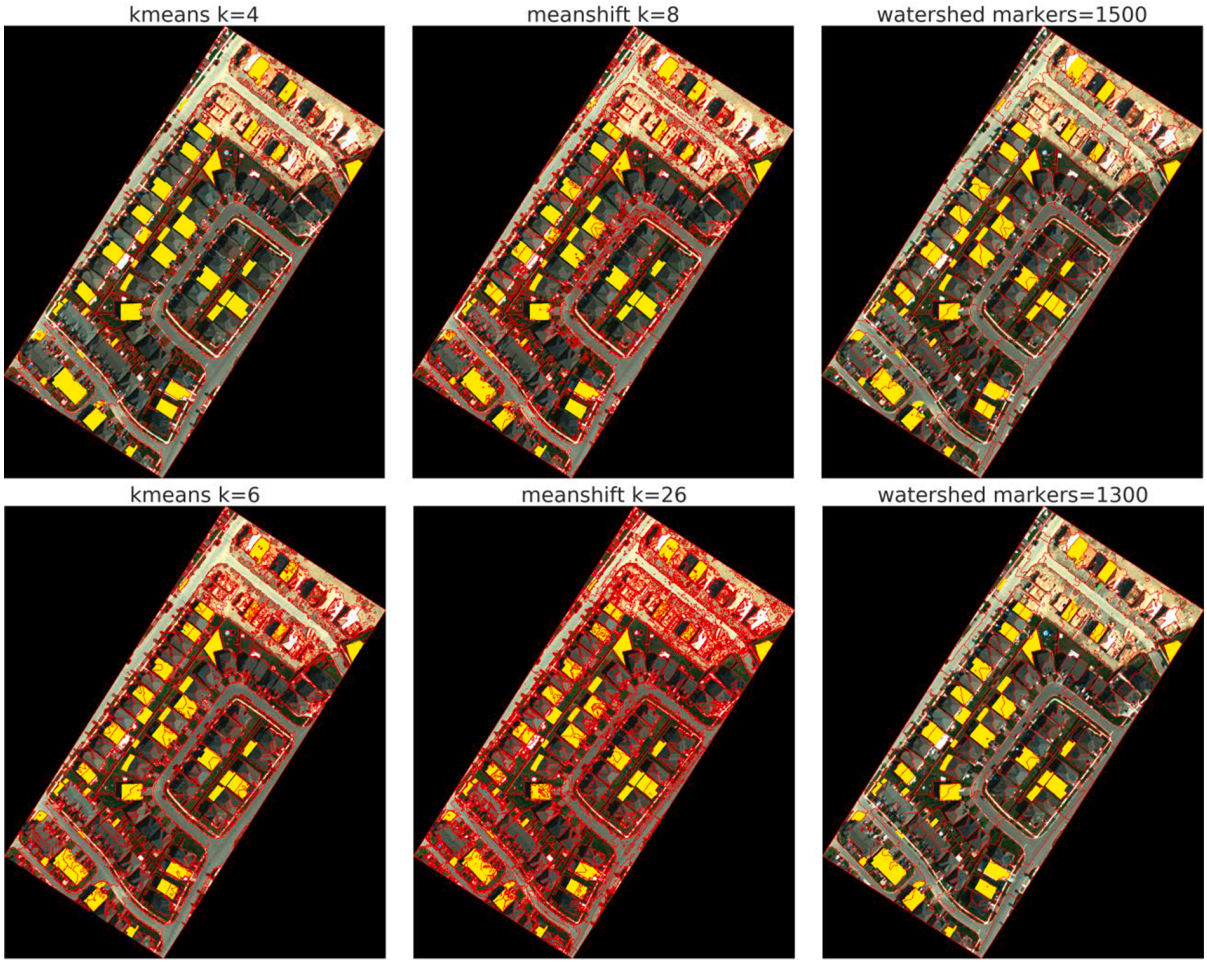

Figure 6.2: Segmentation results of the urban image using optimal scales selected in Table 6.1. From top to bottom are the results of RoC and NN-nRoC methods, respectively. From left to right, the original image with reference polygon, k-means, mean-shift, and watershed segmentation results. Yellow polygons are manually digitized reference polygons for validation.

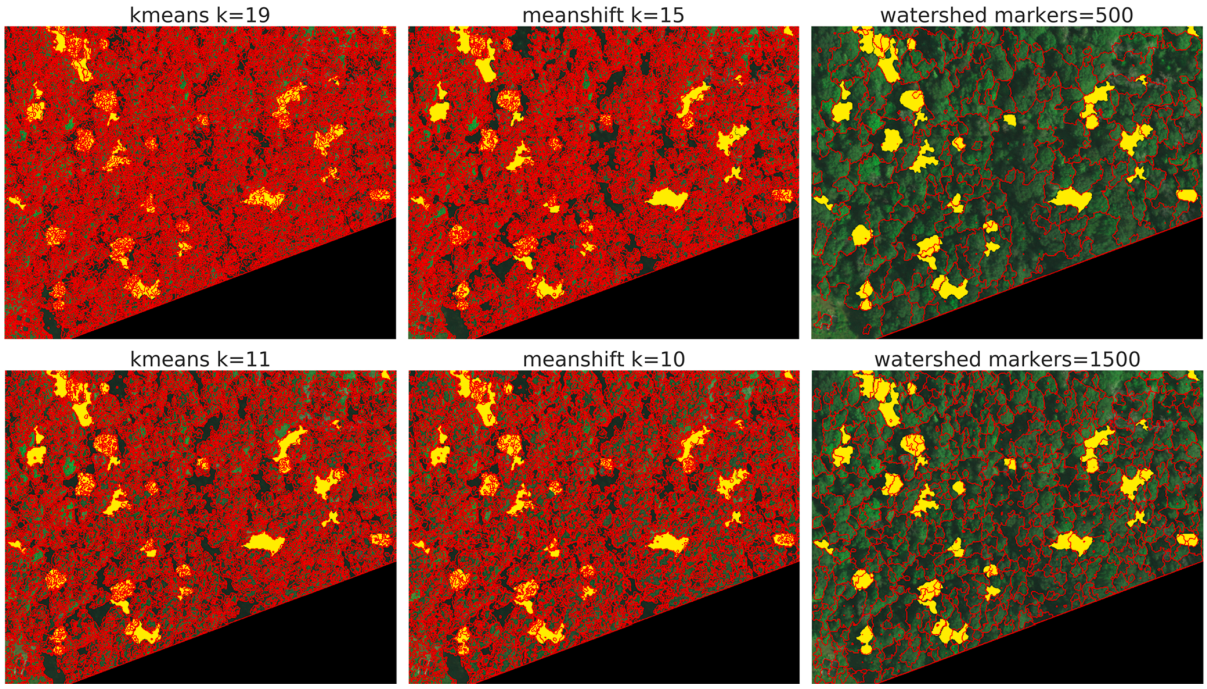

Figure 6.3: Segmentation results of the forest image using optimal scales selected in Table 6.1. From top to bottom are the results of RoC and NN-nRoC methods, respectively. From left to right, the original image with reference polygon, k-means, mean-shift, and watershed segmentation results. Yellow polygons are manually digitized reference polygons for validation.

much more severe compared to the results from the NN-nRoC method.

6.2 Classification at the pixel-level with dimensionality reduction

A standard way to study the performance of different compression algorithms is to recover the original signal from its compressed version as depicted in Figure 6.4. In the following sections, we examine reconstruction errors for PCA, KPCA, ICA, AE and DAE for different compression rates. We also present reconstruction errors both with and without the SG noise reduction preprocessing step. As stated earlier, compressing hyperspectral data is desirable; however, we are also interested in pixel-level classification using the compressed data. We define pixel-based classification as the problem of identifying landcover type, say forest, rooftop, etc., for a given pixel in an hyperspectral image. Within this context, we seek the answer to the following

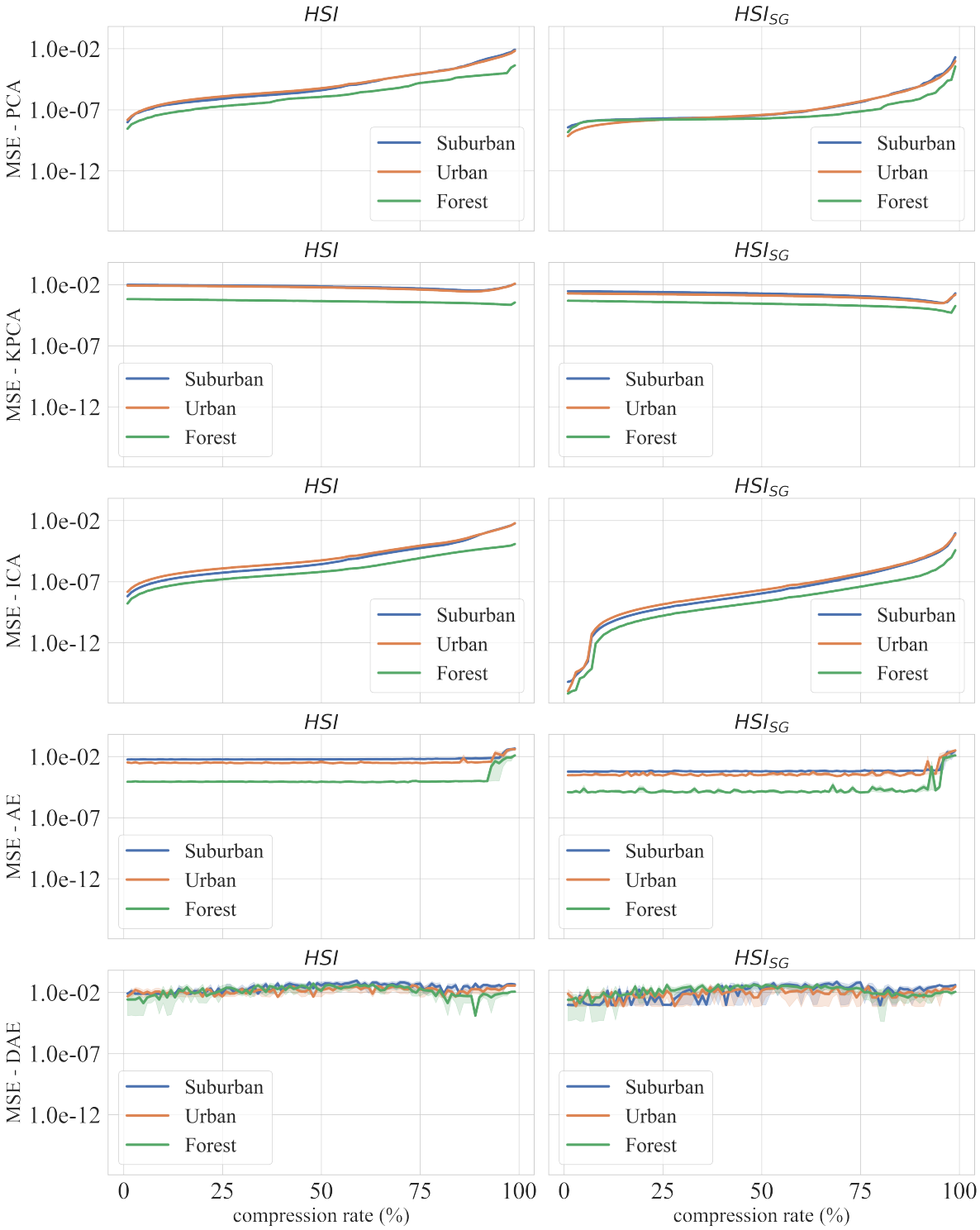

Figure 6.4: MSE (Mean Squared Error) for the 3 datasets and 5 compression algorithms, from top to bottom: a) Principal Component Analysis, b) Independent Component Analysis, c) Kernel Principal Component Analysis, d) AutoEncoder, e) Denoising AutoEncoder

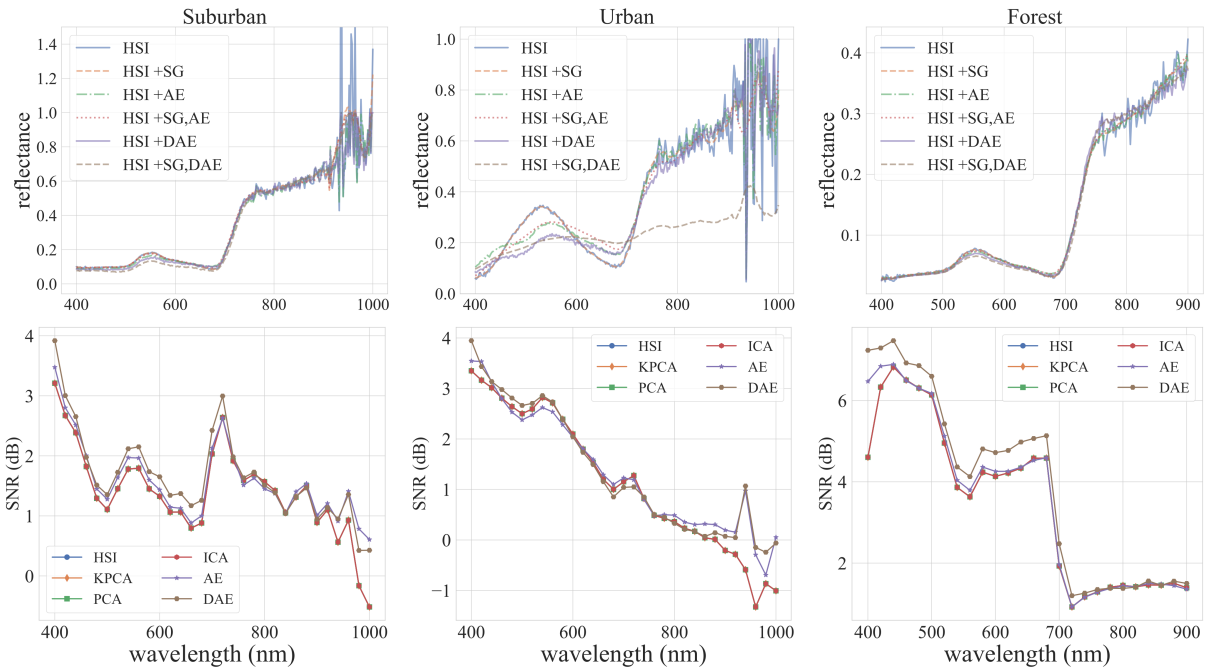

Figure 6.5: **First row:** Spectral reconstructions for a randomly selected pixel in the three images. HSI denotes the original spectral signal. HSI+SG refers to the denoised spectral signal. HSI+AE and HSI+DAE denote reconstructed spectral signals using autoencoder and denoising autoencoder, respectively. HSI+SG+AE and HSI+SG+DAE denote reconstructed spectral signal using transformed encodings (0% Compression rates) from denoised signals (HSI+SG). **Second row:** Signal-to-Noise Ratio of the reconstructed spectra (PCA, KPCA, ICA, AE, DAE) compared to the original pixel (HSI)

two questions: a) how compression rates affect pixel classification scores and b) for a given compression rate, which compression method achieves the highest classification accuracy.

6.2.1 Spectral Reconstruction

Figure 6.4 presents reconstruction losses for different methods for the three datasets. Here compression rates vary from 1% to 99%. For our purposes, the compression rate is defined as the ratio of $(n-d)$ to n, where n is the number of dimensions of the original signal and d is the number of dimensions of the compressed signal. Recall that n is equal to 301 for the datasets used in this paper. Compression rates are related to the memory needed to store the compressed data. The left column of Figure 6.4 shows the results for the original hyperspectral data (HSI), whereas the right column shows the results for the data that have been pre-processed using the

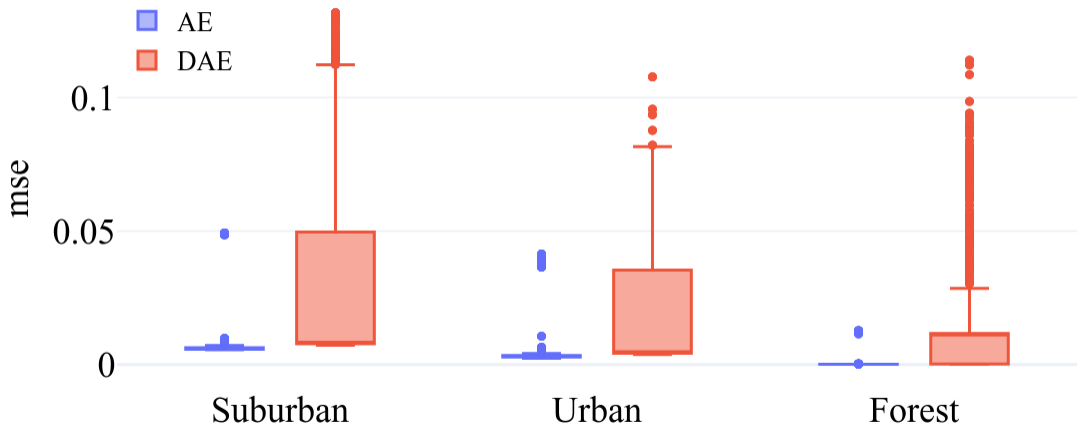

Figure 6.6: Model variance. Reconstruction errors for AE and DAE models for ten training runs.

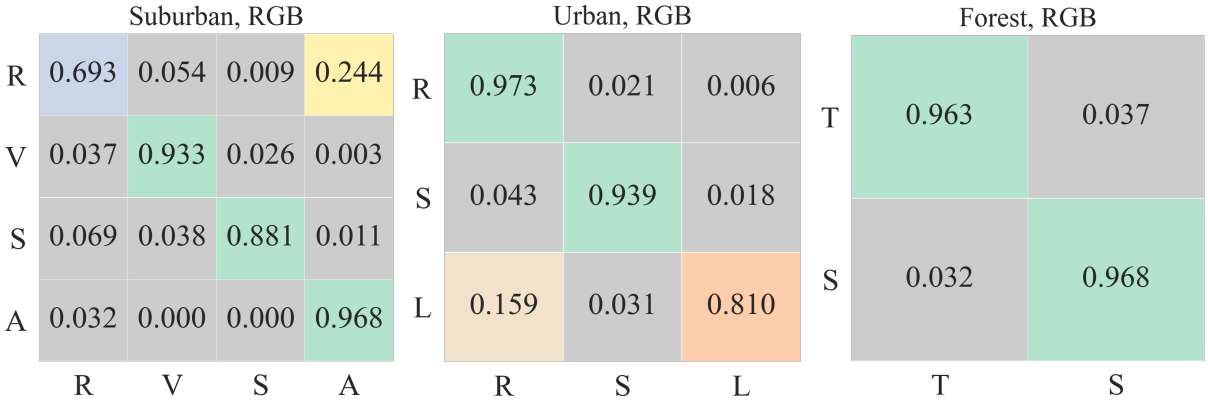

Figure 6.7: Confusion matrix for classification scores for three datasets using RGB data. (Left) R, V, S, and A refer to Rooftop, Vegetation, Shadow, and Asphalt; (Center) R, S, L refer to Rooftop, Shadow and Lawn, respectively; and (Right) T, S refer to Tree and Shadow, respectively

SG filter (HSI+SG).

The reconstruction errors rise as compression rates increase for both PCA and ICA models. Notice, however, other methods—KPCA, AE, and DAE—are able to achieve low reconstruction errors even for high compression rates. This effect can be explained by the fact that non-linear methods are able to better handle non-linearities present in the data while PCA and ICA are linear methods. It is interesting to note that PCA, KPCA, and ICA methods outperform deep learning methods AE and DAE for compression rates less than ninety percent. Furthermore, AE and DAE match the reconstruction performance of PCA, KPCA, and ICA only for compression rates higher than ninety percent.

The difference between reconstruction errors for original data (HSI) and for data preprocessed using SG filter (HSI+SG) falls as compression rate increases. This is noteworthy since it suggests that compression may have a denoising effect on the original spectral signal. Curiously, we also observe a slightly higher variance in reconstruction score for DAE method for pre-processed data (HSI+SG), which merits further investigation, and we leave it as future work. In the following section, we do not apply SG to the original spectra before compression and classification, as it does not show effective as discussed in the Section 5.2.1

6.2.2 Classification

The reconstruction error is a measure of how much information is preserved in the compressed signal, since this information is needed to reconstruct the original signal. The reconstruction error, however, is not a robust measure of classification performance on the compressed signal. In this section we study classification performance on compressed signal for PCA, KPCA, ICA, AE and DAE compression methods and for different compression rates.

Figure 6.7 shows the confusion matrices for landcover classification for RGB data. Here bands corresponding to red ($670nm$), green ($540nm$), and blue ($470nm$) wavelengths are selected to form RGB pixels. These scores provide a baseline for the classification results obtained by using the hyperspectral data. Figure 6.8 shows *f1-scores*, precision, and recall values obtained using 1) RGB, 2) uncompressed, and 3) compressed hyperspectral data. The results shown for compressed data are aggregated over all compression rates. RGB *f1-scores* range between 0.9

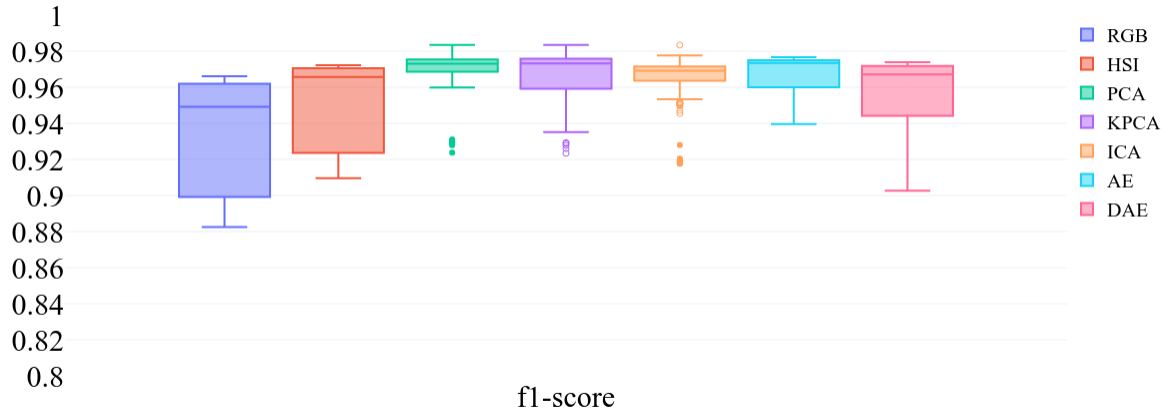

Figure 6.8: F1-scores across all compression rates for all datasets and landcover types using PCA, KPCA, ICA AE and DAE methods. This figure also includes precision, recall and f1 score for all datasets and landcover types when using RGB data for pixel classification

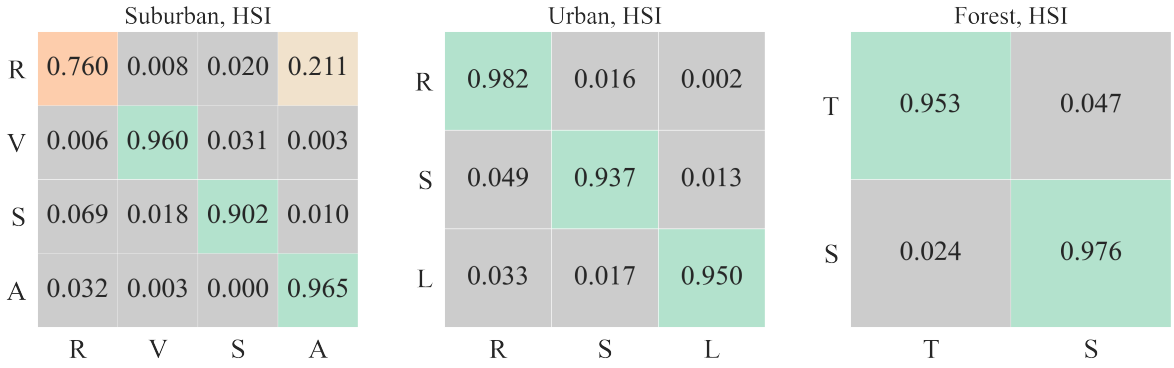

Figure 6.9: Confusion matrix for classification scores for three datasets using HSI data. (Left) R, V, S, and A refer to Rooftop, Vegetation, Shadow, and Asphalt; (Center) R, S, L refer to Rooftop, Shadow and Lawn respectively; and (Right) T, S refer to Tree and Shadow.

and 0.96; however, *f1-scores* obtained by using hyperspectral data fall between 0.96 and 0.98. Figure 6.9 shows *f1-scores*, and it confirms our intuition that the classification results obtained by using hyperspectral data are better than those obtained by using RGB channels. We will return to this later in this section.

We now turn our attention to the case when classification is performed on compressed hyperspectral data. Figure 6.10 plots *f1-scores* for PCA, KPCA, ICA, AE and DAE compression methods vs. compression rates. In each case an XGBoost classifier is used to predict pixel landcover-types. A total of 1470 classifiers is trained, one for each compression rate (98) for every compression method (5) and for each dataset (3), in order to ensure that the differences in classification scores can be explained by the ability of the compression algorithm to encode the relevant information. Each XGBoost classifier is trained using identical meta parameters and training regimes. These experiments then provide a different lens for studying compression algorithms. Specifically, these experiments help us pose the question: is it true that compression algorithms that achieve low reconstruction errors also create a compressed signal that encodes the information necessary to perform pixel-level classifications?

Ideally, we want classification algorithms that operate in the compressed signal domain. It is both computation and space inefficient to have to reconstruct the original signal to perform classification. As expected, classification performance as measured by *f1-scores* drops as compression rates increase. At the same time, however, nearly all methods post *f1-scores* greater than 0.85 even for compression rates greater than eighty percent. This suggests that it is possible to achieve good classification performance when using a compressed hyperspectral signal.

Classification using RGB Data

The RGB classifier only achieves an accuracy of 69% for rooftop landcover type in the suburban dataset. Using hyperspectral data improves upon the classification scores obtained by using RGB data. These results confirm that *f1-scores* for hyperspectral datasets improve upon those for RGB data by around one to two percent. Note also that this improvement is maintained when performing classification using the compressed data. Specifically, our results suggest

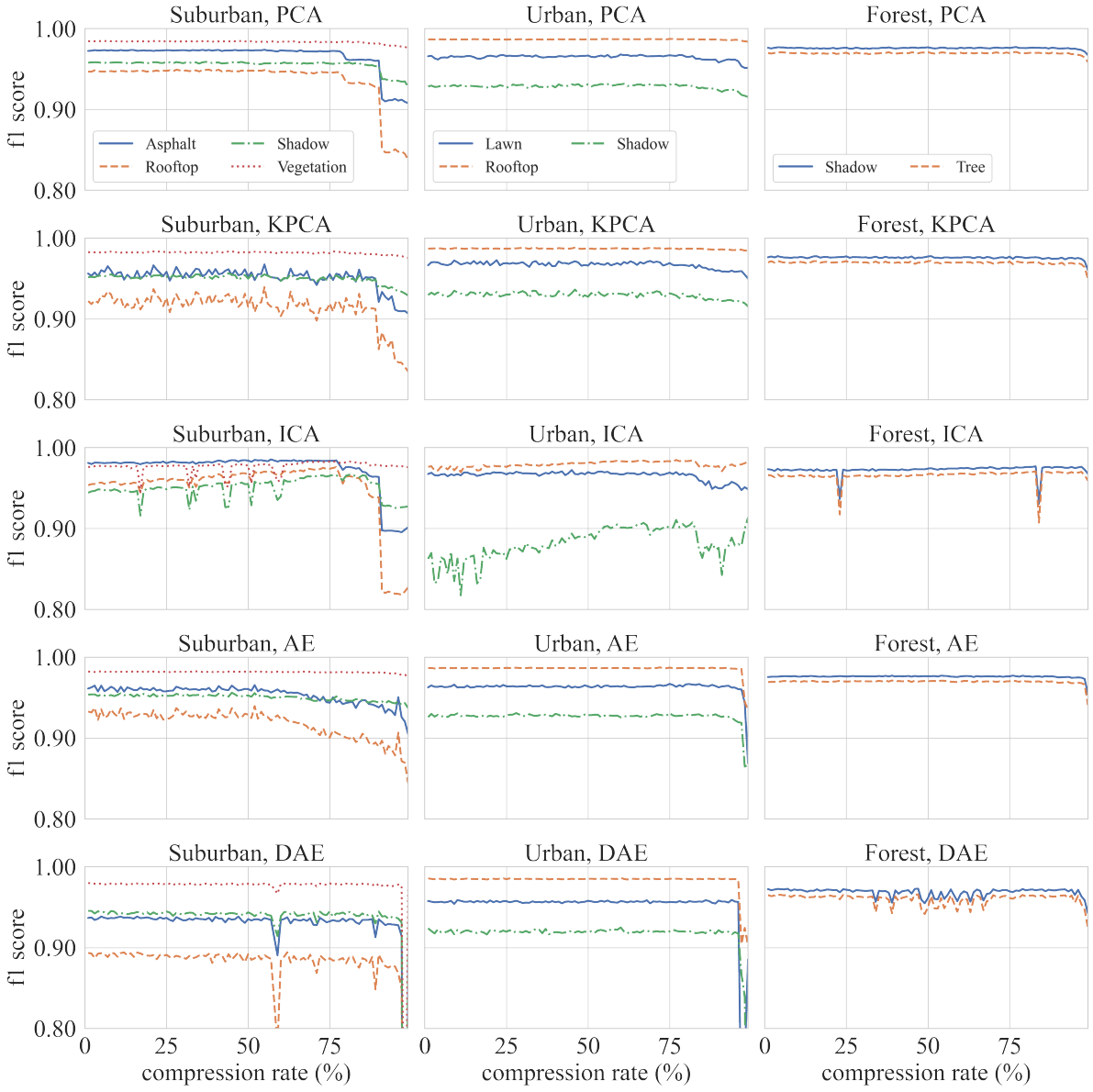

Figure 6.10: Classification f1 scores (using compressed data) vs. compression rates. First column plot results for the Suburban dataset, the second column plot results for the Urban dataset, and the last column plot results for the Forest dataset. The rows group the classification algorithms. The *f1-scores* are plotted for each label present in the dataset

that this improvement holds even at 98% compression rates. At 98% compress rate, each hyperspectral pixel is encoded in a 6-dimensional vector, which is only twice the number that is needed to store an RGB pixel. We believe that classification scores using hyperspectral data, compressed or otherwise, will pull ahead of the scores obtained by RGB data as the number of landcover types (or labels) increases. We currently do not have access to a dataset that is needed to study this issue further.

6.2.3 Classification on compressed data (Suburban Dataset)

Figure 6.11 visualizes classification *f1-score*, recall, and precision values for compression rates between 1% and 99% on Suburban dataset. As expected scores for PCA, KPCA, AE, and DAE compression methods decrease as the compression rate increases. ICA is an outlier. ICA has higher classification performance for compression rates between 63% and 77%. PCA compression achieves best classification performance for compression rates less than 90%. AE compression method achieves best classification performance when compression rate is between 95% and 97%. While classification performance of AE and DAE compression methods is similar to other methods for low compression rates, AE and DAE achieve better classification performance as compared to those obtained by other methods for compression rates between 90% to 95%. Classification accuracy plummets for compression rates greater than 97%. Table 6.2 shows *f1-scores*, precision, and recall for Suburban dataset at 95% compression (the compressed signal is a 15-dimensional vector, down from 301-dimensional original spectral signal). It also includes these scores for the RGB data. AE and DAE methods outperform other methods at this compression rate.

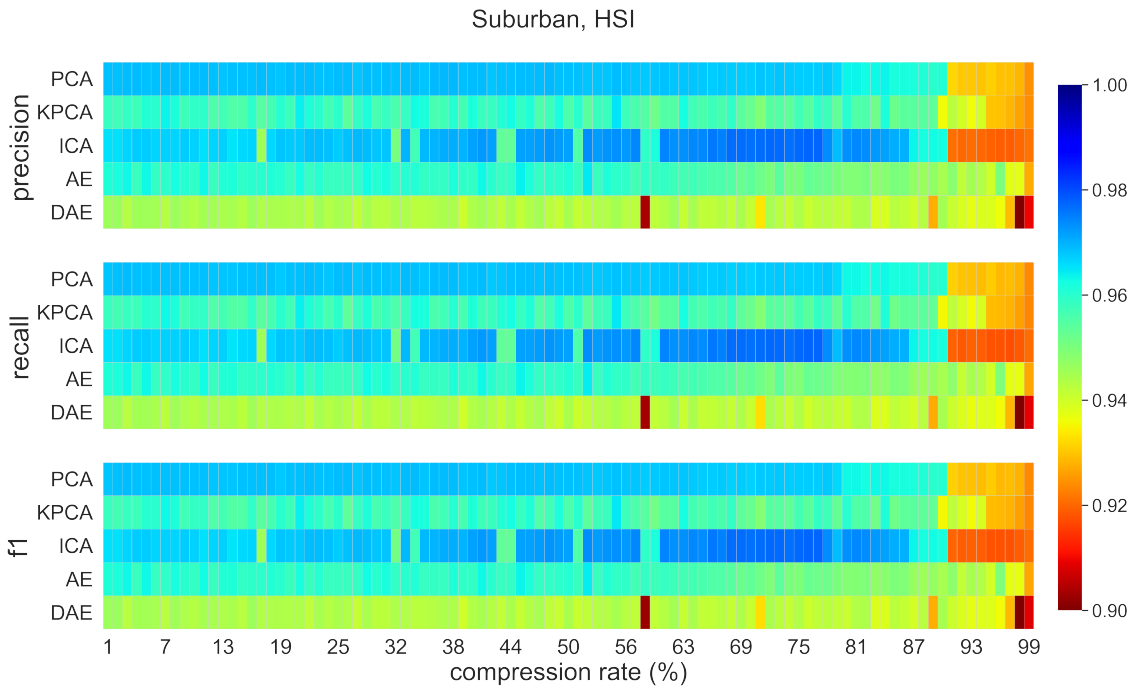

Figure 6.11: Suburban dataset classification scores for all methods for compression rates between 1% to 99%.

Table 6.2: Top classification scores Suburban, HSI, compression rate=95%

label	compression	precision	recall	f1-score
Asphalt	**RGB**	0.809	0.968	0.881
	PCA	0.888	0.939	0.913
	KPCA	0.885	0.939	0.911
	ICA	0.862	0.934	0.897
	AE	0.917	0.950	0.928
	DAE	0.914	0.943	0.928
Rooftop	**RGB**	0.776	0.693	0.732
	PCA	0.856	0.845	0.851
	KPCA	0.855	0.842	0.849
	ICA	0.823	0.816	0.819
	AE	0.881	0.887	0.878
	DAE	0.873	0.880	0.877
Shadow	**RGB**	0.952	0.881	0.915
	PCA	0.955	0.918	0.936
	KPCA	0.954	0.917	0.935
	ICA	0.956	0.896	0.925
	AE	0.960	0.928	0.943
	DAE	1.000	0.923	0.938
Vegetation	**RGB**	0.945	0.933	0.939
	PCA	0.980	0.978	0.979
	KPCA	0.979	0.977	0.978
	ICA	0.978	0.975	0.977
	AE	0.981	0.980	0.980
	DAE	0.977	1.000	0.977

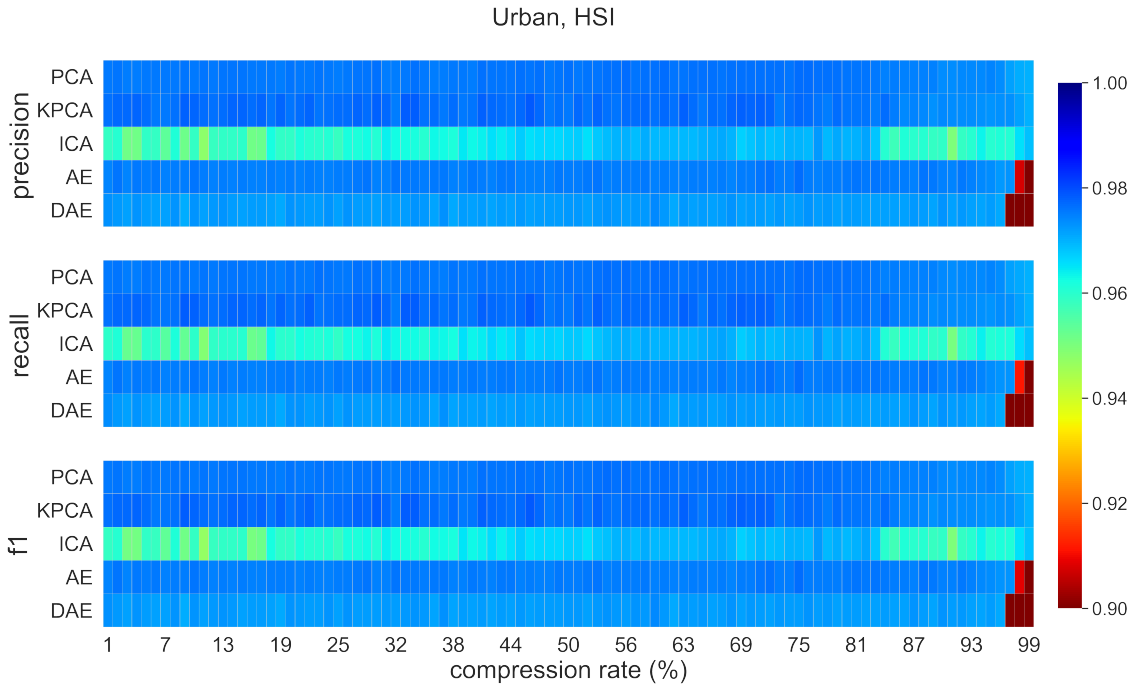

Figure 6.12: Urban dataset classification scores for all methods for compression rates between 1% to 99%.

6.2.4 Classification on compressed data (Urban Dataset)

Figure 6.12 visualizes classification *f1-score*, recall, and precision values for compression rates between 1% and 99% on Urban dataset. PCA and KPCA are the best performing methods; however, AE is able to match the performance of these methods for compression rates less than 96%. ICA method seems to be struggling with this dataset. The performance of the DAE method is inconsistent across the compression rates range. This can be attributed to the stochastic nature of this method. The classification accuracy for data compressed using ICA is higher for compression rates between 63% and 77%. For compression rates of less than 90%, best classification scores are achieved when data are compressed using PCA. AE compression method achieves the best classification scores when the compression rate lies between 95% and 97%. Table 6.3 shows *f1-score*, precision, and recall for the urban data at the 95% compression. The table also includes these scores for the RGB baseline. AE and DAE methods outperform other methods at this level of compression.

Table 6.3: Top classification scores Urban, HSI, compression rate=95%

label	compression	precision	recall	f1-score
Lawn	**RGB**	0.930	0.810	0.866
	PCA	0.963	0.960	0.962
	KPCA	0.966	0.951	0.958
	ICA	0.965	0.947	0.956
	AE	0.967	0.957	0.962
	DAE	0.961	0.954	0.957
Rooftop	**RGB**	0.967	0.973	0.970
	PCA	0.984	0.988	0.986
	KPCA	0.983	0.987	0.985
	ICA	0.971	0.983	0.977
	AE	0.984	0.988	0.986
	DAE	0.983	1.000	0.985
Shadow	**RGB**	0.877	0.939	0.907
	PCA	0.935	0.915	0.924
	KPCA	0.926	0.917	0.921
	ICA	0.901	0.858	0.879
	AE	0.934	0.916	0.922
	DAE	0.935	0.918	0.919

6.2.5 Classification on compressed data (Forest Dataset)

Figure 6.13 visualizes classification *f1-score*, recall, and precision values for compression rates between 1% and 99% on Forest dataset. DAE and ICA perform poorly on this dataset. PCA, KPCA, and AE compression methods achieve good classification performance on this dataset, where AE outperforming PCA and KPCA methods at 97% compression. Classification performance for data compressed using ICA method curiously improves with compression rate. This merits further investigation. Table 6.4 shows *f1-score*, recall, and precision for the Forest dataset at the 95% compression. It also includes these values for the RGB baseline. AE and DAE methods outperform other methods at this level of compression.

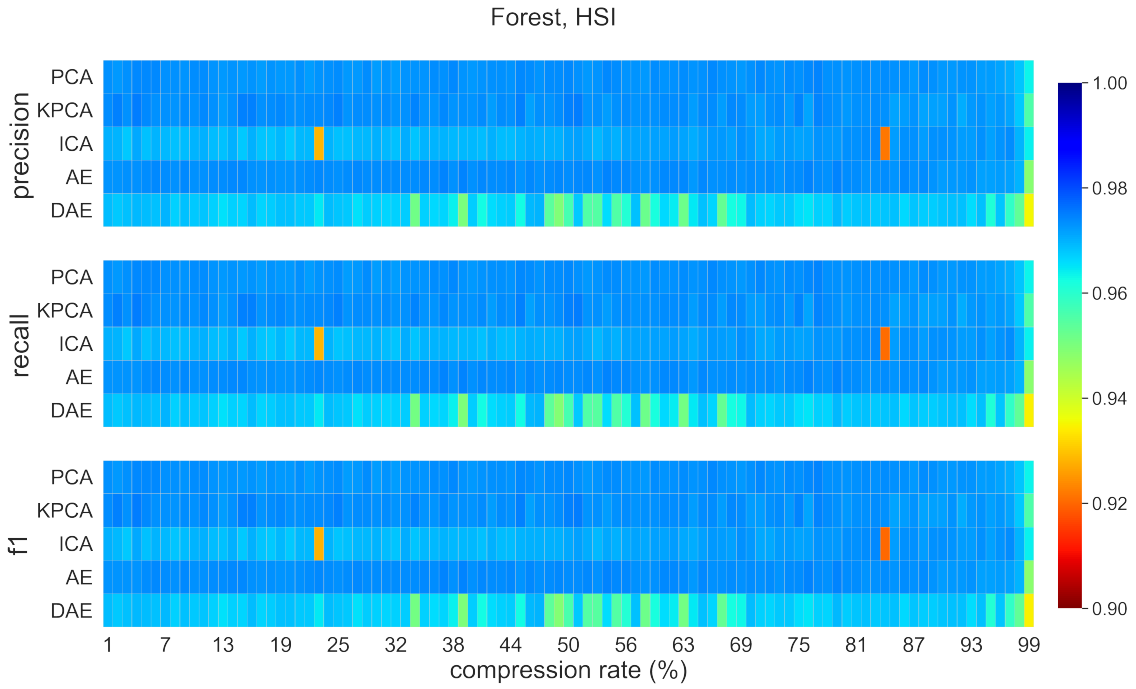

Figure 6.13: Forest dataset classification scores for all methods for compression rates between 1% to 99%.

Table 6.4: Top classification scores Forest, HSI, compression rate=95%

label	compression	precision	recall	f1-score
Shadow	**RGB**	0.971	0.968	0.970
	PCA	0.972	0.978	0.975
	KPCA	0.973	0.977	0.975
	ICA	0.970	0.979	0.975
	AE	0.972	0.979	0.975
	DAE	0.989	1.000	0.965
Tree	**RGB**	0.960	0.963	0.962
	PCA	0.973	0.964	0.969
	KPCA	0.971	0.966	0.968
	ICA	0.974	0.962	0.968
	AE	0.973	0.964	0.968
	DAE	0.964	0.992	0.956

6.2.6 Computational considerations

The compression methods used in this work need to be trained before these can be used to compress the incoming 301-dimensional spectral signal. The methods only need to be trained

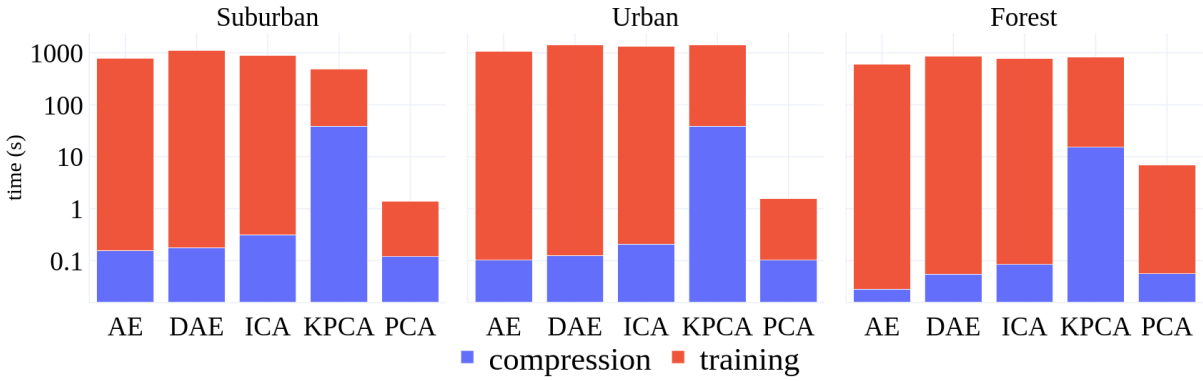

Figure 6.14: Execution times (seconds) for training and compression tasks and all compression algorithms.

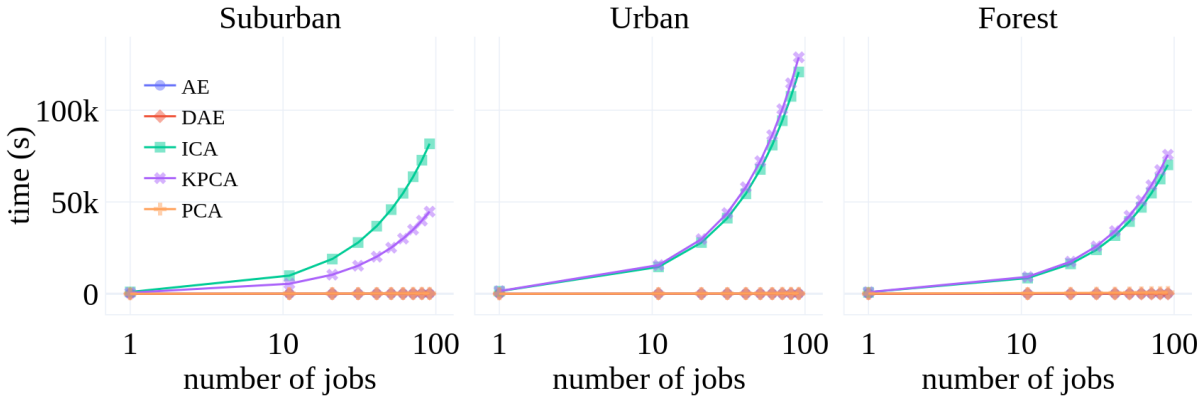

Figure 6.15: Execution times (in seconds) times the number of classification jobs.

once; therefore, the training time is not important as the compression time (the time it takes these methods to encode the incoming signal into a low-dimensional space). Figure 6.14 shows compression and training times for each method. Notice that KPCA has the largest compression time, which suggests that KPCA is not well-suited to applications where smaller runtimes are desirable.

While the training datasets used in this paper fit in memory, one can imagine a situation where the size of data excludes this possibility. It is not easy to use PCA, KPCA, and ICA in situations where training data do not fit in memory. Deep learning methods, such as AE and DAE, can be trained in batches; therefore, these methods can be trained in situations where the entire training data does not fit in memory. Figure 6.15 shows that AE and DAE have better scalability properties than other compression methods. This suggests that AE and DAE

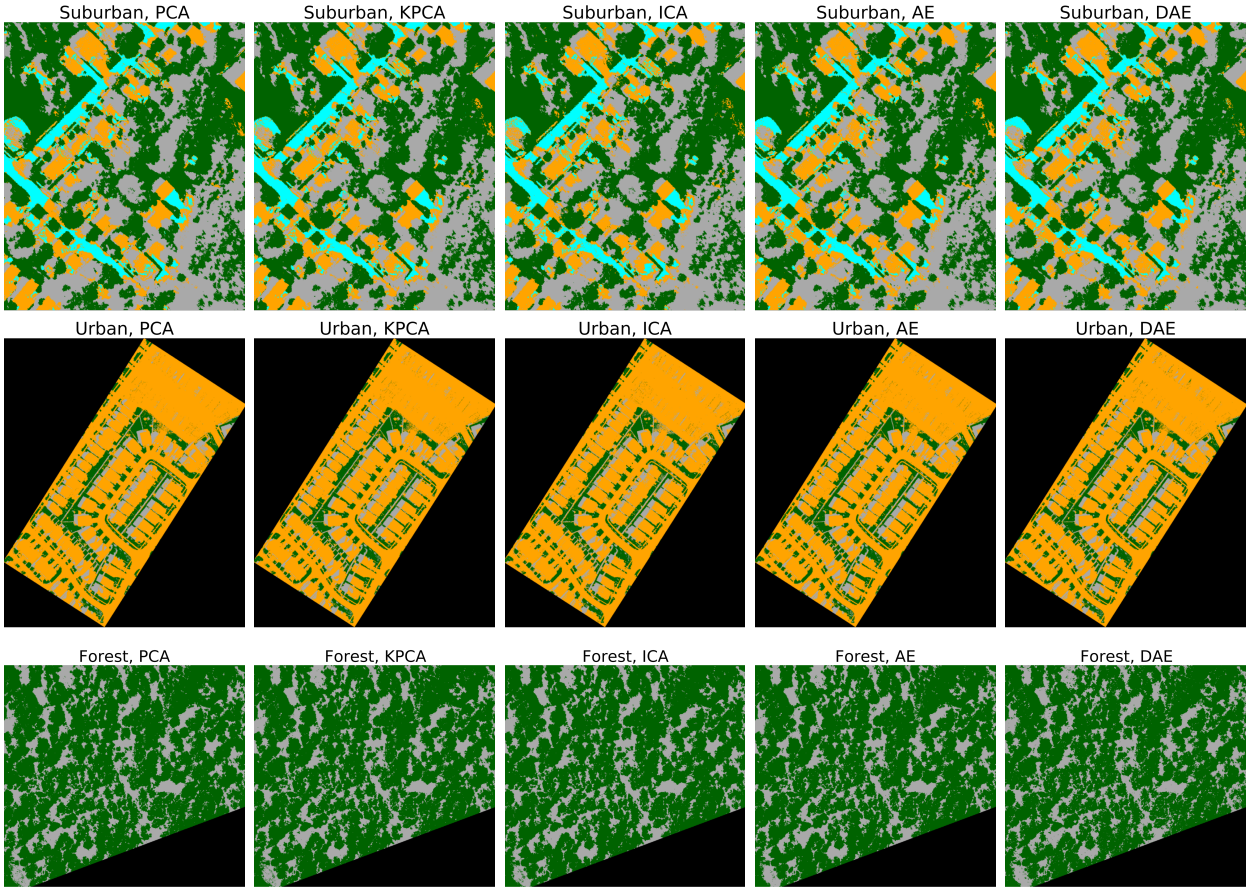

Figure 6.16: Classification images with spectra reduced to 95% of the original size.

methods may be more suited for in-situ applications where computational resources are usually limited. Lastly, Figure 6.16 demonstrates the results of the classification with the reduction rate at 95%.

6.3 Unmixing — classification at the subpixel

We assess our proposed model using the same evaluation methods employed in (Zhang et al., 2022) and (Palsson et al., 2020). We use two metrics: 1) Spectral Angle Distance (SAD) to evaluate endmembers extraction and 2) Root Means Squared Error (RMSE) to evaluate abundance estimation. We also evaluate spectral reconstruction using MSE (Means Squared Error) and SAD. We consider spectral reconstruction an important task for our model, because it underpins endmember extraction. We briefly describe SAD, RMSE, and MSE metrics below.

For a detailed treatment of these metrics, we refer the reader to (Deborah et al., 2015; Zhu, 2017; Borsoi et al., 2020).

6.3.1 Spectral Angle Distance (SAD)

The SAD metric is a distance measurement between two spectral signals:

$$\text{SAD} = \arccos\left(\frac{\hat{\mathbf{x}}_e^T \mathbf{x}_e}{\|\hat{\mathbf{x}}_e^T\|\|\mathbf{x}_e\|}\right), \tag{6.4}$$

where $\hat{\mathbf{x}}_e$ represents the endmember (spectral signal) generated by the decoder stage of LDVAE and $\mathbf{x}_e$ is the reference endmember (ground truth endmember). SAD is used to compute the accuracy of endmember extraction. In our experiments, we used SAD to evaluate the quality of the endmembers extracted for each material present in the dataset. The subscript e denotes that these are spectra corresponding to pure pixels.

6.3.2 Root Mean Squared Error (RMSE)

RMSE is used to evaluate abundance estimation accuracy. Estimated abundances $\hat{\mathbf{z}}$ are generated by the encoder (stage of the LDVAE). The difference between estimated abundances and the ground truth abundances capture abundance estimation accuracy. It is computed as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\mathbf{z}_n - \hat{\mathbf{z}}_n)^2}. \tag{6.5}$$

Here, $\mathbf{z}$ denote the ground truth abundances. N denote the number of pixels used in this computation.

6.3.3 Mean Squared Error (MSE)

RMSE is used to capture spectra reconstruction accuracy. It is defined as follows:

$$\text{MSE} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \hat{\mathbf{x}}_n)^2. \tag{6.6}$$

Table 6.5: Statistics of the reconstruction errors using SAD and MSE to capture the differences between input pixels and the respective reconstructed signal.

	OnTech-HSI-Syn-21		Cuprite		HYDICE Urban		Samson	
	SAD	MSE	SAD	MSE	SAD	MSE	SAD	MSE
mean	0.0349	0.0031	0.0904	0.0120	0.0833	0.0010	0.1241	0.0107
std	0.0578	0.0023	0.0367	0.0035	0.0799	0.0004	0.1322	0.0044
min	0.0013	0.0007	0.0377	0.0031	0.0125	0.0004	0.0113	0.0002
25%	0.0063	0.0011	0.0671	0.0101	0.0380	0.0007	0.0429	0.0082
50%	0.0100	0.0022	0.0779	0.0120	0.0556	0.0009	0.0647	0.0114
75%	0.0204	0.0049	0.0988	0.0147	0.0916	0.0013	0.1574	0.0135
max	0.2850	0.0101	0.2592	0.0185	0.9507	0.0019	1.0919	0.0197

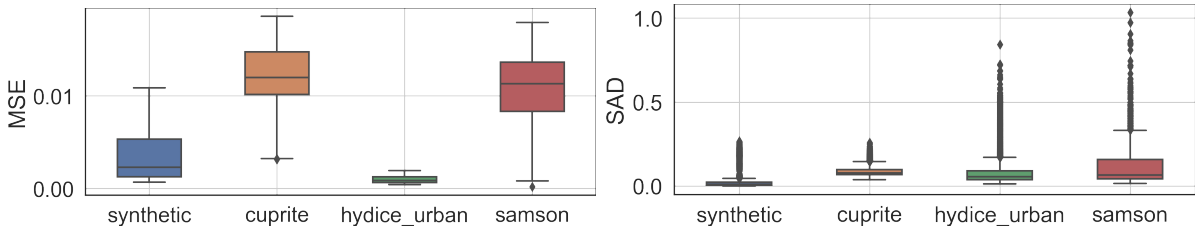

Figure 6.17: Reconstruction errors for all datasets. SAD and MSE measures capture the differences between pixels and their reconstructions.

Here $\hat{\mathbf{x}}$ representes the reconstructed spectrum, $\mathbf{x}$ represents the input, and N denotes the number of pixels used in this computation.

We study the proposed model on three tasks: 1) signal reconstruction, 2) endmember extraction, and 3) abundance estimation. The first task is important to confirm that LDVAE is able to reconstruct the input spectrum from its latent state. Tasks 2 and 3 together capture the performance of the proposed model on the task of hyperspectral pixel unmixing.

6.3.4 Spectral Reconstruction

Table 6.5 lists reconstruction errors using SAD and MSE metrics on OnTech-HSI-Syn-21, Cuprite, Urban (HYDICE), and Samson datasets. These results confirm that the proposed model is able to reconstruct the input spectra from its latent state. Recall that the latent state parameterizes a Dirichlet distribution that encodes abundances. Therefore, we claim that it is possible to use the decoder (stage of trained LDVAE) to generate mixed spectra given

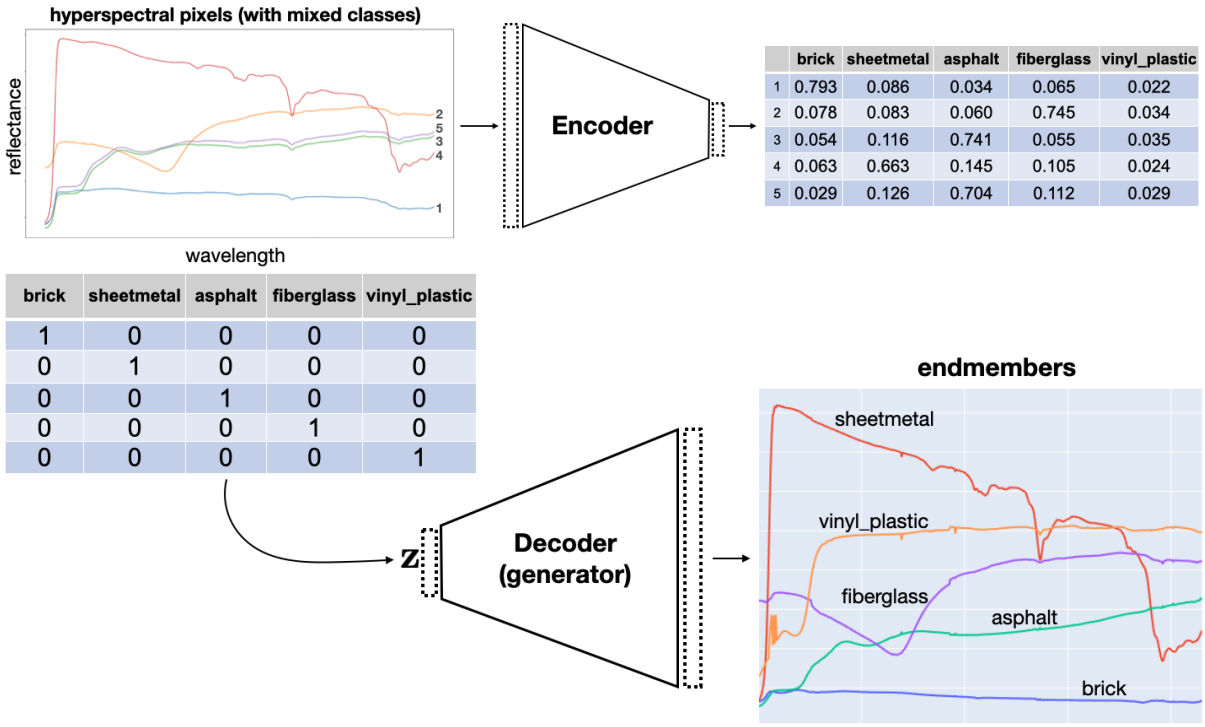

Figure 6.18: Abundance estimation and endmember extraction using the proposed method. The encoder stage of LDVAE estimates the abundances for a given input pixel while the decoder stage extracts endmembers given an abundances vector in the one-hot-encoder format.

Table 6.6: Endmember extraction results for OnTech-HSI-Syn-21 dataset. We use SAD metric to evaluate the distance of extracted endmembers from ground truth endmembers.

SNR	**LDVAE**	SSWNMF Zhang et al. (2022)	SGSNMF Wang et al. (2017)	TV-RSNMF He et al. (2017)	RSNMF He et al. (2017)	GLNMF Lu et al. (2013)	L1/2-NMF Qian et al. (2011)	VCA+FCLS Nascimento and Dias (2005b)
20 dB	0.0224 ±0.01	0.0636 ±0.40	0.0782 ±0.50	0.0679 ±0.30	0.0731 ±0.50	0.0724 ±0.05	0.0744 ±0.40	0.1358 ±0.30
30 dB	0.0138 ±0.01	0.0122 ±0.01	0.0176 ±0.03	0.0131 ±0.03	0.0138 ±0.05	0.0144 ±0.04	0.0142 ±0.04	0.0350 ±0.06
40 dB	0.0081 ±0.00	0.0029 ±0.02	0.0033 ±0.03	0.0036 ±0.02	0.0041 ±0.04	0.0044 ±0.05	0.0037 ±0.04	0.0125 ±0.05
50 dB	0.0082 ±0.00	0.0012 ±0.02	0.0019 ±0.02	0.0014 ±0.03	0.0020 ±0.04	0.0023 ±0.04	0.0024 ±0.03	0.0049 ±0.06
INF	0.0069 ±0.00	-	-	-	-	-	-	-

Table 6.7: Abundances estimation results for OnTech-HSI-Syn-21 dataset. We use RMSE metric to evaluate the distance of estimated abundances vectors and ground truth abundances vectors.

SNR	**LDVAE**	SSWNMF Zhang et al. (2022)	SGSNMF Wang et al. (2017)	TV-RSNMF He et al. (2017)	RSNMF He et al. (2017)	GLNMF Lu et al. (2013)	L1/2-NMF Qian et al. (2011)	VCA+FCLS Nascimento and Dias (2005b)
20 dB	0.0052 ±0.00	0.1339 ±0.20	0.1322 ±0.40	0.1342 ±0.30	0.1426 ±0.40	0.1434 ±0.60	0.1430 ±0.50	0.1704 ±0.03
30 dB	0.0302 ±0.00	0.0386 ±0.20	0.0391 ±0.30	0.0420 ±0.20	0.0426 ±0.30	0.0429 ±0.03	0.0432 ±0.20	0.0548 ±0.20
40 dB	0.0303 ±0.00	0.0122 ±0.03	0.0148 ±0.05	0.0142 ±0.04	0.0147 ±0.05	0.0150 ±0.04	0.0153 ±0.03	0.0164 ±0.10
50 dB	0.0303 ±0.00	0.0041 ±0.02	0.0059 ±0.05	0.0050 ±0.03	0.0055 ±0.03	0.0064 ±0.04	0.0061 ±0.04	0.0087 ±0.08
INF	0.0052 ±0.00	-	-	-	-	-	-	-

abundances. Figure 6.17 shows the reconstruction errors for all datasets. Both SAD and MSE capture the differences between original and the reconstructed pixels.

6.3.5 Pixel Unmixing

Pixel unmixing comprises endmember extraction followed by abundance estimation. For our method, the encoder (stage of the LDVAE) solves abundance estimation for a given input spectrum. The decoder (stage) is able to reconstruct the endmember given a one-hot-encoded abundance vector. Figure 6.18 (top) shows abundance estimation. Here, five pixels are passed to the encoder that estimates their corresponding abundances. Pixel spectra are shown on the left, and each row of the table on the right represents abundances for the corresponding input spectrum. Figure 6.18(bottom) shows endmember extraction. Five one-hot-encoded abundance vectors are used by the decoder to construct the corresponding endmembers (shown on the right). Note that one-hot-encoded abundance vectors represent "pure materials" seen in the hyperspectral image.

Results on OnTech-HSI-Syn-21 Dataset

Table 6.6 shows endmember extraction results for the OnTech-HSI-Syn-21 dataset at different Signal-to-Noise Ratios (SNRs). Similarly, Table 6.7 shows classification results (abundances) for the OnTech-HSI-Syn-21 dataset at different SNRs. Figure 6.20 illustrates the endmember extraction results of LDVAE compared to the ground truth.

Table 6.8: Endmember extraction results for Cuprite dataset. We use SAD metric to evaluate the distance of extracted endmembers from ground truth endmembers.

	LDVAE	SSWNMF Zhang et al. (2022)	SGSNMF Wang et al. (2017)	TV-RSNMF He et al. (2017)	RSNMF He et al. (2017)	GLNMF Lu et al. (2013)	L1/2-NMF Qian et al. (2011)	VCA+FCLS Nascimento and Dias (2005b)
alunite	0.0097 ±0.01	0.1497 ±3.97	0.1238 ±4.01	0.1204 ±4.37	0.1189 ±4.39	0.1353 ±3.83	0.1496 ±3.32	0.1574 ±3.71
andradite	0.0381 ±0.04	-	-	-	-	-	-	-
buddingtonite	0.0051 ±0.01	0.0958 ±4.69	0.1021 ±3.47	0.0903 ±5.08	0.1342 ±4.72	0.1437 ±3.62	0.1441 ±4.16	0.1412 ±3.74
dumortierite	0.1922 ±0.19	-	-	-	-	-	-	-
kaolinite-1	0.0258 ±0.03	0.0885 ±2.94	0.0986 ±3.18	0.1097 ±3.47	0.0955 ±3.07	0.0967 ±4.01	0.0825 ±4.66	0.0736 ±4.42
kaolinite-2	0.0699 ±0.07	0.1206 ±3.67	0.1375 ±3.48	0.1213 ±3.82	0.1396 ±4.11	0.1356 ±3.91	0.1402 ±4.18	0.1420 ±4.16
muscovite	0.0064 ±0.01	0.1024 ±4.24	0.1061 ±3.18	0.1131 ±2.88	0.0997 ±3.46	0.0961 ±3.77	0.0889 ±3.03	0.1007 ±3.31
montmorillonite	0.0496 ±0.05	0.0651 ±3.08	0.0705 ±3.36	0.0783 ±3.95	0.0744 ±3.12	0.0838 ±4.28	0.0876 ±2.91	0.0974 ±3.39
nontronite	0.1048 ±0.10	0.1138 ±4.15	0.1046 ±3.80	0.0911 ±3.49	0.0832 ±4.18	0.0953 ±3.41	0.1038 ±4.46	0.0772 ±2.10
pyrope	0.0156 ±0.02	0.1106 ±3.32	0.1208 ±3.83	0.1253 ±3.10	0.1469 ±3.12	0.1318 ±3.18	0.1123 ±4.91	0.1437 ±3.76
sphene	0.0347 ±0.03	0.1024 ±3.79	0.1179 ±4.02	0.1190 ±2.97	0.1134 ±2.54	0.1291 ±4.21	0.1252 ±5.18	0.1277 ±4.08
chalcedony	0.0055 ±0.01	0.1496 ±4.12	0.1221 ±4.02	0.1387 ±4.01	0.1224 ±4.19	0.1341 ±2.98	0.1520 ±3.43	0.1514 ±3.83
average	0.0465 ±0.05	0.1099 ±3.80	0.1104 ±3.63	0.1107 ±3.71	0.1128 ±3.69	0.1182 ±3.72	0.1186 ±4.02	0.1212 ±3.65

Table 6.9: Endmember extraction results for HYDICE Urban dataset. We use SAD metric to evaluate the distance of extracted endmembers from ground truth endmembers.

	LDVAE	SSWNMF Zhang et al. (2022)	CNNAEU Palsson et al. (2020)	SGSRNMF Palsson et al. (2020)	SHDP Palsson et al. (2020)	MTLAEU Palsson et al. (2020)	VCA+FCLS Nascimento and Dias (2005b)
asphalt road	0.4262 ±0.43	0.0782 ±3.29	0.0575 ± 0.0058	0.2446 ±0.0204	0.2658 ± 0.0751	0.0843 ± 0.0046	0.2246 ±3.44
grass	0.3323 ±0.33	0.1490 ±3.58	0.0366 ± 0.0047	1.3006 ±0.0444	0.5524 ± 0.3172	0.0421 ± 0.0036	0.1981 ±3.39
tree	0.3177 ±0.32	0.1173 ±3.46	0.0321 ± 0.0039	0.0967 ± 0.0113	0.0777 ± 0.0171	0.0539 ± 0.0039	0.2137 ±2.41
roof	0.4393 ±0.44	0.0713 ±3.61	0.0332 ± 0.0066	0.1916 ± 0.0862	0.4117 ± 0.1720	0.0415 ± 0.0045	0.2673 ±3.77
metal	0.7004 ±0.70	0.1241 ±2.76	-	-	-	-	0.1848 ±3.68
dirt	0.2806 ±0.28	0.0802 ±3.17	-	-	-	-	0.1992 ±3.43
average	0.4161 ±0.42	0.1034 ±3.31	0.0398 ± 0.0030	0.4584 ± 0.0148	0.3269 ± 0.0555	0.0555± 0.0019	0.2146 ±3.35

Results on Cuprite Dataset

Table 6.8 lists endmember extraction results for Cuprite dataset. The results suggest that our approach is able to handle situations where pixel-level abundance data is not available for training by leveraging the *transfer learning* paradigm. Note also that endmember extraction results for our method are similar to those achieved by competing approaches. These results also confirm that our method is applicable to real-world scenarios where oftentimes pixel-level abundance information is hard to collect.

Results on Urban (HYDICE) Dataset

Table 6.9 presents the results of endmember extraction obtained from the known Urban Dataset, frequently used as benchmarks for HSI unmixing. Table 6.10 shows the classification results.

Table 6.10: Abundances estimation results for Hydice Urban dataset. We use RMSE metric to evaluate the distance of estimated abundances vectors and ground truth abundances vectors.

	LDVAE	SSWNMF Zhang et al. (2022)	CNNAEU Palsson et al. (2020)	SGSRNMF Palsson et al. (2020)	SHDP Palsson et al. (2020)	MTLAEU Palsson et al. (2020)	VCA+FCLS Nascimento and Dias (2005b)
asphalt road	0.2289 ± 0.00	-	0.1249 ± 0.0400	0.2857 ± 0.0762	0.3015 ± 0.1200	0.151658 ± 0.0316	-
grass	0.1832 ± 0.00	-	0.1256 ± 0.0400	0.4467 ± 0.1015	0.3847 ± 0.2691	0.15 ± 0.0400	-
tree	0.1737 ± 0.00	-	0.0854 ± 0.0387	0.2674 ± 0.1308	0.2886 ± 0.1533	0.0824621 ± 0.0300	-
roof	0.1250 ± 0.00	-	0.0854 ± 0.0387	0.1892 ± 0.0424	0.2729 ± 0.2385	0.0888819 ± 0.0283	-
metal	0.2599 ± 0.00	-	-	-	-	-	-
dirt	0.1334 ± 0.00	-	-	-	-	-	-
average	0.1840 ± 0.00	0.0048 ±0.72	0.1072 ± 0.0316	0.3116 ± 0.0922	0.3150 ± 0.1428	0.122474 ± 0.0283	0.0119 ± 0.66

Table 6.11: Endmember extraction results for Samson dataset. We use SAD metric to evaluate distances between extracted and ground truth endmembers.

	LDVAE	SSWNMF Zhang et al. (2022)	CNNAEU Palsson et al. (2020)	SGSRNMF Palsson et al. (2020)	SHDP Palsson et al. (2020)	MTLAEU Palsson et al. (2020)
soil	0.0959 ±0.10	-	0.0373 ± 0.0210	0.0086 ± 0.0001	0.2147 ± 0.3299	0.0225 ± 0.0060
tree	1.2788 ±1.28	-	0.0397 ± 0.0038	0.0395 ± 0.0019	0.0375 ± 0.0004	0.0371 ± 0.0028
water	0.4022 ±0.40	-	0.0430 ± 0.0092	0.0923 ± 0.0024	0.2064 ± 0.0916	0.0338 ± 0.0031
average	0.5923 ±0.59	-	0.0400 ± 0.0067	0.0468 ± 0.0003	0.1527 ± 0.1390	0.0311 ± 0.0017

Table 6.12: Abundances estimation results for Samson dataset. We use RMSE metric to evaluate the distance of estimated abundances vectors and ground truth abundances vectors.

	LDVAE	CNNAEU Palsson et al. (2020)	SGSRNMF Palsson et al. (2020)	SHDP Palsson et al. (2020)	MTLAEU Palsson et al. (2020)	VCA+FCLS Nascimento and Dias (2005b)	PLMM Borsoi et al. (2020)	ELMM Borsoi et al. (2020)	GLMM Borsoi et al. (2020)	DeepGUn Borsoi et al. (2020)
soil	0.2609 ±0.00	0.2766 ± 0.2600	0.1778 ± 0.0200	0.2853 ± 0.1697	0.0872 ± 0.0374	-	-	-	-	-
tree	0.3461 ±0.00	0.2512 ± 0.2659	0.2400 ± 0.0557	0.2496 ± 0.1204	0.0608 ± 0.0265	-	-	-	-	-
water	0.3165 ±0.00	0.1288 ± 0.1153	0.3503 ± 0.0583	0.3948 ± 0.0949	0.0539 ± 0.0316	-	-	-	-	-
average	0.3078 ±0.00	0.2283 ± 0.2119	0.2657 ± 0.0458	0.3160 ± 0.1095	0.0693 ± 0.0283	0.0545	0.0239	0.0119	0.0006	0.0862

Results on Samson Dataset

Table 6.11 presents the results of endmember extraction and Table 6.12 shows the classification results obtained from Samson Dataset.

6.3.6 Discussion

Unmixing algorithms perform two tasks: endmember extraction and pixel-level abundance estimation. The proposed method is able to extract endmembers by using its decoder to generate spectra corresponding to one-hot encoded abundance vector. Recall that one-hot encoded abundance vectors correspond to pure pixels (see Figure 6.18 bottom). The proposed method performs pixel-level abundance estimation using its encoder that takes in a pixel spectra and outputs the latent space that represents abundances (Figure 6.18 top). The proposed architecture is inherently non-linear, consequently, we surmise that, it is able to capture the non-linear effects present in the unmixing task. These effects are more prevalent in micro-spectroscopy images where each material is composed of several elements. Similarly, these effects are present in low-resolution HSI images captured in a remote sensing setup.

The results show that LDVAE performed well in all scenarios. The results on synthetic data are aggregated over all classes, as we focused on the robustness to the noise present in the signal. The LDVAE model shows consistent performance at all noise levels; however, sometimes it does not achieve the best results. The results on tables 6.6 and 6.7 demonstrate the LDVAE performed similarly to the other methods, however LDVAE's performance does not

degrade when SNR decrease. We attribute this behavior to the fact that LDVAE learned the small variability on pixel values due to the probabilistic approach of variational autoencoders. However, this variability also affected the absolute values of the pixels during reconstruction. In other words, the LDVAE managed to average out the noise, which also explains the reduced reconstruction metrics, but holding up classification performance (Table 6.6). The images in Figure 6.19 show results of classification results compared to the ground truth.

Table 6.8 indicates that LDVAE successfully performs endmember extraction on the Cuprite dataset (see also Figures 6.21 and 6.22). Recall that the model was never trained on the Cuprite dataset. Rather the model was trained on Cuprite Synthetic dataset. Considering the lack of ground truth and the applicability of a synthetic dataset for model training, we observed several opportunities for further research and improvements. This research is explored on the iLDVAE research and described in Sections 5.5.

The worst performance of LDVAE was on the HYDICE Urban dataset, even though it had the larger training set (see Figures 6.23 and 6.24). We used the 50/50 split strategy for training and evaluation of this model. We started with an initial random separation; then, we manually fixed some unbalanced classes. It is noticeable that all machine learning approaches suffer from these issues. We plan to investigate limitations and performance issues surrounding unbalanced input data at another time.

The results on Samson dataset were also satisfactory despite the small amount of ground truth data available for training. We can observe from Figure 6.25, that the abundances estimation could detect the prominent classes, but the proportion of each material was not estimated to the highest accuracy. Figure 6.26 shows a moderate performance on endmember extraction, however noisy in some parts of the spectra (see Figure 6.26, material=tree, bands 100-150).

The RMSE metrics shown in Tables 6.10 and 6.12 are averages of all pixels in each class. The small variance present in the abundances estimations may stem from the model converging toward a local minima as opposed to a global minima, which also results in higher average RMSE values when compared to other methods. These show 1) the validity of our method and 2) that further investigation is necessary to improve the model training and convergence with respect to model capacity, hyperparameters tuning, and feature engineering.

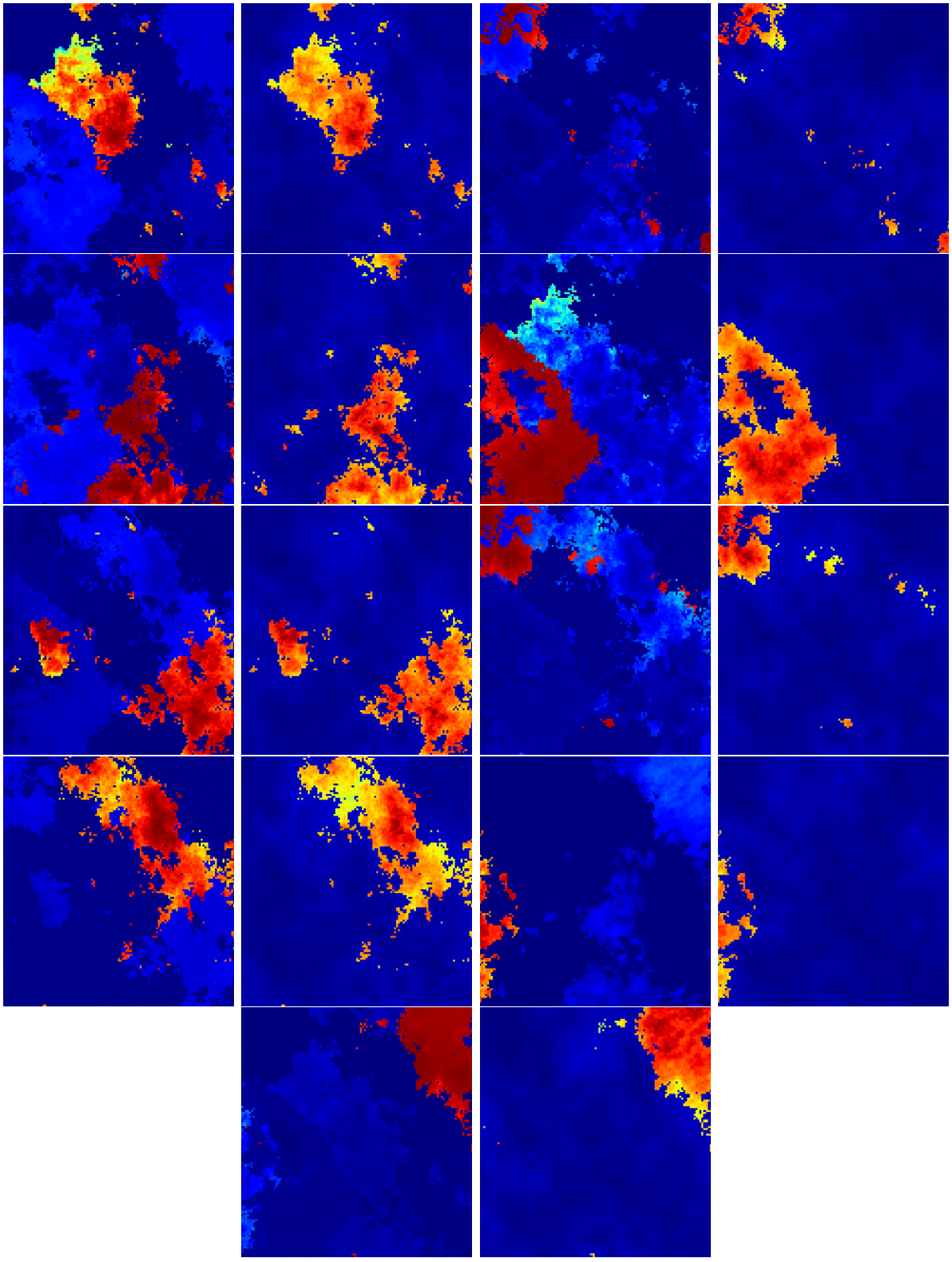

Figure 6.19: Abundances maps of the synthetic dataset; LDVAE (left) and ground truth (right) are side by side. From top to bottom and left to right: Adularia, Jarosite gds99, Jarosite gds101, Anorthite, Calcite, Alunite, Howlite, Corrensite, Fassaite.

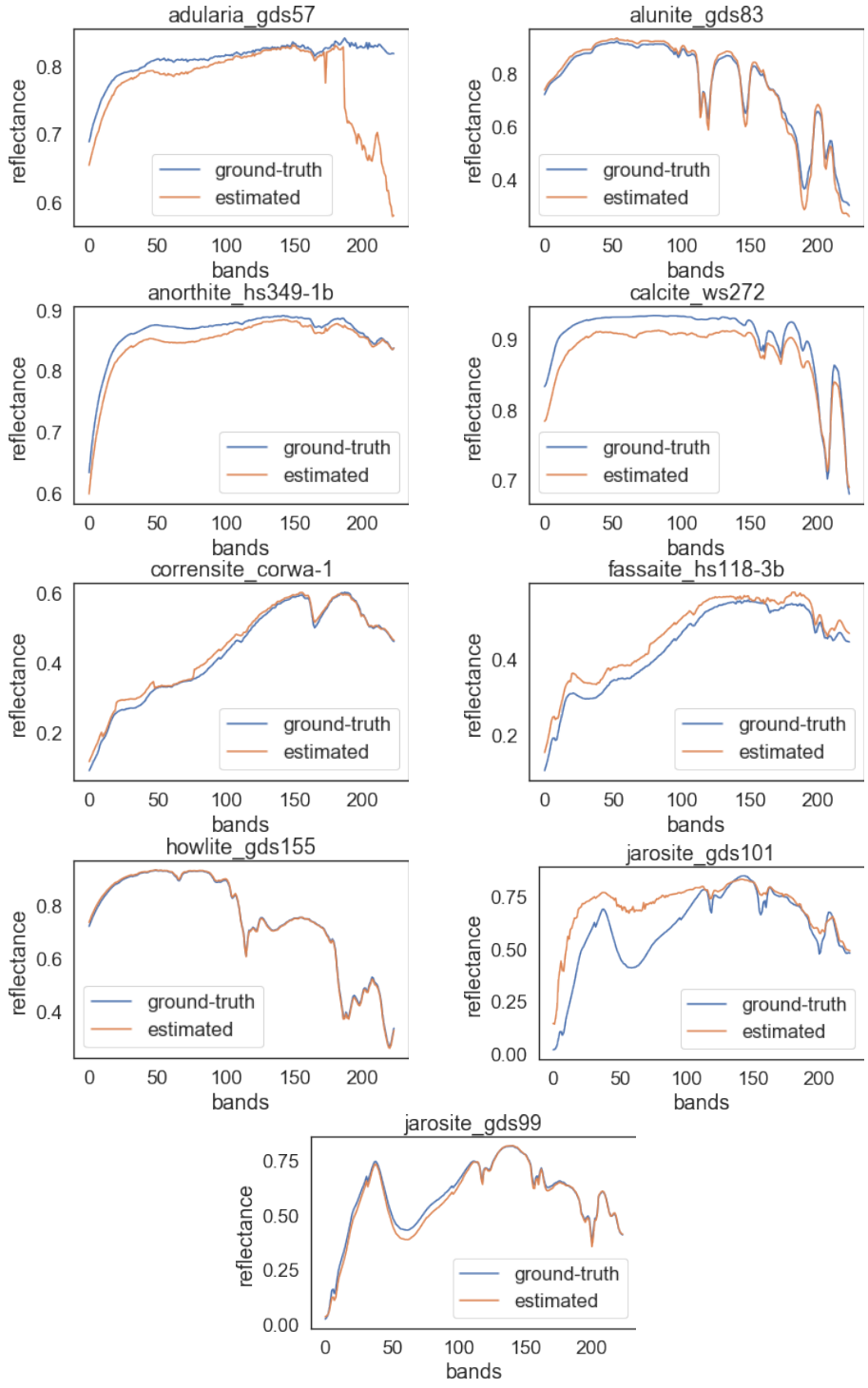

Figure 6.20: Endmembers of the Synthetic dataset generated by LDVAE: comparison with ground truth.

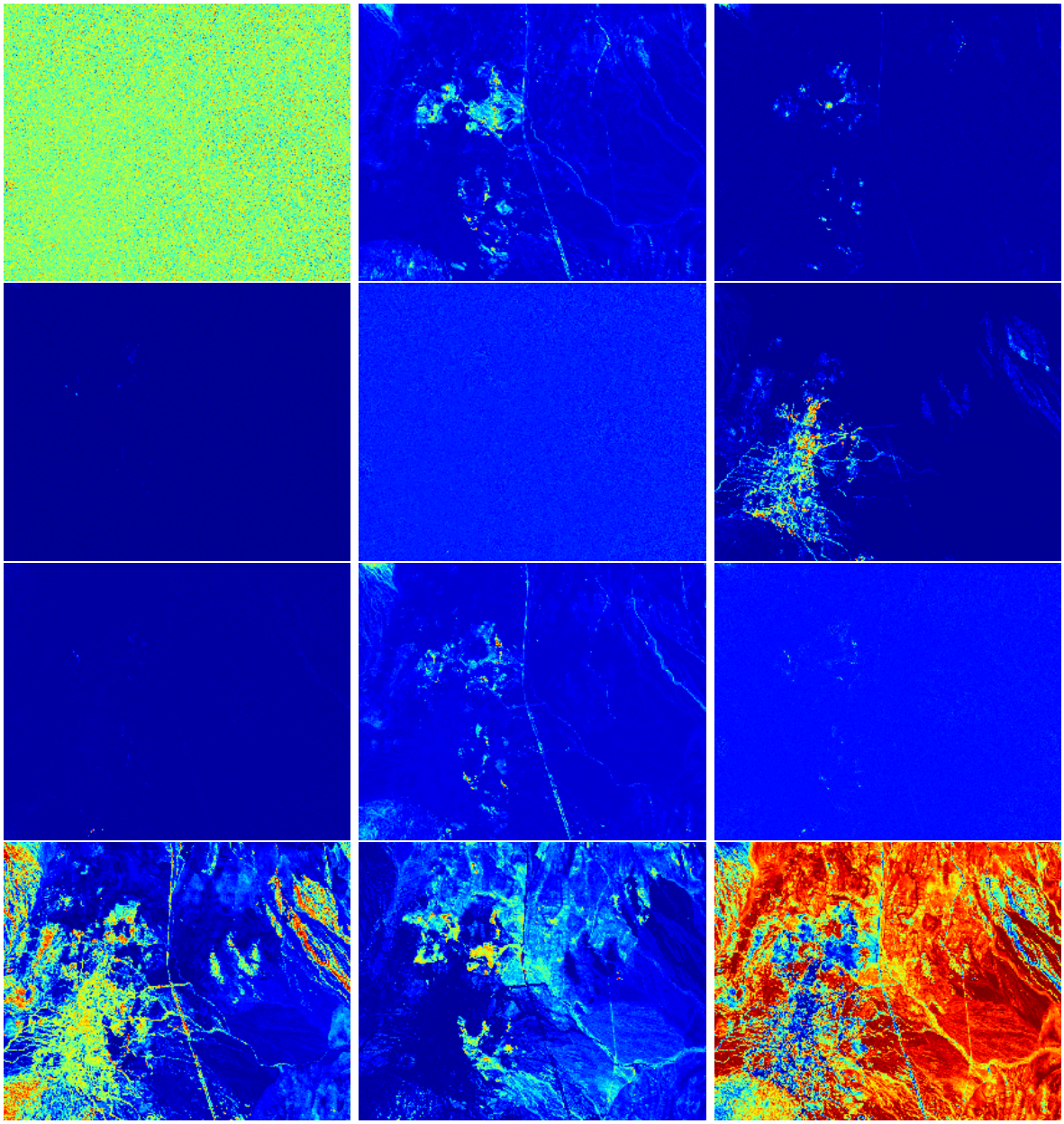

Figure 6.21: Abundances maps of the Cuprite dataset estimated by LDVAE (ground truth not available). From top to bottom and left to right: Alunite, Andradite, Buddingtonite, Chalcedony, Dumortierite, Kaolinite1, Kaolinite2, Montmorillonite, Muscovite Nontronite, Pyrope, and Sphene.

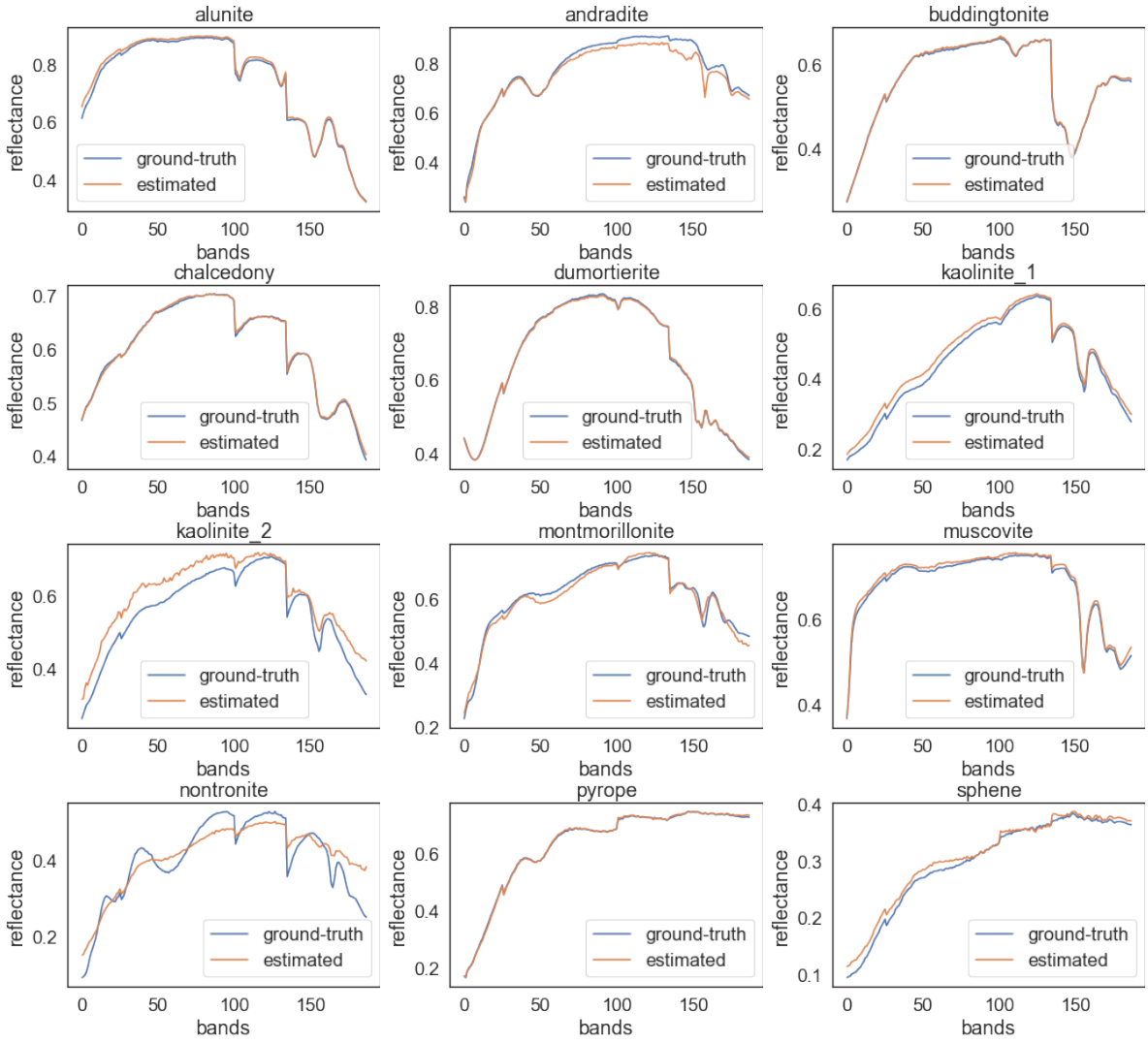

Figure 6.22: Endmembers of the Cuprite dataset generated by LDVAE: comparison with ground truth.

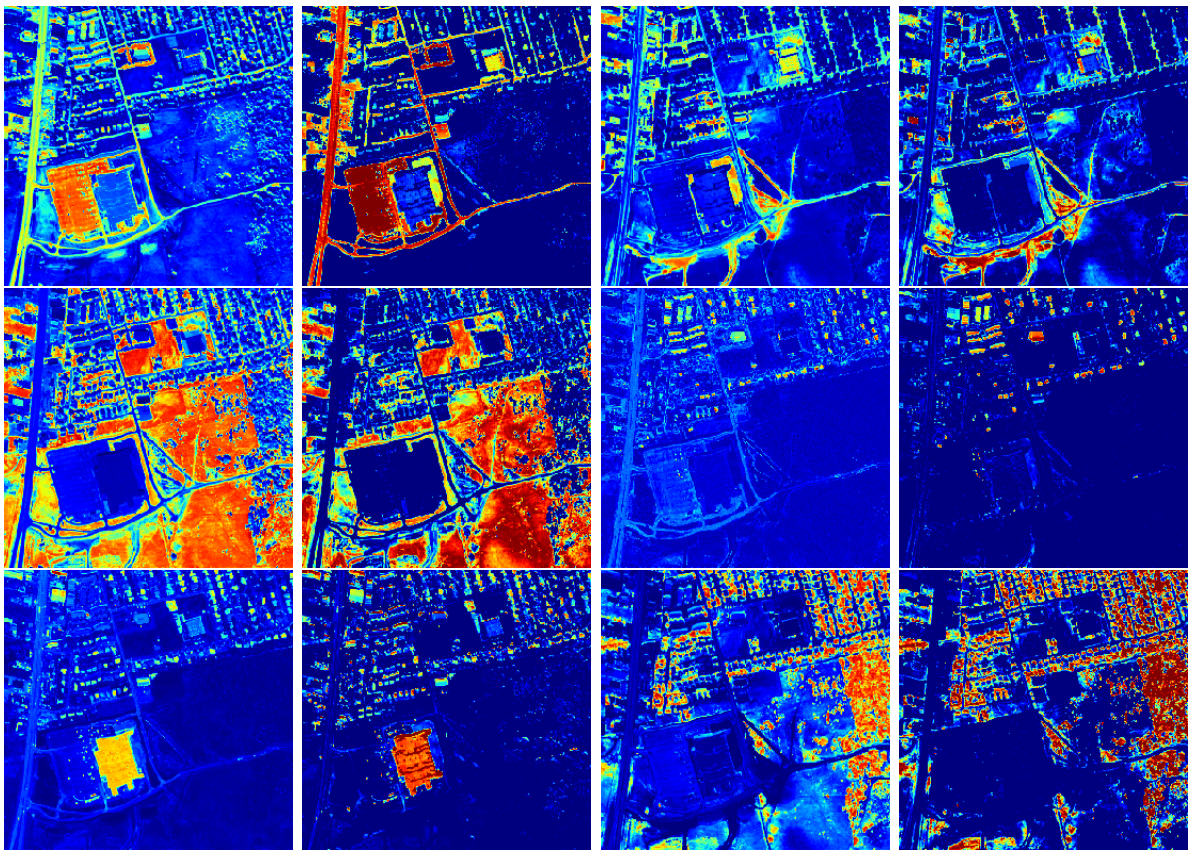

Figure 6.23: Abundances maps of the HYDICE Urban dataset; LDVAE (left) and ground truth (right) are side by side. From top to bottom and left to right: asphalt, dirt, grass, metal, roof, and tree

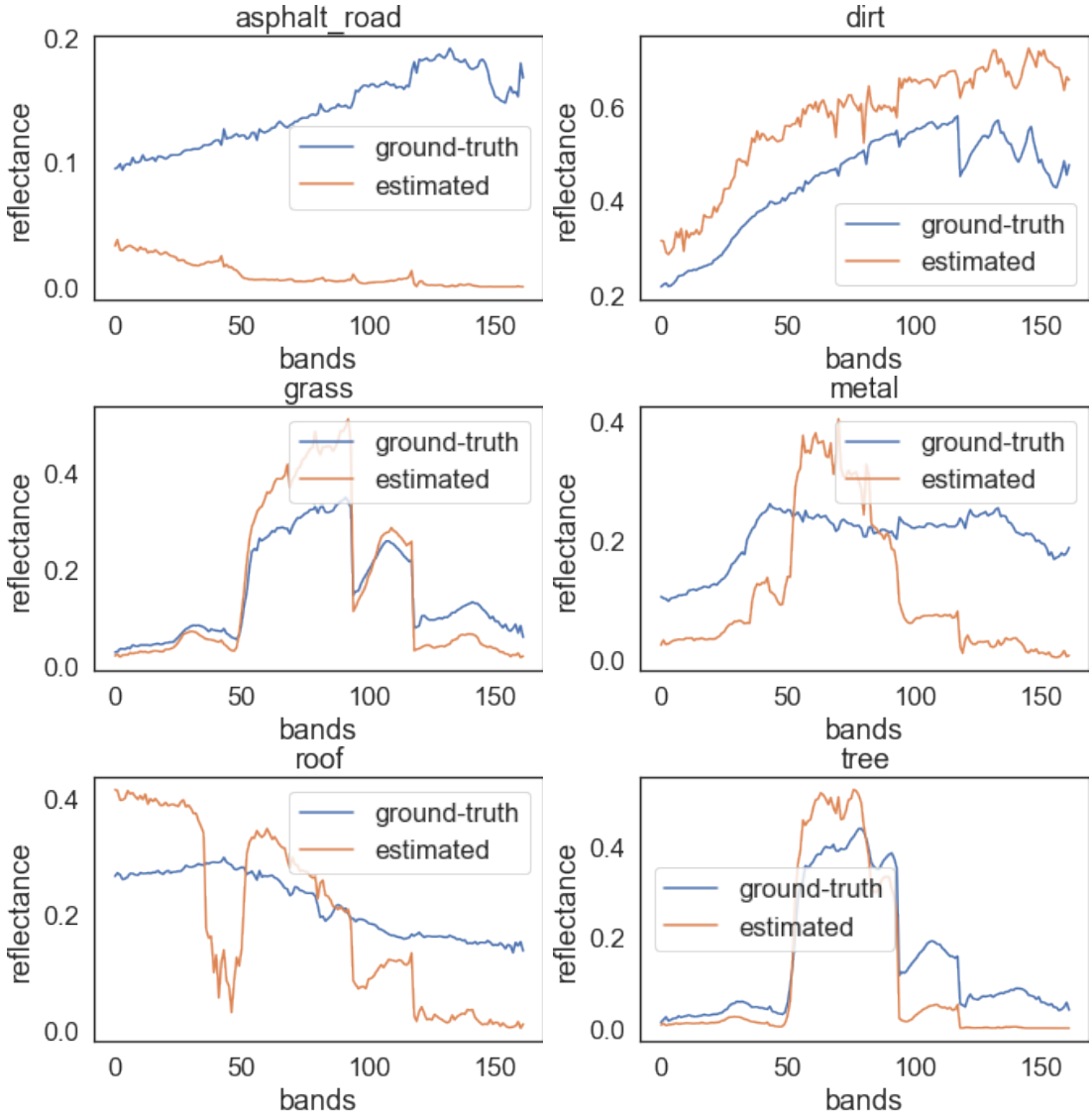

Figure 6.24: Endmembers of the HYDICE Urban dataset generated by LDVAE: comparison with ground truth.

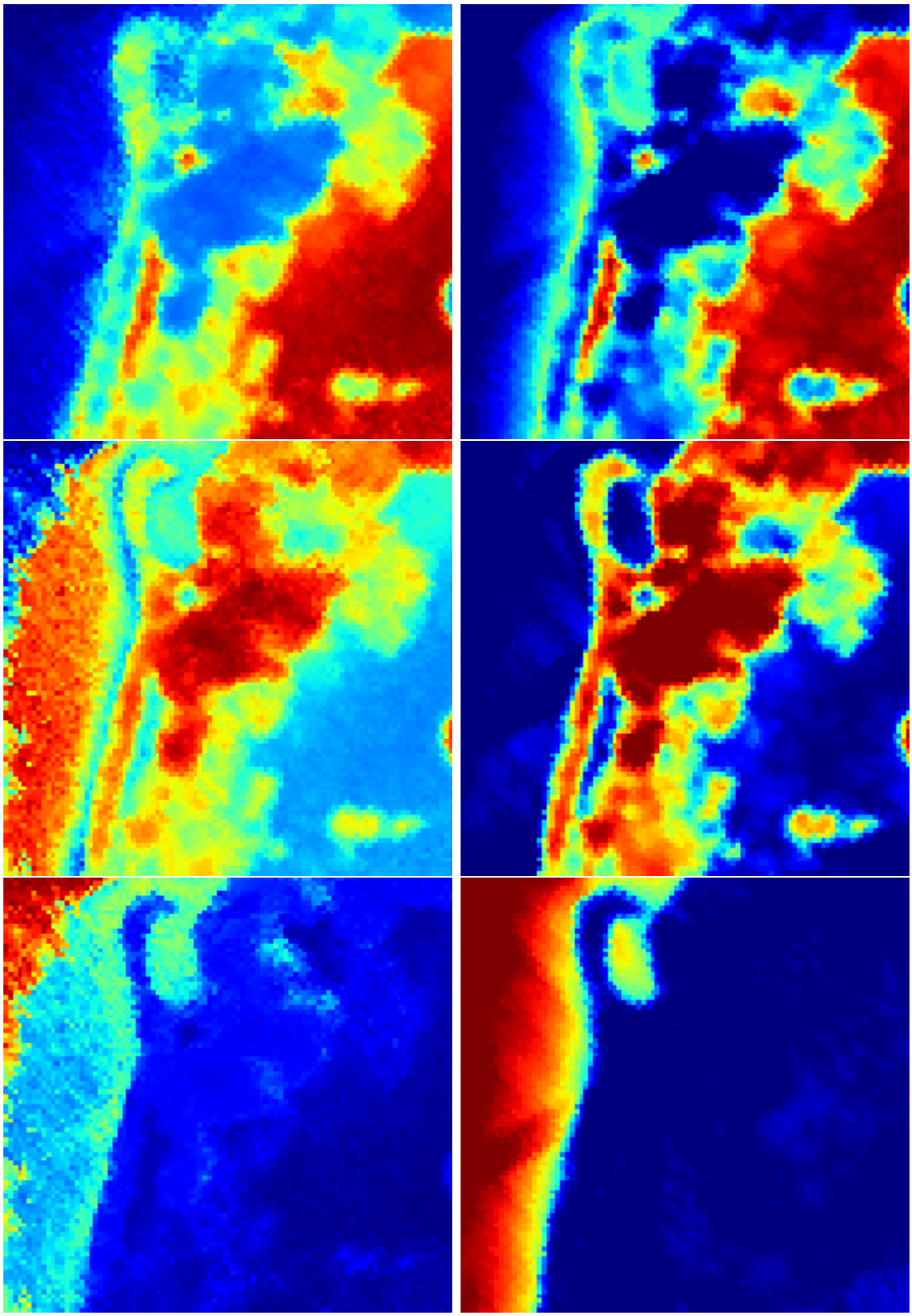

Figure 6.25: Abundances maps of the Samson dataset. Left side is LDVAE and right is ground truth right. From top to botom respectively: Soil, Tree, and Water

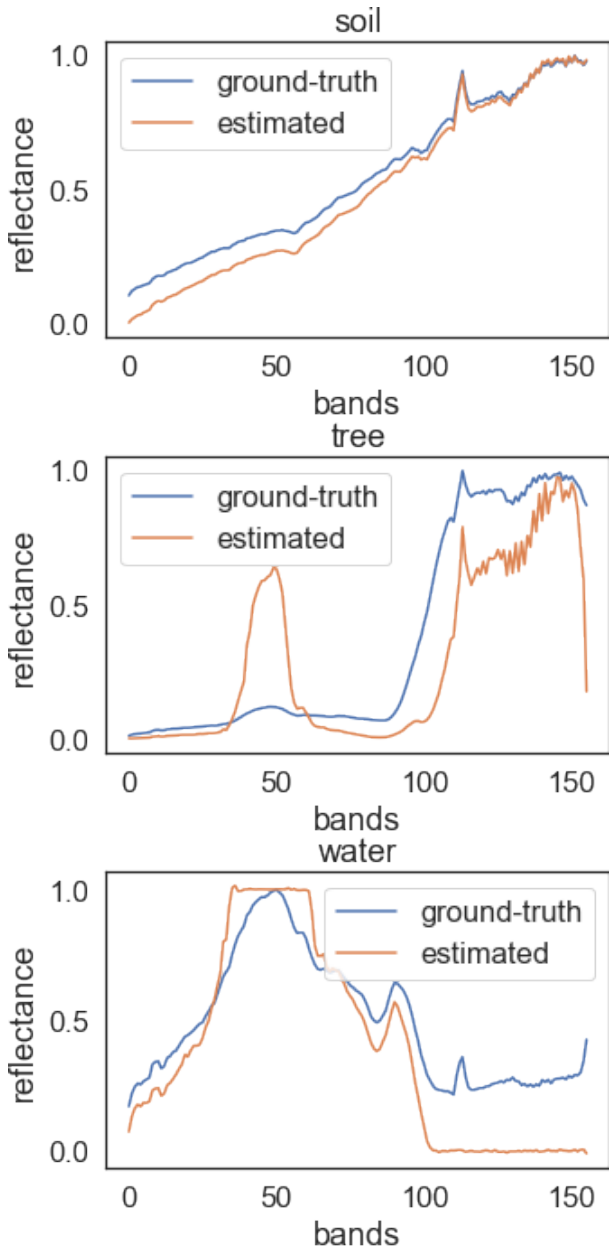

Figure 6.26: Endmembers of the Samson dataset generated by LDVAE: comparison with ground truth. From left to right: Soil, Tree and Water. The curves show the estimated and the ground truth endmembers.

Table 6.13: SAD and RMSE metric, respectively for endmember extraction abundances estimation (OnTech-HSI-Syn-6em dataset).

	SAD	RMSE
sagebrush	0.0447 $\pm 5 \times 10^{-4}$	0.0664 $\pm 1 \times 10^{-4}$
tumbleweed	0.0853 $\pm 5 \times 10^{-4}$	0.0712 $\pm 2 \times 10^{-4}$
grass	0.0233 $\pm 9 \times 10^{-4}$	0.0342 $\pm 1 \times 10^{-4}$
cattail	0.0721 $\pm 2.1 \times 10^{-3}$	0.0899 ± 0.0000
juniper	0.0429 $\pm 1.4 \times 10^{-3}$	0.0796 $\pm 1 \times 10^{-4}$
cactus	0.0810 $\pm 2.3 \times 10^{-3}$	0.0758 $\pm 1 \times 10^{-4}$
average	0.0582 $\pm 2.48 \times 10^{-2}$	0.0695 $\pm 1.91 \times 10^{-2}$

Table 6.14: SAD and RMSE metric, respectively for endmember extraction abundances estimation (HYDICE Urban dataset).

	SAD	RMSE
asphalt	0.0626 $\pm 2.9 \times 10^{-3}$	0.1438 $\pm 2 \times 10^{-5}$
grass	0.1251 $\pm 1.0 \times 10^{-3}$	0.1571 $\pm 4 \times 10^{-5}$
tree	0.1142 $\pm 1.6 \times 10^{-3}$	0.2547 $\pm 4 \times 10^{-5}$
roof	0.0841 $\pm 1.7 \times 10^{-3}$	0.1497 $\pm 2 \times 10^{-5}$
metal	0.5362 $\pm 4.7 \times 10^{-3}$	0.3360 $\pm 2 \times 10^{-5}$
dirt	0.0793 $\pm 4.2 \times 10^{-3}$	0.1490 $\pm 1 \times 10^{-5}$
average	0.1669 $\pm 1.824 \times 10^{-1}$	0.1984 $\pm 7.95 \times 10^{-2}$

6.4 Unmixing with the absence of ground truth

6.4.1 Results: OnTech-HSI-Syn-6em & HYDICE Urban

Ground truth is only available for OnTech-HSI-Syn-6em and HYDICE Urban dataset. We use iLDVAE to estimate endmembers and per-pixel abundances for these datasets and use Spectral Angle Distance (SAD) and Root Mean Squared Error (RMSE) metrics to capture the performance of iLDVAE (Ibarrola-Ulzurrun et al., 2019; Deborah et al., 2015). Tables 6.13 and 6.14 show SAD and RMSE results for OnTech-HSI-Syn-6em and HYDICE Urban datasets. SAD scores capture the quality of the estimated endmembers, whereas RMSE scores record the agreement between ground truth and estimated per-pixel abundances.

In addition, we also use segmentation accuracy

$$\text{acc}_{\text{seg}} = \frac{1}{W \cdot H} \mathbb{1}_{\mathbf{S}(x,y)=\mathbf{S}^{\text{gt}}(x,y)},$$

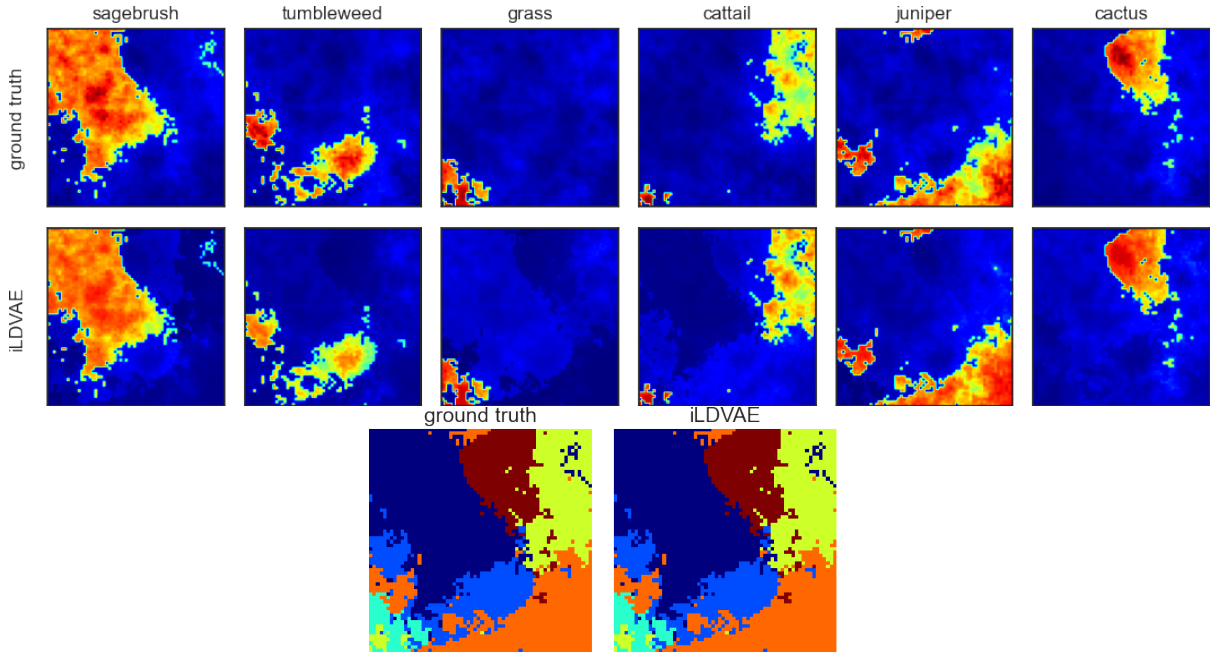

Figure 6.27: Abundances maps and segmentation estimated by iLDVAE (accuracy of segmentation = 1.00).

where W and H represent the width and height of the $\mathbf{I}$, respectively, $\mathbf{S}^{\text{gt}}$ is the ground truth segmentation, and $\mathbb{1}$ is the indicator function. Figures 6.27 and 6.28 show segmentation results for OnTech-HSI-Syn-6em and HYDICE Urban datasets.

6.4.2 Results: Cover Crop USDA

Cover Crop USDA dataset does not contain per-pixel abundances. Rather it only contains species plus soil proportions in each quadrat. For this dataset, we aggregate per-pixel abundances for each quadrat to estimate the proportion of the four species plus soil in that quadrat. We use per-species (plus soil) Pearson's Coefficient of Determination R^2 (Chicco et al., 2021) to see how well our method is able to estimate the proportions. Results are shown in Figure 6.29. Additionally, we compare iLDVAE against Canopeo Green Fraction method (Chung et al., 2017; Patrignani and Ochsner, 2015), which estimates the percentage of vegetation plus soil in each quadrate using RGB information only. Figure 6.30 shows estimated abundances for one quadrat. Most noteworthy is the abundance map of soil, which closely matches the RGB image on the right. Soil is visible in this RGB image that is constructed from the HSI image.

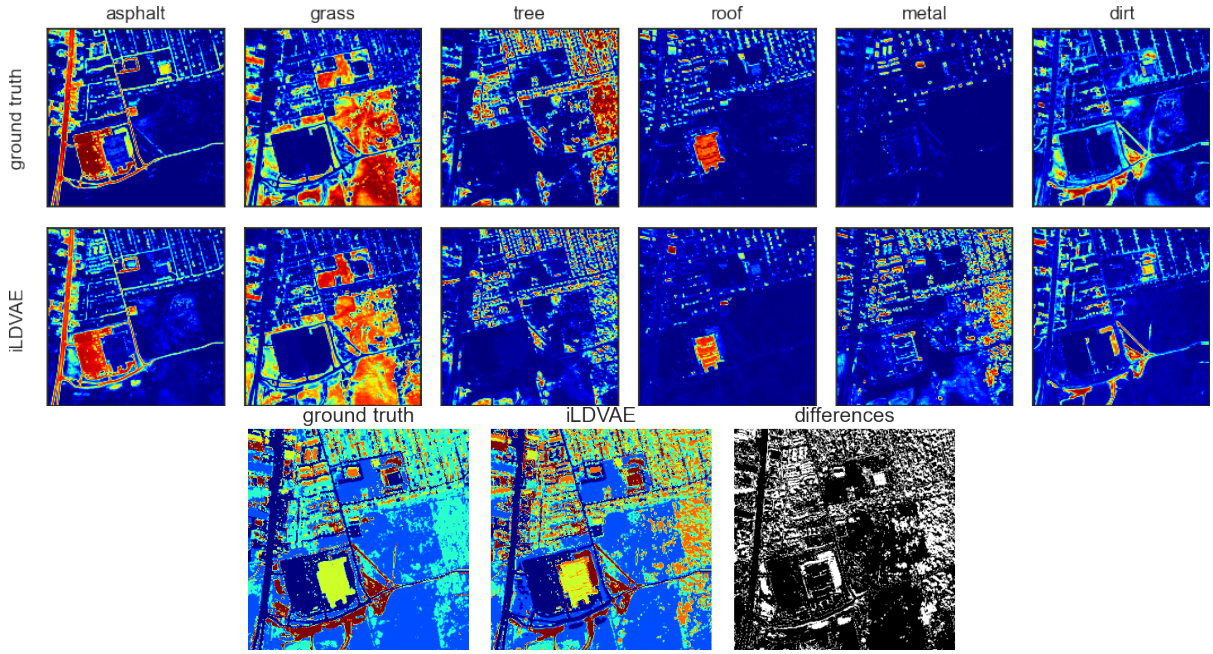

Figure 6.28: Abundances maps estimated and segmentation estimated by iLDVAE (accuracy = 0.7238).

6.4.3 Discussion

The experiments on OnTech-HSI-Syn-6em resulted in an accuracy of 100% on the segmentation task, with an average $SAD = 0.0582$ and $RMSE = 0.0695$, respectively for endmembers extraction abundances estimation (Table 6.13 and Figure 6.27). The results on HYDICE Urban showed an accuracy of 72.38% on the segmentation task, average $SAD = 0.1699$ and $RMSE = 0.1984$ respectively for endmembers extraction and abundances estimation (Table 6.14 and Figure 6.28). The main reason for the lower accuracy on HYDICE Urban dataset is the lack of pixels of higher purity for some of the materials. For Cover Crop USDA, iLDVAE demonstrated a high correlation with the "Percentage Vegetation Cover" metric compared to Canopeo Green Fraction algorithm and ground truth (Coefficient of Determination $R^2 = 0.7$).

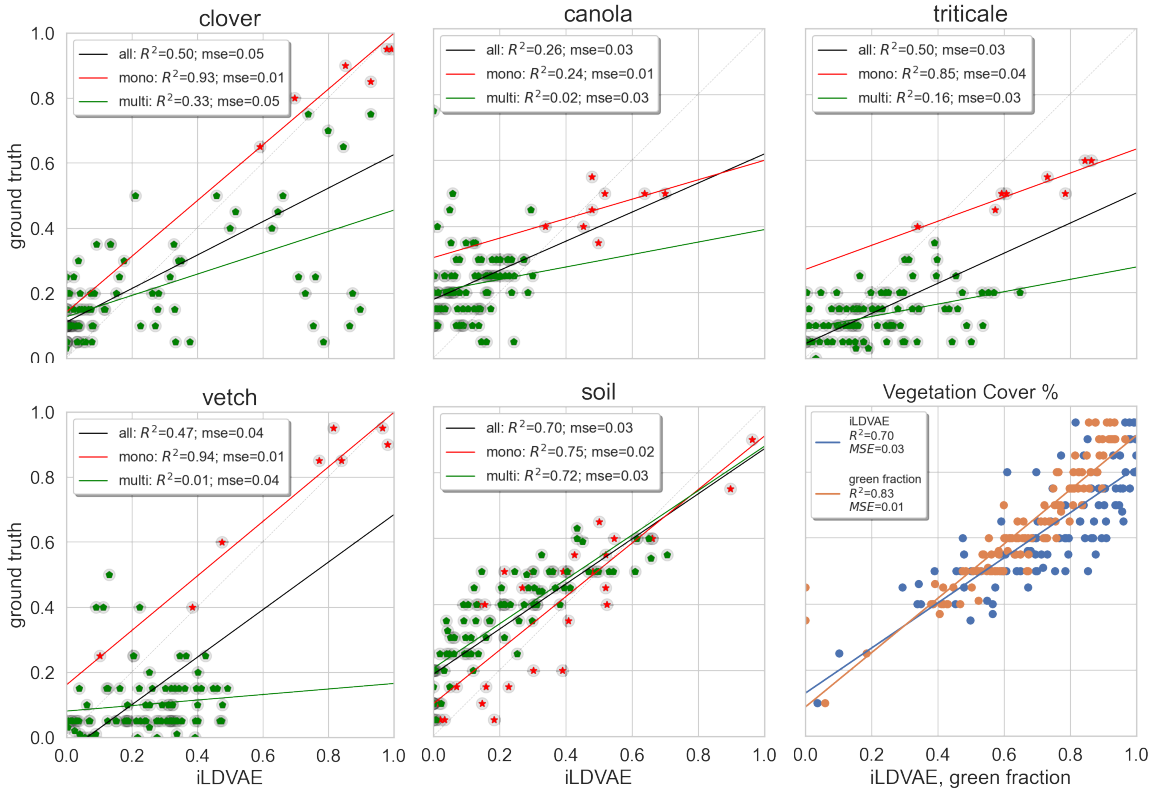

Figure 6.29: Abundances of each species estimated by iLDVAE (x-axis) vs. ground truth (y-axis); bottom-right: Percentage of vegetation cover: iLDVAE-estimated and Canopeo Green Fraction index (Chung et al., 2017; Patrignani and Ochsner, 2015) vs. ground truth (y-axis). Each data point corresponds to one quadrat (total number of quadrats=120).

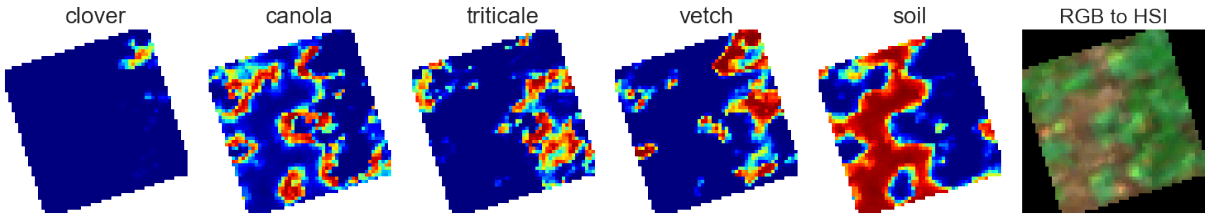

Figure 6.30: iLDVAE-estimated abundance maps and RGB image composite from HSI.

6.5 Incorporating Spatial Features for Pixel Unmixing

6.5.1 Datasets

All the experiments are conducted with four datasets. Samson is $95x95$, 156-channel hyperspectral image (Samson, 2024). This dataset contains three endmembers: soil, tree, and water. UHYDICE Urban is 307×307, 162-channel hyperspectral image covering a 2×2 m^2 (HYDICE, 2024). This dataset has three versions, containing four, five, and six endmembers. In this work, six endmembers are used. 80 : 20 training/testing split is used for both datasets. Cuprite dataset is a 512×614, 188-channel hyperspectral image (Cuprite, 2024). It contains twelve minerals (endmembers). This dataset does not provide ground truth abundances; therefore, we use Cuprite-synthetic dataset for training (Mantripragada and Qureshi, 2024). This showcases transfer learning, where a model trained on synthetic dataset is subsequently used to perform inference on a real world dataset. Lastly, OnTech-HSI-Syn-21 is a synthetic dataset containing nine endmembers. It is 128×128, 224-channel hyperspectral image.

6.5.2 Metrics

Root Mean Square Error (RMSE) is used to evaluate abundance estimation accuracy. It is computed as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\mathbf{z_n} - \hat{\mathbf{z}}_\mathbf{n})^2},$$

where $\mathbf{z}$ is the ground truth abundances, and $\hat{\mathbf{z}}$ is generated abundances. N denotes the number of pixels used in the computation.

Spectral Angle Distance (SAD) is a distance measurement between two spectral signals

$$\text{SAD} = \arccos\left(\frac{\hat{\mathbf{x}}_\mathbf{e}^\mathbf{T}\mathbf{x_e}}{\|\hat{\mathbf{x}}_\mathbf{e}^\mathbf{T}\|\|\mathbf{x_e}\|}\right),$$

where $\hat{\mathbf{x}}_\mathbf{e}$ represents the estimated endmember and $\mathbf{x_e}$ denotes the ground truth endmember. SAD is sometimes referred to as SAM (Chakravarty et al., 2021; Rashmi S and S, 2014). In this work, we use SAD to capture endmember extraction accuracy.

6.5.3 Experimental Settings

SpACNN-LDVAE was implemented using Pyro Python library (Bingham et al., 2019). All models were trained on NVidia V100-SXM2 gpus. Adam optimizer was used with learning rate 0.001 (Kingma and Ba, 2017).

6.5.4 Results

Table 6.15 provides RMSE and SAD metrics (both endmember specific and average values) for Samson dataset. Similarly Table 6.16 lists these metrics for HYDICE Urban dataset. Cuprite lacks abundance ground truth, so only endmember extraction results are provided for this dataset (see Table 6.17). Tables 6.18 and **??** provide endmember extraction (SAD) and abundance estimation (RMSE) results for OnTech-Syn-HSI-21 dataset under various SNR settings, respectively. The proposed method achieves lower RMSE and SAD numbers over MLP-LDVAE method. However, the SAD scores on Cuprite are comparable to those achieved by MLP-LDVAE. Recall that the model was trained on Cuprite Synthetic dataset, which lacks spatial coherence. This is perhaps why the proposed method did not achieve better numbers than MLP-LDVAE. This merits further discussion. In all datasets, the proposed method achieves lower standard deviations.

Table 6.15: Abundance Estimation and Endmember Extraction Results on Samson Dataset

		SpACNN-LDVAE	MLP-LDVAE	VCA+FCLS	PLMM	ELMM	GLMM	DeepGUn	EACNN
Soil	RMSE	0.2522 ± 0.00	0.2609 ± 0.00	-	-	-	-	-	-
	SAD	0.2097 ± 0.01	0.0959 ± 0.10	-	-	-	-	-	0.0328
Tree	RMSE	0.2614 ± 0.00	0.3431 ± 0.00	-	-	-	-	-	-
	SAD	0.5347 ± 0.03	1.2788 ± 1.28	-	-	-	-	-	0.0519
Water	RMSE	0.2098 ± 0.00	0.3165 ± 0.00	-	-	-	-	-	-
	SAD	0.8233 ± 0.04	0.4022 ± 0.40	-	-	-	-	-	0.1026
Average	RMSE	0.2412 ± 0.00	0.3078 ± 0.00	0.0545	0.0239	0.0119	0.0006	0.0862	0.0171
	SAD	0.5525 ± 0.03	0.5923 ± 0.59	-	-	-	-	-	0.0624

MLP-LDVAE: Mantripragada and Qureshi (2024); VCA+FCLS: Nascimento and Dias (2005b); PLMM: Thouvenin et al. (2016); ELMM: Veganzones et al. (2014); Drumetz et al. (2016b); GLMM: Imbiriba et al. (2018); DeepGUn: Borsoi et al. (2020); EACNN: Jin et al. (2022)

Table 6.16: Abundance Estimation and Endmember Extraction Results on HYDICE Urban Dataset

		SpACNN-LDVAE	MLP-LDVAE	SSWNMF	SGSNMF	TV-RSNMF
Asphalt road	RMSE	0.1566 ± 0.00	0.2889 ± 0.00	-	-	-
	SAD	0.2786 ± 0.02	0.4262 ± 0.43	0.0782 ± 3.29	0.0841 ± 4.01	0.0770 ± 2.97
Grass	RMSE	0.1977 ± 0.00	0.1832 ± 0.00	-	-	-
	SAD	0.1936 ± 0.01	0.3323 ± 0.33	0.1490 ± 3.58	0.1516 ± 3.25	0.1495 ± 3.54
Tree	RMSE	0.1632 ± 0.00	0.1737 ± 0.00	-	-	-
	SAD	0.4411 ± 0.04	0.3177 ± 0.32	0.1173 ± 3.46	0.1199 ± 3.36	0.1269 ± 4.02
Roof	RMSE	0.1283 ± 0.00	0.125 ± 0.00	-	-	-
	SAD	0.4502 ± 0.03	0.4393 ± 0.44	0.0713 ± 3.61	0.0731 ± 3.54	0.0746 ± 4.09
Metal	RMSE	0.0992 ± 0.00	0.2599 ± 0.00	-	-	-
	SAD	0.3241 ± 0.02	0.7004 ± 0.70	0.1241 ± 2.76	0.1250 ± 3.81	0.1247 ± 3.53
Dirt	RMSE	0.1894 ± 0.00	0.1334 ± 0.00	-	-	-
	SAD	0.2026 ± 0.01	0.2806 ± 0.28	0.0802 ± 3.17	0.0859 ± 3.91	0.0849 ± 3.92
Average	RMSE	0.1558 ± 0.00	0.1840 ± 0.00	0.0048 ± 0.72	0.0061 ± 0.67	0.0055 ± 0.81
	SAD	0.3151 ± 0.02	0.4161 ± 0.42	0.1034 ± 3.31	0.1060 ± 3.68	0.1063 ± 3.68

		SpACNN-LDVAE	RSNMF	GLNMF	$L_{1/2}$NMF	VCA+FCLS
Asphalt road	RMSE	0.1566 ± 0.00	-	-	-	-
	SAD	0.2786 ± 0.02	0.0869 ± 3.81	0.1008 ±3.19	0.0889 ± 2.88	0.2246 ± 3.44
Grass	RMSE	0.1977 ± 0.00	-	-	-	-
	SAD	0.1936 ± 0.01	0.1594 ± 3.62	0.1531 ± 3.06	0.1452 ± 3.57	0.1981 ± 3.39
Tree	RMSE	0.1632 ± 0.00	-	-	-	-
	SAD	0.4411 ± 0.04	0.1457 ± 4.29	0.1424 ± 3.79	0.1509 ± 3.18	0.2137 ± 2.41
Roof	RMSE	0.1283 ± 0.00	-	-	-	-
	SAD	0.4502 ± 0.03	0.0849 ± 3.90	0.0986 ± 4.62	0.0863 ± 4.06	0.2673 ± 3.77
Metal	RMSE	0.0992 ± 0.00	-	-	-	-
	SAD	0.3241 ± 0.02	0.1324 ± 4.15	0.1370 ± 4.28	0.1334 ± 3.90	0.1848 ± 3.68
Dirt	RMSE	0.1894 ± 0.00	-	-	-	-
	SAD	0.2026 ± 0.01	0.0798 ± 3.77	0.1059 ± 3.96	0.1063 ± 3.54	0.1992 ± 3.43
Average	RMSE	0.1558 ± 0.00	0.0053 ± 0.98	0.0069 ± 0.85	0.0044 ±0.76	0.0119 ± 0.66
	SAD	0.3151 ± 0.02	0.1148 ± 3.92	0.1230 ± 3.52	0.1185 ± 3.52	0.2142 ± 3.35

MLP-LDVAE: Mantripragada and Qureshi (2024); SSWNMF: Zhang et al. (2022); SGSNMF: Wang et al. (2017); TV-RSNMF: He et al. (2017); RSNMF: He et al. (2017); GLNMF: Lu et al. (2013); $L_{1/2}$NMF: Qian et al. (2011); VCA+FCLS: Nascimento and Dias (2005b)

Table 6.17: Endmember Extraction Results on Cuprite Dataset

	SpACNN-LDVAE	MLP-LDVAE	SSWNMF	SGSNMF	TV-RSNMF
Alunite	0.0683 ±0.00	0.0097 ±0.01	0.1497 ± 3.97	0.1238 ± 4.01	0.1204 ± 4.37
Andradite	0.0462 ±0.00	0.0381 ±0.04	-	-	-
Buddingtonite	0.0227 ±0.00	0.0051 ±0.01	0.0958 ± 4.69	0.1021 ± 3.47	0.0903 ± 5.08
Dumortierite	0.0500 ±0.00	0.1922 ±0.19	-	-	-
Kaolinite_1	0.0740 ±0.00	0.0258 ±0.03	0.0885 ± 2.94	0.0986 ± 3.18	0.1097 ± 3.47
Kaolinite_2	0.0249 ±0.00	0.0699 ±0.07	0.1206 ± 3.67	0.1375 ± 3.48	0.1213 ± 3.82
Muscovite	0.0320 ±0.00	0.0064 ±0.01	0.1024 ± 4.24	0.1061 ± 3.18	0.1131 ± 2.88
Montmorillonite	0.0214 ±0.00	0.0496 ±0.05	0.0651 ± 3.08	0.0705 ± 3.36	0.0783 ± 3.95
Nontronite	0.0639 ±0.00	0.1048 ±0.10	0.1138 ± 4.15	0.1046 ± 3.80	0.0911 ± 3.49
Pyrope	0.0342 ±0.00	0.0156 ±0.02	0.1106 ± 3.32	0.1208 ± 3.83	0.1253 ± 3.10
Sphene	0.1030 ±0.00	0.0347 ±0.03	0.1024 ± 3.79	0.1179 ± 4.02	0.1190 ± 2.97
Chalcedony	0.0281 ±0.00	0.055 ±0.01	0.1496 ± 4.12	0.1221 ± 4.02	0.1387 ± 4.01
Average	0.0470 ±0.00	0.0465 ±0.05	0.1099 ± 3.80	0.1104 ± 3.63	0.1107 ± 3.71

	SpACNN-LDVAE	RSNMF	GLNMF	$L_{1/2}$NMF	VCA+FCLS
Alunite	0.0683 ±0.00	0.1189 ± 4.39	0.1353 ± 3.83	0.1496 ± 3.32	0.1574 ± 3.71
Andradite	0.0462 ±0.00	-	-	-	-
Buddingtonite	0.0227 ±0.00	0.1342 ± 4.72	0.1437 ± 3.62	0.1441 ± 4.16	0.1412 ± 3.74
Dumortierite	0.0500 ±0.00	-	-	-	-
Kaolinite_1	0.0740 ±0.00	0.0955 ± 3.07	0.0967 ± 4.01	0.0825 ± 4.66	0.0736 ± 4.42
Kaolinite_2	0.0249 ±0.00	0.1396 ± 4.11	0.1356 ± 3.91	0.1402 ± 4.18	0.1420 ± 4.16
Muscovite	0.0320 ±0.00	0.0997 ± 3.46	0.0961 ± 3.77	0.0889 ± 3.03	0.1007 ± 3.31
Montmorillonite	0.0214 ±0.00	0.0744 ± 3.12	0.0838 ± 4.28	0.0876 ± 2.91	0.0974 ± 3.39
Nontronite	0.0639 ±0.00	0.0832 ± 4.18	0.0953 ± 3.41	0.1038 ± 4.46	0.0772 ± 2.10
Pyrope	0.0342 ±0.00	0.1469 ± 3.12	0.1318 ± 3.18	0.1123 ± 4.91	0.1437 ± 3.76
Sphene	0.1030 ±0.00	0.1134 ± 2.54	0.1291 ± 4.21	0.1252 ± 5.18	0.1277 ± 4.08
Chalcedony	0.0281 ±0.00	0.1224 ± 4.19	0.1341 ± 2.98	0.1520 ± 3.43	0.1514 ± 3.83
Average	0.0470 ±0.00	0.1128 ± 3.69	0.1182 ± 3.72	0.1186 ± 4.02	0.1212 ± 3.65

MLP-LDVAE: Mantripragada and Qureshi (2024); SSWNMF: Zhang et al. (2022); SGSNMF: Wang et al. (2017); TV-RSNMF: He et al. (2017); RSNMF: He et al. (2017); GLNMF: Lu et al. (2013); $L_{1/2}$NMF: Qian et al. (2011); VCA+FCLS: Nascimento and Dias (2005b)

Table 6.18: Endmember Extraction Results on OnTech-Syn-HSI-21 Dataset

SNR	**SpACNN-LDVAE**	MLP-LDVAE	SSWNMF	SGSNMF	TV-RSNMF
20 dB	0.0584 ± 0.00	0.0224 ± 0.01	0.0636 ± 0.40	0.0782 ± 0.50	0.0679 ± 0.30
30 dB	0.0616 ± 0.00	0.0138 ± 0.01	0.0122 ± 0.01	0.0176 ± 0.03	0.0131 ± 0.03
40 dB	0.0613 ± 0.00	0.0081 ± 0.00	0.0029 ± 0.02	0.0033 ± 0.03	0.0036 ± 0.02
50 dB	0.0545 ± 0.00	0.0082 ± 0.00	0.0012 ± 0.02	0.0019 ± 0.02	0.0014 ± 0.03
INF	0.0594 ± 0.00	0.0069 ± 0.00	-	-	-

SNR	**SpACNN-LDVAE**	RSNMF	GLNMF	$L_{1/2}$NMF	VCA+FCLS
20 dB	0.0584 ± 0.00	0.0731 ± 0.50	0.0724 ± 0.05	0.0744 ± 0.40	0.1358 ± 0.30
30 dB	0.0616 ± 0.00	0.0138 ± 0.05	0.0144 ± 0.04	0.0142 ± 0.04	0.0350 ± 0.06
40 dB	0.0613 ± 0.00	0.0041 ± 0.04	0.0044 ± 0.05	0.0037 ± 0.04	0.0125 ± 0.05
50 dB	0.0545 ± 0.00	0.0020 ± 0.04	0.0023 ± 0.04	0.0024 ± 0.03	0.0049 ± 0.06
INF	0.0594 ± 0.00	-	-	-	-

MLP-LDVAE: Mantripragada and Qureshi (2024); SSWNMF: Zhang et al. (2022); SGSNMF: Wang et al. (2017); TV-RSNMF: He et al. (2017); RSNMF: He et al. (2017); GLNMF: Lu et al. (2013); $L_{1/2}$NMF: Qian et al. (2011); VCA+FCLS: Nascimento and Dias (2005b)

Table 6.19: Abundance Estimation Results on OnTech-Syn-HSI-21 Dataset

SNR	**SpACNN-LDVAE**	MLP-LDVAE	SSWNMF	SGSNMF	TV-RSNMF
20 dB	0.0948 ± 0.00	0.0052 ± 0.00	0.1339 ± 0.20	0.1322 ± 0.40	0.1342 ± 0.30
30 dB	0.3356 ± 0.00	0.0302 ± 0.00	0.0386 ± 0.20	0.0391 ± 0.30	0.0420 ± 0.20
40 dB	0.3343 ± 0.00	0.0303 ± 0.00	0.0122 ± 0.03	0.0148 ± 0.05	0.0142 ± 0.04
50 dB	0.3335 ± 0.00	0.0303 ± 0.00	0.0041 ± 0.02	0.0059 ± 0.05	0.0050 ± 0.03
INF	0.0948 ± 0.00	0.0052 ± 0.00	-	-	-

SNR	**SpACNN-LDVAE**	RSNMF	GLNMF	$L_{1/2}$NMF	VCA+FCLS
20 dB	0.0948 ± 0.00	0.1426 ± 0.40	0.1434 ± 0.60	0.1430 ± 0.50	0.1704 ± 0.03
30 dB	0.3356 ± 0.00	0.0426 ± 0.30	0.0429 ± 0.03	0.0432 ± 0.20	0.0548 ± 0.20
40 dB	0.3343 ± 0.00	0.0147 ± 0.05	0.0150 ± 0.04	0.0153 ± 0.03	0.0164 ± 0.10
50 dB	0.3335 ± 0.00	0.0055 ± 0.03	0.0064 ± 0.04	0.0061 ± 0.04	0.0087 ± 0.08
INF	0.0948 ± 0.00	-	-	-	-

MLP-LDVAE: Mantripragada and Qureshi (2024); SSWNMF: Zhang et al. (2022); SGSNMF: Wang et al. (2017); TV-RSNMF: He et al. (2017), RSNMF: He et al. (2017); GLNMF: Lu et al. (2013); $L_{1/2}$NMF: Qian et al. (2011); VCA+FCLS: Nascimento and Dias (2005b).

Chapter 7

Conclusions

This thesis advanced machine learning and deep learning methodologies to overcome critical challenges in hyperspectral image (HSI) analysis, specifically addressing issues of high dimensionality, intrinsic data limitations, complex feature extraction, and the fundamental problem of pixel unmixing. The foundational journey commenced with a comprehensive exploration of established techniques for segmentation, classification, and dimensionality reduction, illuminating the persistent complexities and limitations encountered within traditional hyperspectral data processing paradigms.

7.1 Summary of Key Contributions

This research culminated in the conception and development of novel deep learning architectures, namely the Latent Dirichlet Variational Autoencoder (LDVAE) and its iterative extension, iLDVAE. These architectures stand as the primary intellectual contributions of this work, directly confronting the critical task of pixel unmixing within the HSI domain. By elegantly integrating the Dirichlet distribution to naturally model abundance vectors, both LDVAE and iLDVAE have demonstrated state-of-the-art performance in semi-supervised and unsupervised pixel unmixing tasks, remarkably achieving superior results even in the absence of explicit labeled training data. This capability for unsupervised learning represents a significant advancement over methods reliant on extensive ground truth.

Further pivotal advancements were realized through the incorporation of spatial information, acknowledging the importance of local contextual relationships in accurately discerning the composition of hyperspectral pixels. The proposed SpACNN-LDVAE architecture, featuring a convolutional neural network (CNN) encoder coupled with a spatial attention mechanism, demonstrated the profound effectiveness of seamlessly integrating spatial features. This integration led to improved accuracy in both endmember extraction and abundance estimation, proving that a holistic spectral-spatial perspective is indispensable for robust HSI unmixing.

The research presented in this thesis thus contributes significantly and broadly to the field of hyperspectral image analysis, offering:

- **Novel Deep Learning Architectures for Unmixing:** LDVAE and iLDVAE provide robust, efficient, through semi and unsupervised methods for pixel unmixing. By directly modeling abundance vectors with the Dirichlet distribution, these architectures overcome significant limitations of existing techniques, particularly their reliance on labeled data or restrictive assumptions about endmember variability. This inherent unsupervised capability is critical for real-world HSI applications where ground truth is scarce.

- **Integration of Spatial and Spectral Information for Enhanced Accuracy:** SpACNN-LDVAE stands as an evidence to the potentials of considering spatial context. Its architecture showcases how the integration of spatial features via CNNs and attention mechanisms can lead to improved accuracy in both identifying pure spectral signatures (endmembers) and quantifying their proportions (abundances), offering a more comprehensive understanding of HSI scenes than purely spectral approaches.

- **Addressing Data Scarcity through Transfer Learning and Synthetic Data:** The capability to leverage synthetic data for model training, particularly for the unsupervised LDVAE family, opens critical avenues for applications where acquiring real-world labeled data is prohibitively expensive, time-consuming, or practically impossible. This transfer learning paradigm broadens the applicability and scalability of the developed models.

- **Open-Source Contributions to the Research Community:** The development of versatile synthetic data generation tools and the introduction of a new, realistic synthetic

dataset represent tangible contributions to the broader hyperspectral research community. These resources foster reproducibility, facilitate benchmarking, and significantly lower the barrier to entry for new research in this complex and data-hungry domain.

7.2 Limitations of the Present Work

While the methodologies developed in this thesis represent important progress in addressing core challenges in HSI analysis, it is imperative to acknowledge their inherent limitations, which also serve as motivation for future research:

- **Reliance on Data Quantity for Model Performance:** The efficacy of the LDVAE, particularly when operating in a supervised or semi-supervised context for fine-tuning or specialized tasks, is inherently tied to the quantity and diversity of available training data. This dependency becomes an even more pronounced challenge when extending the LDVAE framework to more complex convolutional neural network (CNN) architectures (e.g., SpACNN-LDVAE), where the increased parameter count necessitates a larger volume of training samples to prevent overfitting and ensure robust generalization across varying scene complexities.

- **Challenges with Spatial Ground Truth for CNNs:** While this thesis introduces SpACNN-LDVAE to leverage spatial information, the development and training of deep learning models, particularly Convolutional Neural Networks (CNNs), for HSI analysis with spatial features face significant hurdles due to the scarcity of comprehensive spatial ground truth datasets. Accurately annotating spatial features like precise object boundaries, textures, and fine-grained sub-pixel structures in HSI is an exceptionally laborious and expensive process, often more so than acquiring spectral-only labels. This inherent limitation in data availability significantly constrained the initial scope and prevailing focus of this research on spectral-only features in the earlier LDVAE models, serving as a foundational step before venturing into the more complex, data-hungry spatial-spectral integration.

- **Purity Assumption in Iterative Unmixing (iLDVAE):** Although iLDVAE effectively addresses the critical problem of the lack of ground truth for endmember extraction through its iterative refinement, its success relies on the fundamental assumption that the hyperspectral scene contains at least some pixels with a high purity index. If the scene is entirely composed of deeply mixed pixels without sufficiently pure endmember representatives, the iterative process may struggle to accurately identify and extract the true constituent endmembers, thereby limiting its direct applicability in highly complex, densely mixed environments.

7.3 Future Work and Research Directions

Building upon the foundations established in this thesis, several promising avenues for future research emerge, aiming to further enhance the capabilities and applicability of deep learning in hyperspectral image analysis:

- **Exploration of Advanced Generative Architectures for Data Synthesis:** Investigate the potential of more sophisticated deep learning models, such as variational autoencoders with full CNN encoders and decoders (e.g., VAE-CNN), and Generative Adversarial Networks (GANs), to generate highly realistic synthetic HSI data. Such models could create synthetic scenes with greater spatial coherence of objects, materials, and pixels, thereby providing invaluable diverse training and pre-training data that realistically mimics complex real-world hyperspectral observations. This would directly address the data scarcity limitation for more complex models.

- **Refined Attention Mechanisms, including Spectral and Channel-wise Attention:** While spatial attention has been successfully incorporated, future work should delve into the integration of spectral or channel-wise attention mechanisms. Approaches like Transformers, specifically designed to capture long-range dependencies, could be adapted to model correlations across the spectral bands (channels). This would allow for a more adaptive and refined feature selection process, dynamically emphasizing the most relevant

spectral characteristics for unmixing or other analysis tasks, leading to potentially more robust and interpretable models.

- **Full End-to-End Deep Architectures for Unmixing:** Explore comprehensive end-to-end deep learning models where both the encoder and decoder components fully leverage convolutional neural networks. This could lead to more efficient and powerful feature representations, potentially improving unmixing performance by learning hierarchical spectral-spatial features directly from the raw HSI data without requiring explicit handcrafted feature engineering.

- **Theoretical Deepening and Algorithmic Optimization:** Conduct further rigorous investigation into the theoretical underpinnings of the LDVAE and its variants. A deeper theoretical understanding could lead to novel optimization strategies, improved regularization techniques, and more robust implementations, ultimately enhancing both performance and generalization capabilities across diverse HSI datasets and unmixing scenarios.

- **Diverse Real-World Applications and Generalizability Studies:** Apply the developed methodologies to a wider array of diverse real-world domains. This includes, but is not limited to, precision agriculture (e.g., crop health monitoring, disease detection), environmental monitoring (e.g., pollution assessment, land cover change), and mineral exploration (e.g., geological mapping, resource identification). Such extensive empirical validation will be crucial for demonstrating the generalizability, scalability, and practical impact of these advanced unmixing techniques.

In the ongoing pursuit of knowledge and innovation within hyperspectral image analysis, this thesis marks a significant step forward, providing robust tools and profound insights for researchers and practitioners alike. As these pioneering techniques are continuously refined, expanded, and applied to ever more challenging scenarios, the inherent potential for unlocking deeper, more nuanced understanding from hyperspectral imagery promises to fundamentally revolutionize our ability to analyze, interpret, and ultimately manage the dynamic changes occurring on our planet through advanced Earth Observation data.

Appendices

Appendix A

Derivation of ELBO function for Dirichlet Distributions

A.1 Evidence Lower Bound (ELBO)

This section details the derivation of the Evidence Lower Bound (ELBO), a fundamental objective in variational inference. The ELBO serves as a tractable lower bound on the marginal likelihood, which is subsequently maximized during model optimization.

This section details the fundamental derivation of the Evidence Lower Bound (ELBO) within the context of Variational Autoencoders (VAEs).

Problem Setup

- **Observed data: x**
- **Latent variable: z**
- **Generative model:** $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z})$

Marginal Likelihood and Intractability

The marginal likelihood of the observed data is given by:

$$p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})\,d\mathbf{z}$$

This integral is generally intractable, making direct optimization or inference challenging.

Introducing an Auxiliary Distribution

To address intractability, an auxiliary distribution $q(\mathbf{z}|\mathbf{x})$ (the encoder) is introduced. This distribution approximates the true posterior $p(\mathbf{z}|\mathbf{x})$.

Deriving the Log Marginal Likelihood

We begin by rewriting the log marginal likelihood:

$$\begin{aligned}
\log p(\mathbf{x}) &= \log \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})\,d\mathbf{z} \\
&= \log \int \frac{q(\mathbf{z}|\mathbf{x})}{q(\mathbf{z}|\mathbf{x})} p(\mathbf{x}|\mathbf{z})p(\mathbf{z})\,d\mathbf{z} \\
&= \log \int q(\mathbf{z}|\mathbf{x}) \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})}\,d\mathbf{z} \\
&= \log \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})}\right]
\end{aligned}$$

Applying Jensen's Inequality

Since the logarithm is a concave function, we can apply Jensen's Inequality:

$$\log \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})}\right] \geq \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\log \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})}\right]$$

Thus, the log marginal likelihood is bounded by:

$$\begin{aligned}\log p(\mathbf{x}) &\geq \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\log \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})}\right] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\log p(\mathbf{x}|\mathbf{z}) + \log p(\mathbf{z}) - \log q(\mathbf{z}|\mathbf{x})\right]\end{aligned}$$

This lower bound is known as the Evidence Lower Bound (ELBO), denoted as $ELBO(\mathbf{x})$.

ELBO Decomposition

The ELBO can be further decomposed:

$$\begin{aligned}ELBO(x) &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\log p(\mathbf{x}|\mathbf{z}) + \log p(\mathbf{z}) - \log q(\mathbf{z}|\mathbf{x})\right] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log p(\mathbf{x}|\mathbf{z})] + \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log p(\mathbf{z}) - \log q(\mathbf{z}|\mathbf{x})] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log p(\mathbf{x}|\mathbf{z})] - \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log q(\mathbf{z}|\mathbf{x}) - \log p(\mathbf{z})]\end{aligned}$$

The Kullback-Leibler (KL) divergence between $q(z|x)$ and $p(z)$ is defined as:

$$KL(q(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})) = \int q(\mathbf{z}|\mathbf{x}) \log \frac{q(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})}\, d\mathbf{z} = \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log q(\mathbf{z}|\mathbf{x}) - \log p(\mathbf{z})]$$

Substituting this into the ELBO expression:

$$ELBO(\mathbf{x}) = \underbrace{\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log p(\mathbf{x}|\mathbf{z})]}_{\text{Reconstruction term}} - \underbrace{\mathrm{KL}(q(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))}_{\text{Regularization term}}$$

The VAE objective is to maximize this ELBO, which implicitly maximizes the log marginal likelihood.

Reconstruction Term Analysis

The reconstruction term, $\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[\log p(\mathbf{x}|\mathbf{z})]$, represents how well the decoder can reconstruct the input $\mathbf{x}$ from the latent variable $\mathbf{z}$ sampled from the approximate posterior $q(\mathbf{z}|\mathbf{x})$.

Decoder Model Assumption

Assume a common decoder model where $p(\mathbf{x}|\mathbf{z})$ is a Gaussian distribution:

$$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}(\mathbf{z}), \sigma^2 I)$$

where:

- $\hat{\mathbf{x}}(\mathbf{z})$: The mean of the Gaussian, representing the decoder output (reconstruction).
- σ^2: A fixed variance.

Each pixel is modeled as an independent Gaussian.

Log-Likelihood for Gaussian Decoder

The log-likelihood for this Gaussian decoder is:

$$\log p(\mathbf{x}|\mathbf{z}) = -\frac{N}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\|\mathbf{x} - \hat{\mathbf{x}}(\mathbf{z})\|^2$$

Ignoring the constant term ($-\frac{N}{2}\log(2\pi\sigma^2)$), the log-likelihood is proportional to:

$$\log p(\mathbf{x}|\mathbf{z}) \propto -\frac{1}{2\sigma^2}\|\mathbf{x} - \hat{\mathbf{x}}(\mathbf{z})\|^2$$

Maximizing the reconstruction term is equivalent to minimizing the mean squared error between the input $\mathbf{x}$ and its reconstruction $\hat{\mathbf{x}}(\mathbf{z})$.

Dirichlet Distribution

The Dirichlet distribution is characterized by a concentration vector $\boldsymbol{\alpha}$.

1. **Sampling components:** Each component y_i is sampled from a Gamma distribution:

$$y_i \sim \text{Gamma}(\alpha_i, 1), \quad \text{for } i = 1, \ldots, k$$

The probability density function (PDF) of a Gamma distribution Gamma(α, β) is:

$$p(y) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} y^{\alpha-1} e^{-\beta y}$$

2. **Generating Dirichlet samples:** The Dirichlet distributed variable x_i is derived from these Gamma samples:

$$x_i = \frac{y_i}{\sum_{j=1}^{k} y_j}$$

A.2 Kullback-Leibler divergence for Dirichlet distribution

This section details the derivation of the Kullback-Leibler (KL) divergence for Dirichlet distributions. This specific form of KL divergence is essential for calculating the variational objective function within models that employ Dirichlet priors, such as the Latent Dirichlet Variational Autoencoder (LDVAE) discussed in this work.

We first need to rewrite Dir$(\mathbf{z}; \alpha)$ as a Multivariate Gamma distribution:

$$z_i \sim \text{MVGamma}(z_i; \alpha_i, \beta) = \frac{\beta^{\alpha_i} z^{\alpha_i - 1} e^{-\beta z_i}}{\Gamma(\alpha_i)}$$

The KL divergence between two Multivariate Gamma distributions can be derived as:

$$\begin{aligned} \text{KL}\left[q(\mathbf{z}, \mathbf{x}; \hat{\boldsymbol{\alpha}}) \| p(\boldsymbol{z}; \alpha)\right] = & \sum \log \Gamma(\alpha_k) \\ & - \sum \log \Gamma(\hat{\alpha}_k) \\ & + \sum (\hat{\alpha}_k - \alpha_k) \frac{d}{dx} \ln \Gamma(\hat{\alpha}_k) \end{aligned} \tag{A.1}$$

As demonstrated by Joo *et al.* (Joo et al., 2019), the proof starts with the derivative of a Gamma function:

$$\frac{d}{d\alpha}\frac{\Gamma(\alpha)}{\beta_\alpha} = \beta^{-\alpha}(\Gamma'(\alpha) - \Gamma(\alpha)\log\beta)\int_0^\infty x^{\alpha-1}e^{-\beta x}\log x dx \tag{A.2}$$

From the definition of KL divergence (Equation **??**), it follows that:

$$\begin{aligned}
\mathrm{KL}\left[Q\|P\right] &= \int_0^\infty \cdots \int_0^\infty \prod Gamma(\hat{\alpha},\beta)\log\frac{\lambda(\hat{\alpha})}{\lambda(\alpha)}d\mathbf{x} \\
&= \int_0^\infty \cdots \int_0^\infty \prod Gamma(\hat{\alpha},\beta)\left[\sum(\hat{\alpha}_k-\alpha_k)\log\beta + \sum\log\Gamma(\alpha_k) - \sum\log\Gamma(\hat{\alpha}_k) + \sum(\hat{\alpha}_k-\alpha_k)\log x_k\right]d\mathbf{x} \\
&= \left[\sum(\hat{\alpha}_k-\alpha_k)\log\beta + \sum\log\Gamma(\alpha_k) - \sum\log\Gamma(\hat{\alpha}_k)\right] + \int_0^\infty \cdots \int_0^\infty \frac{\beta^{\hat{\alpha}_k}}{\prod\Gamma(\hat{\alpha}_k)}e^{-\beta\sum x_k}\prod x_k^{\hat{\alpha}-1}(\sum(\hat{\alpha}_k-\alpha_k)\log x_k)d\mathbf{x} \\
&= \left[\sum(\hat{\alpha}_k-\alpha_k)\log\beta + \sum\log\Gamma(\alpha_k) - \sum\log\Gamma(\hat{\alpha}_k)\right] + \sum(\hat{\alpha}_k-\alpha_k)\beta^{\hat{\alpha}_k}\Gamma^{-1}(\hat{\alpha}_k)\beta^{-\hat{\alpha}_k}(\Gamma'(\hat{\alpha}_k)\log\beta) \\
&= \sum(\hat{\alpha}_k-\alpha_k)\log\beta + \sum\log\Gamma(\alpha_k) - \sum\log\Gamma(\hat{\alpha}_k) + \sum(\hat{\alpha}_k-\alpha_k)(\psi(\hat{\alpha}_k)-\log\beta) \\
&= \sum\log\Gamma(\alpha_k) - \sum\log\Gamma(\hat{\alpha}_k) + \sum(\hat{\alpha}_k-\alpha_k)\psi(\hat{\alpha}_k)
\end{aligned} \tag{A.3}$$

where $\psi(\hat{\alpha}_k)$ is the digamma function:

$$\psi(\hat{\alpha}_k) = \frac{d}{dx}\ln\Gamma(\hat{\alpha}_k) \tag{A.4}$$

For coding purposes, the digamma function can be approximated as follows (Lin, 2016; Spouge, 1994):

$$\psi(\hat{\alpha}_k) \approx \ln\hat{\alpha} - \frac{\hat{\alpha}}{2\hat{\alpha}} \tag{A.5}$$

Therefore, the Equation A.1 becomes:

$$\begin{aligned}
\mathrm{KL}\left[q(\mathbf{z},\mathbf{x};\hat{\alpha})\|p(\mathbf{z};\alpha)\right] = &\sum\log\Gamma(\alpha_k) \\
&- \sum\log\Gamma(\hat{\alpha}_k) \\
&+ \sum(\hat{\alpha}_k-\alpha_k)\ln\hat{\alpha} - \frac{\hat{\alpha}}{2\hat{\alpha}}
\end{aligned} \tag{A.6}$$

and lastly, replacing Equation A.6 into Equation **??**, our final loss function becomes:

$$\begin{aligned}\mathcal{L}(\mathbf{x};\theta,\phi) = \mathbb{E}_{q_\theta}\left[\log p_\phi(\mathbf{x}|\mathbf{z})\right] \\ - \sum \log\Gamma(\alpha_k) \\ - \sum \log\Gamma(\hat{\alpha}_k) \\ + \sum (\hat{\alpha}_k - \alpha_k)\ln\hat{\alpha} - \frac{\hat{\alpha}}{2\hat{\alpha}}\end{aligned}$$

where α is the concentration parameter of the Dirichlet prior, $\hat{\alpha}$ is the concentration parameters of the Dirichlet posterior.

Appendix B

Published Papers

B.1 Segmentation and Classification

In this research I proposed a new method for selecting the optimal segmentation scale for high-resolution hyperspectral images. The key novelties are:

- 1. Applying inverse noise weighting to normalize the spectral bands before segmentation, reducing the impact of noisy bands.
- 2. Detecting and removing outlier segments using the Isolation Forest algorithm before calculating segmentation quality metrics like the coefficient of variation (CV) and its rate of change (RoC).
- 3. Using the noise-normalized non-outlier RoC (NN-nRoC) of the CV as the criteria for optimal scale selection, instead of the original RoC which can be biased by noise and outliers.

The method was tested on three hyperspectral datasets (suburban, urban, forest) using k-means, mean-shift, and watershed segmentation algorithms. Results showed:

- Outlier segments existed across all scales, with more at finer segmentation scales. These outliers significantly impacted the CV and RoC curves.

- The NN-nRoC curves were more robust and reliable for optimal scale selection compared to the original RoC.

- Visual and quantitative evaluation showed the NN-nRoC method produced better segmentation results with lower over/under-segmentation errors across the three datasets and algorithms.

The proposed approach effectively handles noise and outliers in hyperspectral data, enabling more accurate optimal scale selection and improved segmentation performance. It can be applied to other image segmentation problems involving noisy data.

B.1.1 Contributions to the Segmentation Paper

Within the collaborative research presented in the segmentation paper, my contributions were instrumental in advancing the understanding of hyperspectral image segmentation and the impact of spectral band normalization. Specifically, my contributions encompassed the following key aspects:

- **Implementation of Inverse Noise Weighting:** I translated the theoretical concept of Inverse Noise Weighting for spectral band normalization into practical application by developing the code necessary for its implementation. This enabled the quantitative assessment of the technique's effectiveness in subsequent segmentation tasks.

- **Development of Segmentation Algorithms:** I played a crucial role in developing and implementing various segmentation algorithms, allowing for a comprehensive comparison and evaluation of their performance on hyperspectral data. This involved not only the selection of appropriate algorithms but also their adaptation and optimization for the specific characteristics of hyperspectral imagery.

- **Evaluation Methods:** To gauge the efficacy of different segmentation approaches, I spearheaded the development and implementation of robust evaluation metrics. This facilitated a quantitative and objective comparison of the algorithms, providing valuable insights into their strengths and limitations within the context of hyperspectral image analysis.

B.2 Spectral Dimensionality Reduction

In this investigation I presented a systematic study of the effects of hyperspectral pixel dimensionality reduction on the pixel classification task. I implemented five compression methods – Principal Component Analysis (PCA), Kernel PCA (KPCA), Independent Component Analysis (ICA), Autoencoder (AE), and Denoising Autoencoder (DAE) – to compress 634-dimensional hyperspectral pixel vectors. I used these compressed signals to perform pixel-level land cover classification on three hyperspectral image datasets representing urban, suburban, and forest landscapes.

The key findings are:

1. PCA, KPCA, and ICA achieve lower signal reconstruction errors but lower classification accuracy when compression rate exceeds 90

2. AE and DAE methods achieve better classification accuracy at around 90

3. Classification accuracy drops as compression rate increases for all methods, but nearly all methods achieve f1-scores greater than 0.89 even at 80

4. AE and DAE are well-suited for resource-constrained, in-situ applications where computational resources are limited.

The paper provides a systematic evaluation of the interplay between compression methods, compression rates, and the task of hyperspectral pixel classification. The re-

sults suggest both the compression method and rate are important considerations when designing a hyperspectral image analysis pipeline.

B.2.1 Contributions to the Dimensionality Reduction Paper

In the dimensionality reduction paper, I took the lead as the primary investigator, demonstrating my expertise in applying and evaluating various dimensionality reduction techniques within the context of hyperspectral image classification. My contributions were multifaceted and encompassed the following key areas:

- **Comparative Analysis of Dimensionality Reduction Methods:** I undertook a comprehensive investigation of diverse dimensionality reduction techniques, including both linear and non-linear approaches. This involved a thorough review of existing literature, selection of appropriate methods for comparison, and meticulous implementation of each technique.

- **Implementation and Evaluation:** I meticulously implemented each chosen dimensionality reduction method and conducted rigorous evaluations to assess their effectiveness in preserving essential information for hyperspectral image classification. This encompassed the development of appropriate evaluation metrics and a systematic comparison of the results obtained from each technique.

- **Focus on Autoencoders:** Recognizing the potential of deep learning for dimensionality reduction, I placed particular emphasis on exploring the capabilities of autoencoders. This involved delving into the theoretical underpinnings of autoencoders, implementing different autoencoder architectures, and evaluating their performance in comparison to other dimensionality reduction methods.

- **Research Dissemination:** As the lead author, I effectively communicated the research findings through a well-structured manuscript, culminating in the successful publication of the paper in PLOS One, a reputable scientific journal.

My comprehensive investigation and insightful analysis provided valuable insights into the strengths and limitations of various dimensionality reduction methods for hyperspectral image classification. This work not only contributed to the advancement of knowledge in the field but also laid the groundwork for my subsequent research focusing on deep learning-based approaches, specifically variational autoencoders, for hyperspectral image analysis.

B.3 Latent Dirichlet Variational Autoencoder

In this portion of my research, I introduced the Latent Dirichlet Variational Autoencoder (LDVAE), a method for hyperspectral pixel unmixing. The key assumptions are:

- Abundances can be encoded as Dirichlet distributions
- Endmember spectra can be represented as multivariate normal distributions

The LDVAE architecture has a Dirichlet bottleneck layer to model abundances, and the decoder performs endmember extraction. The method can leverage transfer learning, where the model is trained on synthetic data containing endmembers of interest, and then applied to real data containing a subset of those endmembers.

The model achieves state-of-the-art results on several benchmarks: Cuprite, Urban Hydice, and Samson datasets. It also introduces a new synthetic dataset called OnTech-HSI-Syn-21 for studying hyperspectral unmixing methods.

The proposed method can be applied in various domains like agriculture, forestry, mineralogy, and healthcare. A key advantage is that it does not require labeled data for training by leveraging the transfer learning paradigm with synthetic data.

B.3.1 Contributions to the LDVAE - Pixel Unmixing Paper

The LDVAE pixel unmixing paper represents a significant advancement in hyperspectral image analysis, and my contributions were central to its conception, development, and successful dissemination. My innovative approach and dedication led to the following key achievements:

- **Conceptualization of LDVAE:** I developed the idea of leveraging Variational Autoencoders (VAEs) for joint dimensionality reduction and pixel unmixing in hyperspectral imagery. This involved recognizing the potential of VAEs to learn low-dimensional representations while also incorporating the Dirichlet distribution within the latent space to effectively model abundance vectors.

- **Methodological Design and Implementation:** I designed the LDVAE architecture, integrating the Dirichlet distribution as a prior for the latent variables and developing the necessary algorithms for training the model using variational inference. This entailed a deep understanding of both deep learning principles and the mathematical underpinnings of the Dirichlet distribution.

- **Experimental Validation:** I conducted rigorous experiments to evaluate the performance of LDVAE on benchmark hyperspectral datasets, demonstrating its effectiveness in achieving accurate pixel unmixing while simultaneously learning a compressed representation of the spectral information. This involved the selection of appropriate datasets, evaluation metrics, and a comprehensive analysis of the results.

- **Collaboration and Dissemination:** My work attracted significant attention within the remote sensing community, leading to collaborations with researchers from the US Department of Agriculture (USDA) and Microsoft Research (MSR). I effectively communicated the research findings through conference presentations

and a well-structured manuscript, culminating in the publication of the LDVAE paper in the prestigious IEEE Transactions on Geoscience and Remote Sensing.

B.4 Iterative LDVAE

In this extension of my research, I presented an iterative method to address the lack of ground-truth data. I named the method Iterative LDVAE (iLDVAE) as it leverages the Latent Dirichlet Variational Autoencoder (LDVAE), for hyperspectral pixel unmixing, within an analysis-synthesis loop. The key advantages of iLDVAE are:

- It can perform pixel unmixing without requiring labeled training data, which is often costly or infeasible to obtain.

- It iteratively estimates the pure spectra (endmembers) of materials present in the image and their mixing ratios (abundances) in each pixel.

- It uses an LDVAE model within an analysis-synthesis loop, where in each iteration, the LDVAE is trained on a synthesized hyperspectral image with known abundances to estimate endmembers and abundances for the target image.

The proposed method was evaluated on three datasets: a synthetic OnTech-HSI-Syn-6em dataset, the HYDICE Urban benchmark, and a real-world dataset named "Cover Crop" from the USDA. The results demonstrate the effectiveness of iLDVAE in performing accurate pixel unmixing without labeled data, achieving high segmentation accuracy, low spectral angle distance for endmember extraction, and low RMSE for abundance estimation on the synthetic and benchmark datasets. On the USDA dataset, iLDVAE showed a high correlation ($R^2 = 0.7$) with ground truth vegetation cover percentage.

B.4.1 Contributions to the Iterative LDVAE Pixel Unmixing Paper

The iterative LDVAE (iLDVAE) pixel unmixing paper showcases my ability to address challenges posed by real-world datasets and further refine my previously established LDVAE methodology. My key contributions in this work include:

- **Addressing the Lack of Ground Truth:** I identified the limitations posed by the Cover Crop dataset, which lacked per-pixel ground truth labels, a crucial element for traditional supervised learning approaches. To overcome this obstacle, I devised an ingenious iterative approach that leverages the aggregate ground truth information available at the image level.

- **Development of the iLDVAE Algorithm:** Building upon the LDVAE framework, I meticulously designed the iLDVAE algorithm. This involved incorporating an iterative process where the model is initially trained with image-level labels and then refined by generating pseudo-labels at the pixel level based on the model's predictions. These pseudo-labels are subsequently used to further refine the model in subsequent iterations.

- **Experimental Validation and Analysis:** I evaluated the performance of iLDVAE on the Cover Crop dataset, demonstrating its ability to achieve accurate pixel unmixing results despite the absence of per-pixel ground truth labels. I analysis provided valuable insights into the convergence behavior of the iterative process and the impact of different initialization strategies.

- **Contribution to the Remote Sensing Community:** The iLDVAE method presents a valuable tool for researchers working with hyperspectral datasets that lack detailed ground truth information. By disseminating these findings through a publication in the IEEE International Geoscience and Remote Sensing Symposium,

I have made a significant contribution to the remote sensing community.

B.5 SpACNN-LDVAE - Integration of Spatial Soft Attention with LDVAE for Pixel Unmixing

SpACNN-LDVAE, another extension of LDVAE is an extension method which also aims to identify the materials (endmembers) and their proportions (abundances) within each pixel of a hyperspectral image. The key novelty of SpACNN-LDVAE is the incorporation of a **spatial** attention mechanism within the LDVAE framework.

The analysis of hyperspectral images, characterized by high spectral resolution but often limited spatial resolution, relies heavily on the process of pixel unmixing. This process aims to decompose mixed pixels into their constituent materials (endmembers) and their corresponding proportions (abundances). While existing methods, such as my previously proposed the LDVAE, have demonstrated effectiveness, they often neglect the spatial dependencies within the image. This paper introduces SpACNN-LDVAE, an extension of the LDVAE framework that explicitly incorporates local spatial context to enhance both endmember extraction and abundance estimation accuracy.

SpACNN-LDVAE employs a convolutional neural network (CNN) with a spatial attention mechanism as its encoder. This architectural choice allows the model to capture spatial relationships between neighboring pixels, leading to a more informative latent representation of the target pixel. By integrating this spatial information, SpACNN-LDVAE addresses a key limitation of the original LDVAE, which relies solely on spectral features.

Similar to LDVAE, SpACNN-LDVAE assumes a Dirichlet distribution for representing abundances and a multivariate Normal distribution for modeling endmember spectra. The decoder, responsible for reconstructing the spectrum of the central pixel and facilitating endmember extraction, remains largely consistent with the LDVAE architecture.

The efficacy of SpACNN-LDVAE was rigorously evaluated on four distinct hyperspectral image datasets: Samson, HYDICE Urban, Cuprite, and OnTech-HSI-Syn-21. Utilizing Root Mean Square Error (RMSE) as a metric for abundance estimation accuracy and Spectral Angle Distance (SAD) for endmember extraction accuracy, the results demonstrate the superiority of SpACNN-LDVAE over the baseline LDVAE method in most cases. Furthermore, the model's capability to leverage transfer learning was confirmed by training on synthetic data and achieving commendable performance on real-world data, highlighting its potential for scenarios where labeled data is scarce.

In conclusion, SpACNN-LDVAE presents a significant advancement in the field of hyperspectral pixel unmixing. By effectively integrating spatial context into the unmixing process, the model achieves superior accuracy in both endmember extraction and abundance estimation tasks, offering a valuable tool for the analysis and interpretation of hyperspectral imagery.

B.5.1 Contributions to the paper SpACNN-LDVAE

The spatial attention with LDVAE paper represents a further exploration of enhancing the LDVAE framework by incorporating spatial context into the analysis. My contributions in this collaborative work demonstrate a continued commitment to refining and expanding the capabilities of LDVAE:

- **Recognizing the Importance of Spatial Context:** I identified the potential for improving the performance of LDVAE by incorporating spatial information alongside spectral features. This recognition stemmed from the understanding that neighboring pixels in hyperspectral images often exhibit similar spectral characteristics and that capturing these spatial relationships could provide valuable context for pixel unmixing.

- **Collaboration and Guidance:** I played a key role in establishing a collabora-

tion with a student researcher interested in deep learning and hyperspectral image analysis. As a mentor and guide, I provided expert knowledge and supervision throughout the research process, ensuring the successful implementation and evaluation of the proposed method.

- **Evaluation and Analysis of Results:** I actively participated in the evaluation of the spatial attention-based LDVAE model, analyzing its performance in comparison to the original LDVAE and other baseline methods. My insights contributed to a deeper understanding of the impact of spatial attention mechanisms on pixel unmixing accuracy and the overall effectiveness of the proposed approach.

- **Dissemination of Research Findings:** The collaborative effort resulted in a paper accepted for publication in the IEEE International Geoscience and Remote Sensing Symposium. My involvement in the research and manuscript preparation ensured the clear communication of the methodology and its potential impact on the field of hyperspectral image analysis.

Bibliography

Abdi, H. (2010). Coefficient of variation. *Encyclopedia of research design*, 1(5):169–171.

Alaghbari, K. A., Lim, H.-S., Saad, M. H. M., and Yong, Y. S. (2023). Deep autoencoder-based integrated model for anomaly detection and efficient feature extraction in iot networks. *IoT*, 4(3):345–365.

Amigo, J. M., Babamoradi, H., and Elcoroaristizabal, S. (2015). Hyperspectral image analysis. A tutorial. *Analytica Chimica Acta*, 896:34–51.

Aroma, R. J. and Raimond, K. (2020). A wavelet transform applied spectral index for effective water body extraction from moderate-resolution satellite images. In *Artificial Intelligence Techniques for Satellite Image Analysis*, pages 255–274. Springer.

Arthur, D. and Vassilvitskii, S. (2006). k-means++: The advantages of careful seeding. Technical report, Stanford.

Azzolini, M., Ridzel, O. Y., Kaplya, P. S., Afanas'Ev, V., Pugno, N. M., Taioli, S., and Dapor, M. (2020). A comparison between monte carlo method and the numerical solution of the ambartsumian-chandrasekhar equations to unravel the dielectric response of metals. *Computational Materials Science*, 173:109420.

Baatz, M. and Schäpe, A. (2000). Multiresolution Segmentation: An optimization approach for high quality multi-scale image segmentation. *Angewandte geographische informationsverarbeitung*, page 12.

Ball, J. E. and Wei, P. (2018). Deep learning hyperspectral image classification using multiple class-based denoising autoencoders, mixed pixel training augmentation, and morphological operations. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 6903–6906. IEEE.

Bellens, R., Gautama, S., Martinez-Fonte, L., Philips, W., Chan, J. C.-W., and Canters, F. (2008). Improved classification of vhr images of urban areas using directional morphological profiles. *IEEE Transactions on Geoscience and Remote Sensing*, 46(10):2803–2813.

Belwalkar, A., Nath, A., and Dikshit, O. (2018). SPECTRAL-SPATIAL CLASSIFICATION OF HYPERSPECTRAL REMOTE SENSING IMAGES USING VARIATIONAL AUTOENCODER AND CONVOLUTION NEURAL NETWORK. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-5:613–620.

Benediktsson, J. A., Chanussot, J., and Moon, W. M. (2012). Very High-Resolution Remote Sensing: Challenges and Opportunities [Point of View]. *Proceedings of the IEEE*, 100(6):1907–1910.

Beucher, S. (1979). Use of watersheds in contour detection. In *Proc. Int. Workshop on Image Processing, Sept. 1979*, pages 17–21.

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. (2019). Pyro: Deep universal probabilistic programming. *Journal of machine learning research*, 20(28):1–6.

Bioucas-Dias, J. M., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P., and Chanussot, J. (2012). Hyperspectral Unmixing Overview: Geometrical, Statistical, and Sparse Regression-Based Approaches.

Blaschke, T., editor (2008). *Object-Based Image Analysis: Spatial Concepts for Knowledge-Driven Remote Sensing Applications*. Lecture Notes in Geoinformation and Cartography. Springer, Berlin Heidelberg.

Blaschke, T. (2010). Object based image analysis for remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 65(1):2–16.

Blei, D., Ng, A., and Jordan, M. (2001). Latent dirichlet allocation. *Advances in neural information processing systems*, 14.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of machine Learning research*, page 30.

Borsoi, R. A., Imbiriba, T., and Bermudez, J. C. M. (2020). Deep Generative Endmember Modeling: An Application to Unsupervised Spectral Unmixing. *IEEE Transactions on Computational Imaging*, 6:374–384.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Cao, L., Chua, K., Chong, W., Lee, H., and Gu, Q. (2003). A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine. *Neurocomputing*, 55(1-2):321–336.

Chakravarty, S., Paikaray, B. K., Mishra, R., and Dash, S. (2021). Hyperspectral image classification using spectral angle mapper. In *2021 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE)*, pages 87–90. IEEE.

Chen, S., Yang, X., You, Z., and Wang, M. (2016). Innovation of aggregate angularity characterization using gradient approach based upon the traditional and modified sobel operation. *Construction and Building Materials*, 120:442–449.

Cheriyadat, A. and Bruce, L. (2003). Why principal component analysis is not an appropriate feature extraction method for hyperspectral data. In *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No.03CH37477)*, volume 6, pages 3420–3422, Toulouse, France. IEEE.

Chicco, D., Warrens, M. J., and Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*.

Chitnis, S., Mantripragada, K., and Qureshi, F. Z. (2024). Spacnn-ldvae: Spatial attention convolutional latent dirichlet variational autoencoder for hyperspectral pixel unmixing. In *IGARSS 2024-2024 IEEE International Geoscience and Remote Sensing Symposium*, pages 7714–7719. IEEE.

Chung, Y. S., Choi, S. C., Silva, R. R., Kang, J. W., Eom, J. H., and Kim, C. (2017). Case study: Estimation of sorghum biomass using digital image analysis with canopeo. *Biomass and Bioenergy*.

Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):603–619.

Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, 36(3):287–314.

Crane, M., Clayton, T., Raabe, E., Stoker, J., Handley, L., Bawden, G., Morgan, K., and Queija, V. (2004). Report of the U.S. Geological Survey Lidar Workshop Sponsored by the Land Remote Sensing Program and held in St. Petersburg, FL, November 2002. Technical report, US Geological Survey.

Cuprite (2024). Cuprite dataset. Technical report.

Dao, P. D., He, Y., and Lu, B. (2019a). Maximizing the quantitative utility of airborne hyperspectral imagery for studying plant physiology: An optimal sensor exposure setting procedure and empirical line method for atmospheric correction. *International Journal of Applied Earth Observation and Geoinformation*, 77:140–150.

Dao, P. D. and Liou, Y.-A. (2015). Object-based flood mapping and affected rice field estimation with landsat 8 oli and modis data. *Remote Sensing*, 7(5):5077–5097.

Dao, P. D., Mantripragada, K., He, Y., and Qureshi, F. Z. (2021). Improving hyperspectral image segmentation by applying inverse noise weighting and outlier removal for optimal scale selection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 171:348–366.

Dao, P. D., Mong, N. T., and Chan, H.-P. (2019b). Landsat-modis image fusion and object-based image analysis for observing flood inundation in a heterogeneous vegetated scene. *GIScience & remote sensing*, 56(8):1148–1169.

Data, M. N. (2019). Electromagnetic Spectrum Diagram. https://mynasadata.larc.nasa.gov/basic-page/electromagnetic-spectrum-diagram.

Datta, A., Ghosh, S., and Ghosh, A. (2018). Pca, kernel pca and dimensionality reduction in hyperspectral images. In *Advances in Principal Component Analysis*, pages 19–46. Springer.

Deborah, H., Richard, N., and Hardeberg, J. Y. (2015). A Comprehensive Evaluation of Spectral Distance Functions and Metrics for Hyperspectral Image Processing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):3224–3234.

Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.-E., Lomeli, M., Hosseini, L., and Jégou, H. (2024). The faiss library. *arXiv preprint arXiv:2401.08281*.

Drăguţ, L., Csillik, O., Eisank, C., and Tiede, D. (2014). Automated parameterisation for multi-scale image segmentation on multiple layers. *ISPRS Journal of photogrammetry and Remote Sensing*, 88:119–127.

Drumetz, L., Chanussot, J., and Jutten, C. (2020). Spectral Unmixing: A Derivation of the Extended Linear Mixing Model from the Hapke Model. *IEEE Geoscience and Remote Sensing Letters*, 17(11):1866–1870.

Drumetz, L., Meyer, T. R., Chanussot, J., Bertozzi, A. L., and Jutten, C. (2019). Hyperspectral Image Unmixing with Endmember Bundles and Group Sparsity Inducing Mixed Norms. *IEEE Transactions on Image Processing*, 28(7):3435–3450.

Drumetz, L., Veganzones, M.-A., Henrot, S., Phlypo, R., Chanussot, J., and Jutten, C. (2016a). Blind Hyperspectral Unmixing Using an Extended Linear Mixing Model to Address Spectral Variability. *IEEE Transactions on Image Processing*, 25(8):3890–3905.

Drumetz, L., Veganzones, M.-A., Henrot, S., Phlypo, R., Chanussot, J., and Jutten, C. (2016b). Blind hyperspectral unmixing using an extended linear mixing model to address spectral variability. *IEEE Transactions on Image Processing*, 25(8):3890–3905.

Drăguţ, L., Tiede, D., and Levick, S. R. (2010). Esp: a tool to estimate scale parameter for multiresolution image segmentation of remotely sensed data. *International Journal of Geographical Information Science*, 24(6):859–871.

Du, H., Qi, H., Wang, X., Ramanath, R., and Snyder, W. E. (2003). Band selection using independent component analysis for hyperspectral image processing. In *32nd Applied Imagery Pattern Recognition Workshop, 2003. Proceedings.*, pages 93–98. IEEE.

Eismann, M. T. (2012). *Hyperspectral Remote Sensing.* SPIE Press, Bellingham, Wash.

Epshtein, B., Ofek, E., and Wexler, Y. (2010). Detecting text in natural scenes with stroke width transform. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 2963–2970. IEEE.

Fox, C. W. and Roberts, S. J. (2012). A tutorial on variational bayesian inference. *Artificial intelligence review*, 38(2):85–95.

Fukunaga, K. and Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, 21(1):32–40.

Gao, Q., Lim, S., and Jia, X. (2018). Hyperspectral image classification using convolutional neural networks and multiple feature learning. *Remote Sensing*, 10(2):299.

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.* O'Reilly Media.

Ghamisi, P. (2017). Advances in Hyperspectral Image and Signal Processing - A comprehensive overview of the state of the art. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):C2–C2.

Gondara, L. and Wang, K. (2018). Mida: Multiple imputation using denoising autoencoders. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 260–272. Springer.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.

Günter, S., Schraudolph, N. N., and Vishwanathan, S. (2007). Fast iterative kernel principal component analysis. *Journal of Machine Learning Research*, 8(Aug):1893–1918.

Guo, B., Gunn, S., Damper, R., and Nelson, J. (2006). Band Selection for Hyperspectral Image Classification Using Mutual Information. *IEEE Geoscience and Remote Sensing Letters*, 3(4):522–526.

Guo, X., Liu, X., Zhu, E., and Yin, J. (2017). Deep clustering with convolutional autoencoders. In *Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14-18, 2017, Proceedings, Part II 24*, pages 373–382. Springer.

Gupta, S. and Mazumdar, S. G. (2013). Sobel edge detection algorithm. *International journal of computer science and management Research*, 2(2):1578–1583.

Hapke, B. (2012). *Theory of Reflectance and Emittance Spectroscopy*. Cambridge University Press, Cambridge, UK ; New York, 2nd ed edition.

He, C., Li, S., Xiong, D., Fang, P., and Liao, M. (2020). Remote sensing image semantic segmentation based on edge information guidance. *Remote Sensing*, 12(9):1501.

He, W., Zhang, H., and Zhang, L. (2017). Total Variation Regularized Reweighted Sparse Nonnegative Matrix Factorization for Hyperspectral Unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3909–3921.

Heylen, R., Parente, M., and Gader, P. (2014). A Review of Nonlinear Hyperspectral Unmixing Methods. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6):1844–1868.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–507.

Hossain, M. D. and Chen, D. (2019). Segmentation for Object-Based Image Analysis (OBIA): A review of algorithms and challenges from remote sensing perspective. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150:115–134.

HYDICE (2024). Hydice urban hsi dataset. Technical report.

Hyperspectral Imaging Solutions, R. (2023). Hyperspectral Sensors — Airborne Remote Systems — Resonon. https://resonon.com/hyperspectral-airborne-remote-sensing-system.

Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430.

Ibarrola-Ulzurrun, E., Drumetz, L., Marcello, J., Gonzalo-Martin, C., and Chanussot, J. (2019). Hyperspectral Classification Through Unmixing Abundance Maps Addressing Spectral Variability. *IEEE Transactions on Geoscience and Remote Sensing*, 57(7):4775–4788.

ICSynthesis (2024). Hyperspectral imagery synthesis (EIAs) toolbox. Technical report, Grupo de Inteligencia Computacional, Universidad del País Vasco / Euskal Herriko Unibertsitatea (UPV/EHU), Spain.

Imbiriba, T., Borsoi, R. A., and Moreira Bermudez, J. C. (2018). Generalized Linear Mixing Model Accounting for Endmember Variability. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1862–1866, Calgary, AB. IEEE.

Janiczek, J., Thaker, P., Dasarathy, G., Edwards, C. S., Christensen, P., and Jayasuriya, S. (2020). Differentiable Programming for Hyperspectral Unmixing using a Physics-based Dispersion Model.

Jin, B., Zhu, Y., Huang, W., Chen, Q., and Li, S. (2022). An efficient attention-based convolutional neural network that reduces the effects of spectral variability for hyperspectral unmixing. *Applied Sciences*, 12(23):12158.

Johnson, J., Douze, M., and Jégou, H. (2019). Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Jolliffe, I. (2011). *Principal Component Analysis*, pages 1094–1096. Springer Berlin Heidelberg, Berlin, Heidelberg.

Joo, W., Lee, W., Park, S., and Moon, I.-C. (2019). Dirichlet Variational Autoencoder.

Joo, W., Lee, W., Park, S., and Moon, I.-C. (2020). Dirichlet variational autoencoder. *Pattern Recognition*, 107:107514.

Karl, J. W. and Maurer, B. A. (2010). Spatial dependence of predictions from image segmentation: A variogram-based method to determine appropriate scales for producing land-management information. *Ecological Informatics*, 5(3):194–202.

Khan, M. J., Khan, H. S., Yousaf, A., Khurshid, K., and Abbas, A. (2018). Modern Trends in Hyperspectral Image Analysis: A Review. *IEEE Access*, 6:14118–14129.

Kim, H. and Kim, H. (2023). Contextual anomaly detection for high-dimensional data using dirichlet process variational autoencoder. *IISE Transactions*, 55(5):433–444.

Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization. *(No Title)*.

Kingma, D. P. and Welling, M. (2014). Auto-Encoding Variational Bayes.

Kozintsev, B. (1999). Computations with gaussian random fields. *research directed by Dept. of Mathematics.University of Maryland, College Park*.

Kruse, F. A., Lefkoff, A., Boardman, y. J., Heidebrecht, K., Shapiro, A., Barloon, P., and Goetz, A. (1993). The spectral image processing system (sips)—interactive visualization and analysis of imaging spectrometer data. *Remote sensing of environment*, 44(2-3):145–163.

Kurnaz, M. N., Dokur, Z., and Ölmez, T. (2005). Segmentation of remote-sensing images by incremental neural network. *Pattern recognition letters*, 26(8):1096–1104.

Kussul, N., Lavreniuk, M., Skakun, S., and Shelestov, A. (2017). Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data. *IEEE Geoscience and Remote Sensing Letters*, 14(5):778–782.

Landgrebe, D. (2002). Hyperspectral image data analysis. *IEEE Signal Processing Magazine*, 19(1):17–28.

Längkvist, M., Kiselev, A., Alirezaie, M., and Loutfi, A. (2016). Classification and segmentation of satellite orthoimagery using convolutional neural networks. *Remote Sensing*, 8(4):329.

Li, F., Zhang, S., Liang, B., Deng, C., Xu, C., and Wang, S. (2021). Hyperspectral Sparse Unmixing With Spectral-Spatial Low-Rank Constraint. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:6119–6130.

Li, J., Yu, T., Li, J., Zhang, H., Zhao, K., Rong, Y. U., Cheng, H., and Huang, J. (2020). Dirichlet Graph Variational Autoencoder.

Liao, W., Pizurica, A., Philips, W., and Pi, Y. (2010). A fast iterative kernel pca feature extraction for hyperspectral images. In *2010 IEEE International Conference on Image Processing*, pages 1317–1320. IEEE.

Licciardi, G. and Chanussot, J. (2018). Spectral transformation based on nonlinear principal component analysis for dimensionality reduction of hyperspectral images. *European Journal of Remote Sensing*, 51(1):375–390.

Lin, J. (2016). *On The Dirichlet Distribution*. PhD thesis, Department of Mathematics and Statistics, Queens University.

Liu, F. T., Ting, K. M., and Zhou, Z.-H. (2012). Isolation-Based Anomaly Detection. *ACM Transactions on Knowledge Discovery from Data*, 6(1):1–39.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.

Lu, B., Dao, P. D., Liu, J., He, Y., and Shang, J. (2020). Recent advances of hyperspectral imaging technology and applications in agriculture. *Remote Sensing*, 12(16):2659.

Lu, X., Wu, H., Yuan, Y., Yan, P., and Li, X. (2013). Manifold Regularized Sparse NMF for Hyperspectral Unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 51(5):2815–2826.

Ma, Y., Wu, H., Wang, L., Huang, B., Ranjan, R., Zomaya, A., and Jie, W. (2015). Remote sensing big data computing: Challenges and opportunities. *Future Generation Computer Systems*, 51:47–60.

Maggiori, E., Plaza, A., and Tarabalka, Y. (2018). Models for hyperspectral image analysis: From unmixing to object-based classification. In Moser, G. and Zerubia, J., editors, *Mathematical Models for Remote Sensing Image Processing: Models and Methods for the Analysis of 2D Satellite and Aerial Images*, pages 37–80. Springer International Publishing, Cham.

Mahabir, R., Croitoru, A., Crooks, A., Agouris, P., and Stefanidis, A. (2018). A Critical Review of High and Very High-Resolution Remote Sensing Approaches for Detecting and Mapping Slums: Trends, Challenges and Emerging Opportunities. *Urban Science*, 2(1):8.

Mantripragada, K., Adler, P. R., Olsen, P. A., and Qureshi, F. Z. (2023). An iterative method for hyperspectral pixel unmixing leveraging latent dirichlet variational autoencoder. In *IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2023, Pasadena, CA, USA, July 16-21, 2023*, pages 7527–7530. IEEE.

Mantripragada, K., Dao, P. D., He, Y., and Qureshi, F. Z. (2022). The effects of spectral dimensionality reduction on hyperspectral pixel classification: A case study. *PLOS ONE*, 17(7):1–24.

Mantripragada, K., Qureshi, F., and Chitnis, S. (2021). Hyperspectral data loader. https://github.com/kiranmantri/hyperspectral-data-loader.

Mantripragada, K. and Qureshi, F. Z. (2024). Hyperspectral pixel unmixing with latent dirichlet variational autoencoder. *IEEE Trans. Geosci. Remote. Sens.*, 62:1–12.

Martin, D. R., Fowlkes, C. C., and Malik, J. (2004). Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE transactions on pattern analysis and machine intelligence*, 26(5):530–549.

Ming, D., Li, J., Wang, J., and Zhang, M. (2015). Scale parameter selection by spatial statistics for geobia: Using mean-shift based multi-scale segmentation as an example. *ISPRS Journal of Photogrammetry and Remote Sensing*, 106:28–41.

Mitra, P., Shankar, B. U., and Pal, S. K. (2004). Segmentation of multispectral remote sensing images using active support vector machines. *Pattern recognition letters*, 25(9):1067–1074.

Moser, G. and Zerubia, J., editors (2018). *Mathematical Models for Remote Sensing Image Processing*. Signals and Communication Technology. Springer International Publishing, Cham.

Mou, L., Ghamisi, P., and Zhu, X. X. (2017). Deep Recurrent Neural Networks for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3639–3655.

Mou, L., Ghamisi, P., and Zhu, X. X. (2018). Unsupervised Spectral-Spatial Feature Learning via Deep Residual Conv-Deconv Network for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 56(1):391–406.

Myint, S. W., Gober, P., Brazel, A., Grossman-Clarke, S., and Weng, Q. (2011). Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. *Remote Sensing of Environment*, 115(5):1145–1161.

Mylonas, S. K., Stavrakoudis, D. G., Theocharis, J. B., and Mastorocostas, P. A. (2015). A region-based genesis segmentation algorithm for the classification of remotely sensed images. *Remote Sensing*, 7(3):2474–2508.

Nagirner, D. I. and Ivanov, V. V. (2020). Chandrasekhar's h-function revisited. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 246:106914.

Nalepa, J., Myller, M., Imai, Y., Honda, K., Takeda, T., and Antoniak, M. (2019). Unsupervised Segmentation of Hyperspectral Images Using 3-D Convolutional Autoencoders. *IEEE Geoscience and Remote Sensing Letters*, pages 1–5.

NASA, G. S. F. C. (2023). Electromagnetic Spectrum - Introduction. https://imagine.gsfc.nasa.gov/science/toolbox/emspectrum1.html.

Nascimento, J. and Dias, J. (2005a). Does independent component analysis play a role in unmixing hyperspectral data? *IEEE Transactions on Geoscience and Remote Sensing*, 43(1):175–187.

Nascimento, J. and Dias, J. (2005b). Vertex component analysis: A fast algorithm to unmix hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 43(4):898–910.

Neubert, P. and Protzel, P. (2014). Compact watershed and preemptive slic: On improving trade-offs of superpixel segmentation algorithms. In *2014 22nd international conference on pattern recognition*, pages 996–1001. IEEE.

Palsson, B., Sigurdsson, J., Sveinsson, J. R., and Ulfarsson, M. O. (2018). Hyperspectral Unmixing Using a Neural Network Autoencoder. *IEEE Access*, 6:25646–25656.

Palsson, B., Ulfarsson, M. O., and Sveinsson, J. R. (2020). Convolutional autoencoder for spectral–spatial hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 59(1):535–549.

Palsson, B., Ulfarsson, M. O., and Sveinsson, J. R. (2022). Synthetic hyperspectral images with controllable spectral variability and ground truth. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5.

Park, S., Gil, M.-S., Im, H., and Moon, Y.-S. (2019). Measurement noise recommendation for efficient kalman filtering over a large amount of sensor data. *Sensors*, 19(5):1168.

Patrignani, A. and Ochsner, T. E. (2015). Canopeo: A powerful new tool for measuring fractional green canopy cover. *Agronomy Journal*.

Pearson, K. (1901). Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.

Pinheiro Cinelli, L., Araujo Marins, M., Barros da Silva, E. A., and Lima Netto, S. (2021). *Variational Autoencoder*, pages 111–149. Springer International Publishing, Cham.

Qian, Y., Jia, S., Zhou, J., and Robles-Kelly, A. (2011). Hyperspectral Unmixing via $L_{1/2}$ Sparsity-Constrained Nonnegative Matrix Factorization. *IEEE Transactions on Geoscience and Remote Sensing*, 49(11):4282–4297.

Rashmi S, Swapna Addamani, V. S. and S, R. (2014). Spectral angle mapper algorithm for remote sensing image classification.

Rasti, B., Scheunders, P., Ghamisi, P., Licciardi, G., and Chanussot, J. (2018). Noise Reduction in Hyperspectral Imagery: Overview and Application. *Remote Sensing*, 10(3):482.

Richards, J. A. and Jia, X. (2006). *Remote Sensing Digital Image Analysis: An Introduction*. Springer, Berlin, 4th ed edition.

Ruffin, C. and King, R. L. (1999). The analysis of hyperspectral data using savitzky-golay filtering-theoretical basis. 1. In *IEEE 1999 International Geoscience and Remote Sensing Symposium. IGARSS'99 (Cat. No. 99CH36293)*, volume 2, pages 756–758. IEEE.

Samson (2024). Samson hsi dataset. Technical report.

Shahid, K. T. and Schizas, I. D. (2021). Unsupervised hyperspectral unmixing via nonlinear autoencoders. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13.

Sorensen, A. T. (2000). Equilibrium price dispersion in retail markets for prescription drugs. *Journal of Political Economy*, 108(4):833–850.

Spouge, J. L. (1994). Computation of the Gamma, Digamma, and Trigamma Functions. *SIAM Journal on Numerical Analysis*, 31(3):931–944.

Stone, J. V. (2004). *Independent component analysis: a tutorial introduction*. MIT press.

Su, Y., Li, J., Plaza, A., Marinoni, A., Gamba, P., and Chakravortty, S. (2019). DAEN: Deep Autoencoder Networks for Hyperspectral Unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 57(7):4309–4321.

Sun, L. and Lucey, P. G. (2021). Unmixing Mineral Abundance and Mg# With Radiative Transfer Theory: Modeling and Applications. *Journal of Geophysical Research: Planets*, 126(2).

Sun, W. and Du, Q. (2019). Hyperspectral Band Selection: A Review. *IEEE Geoscience and Remote Sensing Magazine*, 7(2):118–139.

Tarabalka, Y., Chanussot, J., and Benediktsson, J. (2010). Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognition*, 43(7):2367–2379.

Thouvenin, P.-A., Dobigeon, N., and Tourneret, J.-Y. (2016). Hyperspectral Unmixing With Spectral Variability Using a Perturbed Linear Mixing Model. *IEEE Transactions on Signal Processing*, 64(2):525–538.

U. S. Geological Survey, Kokaly, R. F., Clark, R. N., Swayze, G. A., Livo, K. E., Hoefen, T. M., Pearson, N. C., Wise, R. A., Benzel, W. M., Lowers, H. A., Driscoll, R. L., and Klein, A. J. (2017). Usgs spectral library version 7. Technical report, U. S. Geological Survey, Reston, VA.

Vaiphasa, C. (2006). Consideration of smoothing techniques for hyperspectral remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 60(2):91–99.

Vasilev, I., Slater, D., Spacagna, G., Roelants, P., and Zocca, V. (2019). *Python Deep Learning: Exploring Deep Learning Techniques and Neural Network Architectures with PyTorch, Keras, and TensorFlow*. Packt Publishing Limited, Birmingham Mumbai, second edition edition.

Veganzones, M., Drumetz, L., Tochon, G., Dalla Mura, M., Plaza, A., Bioucas-Dias, J., and Chanussot, J. (2014). A new extended linear mixing model to address spectral variability.

In *2014 6th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–4.

Vidal, M. and Amigo, J. M. (2012). Pre-processing of hyperspectral images. Essential steps before image analysis. *Chemometrics and Intelligent Laboratory Systems*, 117:138–148.

Vincent, L. and Soille, P. (1991). Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 13(06):583–598.

Wang, X., Zhong, Y., Zhang, L., and Xu, Y. (2017). Spatial Group Sparsity Regularized Nonnegative Matrix Factorization for Hyperspectral Unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 55(11):6287–6304.

Wang, Y., Yao, H., and Zhao, S. (2016). Auto-encoder based dimensionality reduction. *Neurocomputing*, 184:232–242.

Wang, Z., Song, C., Wu, Z., and Chen, X. (2005). Improved watershed segmentation algorithm for high resolution remote sensing images using texture. In *Proceedings. 2005 IEEE International Geoscience and Remote Sensing Symposium, 2005. IGARSS'05.*, volume 5, pages 3721–3723. IEEE.

Weber, E. U., Shafir, S., and Blais, A.-R. (2004). Predicting risk sensitivity in humans and lower animals: risk as variance or coefficient of variation. *Psychological review*, 111(2):430.

Winter, M. E. (1999). N-FINDR: An algorithm for fast autonomous spectral end-member determination in hyperspectral data. In Descour, M. R. and Shen, S. S., editors, *SPIE's International Symposium on Optical Science, Engineering, and Instrumentation*, pages 266–275, Denver, CO.

Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). Cbam: Convolutional block attention module.

Xu, K., Fan, W., and Liu, X. (2023). Unsupervised disentanglement learning via dirichlet variational autoencoder. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 341–352. Springer.

Yang, J., He, Y., and Weng, Q. (2015). An automated method to parameterize segmentation scale by enhancing intrasegment homogeneity and intersegment heterogeneity. *IEEE Geoscience and Remote Sensing Letters*, 12(6):1282–1286.

Yang, J., Li, P., and He, Y. (2014). A multi-band approach to unsupervised scale parameter selection for multi-scale image segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 94:13–24.

Yin, D., Du, S., Wang, S., and Guo, Z. (2015). A direction-guided ant colony optimization method for extraction of urban road information from very-high-resolution images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(10):4785–4794.

Yu, Q., Gong, P., Clinton, N., Biging, G., Kelly, M., and Schirokauer, D. (2006). Object-based detailed vegetation classification with airborne high spatial resolution remote sensing imagery. *Photogrammetric Engineering & Remote Sensing*, 72(7):799–811.

Zhang, S., Zhang, G., Li, F., Deng, C., Wang, S., Plaza, A., and Li, J. (2022). Spectral-Spatial Hyperspectral Unmixing Using Nonnegative Matrix Factorization. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13.

Zhang, X., Xiao, P., Song, X., and She, J. (2013). Boundary-constrained multi-scale segmentation method for remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 78:15–25.

Zhong, P., Gong, Z., Li, S., and Schönlieb, C.-B. (2017). Learning to diversify deep belief networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(6):3516–3530.

Zhou, S., Xue, Z., and Du, P. (2019). Semisupervised Stacked Autoencoder With Cotraining for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(6):3813–3826.

Zhu, F. (2017). Hyperspectral Unmixing: Ground Truth Labeling, Datasets, Benchmark Performances and Survey.