Design and implementation of new methods for inference tasks using compressively sensed hyperspectral images

M.Sc. Hector Miguel Vargas García

Doctoral thesis to qualify for the Doctor of Engineering degree, electronic area

Advisor

Henry Arguello

Ph.D. in Electrical and Computer Engineering

Universidad Industrial de Santander

School of Physical-Mechanical Engineering

Department of Electrical, Electronics, and Telecommunication Engineering.

Bucaramanga

2023

Dedicatoría

Dedico este trabajo, a mi hija hermosa Gabriela Vargas Hoyos.

A mi esposa incondicional Luisa Marcela Hoyos Marin, por su apoyo y compresión, sin ella esto no hubiese sido posible.

A mi padre Héctor Miguel Vargas, por sus lecciones, el valor del arduo trabajo y sus inspiradoras palabras que me guiaron hacia la excelencia.

A mi madre Ligia García Garnica, por su ejemplo y principios enseñados a lo largo de esta travesía.

A mis hermanos Cesar Augusto Vargas García y Andrea Carolina Vargas García, por su apoyo.

A todos mis compañeros y profesores que creyeron en mis capacidades y las respetaron, sus apoyos fueron fundamentales en cada etapa de mi formación doctoral.

Agradecimientos

Un agradecimiento al grupo de investigación High Dimensional Signal Processing (HDSP) por brindar el apoyo científico como equipo, dado a su gran fortaleza de interdisciplinaridad y excelentes miembros que facilitaron el desarrollo de este trabajo de investigación.

Gracias al profesor Henry Arguello Fuentes por su paciencia y enseñanzas, además de permitirme crecer académicamente en su grupo. Agradezco a mis amigos y compañeros que me apoyaron en este camino, en especial a Ariolfo Camacho, quien me apoyo en los momentos difíciles. Así mismo y no menos importantes infinitas gracias a Gladis Noriega, Samuel Pinilla, Claudia, Laura, Ana y Hoover, quienes aportaron en su medida a mi formación académica.

También agradezco a la Universidad Industrial de Santander por la formación, docencia y su apoyo en el desarrollo de la presente tesis. Además, un reconocimiento especial a la UIS por el crédito condonable que permitió el sostenimiento financiero propio y de mi familia.

Content

page

List of Figures

List of Tables

page

List of Abbreviations

ADMM Alternating direction method of multipliers

CASSI Coded aperture snapshot spectral imager

CS Compressive sensing

DMD Digital micro-mirror device

FOV Field of view

LR Low rank

HSI Hyperspectral image

i.i.d. Independently and identically distributed

PCA Principal component analysis

PSNR Peak signal-to-noise ratio

SVD Singular value decomposition

TV Total variation

Resumen

Título: Diseño e implementación de nuevos métodos para tareas de inferencia usando imágenes hiperespectrales sensadas por compresión [*]

Autor: Héctor Miguel Vargas García [**]

Palabras Clave: Adquisición compresiva, imágenes espectrales, extracción y fusión de características, optimización numérica y clasificación.

Descripción: La escasez, representada por un conjunto reducido de coeficientes en un diccionario dado, es clave en tareas de procesamiento de señales. La adquisición compresiva utiliza proyecciones aleatorias para aprovechar la escasez de las señales en sistemas con recursos limitados, como sensores, con una costosa reconstrucción. Una alternativa es transformar la reconstrucción costosa en un método de procesamiento de señales más económico, estimando un número reducido de características. Por otro lado, las proyecciones aleatorias computacionales se emplean para integrarse eficientemente con métodos de inferencia tradicionales. Se asume que ciertas proyecciones preservan el subespacio de datos, utilizándose en métodos basados en subespacios para reducir ruido y dimensión de la información. En esta tesis, se propone una metodología para adquirir imágenes hiperespectrales de manera compresiva. Se utiliza un sistema óptico multimodal con cámaras hiperespectral y RGB. La extracción de características se realiza sin reconstruir todo el cubo de datos, mediante una estrategia de optimización numérica. Este enfoque demuestra la posibilidad de obtener características discriminatorias sin reconstruir todos los datos en términos de precisión de clasificación.

[*] Tesis de doctorado

[**] Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Eléctrica, Electrónica y de Telecomunicaciones. Doctorado en Ingeniería, Área Electrónica. Director: Henry Arguello Fuentes

Abstract

Title: Design and implementation of new methods for inference tasks using compressively sensed hyperspectral images [*]

Author: Hector Miguel Vargas García [**]

Keywords: Compressive hyperspectral imaging, feature extraction and fusion, numerical optimization and land-cover classification..

Description: Sparsity, the capacity to represent signals with minimal coefficients in a given dictionary, proves beneficial for signal processing applications in communication and storage. Compressive sensing harnesses signal sparsity via random projections, particularly advantageous for resource-constrained sensor systems such as environmental sensors, surveillance systems, and scanners. While compressive sampling in hyperspectral imaging reduces data dimensionality, it necessitates costly signal reconstruction before traditional processing. To address this challenge, ongoing efforts in computational random projection focus on efficiently reducing data and integrating with classical methods. This thesis introduces a novel framework for acquiring hyperspectral images efficiently, employing a multimodal optical system comprising a compressive hyperspectral camera and a high-resolution RGB camera. The proposed compressive feature extraction method relies on preserving subspaces during acquisition. This innovative approach enables the extraction of spatial features directly from compressed measurements, eliminating the need for full data reconstruction. The study showcases the feasibility of obtaining discriminatory features without achieving complete recovery in terms of classification accuracy.

[*] Doctoral Thesis

[**] School of Physical-Mechanical Engineering. Department of Electrical, Electronics, and Telecommunication Engineering. Doctoral degree in Engineering, electronic area. Advisor: Henry Arguello Fuentes

Introduction

Remote sensing is a technology that has revolutionized our ability to acquire and interpret data from the distance, enabling us to understand and monitor our planet with high precision. One of the most powerful tools in the remote sensing is hyperspectral imaging, which captures information across a large range of the electromagnetic spectrum. However, acquiring, transmitting, and processing hyperspectral images presents significant challenges due to the huge volume of data involved. In this introduction, we will explore into these challenges and present how they have been effectively addressed through a groundbreaking technique known as compressive feature extraction.

The Challenges of Hyperspectral Remote Sensing: Hyperspectral remote sensing involves capturing data across hundreds or even thousands of narrow spectral bands, providing rich information about the Earth's surface and atmosphere Bioucas-Dias et al. (2012). The spectral information contains a wealth of data that can be used for inference tasks as object identification or classification in various remote sensing applications, including food quality assessment, precision agriculture, and material identification Lorente et al. (2012); Dale et al. (2013); Bioucas-Dias et al. (2013). While this wealth of data is immensely valuable, it also poses a multitude of challenges. First, acquiring hyperspectral images can be a complex and resource-intensive task. It often requires advanced sensor technology and may demand specialized platforms, such as satellites, drones, or aircraft, to cover large areas. Furthermore, these sensors generate vast amounts of data, which must be efficiently transmitted to processing centers Lopez et al. (2013); Báscones et al. (2018). Traditional methods involve capturing and storing all the spectral information, creating massive data sets that effort computing resources. In addition to acquisition and transmission challenges, processing hyperspectral data is a computational and storage nightmare. Once on the ground, the processing of hyperspectral data involves dimentional reduction through feature extraction, as well spectral unmixing, object detection or classification algorithms to extract valuable information about land cover, vegetation health, mineral composition, and more. However, the size of these data sets makes it difficult to analyze in a timely and cost-effective manner. As a result, these challenges have been a significant bottleneck in the effective use of hyperspectral imaging for environmental monitoring, agriculture, mineral exploration, and other crucial applications.

Overcoming Challenges with Compressive Sampling: Recently, the philosophy behind all our current technologies for digital data acquisition has been changing due to a large amount of information. New acquisition systems are designed with the expectation of requiring a large number of samples, improving the probability of collecting more discriminating information. For spectral imaging, increasing the spectral resolution improves the probability of collect features that enhance the differentiation of materials in a scene. While not all hyperspectral applications require a large number of spectral bands for effective operation, in the absence of prior constraints, a small number of acquired bands may be enough to characterize one scenario but not another. Therefore, hyperspectral systems that upsample the spectral details of a scene allow additional information to accommodate any acquisition variability. Besides, it brings flexibility to the growing interest in extending the use of hyperspectral data to various fields of science. However, the advantages of obtaining large amounts of information come at the cost of a substantial increase in overhead for acquisition, storage, communication, and processing.

To address the hurdles associated with hyperspectral remote sensing, compressive sampling, also known as compressed sensing or CS, has emerged as a revolutionary technique. Compressive sampling is a signal processing approach that exploits the inherent structure and redundancy within hyperspectral data, allowing for efficient data acquisition Li et al. (2012); Lin et al. (2014); Wang et al. (2018); Garcia et al. (2018). This innovative method relies on the principle that many natural signals are inherently sparse or compressible in some domain. In the case of hyperspectral images, this means that the valuable information can be accurately reconstructed from a significantly smaller set of measurements than traditional methods would require. By strategically sampling the signal, compressive sampling reduces the amount of data acquired and transmitted, making it more feasible for resource-constrained platforms. This results in a more efficient and cost-effective workflow, enabling a more responsive time analysis for a wide range of applications.

New Challenges and Research Question: Traditional hyperspectral data is voluminous and high-dimensional, and compressing it significantly reduces the available information Nascimento et al. (2020). However, reconstructing hyperspectral data from compressive measurements for inference tasks, poses some challenges due to the unique characteristics of hyperspectral imagery. When compressive measurements are employed to reduce data volume, critical spectral information may be lost, making it difficult to accurately identify and classify objects. Moreover, hyperspectral data is highly susceptible to noise and atmospheric artifacts, which can further complicate the reconstruction process. The computational demands for reconstructing hyperspectral data are substantial, as they require complex algorithms and substantial processing power. Balancing the trade-off between data compression and the preservation of critical spectral details, while ensuring efficient processing for object detection and classification remains a persistent challenge in the field of hyperspectral remote sensing. This doctoral thesis aims to investigate the effect of number of measurements on classifcation accuracy. The central research question guiding this study is

how can precision be preserved when inferring pixels in hyperspectral images that have been acquired compressively without needing to reconstruct the entire data?

As we delve deeper into the realm of compressive hyperspectral imaging, a shift in perspective is needed. Instead of focusing solely on reconstructing the entire data cube, we must explore methods that extract valuable features from the compressed data. Feature extraction is an essential concept that enables the retrieval of critical information while mitigating the computational and storage challenges associated with data cube reconstruction. Feature extraction from compressive measures offers several advantages over traditional hyperspectral reconstruction for inference tasks such as object detection, or classification. Firstly, it significantly reduces the computational burden by avoiding the need to fully reconstruct high-dimensional hyperspectral data, which can be time-consuming and resource-intensive. This allows efficient data processing, making it suitable for applications that require rapid decision-making. Secondly, feature extraction can enhance the robustness of the inference task by extracting relevant information directly from the compressed measurements, often resulting in more compact and discriminative feature representations. This can lead to improved accuracy in tasks like target detection or classification, where relevant information can be buried within the hyperspectral data. Overall, feature extraction from compressive measures offers a more efficient and effective approach to hyperspectral data analysis, making it a valuable technique in various remote sensing and image processing applications.

This thesis represents a contribution to the effort to bring hyperspectral processing methods to real applications through the analysis of the necessary aspects that allow the incorporation of compressive sensing theory in hyperspectral images.

Related work: Previous works have developed strategies to perform inference tasks in the compressed measurements. For instance, target detection and classification from the compressed measurements are addressed in Yang et al. (2013, 2014). Also, the assumption that the HSIs live in a low dimensional subspace has been recently exploited in the recovery Golbabaee et al. (2013); Martín et al. (2015); Yang et al. (2015a). In Golbabaee et al. (2013), the subspace is assumed known and a spatial regularization term is used to promote the spatial smoothness. In Martín et al. (2015); Yang et al. (2015a), the subspace is assumed known (endmembers, dictionary, etc) or randomly initialized and updated using an alternating optimization (AO) strategy. The use of AO strategy and the ℓ_1 gradient regularization as a form of spatial regularization to extract features has been demonstrated in Rasti et al. (2017). However, all of these methods require the knowledge of the subspace, which, may compromise its applicability. Even more, if the subspace is randomly initialized, it may result in a local minimum solution. Also, sensing strategies have been designed to preserve the subspace and estimate it from the compressed measurements Chen et al. (2014); Yang et al. (2015b); Martín and Bioucas-Dias (2016); Bacca et al. (2019). In Chen et al. (2014), a partition sampling is proposed to estimate the subspace from compressed measurements supported by the Rayleigh-Ritz theory. In Yang et al. (2015b), spatial-spectral tensor sampling is adopted to preserve the structure of the data. In Martín and Bioucas-Dias (2016); Bacca et al. (2019), a measurement strategy is designed on the spectral domain, and thus learn the subspace the from compressed measurements.

On the other hand, recent research has shifted its attention to the integration of multi-sensor data for land cover classification through feature fusion. One notable approach in this domain is the subspace feature fusion (SubFus) technique, which involves extracting spatial features from multimodal images and achieving fused features by solving an optimization problem based on subspaces Rasti et al. (2019); Rasti and Ghamisi (2020). This technique has been studied extensively in the context of hyperspectral (HS) and high-resolution (HR) images to improve classification accuracy. Additionally, methods have been proposed to fuse features of compressive spectral sensor systems with complementary RGB sensors. These methods synthesize sampling matrices that describe compressive measurements as projections of the fused features for different CSI optical architectures, such as the coded aperture snapshot spectral imaging (CASSI) system Ramirez and Arguello (2019); Ramirez and Arguello (2019); Ramirez et al. (2021). The process of feature fusion is framed as the resolution of a regularized inverse problem, and several studies have demonstrated that employing ℓ_1 gradient regularization can serve as an effective preprocessing technique for segmenting regions and improving classification performance. It's worth noting, however, that in contrast to the widespread use of ℓ_1 gradient regularization as a form of spatial regularization, the ℓ_0 gradient regularization stands out by yielding significantly superior results when applied to piecewise constant images. This observation has been supported by various works, as cited in Cascarano et al. (2021); Wang et al. (2021).

General Objective: Design and implement methods that preserve precision during the inference process for compressively captured hyperspectral images.

Specific Objectives: Develop a compressive sampling method for hyperspectral images that preserves classification accuracy during pixel inference.

Design a computational method that preserve inference precision, especially in classification, when working with compressively acquired spectral images.

Implement an optical prototype in the optics laboratory using a Digital Mirror Device (DMD) based on the developed sampling method.

Evaluate the designed methods and compare their performance with state-of-the-art techniques.

Summary of Contributions: A list of the original contributions made within this thesis is as follows:

Chapter 3: One recovery model for improved speed reconstruction of HSI from CS measurements using LR matrix approximation (Section 2.2).

An accompanying recovery algorithm, based on the combination of alternating optimization (AO) and alternating direction method of multipliers (ADMM) (Section 2.3).

Detailed performance analyses of the proposed framework (Section 2.4).

Chapter 4: One recovery model for feature extraction of HSI from CS measurements using multimodal model (Section 3.2).

An accompanying recovery algorithm, based on the combination of alternating optimization (AO) and alternating direction method of multipliers (ADMM) (Section 3.3).

Detailed performance analyses of the proposed multimodal model by means of classification accuracy (Section 3.4).

Journals: Vargas, H., Ramirez, J., & Arguello, H. (2020). ADMM-based $\ell_1 - \ell_1$ optimization algorithm for robust sparse channel estimation in OFDM systems. Signal Processing, 167, 107296.

Vargas, H., & Arguello, H. (2019). A low-rank model for compressive spectral image classification. IEEE Transactions on Geoscience and Remote Sensing, 57(12), 9888-9899.

Conferences: Ramírez, J., Vargas, H., Martínez, J. I., & Arguello, H. (2021, July). Subspace-Based Feature Fusion from Hyperspectral and Multispectral Images for Land Cover Classification. In 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS (pp. 3003-3006). IEEE.

Vargas, H., Fonseca, Y., & Arguello, H. (2018, September). Object detection on compressive measurements using correlation filters and sparse representation. In 2018 26th European Signal Processing Conference (EUSIPCO) (pp. 1960-1964). IEEE.

1. Theoretical Aspects

This chapter presents spectral images, the concept of spatial and spectral resolution are explained along with traditional acquisition methods of hyperspectral imaging systems. On the other hand, compressive sensing, and main approaches used in reconstruction methods are reviewed.

1.1. Notation and Symbols

The mathematical notations used throughout this thesis are as follow. Scalars are represented using letters without bold, e.g., $(x, y, z, \cdots)$, vectors are represented using lower-case bold letters, e.g., $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \cdots)$, and matrices are represented using upper-case bold letters, e.g., $(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \cdots)$. If an individual element of a vector is required, it is indexed by a scalar placed as a lower subscript, e.g., x_i or $\boldsymbol{x}_{(i)}$ which represents the i-th element of vector $\boldsymbol{x}$. Unless specified otherwise, all vectors are by default column vectors. If a row vector is required, it is written as the transpose, e.g., $\boldsymbol{x}^{\mathsf{T}}$. Indexing columns of a matrix are represented by lower subscripts and colon, e.g., $\boldsymbol{x}_j$ or $\boldsymbol{X}_{(:,j)}$ which denotes the j-th column of $\boldsymbol{X}$. Similarly, rows of a matrix are denoted by the corresponding subscripted row vectors, e.g., $\boldsymbol{x}_i^{\mathsf{T}}$ or $\boldsymbol{X}_{(i,:)}$ is the i-th row of $\boldsymbol{X}$. If an individual element of a matrix is required, it is represented as a scalar with two comma separated subscript indices so that $x_{i,j}$, which is the element in the i-th row and j-th column of $\boldsymbol{X}$. Alternatively, it can be written as $\boldsymbol{X}_{(i,j)}$. When manipulating matrices with complex elements, the conjugate (or Hermitian) transpose is denoted by $\boldsymbol{X}^{\mathsf{H}}$. The expression $\boldsymbol{I}_n$ means the identity matrix with dimensions $n \times n$, $\mathbf{0}_{n \times m}$ means a matrix of zeros with dimensions $n \times m$, and $\mathbf{1}_{n \times m}$ means a vector with all components unitary and dimensions $n \times m$. Prior to the ensuing presentation, let us define the symbols in the Table 1 for the ease of later use.

Table 1
Symbols

Symbol	**Description**
$\mathbb{R}$, $\mathbb{C}$	Set of real and complex numbers respectively
$\mathbb{R}^n$, $\mathbb{C}^n$	An n-dimensional vector space defined over real or complex numbers respectively.
$\otimes$	Kronecker operator. Details are presented in Appendix 3.5.
$\text{vec}(\cdot)$	Vectorization operator. Convert from $\boldsymbol{X} \in \mathbb{R}^{n\times m}$ to $\boldsymbol{x} \in \mathbb{R}^{nm}$.
$\text{mat}(\cdot,\ n,\ m)$	Matrization operator. Convert from $\boldsymbol{x} \in \mathbb{R}^{nm}$ to $\boldsymbol{X} \in \mathbb{R}^{n\times m}$.
$\lVert\cdot\rVert_p$	ℓ_p-norm ($p = 1,2,\infty$). Details are presented in Appendix 3.5.
$\lVert\cdot\rVert_{p,q}$	$\ell_{p,q}$ mixed-norm. Details are presented in Appendix 3.5
$\lVert\cdot\rVert_{\mathsf{F}}$	Frobenius norm
$\{i, j, k, \cdots\}$	General-purpose set index of arrays
$\{N_i, N_j, N_k, \cdots\}$	General-purpose set length of arrays

1.2. Hyperspectral Imaging

The Spectral Image (SI) is a data set consisting of spatial and spectral information. Spectral information represents the response to the absorption or emission of electromagnetic radiation to certain wavelengths with spatial coordinates. These images are captured by spectrometer systems that measure the intensity of light waves to study the interaction of electromagnetic radiation with matter to analyze their characteristics.

Electromagnetic spectrum: Imaging sensors primarily utilize optical imaging systems, thermal imaging systems, or synthetic-aperture radar (SAR) Zhu et al. (2018). Figure 1 depicts the electromagnetic spectrum (EMS), spanning from gamma rays to radio waves. Optical imaging systems are designed to capture imagery within the visible, near-infrared, and shortwave infrared spectrums, typically generating panchromatic, multispectral, and hyperspectral images. In the field of remote sensing, common applications involve the visible light (VIS) range 380 – 780 mm, the infrared (IR) range 780 – 0.1 mm, and the microwave range 0.1 – 1 m. Within the VIS range, specific spectral bands such as blue 450 - 495 nm, green 495 - 570 nm, and red 620 – 750 nm are employed for panchromatic, multispectral, or hyperspectral imaging purposes. The red spectrum, in conjunction with near-infrared (NIR), is typically harnessed for applications related to vegetation. For instance, the Normalized Difference Vegetation Index (NDVI) serves as a tool for assessing targets, whether they contain live green vegetation or not. Typically, the infrared spectrum is categorized into distinct regions: Near-Infrared (NIR, 0.78 – 3 μm), Mid-Infrared (MIR, 3 - 50 μm), and Far-Infrared (FIR, 50 – 1000 μm) Zhu et al. (2018). Conversely, Synthetic Aperture Radar (SAR) employs microwaves to illuminate ground targets, gauging the backscatter and travel time of transmitted waves as they bounce off objects on the ground Zhu et al. (2018). Typically, SAR can be categorized into Single frequency (L-band, C-band, or X-band)and Multiple frequency (Combination of two or more frequency bands).

The fundamental factors for designing and operating a Synthetic Aperture Radar (SAR) enclose the following key parameters: electromagnetic energy power, frequency, phase, polarization, incident angle, and spatial resolution.

Acquisition methods: Figure 2 presents an illustration of three distinct acquisition methods. Among these methods, two are spatially scanned approaches: the slit spectrometer and the whiskbroom scanner introduced by Golay in Golay (1949) and discussed further by Breuer in Breuer and Albertz (2000). These techniques involve scanning the spatial coordinates (x,y) of a scene to capture the 3-dimensional (3D) hypercube (x,y,λ). However, it's important to note that

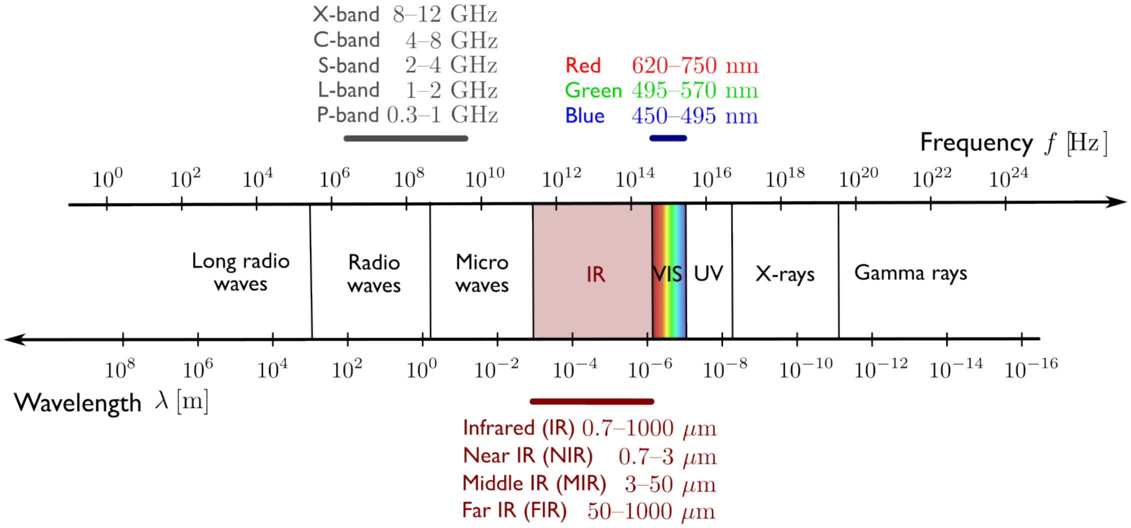


Figure 1. Classification of sensors based on spectral bandwidth.

spatial measurements obtained using these methods may be impacted by temporal misalignment due to object or scene motion. The third method depicted is the spectral scanned spectrometer, featuring a filtered camera, as described by Gat in Gat (2000). This particular sensor measures the wavelength (λ) by sequentially adjusting a spectral bandpass filter over time. Unlike the spatially scanned modalities, spectral measurements are susceptible to potential temporal misregistration. It's worth mentioning that various acquisition methods may exhibit different types of spatial artifacts, depending on the specific imaging technique employed. Additionally, the spatial and spectral resolution can vary between different sensors. Developers are continually working to enhance the quality and increase the resolution of these sensors.

Spatial resolution: As shown in Figure 3, the spatial resolution is the ability to distinguish features in an image and it can be expressed as the minimum distance by which two separate objects are perceived as disjoint. In optical image sensors, the spatial resolution is generally related to the field of view (FOV) of the sensor Brady (2009). Scene elements located at different positions FOV

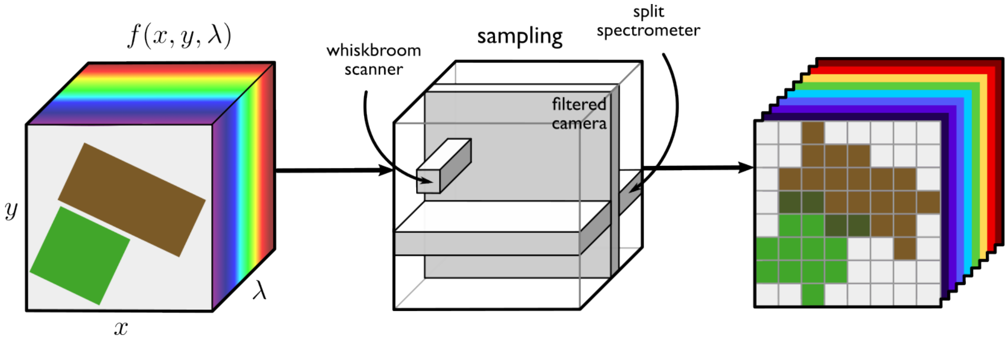


Figure 2. The hypercube that is sampled by traditional spectral imaging devices through spatial or spectral scanning.

show a different spatial resolution. This last aspect is very important for airborne platforms (large FOV) and negligible for satellite sensors (small FOV).

Spectral resolution: As shown in Figure 3, the spectral resolution is the ability of a sensor to respond to a range of specific wavelengths (spectral bands). The sensors can be classified as panchromatic (PAN), multispectral (MS), hyperspectral (HS). The best example for PAN is the sensor in the visible spectral range and near-infrared (NIR-V), where the detector is typically in the range of [400-1000] nanometers. The MS sensors operate in multiple wavelength ranges. The number of bands used is 3-10 in the visible range with wavelengths ranges of about 50 nanometers. When the spectral resolution is better than 10 nanometers, the sensors are denoted as HS and may contain hundreds of bands Brady (2009).

Radiometric resolution: The radiometric resolution of an imaging system es the capacity for distinguishing variations in energy levels. A sensor with higher radiometric resolution will exhibit greater sensitivity in detecting small differences in reflected or emitted energy.

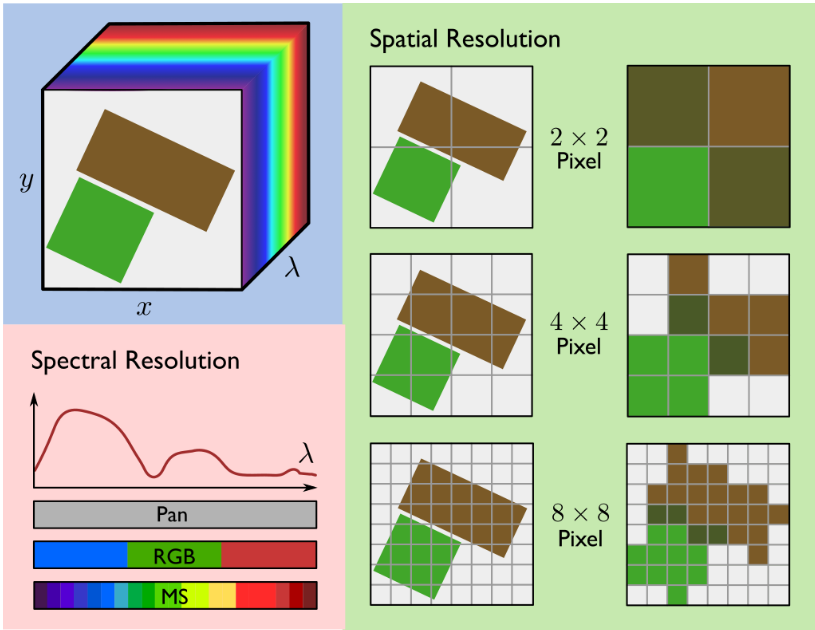


Figure 3. Spatial resolution examples and classification of sensors based on spectral bandwidth.

1.3. Processing Pipeline

In practical hyperspectral imaging applications, the quality of the observed hyperspectral image (HSI) can be adversely affected by various factors, such as the imaging technology, system, and environmental conditions. Consequently, it becomes necessary to estimate a clean, noise-free version of the HSI Rasti et al. (2018). When noise sources degrade the observed signal, this process is commonly referred to as "denoising." To represent a hyperspectral image, we reshape it into a two-dimensional matrix form by flattening its spatial dimensions for each spectral band. Specifically, we represent the hyperspectral image as a matrix denoted by $\boldsymbol{Z} \in \mathbb{R}^{n \times N_\lambda}$, where $n = N_x \cdot N_y$. Here, N_x and N_y represent the spatial dimensions, and N_λ denotes the number of spectral bands. In the matrix representation, the degraded HSI is expressed as a sum of a true unknown signal and an additive noise component, as shown below:

$$\boldsymbol{Y} = \boldsymbol{Z} + \boldsymbol{H}, \tag{1}$$

where $\boldsymbol{Y} \in \mathbb{R}^{n \times N_\lambda}$ containing the vectorized observed image at band i in its ith column, $\boldsymbol{Z} \in \mathbb{R}^{n \times N_\lambda}$ is the true unknown signal, and $\boldsymbol{H} \in \mathbb{R}^{n \times N_\lambda}$ is the matrix representing the noise. The Gaussian noise model is a widely employed approach in the field of remote sensing for modeling HSI. Consequently, the elements in $\boldsymbol{H}_{(i,j)}$ are assumed to be independent and identically distributed (i.i.d.) Gaussian variables with a zero mean and a variance of σ^2.

1.3.1. Feature extraction. While hyperspectral images typically contain abundance data, a significant portion of this information can be redundant or irrelevant, often referred to as noise. Therefore, it becomes crucial to identify the most informative aspects of these images. This process is known as feature extraction, which involves converting the data into numerical features of reduced dimensionality while retaining the most important information from the original dataset. Feature extraction offers several advantages, such as reducing computational complexity and simplifying models. In the field of HSI analysis, linear regression modeling is widely employed for tasks like dimensionality reduction, feature extraction, denoising, and compression. In this context, an HSI represented as $\boldsymbol{Z}$ is commonly modeled as a low-rank (LR) matrix. Consequently, if n vectors, each of dimension N_λ, are confined to a subspace of much lower dimension, denoted as $N_r \ll N_\lambda$, each N_λ-long vector exhibits only N_r degrees of freedom. The singular value decomposition (SVD) provides a means of achieving a LR decomposition of $\boldsymbol{Z}$. This decomposition leads to the noisy low-rank model:

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{Z}+\boldsymbol{H}, \\ &= \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^{\mathsf{T}}+\boldsymbol{H}, \end{aligned} \tag{2}$$

where the columns of $\boldsymbol{U} \in \mathbb{R}^{n\times N_r}$, $\boldsymbol{V} \in \mathbb{R}^{N_\lambda \times N_r}$ are respectively a set of vectorized eigen-images and the associated spectral components, and $\boldsymbol{S} \in \mathbb{R}^{N_r\times N_r}$ is a diagonal singular value matrix. In this variant, $\boldsymbol{U}$ and $\boldsymbol{V}$ are semi-unitary matrices such that $\boldsymbol{U}^{\mathsf{T}}\boldsymbol{U} = \boldsymbol{V}^{\mathsf{T}}\boldsymbol{V} = \boldsymbol{I}_{N_r}$. Note that SVD is an approximation modeled by an error which in this case is Gaussian. In practical scenarios, an efficient estimation of the noise level can be achieved when assuming that the original image exhibits LR characteristics and follows a Gaussian distribution with regards to the elements of the noise matrix, denoted as $\boldsymbol{H}$. Specifically, if $\boldsymbol{H} = \boldsymbol{0}$ and all the eliminated singular values are zero, this representation effectively covers the genuine signal subspace. When the noise condition concerning $\boldsymbol{H}$ is independently and identically distributed (i.i.d.), this representation aligns with the maximum likelihood estimate of the said subspace.

Principal Component Analysis Decomposition: PCA is based on the fact that the neighboring bands of the hyperspectral image are highly correlated and contain similar information. PCA was developed from the point of view of analysis of variance. Hence, the initial principal component (PC) can be regarded as a linear combination of variables that maximizes variance. The subsequent PC represents the linear combination that maximizes variance while satisfying the requirement that its loading vector is orthogonal to the loading vector of the first PC, and so forth. From the geometric point of view, which is closely related to the SVD, PCA seeks to find an (affine) subspace $\mathscr{S} = \{\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_{N_r}\}$ of dimension N_r where $\boldsymbol{q}_i \in \mathbb{R}^{N_\lambda}$ that best fits a set of points $\boldsymbol{Z}_{(i,:)}$ for all $i = \{1, 2, \ldots, n\}$. Therefore, if those points are contaminated by additive noise, then

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{Z} + \boldsymbol{H}, \\ &= \boldsymbol{C}\boldsymbol{Q} + \boldsymbol{H}, \end{aligned} \tag{3}$$

where $\boldsymbol{C} \in \mathbb{R}^{n \times N_r}$ is the feature matrix, $\boldsymbol{Q} \in \mathbb{R}^{N_r \times N_\lambda}$ is an unknown subspace basis, and $\boldsymbol{H}_{(:,i)}$ is i.i.d. Gaussian noise with covariance matrix $\sigma^2 \boldsymbol{I}_n$. In the model (3), the matrix $\boldsymbol{Q}$ contains the eigenvectors of the covariance matrix from the observation matrix $\boldsymbol{Y}$. For this, let $\overline{\boldsymbol{Y}} = \boldsymbol{Y} - \boldsymbol{1}_n \boldsymbol{\mu}^\top$ be the observed data centered along of the columns, where $\boldsymbol{\mu} \in \mathbb{R}^{N_\lambda}$ is the mean vector of the columns of $\boldsymbol{Y}$. The eigen-decomposition of the covariance matrix is given by:

$$\begin{aligned} \overline{\boldsymbol{Y}}^\top \overline{\boldsymbol{Y}} &= \left(\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top\right)^\top \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top, \\ &= \boldsymbol{V}\boldsymbol{S}^\top \boldsymbol{U}^\top \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top, \\ &= \boldsymbol{V}\boldsymbol{S}^2\boldsymbol{V}^\top, \end{aligned}$$

where $\boldsymbol{S}^2$ is the diagonal part of matrix $\boldsymbol{S}$ with every element on the diagonal squared. The eigenvectors of $\overline{\boldsymbol{Y}}^\top \overline{\boldsymbol{Y}}$, can be obtained either by doing an eigen-decomposition of $\overline{\boldsymbol{Y}}^\top \overline{\boldsymbol{Y}}$, or by doing a singular value decomposition from $\overline{\boldsymbol{Y}}$. The pseudo-code of PCA for any matrix is summarized in the Algorithm 1.1.

Algorithm 1.1 Principal component analysis (PCA).

```
function PCA($\boldsymbol{X}$, $N_k$)
    // Let be $\boldsymbol{X} \in \mathbb{R}^{n\times m}$ and $\mathbf{1}_n \in \mathbb{R}^{n}$, then
    $\boldsymbol{\mu} = \left(\frac{1}{n}\right)\boldsymbol{X}^{\top}\mathbf{1}_n$        // where $\boldsymbol{\mu} \in \mathbb{R}^{m}$
    $\overline{\boldsymbol{X}} = \boldsymbol{X} - \mathbf{1}_n\boldsymbol{\mu}^{\top}$
    $[\boldsymbol{U}, \boldsymbol{S}, \boldsymbol{V}] = \text{svd}\left(\overline{\boldsymbol{X}}^{\top}\overline{\boldsymbol{X}}\right)$
    $\boldsymbol{Q} = \left(\boldsymbol{V}_{(:,1:N_k)}\right)^{\top}$        // where $\boldsymbol{Q} \in \mathbb{R}^{N_k\times m}$
    return $\boldsymbol{Q}$
end function
```

1.3.2. Inference Task. HSI finds its primary applications in the domains of environment Smith et al. (2001), agriculture Guan et al. (2004), pharmacology, and more, where it plays a crucial role in detecting, classifying, or identifying objects and their properties based on their spectral characteristics Gehm et al. (2008). Inference, in this context, involves extracting specific information from spectral measurements. Tasks such as unmixing, detection, and classification in spectral imaging have evolved from years of research in Remote Sensing Applications. Remote Sensing (RS) takes advantage of these images to measure, analyze, and interpret objects within a scene, whether at a short, medium, or long distance, achieved through aerial or satellite sensors, as highlighted by Plaza in a recent study Plaza et al. (2009).

Unmixing: Sensors capture scenes in which a single pixel can contain spectral information of different materials. In remote sensing it is very common that more than one material can be within the spatial boundaries of one pixel. The spatial coordinates that contain multiple material are called mixed pixels, in contrast to Pure Pixels (PP) that are pixels containing only one material Keshava and Mustard (2002). Each pixel in the grid is thus a sum of spectral reflectance from the different materials within the spatial boundaries of the pixel. Let $\boldsymbol{z}_i \in \mathbb{R}^{N_\lambda}$ be a spectral vector.

Each vector in the observed spectral image can be represented by a linear combination of basis vectors $\boldsymbol{E} \in \mathbb{R}^{N_\lambda \times N_e}$ (N_e is the number of basis) as

$$\boldsymbol{z}_i = \boldsymbol{E}\boldsymbol{a}_i, \tag{4}$$

where $\boldsymbol{a}_i \in \mathbb{R}^{N_e}$ is the basis coefficients of $\boldsymbol{z}_i$ with respect to $\boldsymbol{E}$. The basis vectors can be seen as the mixing matrix, which contains the spectral characteristics of the endmembers, while $\boldsymbol{a}_i$ represents the abundance coefficients for the spectral vector $\boldsymbol{z}_i$. Due to physical considerations, the abundance coefficients should satisfy two constraints: The abundance non-negativity constraint (ANC) and the abundance sum-to-one constraint (ASC) given by

$$\boldsymbol{a}_i \geq 0, \quad \text{and} \quad \sum_{k=1}^{N_e} \boldsymbol{a}_{i(k)} = 1.$$

In some cases, researchers might anticipate that the combined abundance fractions do not necessarily add up to one, as certain algorithms may not fully capture all materials within a pixel. This is particularly relevant because many algorithms are based on either a geometrical or a statistical framework, as discussed by Bioucas-Dias et al. in their work on hyperspectral imaging Bioucas-Dias et al. (2012).

Classification and Target Detection: In classification, as discussed in Chen et al. (2011) Chen et al. (2011a), a spectral signature can be expressed as a linear combination of a select few elements from an overcomplete dictionary, which is composed of training data from various class categories. This representation can effectively reveal class-specific information, particularly when signals from distinct classes occupy disparate subspaces. It is assumed that the spectral signatures of pixels within the same class approximately share a low-dimensional subspace. Suppose we have N_c distinct classes and the k-th class has N_t training samples $\boldsymbol{D}_k \in \mathbb{R}^{N_\lambda \times N_t}$. If $\boldsymbol{z}_i$ belongs to the k-th class, then its spectrum approximately lies in a low-dimensional subspace spanned by the training samples in the k-th class. Building upon the sparsity assumption mentioned earlier, we can represent an unfamiliar test sample as belonging to the collective set of N_c subspaces that correspond to the N_c classes. By combining the subdictionaries for each class, the test sample, denoted as $\boldsymbol{z}_i$, can be expressed as a linear combination of the training samples in a sparse manner:

$$\boldsymbol{z}_i = \sum_{k=1}^{N_c} \boldsymbol{D}_k \boldsymbol{c}_{i,k},$$

where $\boldsymbol{c}_{i,k} \in \mathbb{R}^{N_t}$ is a coefficients vector associated with each class subdictionary. Ideally, if $\boldsymbol{z}_i$ belongs to the k-th class, then $\boldsymbol{c}_{i,\overline{k}} = \boldsymbol{0}$, where $1 \leq \overline{k} \leq N_c$ and $\overline{k} \neq k$. In supervised classification, it is assumed that N_t training labels per class are known, from which the remaining labels are estimated. Assuming that there are N_c different classes, N_t is the number of training samples per class and every single observation vector belongs to one of the given classes, the problem of classifying each test vector consists in finding the class whose training vector is the nearest to the test vector in the Euclidean distance sense. In the case of target detection Chen et al. (2011b), typically the dictionary consists of the training samples from the target and background subdictionaries represented by $\boldsymbol{D} = [\boldsymbol{D}_{\mathrm{t}}\, \boldsymbol{D}_{\mathrm{d}}]$ and sparse representation vector $\boldsymbol{c}_i = \left[(\boldsymbol{c}_{\mathrm{t}})_i^{\mathsf{T}}\ (\boldsymbol{c}_{\mathrm{b}})_i^{\mathsf{T}}\right]^{\mathsf{T}}$, where $(\boldsymbol{c}_{\mathrm{t}})_i$ and $(\boldsymbol{c}_{\mathrm{b}})_i$

represent the sparse coefficient vectors corresponding to the target and background dictionaries, respectively.

1.4. Compressive Spectral Imaging

Traditional approaches to sampling signals and images are based on the Nyquist / Shannon theorem which states that the sampling rate must be greater than twice the bandwidth of an input signal Vaidyanathan (2001). However, in Candes and Tao (2006), a new concept called Compressive Sampling (CS) was proposed as a signal acquisition and compression method. Generally, CS states that it is possible to obtain images or signals from a reduced number of data samples than the criterion of Nyquist / Shannon Romberg (2008); Duarte et al. (2008); Tropp et al. (2006). The success of this technique is that sensing and compression processes are carried out simultaneously and the number of samples required is significantly reduced. CS requires two conditions under which recovery is possible Candes and Romberg (2007). There are two key concepts to consider. The first is "sparsity," which demand that the signal exhibit sparsity in a specific domain. The second concept is "incoherence," which is enforced by the isometric property and is adequate for sparse signals.

1.4.1. Mathematical model.

In order to describe the compressive hyperspectral image acquisition process, suppose the i-th image band is represented by the vector $\boldsymbol{z}_i \in \mathbb{R}^n$, that is a column-wise concatenation of the pixels in the i-th band. A hyperspectral image with N_λ bands can then be represented as a matrix $\boldsymbol{Z} \in \mathbb{R}^{n \times N_\lambda}$. Let us define the operator $\phi : \mathbb{R}^{n \times N_\lambda} \rightarrow \mathbb{R}^m$. Since compressive sensing theory deals with linear sampling schemes, any compressive hyperspectral image acquisition process can therefore be modelled by:

$$\boldsymbol{y} = \phi(\boldsymbol{Z}) + \boldsymbol{\eta}, \tag{5}$$

where $\boldsymbol{y} \in \mathbb{R}^m$ is the vector containing m linear measurements of $\boldsymbol{Z}$, corrupted by the noise vector $\boldsymbol{\eta} \in \mathbb{R}^m$ inherent to the acquisition process. By assuming that $\phi(\cdot)$ is a linear operator, one can also write (5) equivalently as the more familiar matrix-vector product:

$$\boldsymbol{y} = \boldsymbol{\Phi}\text{vec}\,(\boldsymbol{Z}) + \boldsymbol{\eta}, \tag{6}$$

where the operator $\mathscr{A}$ is represented as the matrix $\boldsymbol{\Phi} \in \mathbb{R}^{m \times nN_\lambda}$ so that each element in $\boldsymbol{y}$ corresponds to an inner product between the hyperspectral image $\boldsymbol{Z}$, and the corresponding row of $\boldsymbol{\Phi}$. Consider the following example in the Coded Aperture Snapshot Spectral Imager (CASSI): The imaging process involves encoding the spatial dimension and employing dispersive components. CASSI's operational principle relies on the use of a coded aperture and a dispersive component to control the optical field originating from the scene. This involves projecting the object through a coded aperture, a dispersive element, and several sets of relay lenses onto the detector, resulting in a multiplexed projection of the three-dimensional data cube. To obtain compressive measurements across the Focal Plane Array (FPA), the optical field is integrated over the detector's spectral range. Alternatively, the Compressive Whiskbroom is the solution based on the work Fowler (2014). The measurement system operates directly within the instrument optics, eliminating the necessity to capture data in its entirety in spectral resolution. The optical system comprises three primary components: the initial element split the light into various wavelengths, the second element employs a programmable DMD to execute signal-random element multiplications, and the final element, a cylindrical lens, combines the reflected light from the DMD producing the final randomly projected observations.

1.4.2. Sparse representation. Spectral data cubes are sparse and they admit a representation in a given basis or frame in which most of the coefficients are small and they are well approximated with a small number of large coefficients. Kronecker dictionaries, those that can be written as a Kronecker product of elementary matrices, play a key role in the sparse representation of higher dimensional data. In order to introduce them here, let us consider the simplest case of a hyperspectral image $\boldsymbol{Z} \in \mathbb{R}^{n \times N_\lambda}$ for which a separable transform can be applied as follows:

$$\boldsymbol{C} = \boldsymbol{\Psi}_{\text{2D}} \boldsymbol{Z} \boldsymbol{\Psi}_{\text{1D}}, \tag{7}$$

with $\boldsymbol{\Psi}_{\text{2D}} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{\Psi}_{\text{1D}} \in \mathbb{R}^{N_\lambda \times N_\lambda}$ being matrices associated with the transforms of columns and rows, respectively, and $\boldsymbol{C} \in \mathbb{R}^{n \times N_\lambda}$ is the matrix of coefficients. Also, the original hyperspectral image $\boldsymbol{Z}$ can be recovered by applying the inverse transform,

$$\boldsymbol{Z} = \boldsymbol{\Psi}_{\text{2D}}^{-1} \boldsymbol{C} \boldsymbol{\Psi}_{\text{1D}}^{-1}. \tag{8}$$

Then, the signal $\boldsymbol{Z}$ has a N_s-sparse representation with respect to the dictionary $\boldsymbol{\Psi} = (\boldsymbol{\Psi}_{\mathrm{1D}}^{-1})^{\mathsf{T}} \otimes \boldsymbol{\Psi}_{\mathrm{2D}}^{-1}$ if the following relation holds:

$$\mathrm{vec}(\boldsymbol{Z}) = \boldsymbol{\Psi}\boldsymbol{c}, \text{ with } \|\boldsymbol{c}\|_0 \leq N_s, \tag{9}$$

where $\boldsymbol{c} = \mathrm{vec}(\boldsymbol{C})$ with dimensions n. The functional $\|\boldsymbol{x}\|_0 = \{\#i \mid x_i \neq 0\}$ is the ℓ_0 pseudo-norm of the vector $\boldsymbol{x} \in \mathbb{R}^n$ obtained by counting the number of nonzero entries. Traditional basis functions are the wavelet transform, cosine transform, and pre-trained dictionaries Arguello and Arce (2011, 2013). One important goal of CS is to recover the signal $\boldsymbol{c}$ from the fewest possible measurements $\boldsymbol{y}$. Many vector $\mathrm{vec}(\boldsymbol{C})$ can yield the measurements $\boldsymbol{y}$ due to the rank deficiency of matrix $\boldsymbol{A} = \boldsymbol{\Phi}\boldsymbol{\Psi}^{\mathsf{T}}$. The coherence Candes and Romberg (2007) between the sampling matrix $\boldsymbol{\Phi}$ and the sparsifying basis $\boldsymbol{\Psi}$ is defined as

$$\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) = \sqrt{nN_\lambda} \left(\max_{1 \leq i,j \leq nN_\lambda} |\langle \boldsymbol{\phi}_i, \boldsymbol{\psi}_j \rangle| \right),$$

where $|\langle \boldsymbol{\phi}_i, \boldsymbol{\psi}_j \rangle|$ represents the inner product between the i-th column of $\boldsymbol{\Phi}$ and the j-th column of $\boldsymbol{\Psi}$. The coherence metric quantifies the strongest correlation between any two elements within $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$. The coherence value, denoted as $\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi})$, falls within the range of $[1, \sqrt{nN_\lambda}]$. When $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ exhibit correlated elements, the coherence value is high; conversely, it is low when the elements are uncorrelated. In the context of compressive sampling, the primary focus is on pairs with low coherence, essentially incoherent pairs. Conversely, an effective approach to constructing the matrix $\boldsymbol{\Phi}$ is to explicitly sample measurements from an orthonormal basis. When selecting $\boldsymbol{\Psi}$ and $\boldsymbol{\Phi}$, it is possible to implement the matrix $\boldsymbol{A}$ efficiently, eliminating the need for explicit matrix multiplications. This optimization aids in the development of rapid algorithms for solving the reconstruction problem.

1.4.3. Reconstruction algorithms.

The previous theoretical results correspond to a noiseless scenario, but in practical situations, noise is always present. When $\boldsymbol{\eta}$ represents additive Gaussian noise with limited energy, the inversion of the process in (6) can be approached in various ways. If we lack any prior information about the unknown, Maximum Likelihood (ML) estimation recommends identifying the $\boldsymbol{Z}$ that yields the most probable set of measurements $\boldsymbol{y}$. However, this approach is often filled with difficulties since most inverse problems are ill-posed. A more stable solution to the aforementioned inverse problem is offered by the Maximum-A posteriori Probability (MAP) estimator, which introduces regularization into the estimation process by assuming a prior distribution over the signal space. When exploring the extensive body of published work in this field, two primary types of priors emerge Elad et al. (2007).

MAP synthesis approach: The first type of prior arises from employing a synthesis-based approach. Suppose that our clean hyperspectral signal $\boldsymbol{Z} \in \mathbb{R}^{n \times N_\lambda}$ in the model (6) is replaced using the linear combination of (9) as:

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{\Phi}\text{vec}(\boldsymbol{Z}) + \boldsymbol{\eta} \\ &= \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{c} + \boldsymbol{\eta} \\ &= \boldsymbol{A}\boldsymbol{c} + \boldsymbol{\eta}. \end{aligned} \tag{10}$$

We assume that for the clean signal $\boldsymbol{Z}$, its representation $\boldsymbol{c}$ is sparse in the transform basis $\boldsymbol{\Psi} \in \mathbb{R}^{nN_\lambda \times nN_\lambda}$, implying that only a few coefficients are involved in its construction. Also, we assume that the transformation matrix $\boldsymbol{\Psi}$ satisfies the property $\boldsymbol{\Psi}^\top\boldsymbol{\Psi} = \boldsymbol{\Psi}\boldsymbol{\Psi}^\top = \boldsymbol{I}_{nN_\lambda}$. By using the statistical properties of the noise vector $\boldsymbol{\eta}$, which is assumed normally distributed with zero mean, and variance σ^2, then the following is the MAP synthesis option for the recovery of $\boldsymbol{Z}$:

$$\boldsymbol{c} \in \underset{\boldsymbol{c}}{\arg\min}\ f(\boldsymbol{c}) + g(\boldsymbol{c}), \tag{11}$$

where

$$f(\boldsymbol{c}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{c}\|_2^2, \quad g(\boldsymbol{c}) = \lambda\|\boldsymbol{c}\|_1,$$

with $\|\boldsymbol{x}\|_1$ being the ℓ_1-norm of the vector $\boldsymbol{x}$, and $\lambda > 0$ a regularization parameter that balances the data-fidelity term and code sparsity. Note that f and g are convex functional and f is differentiable. The proximal gradient method Parikh et al. (2014):

$$\begin{aligned}\boldsymbol{c}^{(t)} &= \operatorname{prox}_{\rho,g}\left(\boldsymbol{c}^{(t-1)} - \rho\nabla f(\boldsymbol{c}^{(t-1)})\right)\\ &= \operatorname{soft}\left(\boldsymbol{c}^{(t-1)} - \rho\boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{c}^{(t-1)} - \boldsymbol{y}\right),\, \lambda\rho\right),\end{aligned}$$

is a fixed point iteration that converges to the unique minimizer of the objective function $f(\boldsymbol{c}) + g(\boldsymbol{c})$ for a fixed step-size $\rho \in (0, 1/L]$, where ∇f is the gradient of f, and L is the Lipschitz constant of ∇f. The Lipschitz constant in the problem (11) is approximate as $L = \lambda_{\max}(\boldsymbol{A}^\top\boldsymbol{A})$, which is the largest eigen-value of gram matrix $\boldsymbol{A}^\top\boldsymbol{A}$. The operator $\operatorname{prox}_{\lambda,g}(\boldsymbol{x})$ is the proximal operator with parameter λ for functional g. The proximal operator for the ℓ_1-norm is the element-wise soft-thresholding operator, which is defined as $\text{soft}(x,\tau) = \text{sign}(x)\max(|x|-\tau,\ 0)$. Details of the proximal gradient method are described in Algorithm 1.2. When the problem in (11) is solved, the hyperspectral image is estimated as:

$$\boldsymbol{Z}_{\text{map-s}} = \text{mat}\left(\boldsymbol{\Psi}\hat{\boldsymbol{c}},\ n,\ N_\lambda\right).$$

The iterative soft thresholding algorithm (ISTA) is by no means the best solution for solving the (11) problem, however, it is very simple, and easy to implement. Alternatively, Fast ISTA (FISTA) is a fast version of the proximal gradient method which adaptively change the step-size of the gradient to improve its convergence Beck and Teboulle (2009).

Algorithm 1.2 Proximal gradient ℓ_1-norm.

```
Input: $\boldsymbol{y} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{m \times nN_\lambda}$, $\rho \in (0, 1/L)$, $\lambda > 0$ and $\varepsilon$.
Initialize: $\boldsymbol{c}^{(0)} = \boldsymbol{0}$, error $= 1$
for $t = 1, 2, \ldots$ to MAXITER or (error $< \varepsilon$) do
    $\boldsymbol{c}^{(t)} = \text{soft}\left(\boldsymbol{c}^{(t-1)} - \rho\boldsymbol{A}^\top\left(\boldsymbol{A}\boldsymbol{c}^{(t-1)} - \boldsymbol{y}\right),\ \lambda\rho\right)$
    error $= \|\boldsymbol{c}^{(t)} - \boldsymbol{c}^{(t-1)}\|_2^2$
end for
Set $\hat{\boldsymbol{c}} = \boldsymbol{c}^{(t)}$
Output: $\hat{\boldsymbol{c}}$
```

MAP analysis approach: The second method employs an analysis-based approach, wherein it computes the likelihood of a signal by applying a series of forward transforms to it. These prior probabilities serve as fundamental components in numerous traditional and contemporary algorithms, often serving as regularization factors in optimization tasks.

$$\boldsymbol{Z}_{\text{map-a}} \in \arg\min_{\boldsymbol{Z}} \; f(\boldsymbol{Z}) + g(\boldsymbol{Z}), \tag{12}$$

where

$$\begin{aligned} f(\boldsymbol{Z}) &= \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\text{vec}(\boldsymbol{Z})\|_2^2 \\ g(\boldsymbol{Z}) &= \lambda_{\text{2D}} \sum_{i=1}^{N_\lambda} \left(\left\|\boldsymbol{D}_\text{h}\boldsymbol{Z}_{(:,i)}\right\|_1 + \left\|\boldsymbol{D}_\text{v}\boldsymbol{Z}_{(:,i)}\right\|_1 \right) + \lambda_{\text{1D}} \sum_{i=1}^{n} \left\|\boldsymbol{Z}_{(i,:)}\boldsymbol{D}_\lambda\right\|_1 \end{aligned}$$

are lower semi-continuous functions. The matrices $\boldsymbol{D}_\text{h}$, $\boldsymbol{D}_\text{v}$ and $\boldsymbol{D}_\lambda$ are the spatial and spectral sparsity operators, respectively, with their own regularization parameters λ_{2D} and λ_{1D}. Details of the of $\boldsymbol{D}_\text{h}$ and $\boldsymbol{D}_\text{v}$ are consigned in the Appendix 3.5.

On the othder hand, alternating direction method of multiplier (ADMM) is a variant of the family of algorithms known as the augmented Lagrangian methods Boyd et al. (2011) that solves problems in the form of (12), by rewritten it as a contrained optimization problem

$$\min_{\boldsymbol{Z}_1, \boldsymbol{z}_2} \quad f(\boldsymbol{Z}_1) + g_z(\boldsymbol{z}_2) \quad \text{s.t.} \quad \boldsymbol{D}\text{vec}(\boldsymbol{Z}_1) = \boldsymbol{z}_2, \tag{13}$$

where

$$g_z(\boldsymbol{z}_2) = g_z\left(\boldsymbol{z}_{2,1}, \boldsymbol{z}_{2,2}, \boldsymbol{z}_{2,3}\right) = \lambda_{2\mathrm{D}}\left(\left\|\boldsymbol{z}_{2,1}\right\|_1 + \left\|\boldsymbol{z}_{2,2}\right\|_1\right) + \lambda_{1\mathrm{D}}\left\|\boldsymbol{z}_{2,3}\right\|_1$$

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{I}_{N_\lambda} \otimes \boldsymbol{D}_\mathrm{h} \\ \boldsymbol{I}_{N_\lambda} \otimes \boldsymbol{D}_\mathrm{v} \\ (\boldsymbol{D}_\lambda)^\mathsf{T} \otimes \boldsymbol{I}_n \end{bmatrix}, \quad \boldsymbol{z}_2 = \begin{bmatrix} \boldsymbol{z}_{2,1} \\ \boldsymbol{z}_{2,2} \\ \boldsymbol{z}_{2,3} \end{bmatrix},$$

and $\boldsymbol{z}_{2,1}, \boldsymbol{z}_{2,2}, \boldsymbol{z}_{2,3} \in \mathbb{R}^{nN_\lambda}$ are auxiliary variables. Despite this seemingly trivial change, the MM now solves (13) by forming the so-called augmented Lagrangian of (13):

$$\mathscr{L}\left(\boldsymbol{Z}_1, \boldsymbol{z}_2, \boldsymbol{z}_3,\right) = f(\boldsymbol{Z}_1) + g_z(\boldsymbol{z}_2) + \frac{\rho}{2}\left\|\boldsymbol{D}\mathrm{vec}(\boldsymbol{Z}_1) - \boldsymbol{z}_2 + \boldsymbol{z}_3\right\|_2^2,$$

where $\boldsymbol{z}_3$ is the associated Lagrange multiplier and $\rho > 0$ is a scalar constant. The ADMM finds the solution to (13) by iterating between minimizing $\mathscr{L}\left(\boldsymbol{Z}_1, \boldsymbol{z}_2, \boldsymbol{z}_3\right)$ with respect to $\left(\boldsymbol{Z}_1, \boldsymbol{z}_2\right)$ while keeping $\boldsymbol{z}_3$ fixed, and updating $\boldsymbol{z}_3$ for the given $\boldsymbol{Z}_1$ and $\boldsymbol{z}_2$ until the designated stopping criterion is satisfied. The resulting algorithm is presented in Algorithm 1.3.

For the termination of Algorithm 1.3, the stopping criterion described in (Boyd et al., 2011, sec. 3.3.1) is adopted. Then, the Algorithm 1.3 ends when r_res and s_res are smaller than some

Algorithm 1.3 Alternating direction method of multiplier (ADMM).

```
Input: $\boldsymbol{y} \in \mathbb{R}^m$, $\boldsymbol{\Phi} \in \mathbb{R}^{m \times nN_\lambda}$, $\rho, \lambda > 0$ and $\varepsilon$.
Initialize: $\boldsymbol{Z}_2^{(0)} = \mathbf{0}$, $\boldsymbol{z}_3^{(0)} = \mathbf{0}$
for $t = 1, 2, \ldots$ to MAXITER or $(\text{error} < \varepsilon)$ do
    $\boldsymbol{Z}_1^{(t)} \in \arg\min_{\boldsymbol{Z}_1} \; f(\boldsymbol{Z}_1) + (\rho/2) \left\| \boldsymbol{D}\text{vec}(\boldsymbol{Z}_1) - \boldsymbol{z}_2^{(t-1)} + \boldsymbol{z}_3^{(t-1)} \right\|_2^2$
    $\boldsymbol{z}_2^{(t)} \in \arg\min_{\boldsymbol{z}_2} \; g_z(\boldsymbol{z}_2) + (\rho/2) \left\| \boldsymbol{D}\text{vec}\left(\boldsymbol{Z}_1^{(t)}\right) - \boldsymbol{z}_2 + \boldsymbol{z}_3^{(t-1)} \right\|_2^2$
    $\boldsymbol{z}_3^{(t)} = \boldsymbol{z}_3^{(t-1)} + \boldsymbol{D}\text{vec}\left(\boldsymbol{Z}_1^{(t)}\right) - \boldsymbol{z}_2^{(t)}$
    $\text{error} = r_{\text{res}} + s_{\text{res}}$
end for
Set $\hat{\boldsymbol{Z}} = \boldsymbol{Z}_1^{(t)}$
Output: $\hat{\boldsymbol{Z}}$
```

threshold ε, where

$$r_{\text{res}} = \left\| \boldsymbol{D}\text{vec}\left(\boldsymbol{Z}_1^{(t)}\right) - \boldsymbol{z}_2^{(t)} \right\|_2^2 \Big/ \max\left(\left\| \boldsymbol{D}\text{vec}(\boldsymbol{Z}_1^{(t)}) \right\|_2^2, \left\| \boldsymbol{z}_2^{(t)} \right\|_2^2 \right),$$

$$s_{\text{res}} = \left\| \rho \boldsymbol{D}^\top \left(\boldsymbol{z}_2^{(t)} - \boldsymbol{z}_2^{(t-1)} \right) \right\|_2^2 \Big/ \left\| \boldsymbol{D}^\top \boldsymbol{z}_3^{(t)} \right\|_2^2,$$

are the relative primal residual and dual residual, respectively.

2. Compressive Hyperspectral Image Acquisition, Reconstruction, and Classification

Previous sections described some compressive spectral imaging systems that allow acquiring spectral images with a small number of measurements. These acquisition systems are designed according to CS theory, which contains enough information for an accurate SI reconstruction. The benefit of these implementations comes from CS sampling theory with a significant reduction in acquisition complexity (from a hardware point of view) at the expense of additional recovery procedures that the decoder must perform to recover the full-dimensional image. However, reconstruction may not be necessary in certain applications such as land-cover classification. Instead of knowing the full image, researchers are interested in features that could be extracted directly from the compressed measurements, which provide high inference capabilities. LR matrix approximation has been widely used in feature extraction, because it reduces the data dimension and computational cost. Therefore, in this thesis, compressive hyperspectral imaging and feature extraction are combined in a framework for HSI classification using a LR matrix approximation model. In the proposed framework, the compressed measurements are acquired from a single pixel spectrometer. Instead of using the traditional high-complexity reconstruction model, a LR matrix factorization problem is formulated. The LR problem maximizes the posterior distribution with respect to the feature space and coefficients, and it is numerically solved based on an alternating optimization strategy. By incorporating spatial information, the numerical procedure minimizes the total variational of the feature coefficients subject to an orthogonality constraint for the feature space. Experiments on real HSI show that the proposed approach can provide equally competitive classification results when compared to the traditional approach that performs feature extraction and classification on the recovered from the compressive measurements.

2.1. Single Pixel Hyperspectral Camera

Figure 4 illustrates the acquisition process of the Compressive Hyperspectral Imaging (CHSI) system used in this work Li et al. (2012). The continuous model of the power spectral density through the spatial light modulator and optics at the detector is defined as follows. The spectral density entering the instrument is denoted as $z(x,y,\lambda)$. Immediately after the spatial light modulator, the spectral density is given by

$$z_1^{(k)}(x,y,\lambda) = z(x,y,\lambda)\phi^{(k)}(x,y), \tag{14}$$

where $\phi^{(k)}(x,y)$ is the transmission function of the spatial modulation and k indexes the number of random patterns, $k = 1,\ldots,m$. Let $\phi_{i,j}^{(k)} \in \{0,1\} \mid i = 1,\ldots,N_x,\ j = 1,\ldots,N_y$, be the discretization of the spatial modulation function $\phi^{(k)}(x,y)$ such that

$$\phi^{(k)}(x,y) = \sum_{i,j} \phi_{i,j}^{(k)} \mathrm{rect}\left(\frac{x}{\Delta} - i, \frac{y}{\Delta} - j\right), \tag{15}$$

where

$$\mathrm{rect}(x,y) = \begin{cases} 1 & \text{if } 0 \leq x \leq 1,\ 0 \leq y \leq 1 \\ 0 & \text{otherwise} \end{cases},$$

and the spatial modulator has pixel size Δ. Similarly, the spatial discretization of the spectral data cube is defined as

$$z_{i,j}(\lambda) = \iint z(x,y,\lambda)\,\mathrm{rect}\left(\frac{x}{\Delta} - i, \frac{y}{\Delta} - j\right) dxdy. \tag{16}$$

The continuous model for the spectral density through the spatial light modulator and the optics, before it impinges the sensor array is given by

$$z^{(k)}(\lambda) = \iint z_1^{(k)}(x,y,\lambda)dxdy. \tag{17}$$

Using (14), (15), and (16), Equation (17) can be rewritten as

$$\begin{aligned} z^{(k)}(\lambda) &= \iint z(x,y,\lambda) \sum_{i,j} \phi_{(i,j)}^{(k)} \mathrm{rect}\left(\frac{x}{\Delta} - i, \frac{y}{\Delta} - j\right) dxdy \\ &= \sum_{i,j} \phi_{i,j}^{(k)} z_{i,j}(\lambda). \end{aligned} \tag{18}$$

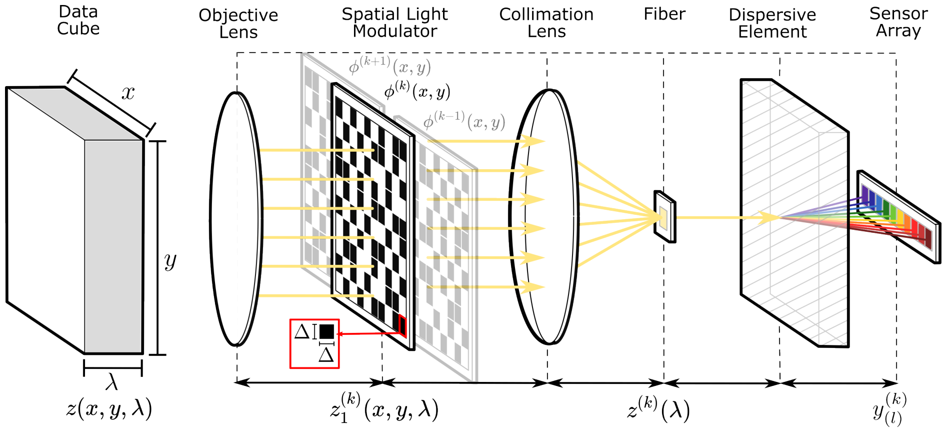


Figure 4. Schematic of the single pixel camera for hyperspectral data acquisition Li et al. (2012).

For the discretization in the spectral domain, the spectral range is partitioned into a finite number of subintervals, or channels. Let the discretization of the spectral axis be λ_l for $l = 1,\ldots,N_\lambda$. The range of the l-th spectral band is $[\lambda_l, \lambda_{l+1}]$ where λ_l is the solution to the equation given by

$$\mathscr{S}(\lambda_{l+1}) - \mathscr{S}(\lambda_l) = \Delta_\lambda, \quad l = 1,\ldots,N_\lambda, \tag{19}$$

where $\mathscr{S}(\lambda)$ is the dispersion induced by the dispersive element. In practice, the dispersive element and sensor operations are performed by a spectrometer. Thus, in the presence of noise, the measurements on the detector can be represented as

$$x_l^{(k)} = \int z^{(k)}(\lambda) \operatorname{rect}\left(\frac{\lambda}{\Delta_\lambda(l)} - l\right) d\lambda + w^{(k)}, \tag{20}$$

where $w^{(k)}$ is additive noise in the sensor, and $\Delta_\lambda(l) = \lambda_{l+1} - \lambda_l$ is the range of the l-th spectral band. As a result, the sensor array measurements can be written as

$$x_l^{(k)} = \sum_{i,j} \phi_{i,j}^{(k)} z_{i,j,l} + w^{(k)}, \tag{21}$$

where

$$z_{i,j,l} = \int z_{i,j}(\lambda) \operatorname{rect}\left(\frac{\lambda}{\Delta_\lambda(l)} - l\right) d\lambda.$$

Equation (21) can be expressed as the linear matrix-vector system. By converting from row-column subscripts into linear indexing with $n = N_x N_y$, the CHSI can be expressed as a linear matrix system given by

$$\begin{bmatrix} \boldsymbol{x}_1^{\mathsf{T}} \\ \boldsymbol{x}_2^{\mathsf{T}} \\ \vdots \\ \boldsymbol{x}_m^{\mathsf{T}} \end{bmatrix} = \begin{bmatrix} \phi_1^{(1)} & \phi_2^{(1)} & \cdots & \phi_n^{(1)} \\ \phi_1^{(2)} & \phi_2^{(2)} & \cdots & \phi_n^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1^{(m)} & \phi_2^{(m)} & \cdots & \phi_n^{(m)} \end{bmatrix} \begin{bmatrix} \boldsymbol{z}_1^{\mathsf{T}} \\ \boldsymbol{z}_2^{\mathsf{T}} \\ \vdots \\ \boldsymbol{z}_n^{\mathsf{T}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{w}_1^{\mathsf{T}} \\ \boldsymbol{w}_2^{\mathsf{T}} \\ \vdots \\ \boldsymbol{w}_m^{\mathsf{T}} \end{bmatrix}, \tag{22}$$

where $\boldsymbol{x} \in \mathbb{R}^{N_\lambda}$ is the k-th sample for all bands, and $\boldsymbol{z}_i \in \mathbb{R}^{N_\lambda}$ is the i-th hyperspectral pixel for all bands. In matrix form would be

$$\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{Z} + \boldsymbol{W}, \tag{23}$$

where $\boldsymbol{\Phi} \in \{0,1\}^{m\times n}$ represents the acquisition HSI system, $\boldsymbol{X} \in \mathbb{R}^{m\times N_\lambda}$ denotes the observation matrix with m measurements and $\boldsymbol{W}$ is the noise matrix. For practical implementation purposes, the real system can be implementing by first capturing $\boldsymbol{X}_1 = \boldsymbol{\Phi}_1\boldsymbol{Z} + \boldsymbol{W}_1$ where $\boldsymbol{\Phi}_1 \in \{1\}^{m\times n}$ is a matrix of ones, $\boldsymbol{Y}_1 \in \mathbb{R}^{m\times N_\lambda}$, and $\boldsymbol{W}_1 \in \mathbb{R}^{m\times N_\lambda}$ is the noise. Note that just one measurement is needed to form $\boldsymbol{Y}_1$ since all the projections on $\boldsymbol{\Phi}_1$ are the same. Second, the m measurements $\boldsymbol{X}$ are obtained from Equation (23). Thus, the measurements corresponding to the matrix $\boldsymbol{\Phi}_{2\mathrm{d}} \in \{-1,1\}^{m\times n}$ are obtained by performing the following operations

$$\begin{aligned} (2\boldsymbol{X} - \boldsymbol{X}_1) &= (2\boldsymbol{\Phi} - \boldsymbol{\Phi}_1)\boldsymbol{Z} + (2\boldsymbol{W} - \boldsymbol{W}_1) \\ \boldsymbol{Y} &= \boldsymbol{\Phi}_{2\mathrm{d}}\boldsymbol{Z} + \boldsymbol{H}. \end{aligned} \tag{24}$$

However, the projection matrix $\boldsymbol{\Phi}_{2\mathrm{d}}$ requires mn memory units for storage and $\mathscr{O}(mnN_\lambda)$ operations, which quickly reaches practical computational limits. To overcome these drawbacks, Structurally Random Matrix (SRM) coding strategy is used, due to its optimal sensing performance, fast transformation and hardware/optics implementation Do et al. (2012). The SRM is defined as

$$\boldsymbol{\Phi}_{\mathrm{2d}} = \left(\sqrt{n/m}\right)\boldsymbol{R}\boldsymbol{H}_{\mathrm{h}}\boldsymbol{P},$$

where $\boldsymbol{P} \in \{0,-1,1\}^{n\times n}$ is a diagonal matrix whose diagonal entries have independent random signs, $\boldsymbol{H}_{\mathrm{h}} \in \{-1,1\}^{n\times n}$ is the Walsh–Hadamard transform, and $\boldsymbol{R} \in \{0,1\}^{m\times n}$ is a subset of rows from the identity matrix. The scale coefficient $(\sqrt{n/m})$ normalizes the transform so that the energy of the measurement vector is almost similar to that of the input signal vector. The minimal number of measurements for N_s-sparse vectors is given by $m = \mathscr{O}(N_s \log(n))$ when $\boldsymbol{\Phi}_{\mathrm{2d}}$ is a SRM, (Do et al., 2012, Theorem IV.2). Note that coding strategy presented in Do et al. (2012), which is named Structurally Random Matrix (SRM), is the same coding strategy used in Boutsidis and Gittens (2013), which is named Subsampled Randomized Hadamard Transform (SRHT). The purpose of the coding strategy used in Boutsidis and Gittens (2013) is to preserve the subspace matrix in randomized low-rank approximation algorithms. From this, we can assume that this coding strategy, proposed in Do et al. (2012) and used in Boutsidis and Gittens (2013), is good for sparse recovery and subspace matrix preservation. The bound of number of measurements is analyzed from the coherence property of the sensing matrix and not from the RIP property.

2.2. Problem Statement

In general, the degradation model (24) represents the discrete approximation of the components of data acquisition systems (include sensors, filters, signal conditioning, data acquisition hardware, and software applications) for the observed images. As denoted in previous chapter, a commonly used approximation in most real HSI is the LR model Bioucas-Dias et al. (2012). Then, if n vectors of dimension N_λ lie in a subspace of dimension $N_r \ll N_\lambda$, each N_λ-long vector has only N_r degrees of freedom. In order to get significant reduction in complexity, (24) can be approximated as

$$\boldsymbol{Y} = \boldsymbol{\Phi}_{\text{2d}}\boldsymbol{C}\boldsymbol{Q} + \boldsymbol{H}, \tag{25}$$

where $\boldsymbol{C} \in \mathbb{R}^{n \times N_r}$ is the feature matrix (the columns of $\boldsymbol{C}$ are known as feature maps), $\boldsymbol{Q} \in \mathbb{R}^{N_r \times N_\lambda}$ is an unknown subspace basis and N_r is the numerical rank of $\boldsymbol{Z}$. The matrix $\boldsymbol{H}$ is additive term that include both modeling errors and sensor noises.

2.2.1. Constrained optimization.

The estimation of the matrices $\boldsymbol{C}$ and $\boldsymbol{Q}$ from the measurements $\boldsymbol{Y}$ in (25) can be done from a Bayesian point of view. In this work, the matrix $\boldsymbol{Q}$ belongs the set $\mathscr{Q} = \{\boldsymbol{Q} \in \mathbb{R}^{N_r \times N_\lambda} \mid \boldsymbol{Q}\boldsymbol{Q}^\mathsf{T} = \boldsymbol{I}_{N_r}\}$. Using the statistical properties of the noise matrices $\boldsymbol{H}$ and $\boldsymbol{Y}$, have matrix Gaussian distributions, i.e.,

$$p(\boldsymbol{Y}|\boldsymbol{C},\boldsymbol{Q}) = \mathscr{M}\mathscr{N}_{n \times N_\lambda}\left(\boldsymbol{\Phi}_{\text{2D}}\boldsymbol{C}\boldsymbol{Q},\ \sigma^2\boldsymbol{I}_n,\ \boldsymbol{I}_{N_\lambda}\right),$$

with variance σ^2. The posterior of $\boldsymbol{C}$ and $\boldsymbol{Q}$ is given by

$$p(\boldsymbol{C},\boldsymbol{Q}|\boldsymbol{Y}) \propto p(\boldsymbol{Y}|\boldsymbol{C},\boldsymbol{Q})p(\boldsymbol{C})p(\boldsymbol{Q}). \tag{26}$$

One can then obtain stable reconstructed features by computing the maximum of the posterior density, i.e. the Maximum a Posteriori (MAP). The constraints for the matrix $\boldsymbol{Q}$ admit a uniform distribution on the feasible region $\mathscr{Q}$ Wen and Yin (2013) such that

$$p(\boldsymbol{Q}) = \begin{cases} 1/\text{vol}(\mathscr{Q}), & \text{if } \boldsymbol{Q} \in \mathscr{Q} \\ 0 & \text{elsewhere} \end{cases},$$

where $\text{vol}(\mathscr{Q}) = \int_{\boldsymbol{Q}\in\mathscr{Q}} d\boldsymbol{Q}$ is the volume of the set $\mathscr{Q}$. Then, the MAP of Equation (26) is defined as the mode of the posterior distribution given by:

$$\begin{aligned} \max_{\boldsymbol{C},\boldsymbol{Q}} p(\boldsymbol{C},\boldsymbol{Q}|\boldsymbol{Y}) &= \max_{\boldsymbol{C},\boldsymbol{Q}} p(\boldsymbol{Y}|\boldsymbol{C},\boldsymbol{Q})p(\boldsymbol{C})p(\boldsymbol{Q}) \\ &= \max_{\boldsymbol{C},\boldsymbol{Q}} \begin{cases} p(\boldsymbol{Y}|\boldsymbol{C},\boldsymbol{Q})p(\boldsymbol{C}) & \text{if } \boldsymbol{Q} \in \mathscr{Q} \\ 0 & \text{elsewhere} \end{cases} \\ &= \max_{\boldsymbol{C},\boldsymbol{Q}\in\mathscr{Q}} p(\boldsymbol{Y}|\boldsymbol{C},\boldsymbol{Q})p(\boldsymbol{C}). \end{aligned}$$

On the other hand, the prior $\boldsymbol{C}$ is obtained by assuming that the increment, are independently and identically distributed following a Laplacian distribution Bardsley (2012). The prior of $\boldsymbol{C}$ is defined as

$$p(\boldsymbol{C}) \propto \exp\left(-\lambda_c \sum_{i=1}^{N_r} \|\boldsymbol{D}\boldsymbol{C}_{(:,i)}\|_1\right), \tag{27}$$

where $\boldsymbol{D} = [\boldsymbol{D}_\mathrm{v};\ \boldsymbol{D}_\mathrm{h}]$ is the TV operator, and the matrices $\boldsymbol{D}_\mathrm{v}$ and $\boldsymbol{D}_\mathrm{h}$ are the 2D discretized vertical and horizontal derivatives, respectively, and $\lambda_c > 0$ is the shape parameter of the distribution. Calculation of the matrix operators for the vertical and horizontal differences to apply on a vectorized image can be defined as follow. Assume that we have an $N_x \times N_y$ image $\boldsymbol{X}$. Now, apply a vertical difference matrix on $\boldsymbol{X}$, i.e., $\boldsymbol{D}_x\boldsymbol{X}$, where $\boldsymbol{D}_x$ is an $N_x \times N_x$ matrix given by

$$\boldsymbol{D}_x = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & -1 \\ -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix},$$

where $\boldsymbol{D}_x$ is the circular convolution matrix of the kernel $\boldsymbol{k} = [1, -1]$. Now vectorize $\boldsymbol{D}_x\boldsymbol{X}$, i.e.,

$$\mathrm{vec}(\boldsymbol{D}_x\boldsymbol{X}) = \left(\boldsymbol{I}_{N_y} \otimes \boldsymbol{D}_x^\mathsf{T}\right)\mathrm{vec}(\boldsymbol{X}) = \boldsymbol{D}_\mathrm{v}\boldsymbol{x},$$

where $\boldsymbol{x}$ is the vectorized image of length $n = N_xN_y$. This shows that $\boldsymbol{D}_\mathrm{v}\boldsymbol{x}$ contains a vertical difference of an image $\boldsymbol{X}$. Moreover, with a similar argument, $\boldsymbol{D}_\mathrm{h}\boldsymbol{x}$ contains a horizontal difference of an image $\boldsymbol{X}$. By taking the negative logarithm of $p(\boldsymbol{C},\boldsymbol{Q}|\boldsymbol{Y})$ in (26), i.e.,

$$\min_{\boldsymbol{C},\boldsymbol{Q}\in\mathscr{Q}} -\log\left(p(\boldsymbol{Y}|\boldsymbol{C},\boldsymbol{Q})\right)-\log(p(\boldsymbol{C})),$$

the MAP estimator of $(\boldsymbol{C},\boldsymbol{Q})$ can be obtained by solving a constrained optimization problem as

$$\min_{\boldsymbol{C},\boldsymbol{Q}} \quad J(\boldsymbol{C},\boldsymbol{Q}) = f(\boldsymbol{C},\boldsymbol{Q})+g(\boldsymbol{C}) \quad \text{s.t.} \quad \boldsymbol{Q}\boldsymbol{Q}^{\mathsf{T}} = \boldsymbol{I}_{N_r}, \tag{28}$$

where

$$f(\boldsymbol{C},\boldsymbol{Q}) = \frac{1}{2}\|\boldsymbol{Y}-\boldsymbol{\Phi}_{2\mathrm{D}}\boldsymbol{C}\boldsymbol{Q}\|_{\mathsf{F}}^2, \quad g(\boldsymbol{C}) = \lambda\sum_{i=1}^{N_r}\|\boldsymbol{D}\boldsymbol{C}_{(i,:)}\|_1,$$

$\lambda = \lambda_c\sigma^2$, $\lambda > 0$ is a regularization parameter, and $\|\cdot\|_{\mathsf{F}}$ is the Frobenius norm. The vector $\boldsymbol{C}_{(i,:)} \in \mathbb{R}^n$ represents the i-th feature. In the problem (28), the ℓ_1-norm on the differences between adjacent pixels (TV operator) offers some desirable properties. First, it encourages sparsity of the coefficients and also sparsity of their differences, which is very popular in CS recovery techniques. Second, it ensures some spatial regularity and preserves the edges, which are boundaries of objects that are used to obtain a preliminary classification procedure. Therefore, TV combined with the ℓ_1-norm has a strong geometrical meaning that makes it useful for feature selection and denoising of shapes Bardsley (2012). Notice that, the problem (28) is non-convex with respect to $\boldsymbol{C}$ and $\boldsymbol{Q}$, simultaneously. However, the sub-problem of $\boldsymbol{C}$ is a generalized lasso optimization problem Boyd et al. (2011), which is convex. While the sub-problem of $\boldsymbol{Q}$ minimizes a quadratic function with orthogonality constraint Wen and Yin (2013), which is non-convex. Therefore, each sub-problem can be efficiently solved via ADMM.

2.2.2. Subspace preservation. Because the problem (28) is nonconvex with respect to both variables, the initialization drastically impacts the results. Note that the matrix $\boldsymbol{\Phi}$ preserves the subspace matrix of $\boldsymbol{Z}$, with high probability after projection Boutsidis and Gittens (2013). Then, for initializing the solution of (28), the eigenvectors of the matrix $\boldsymbol{Y}^\top\boldsymbol{Y}$ are used since they are close to the eigenvectors of the matrix $\boldsymbol{Z}^\top\boldsymbol{Z}$. Let $\boldsymbol{Z} = \boldsymbol{USV}^\top$ be the Singular Value Decomposition (SVD). Note that

$$\begin{aligned}\boldsymbol{Y}^\top\boldsymbol{Y} &= \boldsymbol{VSU}^\top\left(\boldsymbol{\Phi}_{2\text{D}}^\top\boldsymbol{\Phi}_{2\text{D}}\right)\boldsymbol{USV}^\top \\ &= \boldsymbol{VSU}^\top\left(\boldsymbol{I}_n + \boldsymbol{\Phi}_{2\text{D}}^\top\boldsymbol{\Phi}_{2\text{D}} - \boldsymbol{I}_n\right)\boldsymbol{USV}^\top \\ &= \boldsymbol{VS}^2\boldsymbol{V}^\top + \boldsymbol{VSU}^\top\left(\boldsymbol{\Phi}_{2\text{D}}^\top\boldsymbol{\Phi}_{2\text{D}} - \boldsymbol{I}_n\right)\boldsymbol{USV}^\top,\end{aligned} \tag{29}$$

where the first term $\boldsymbol{VS}^2\boldsymbol{V}^\top$ is the eigen-decomposition of the matrix $\boldsymbol{Z}^\top\boldsymbol{Z}$ and the second term $\boldsymbol{VSU}^\top(\boldsymbol{\Phi}_{2\text{D}}^\top\boldsymbol{\Phi}_{2\text{D}} - \boldsymbol{I}_n)\boldsymbol{USV}^\top$ is the perturbation matrix. Then, when the second term is small in some sense, it would be reasonable to expect $\boldsymbol{Y}^\top\boldsymbol{Y}$ to have approximately the same spectral information of $\boldsymbol{Z}^\top\boldsymbol{Z}$. From equation (29), the symmetric matrix $\boldsymbol{Y}^\top\boldsymbol{Y}$ can be modeled as the summation of some original symmetric matrix $\boldsymbol{Z}^\top\boldsymbol{Z}$ and a perturbation matrix $\boldsymbol{E}$, such that,

$$\boldsymbol{Y}^\top\boldsymbol{Y} = \boldsymbol{Z}^\top\boldsymbol{Z} + \boldsymbol{E}, \quad \text{where } \boldsymbol{E} = \boldsymbol{Z}^\top(\boldsymbol{\Phi}_{2\text{D}}^\top\boldsymbol{\Phi}_{2\text{D}} - \boldsymbol{I}_n)\boldsymbol{Z}. \tag{30}$$

Since $\boldsymbol{\Phi}_{2\text{D}} = \boldsymbol{RH}_h\boldsymbol{P}$ (details about $\boldsymbol{R}, \boldsymbol{H}_h$ and $\boldsymbol{P}$ are reported in Section 2.1), each element of the matrix $\boldsymbol{\Phi}_{2\text{D}(i,j)} \in \{-1/\sqrt{m},\ 1/\sqrt{m}\}$ can be approximated as $(2\mathscr{B}(1,0.5)-1)/\sqrt{m}$ where $\mathscr{B}(\cdot)$ is a Binomial distribution. The scale coefficient $\sqrt{m}$ is used to normalize the transformation so that $\mathrm{diag}(\boldsymbol{\Phi}_{2\mathrm{D}}^{\mathsf{T}}\boldsymbol{\Phi}_{2\mathrm{D}}) = \boldsymbol{I}_n$ for any value of m. Note that

- when $m = n$, $\boldsymbol{\Phi}_{2\mathrm{D}}^{\mathsf{T}}\boldsymbol{\Phi}_{2\mathrm{D}} = \boldsymbol{I}_n$ and the entries of $(\boldsymbol{\Phi}_{2\mathrm{D}}^{\mathsf{T}}\boldsymbol{\Phi}_{2\mathrm{D}} - \boldsymbol{I}_n)$ are 0.
- when $m < n$, $\boldsymbol{\Phi}_{2\mathrm{D}}^{\mathsf{T}}\boldsymbol{\Phi}_{2\mathrm{D}} \approx \boldsymbol{I}_n$ and the entries of $\boldsymbol{\Phi}_{2\mathrm{D}}^{\mathsf{T}}\boldsymbol{\Phi}_{2\mathrm{D}} - \boldsymbol{I}_n$ are small.

Also, this expression can be understood as the amount of error between the eigenvectors of $\boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z}$ and those of $\boldsymbol{Y}^{\mathsf{T}}\boldsymbol{Y}$ under the additive perturbation $\boldsymbol{E}$. Let

$$\boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z} = \boldsymbol{V}\boldsymbol{S}^2\boldsymbol{V}^{\mathsf{T}} = \sum_{i=1}^{N_\lambda} \boldsymbol{s}_i^2 \boldsymbol{v}_i \boldsymbol{v}_i^{\mathsf{T}}, \quad \boldsymbol{Y}^{\mathsf{T}}\boldsymbol{Y} = \boldsymbol{V}_y\boldsymbol{S}_y^2\boldsymbol{V}_y^{\mathsf{T}} = \sum_{i=1}^{N_\lambda} (\boldsymbol{s}_y)_i^2 (\boldsymbol{v}_y)_i (\boldsymbol{v}_y)_i^{\mathsf{T}},$$

where $\boldsymbol{v}_i$ and $(\boldsymbol{v}_y)_i$ are the original and perturbed eigenvectors, respectively and $\boldsymbol{s}_i^2$ and $(\boldsymbol{s}_y)_i^2$ are the original and perturbed eigenvalues, respectively. Then, the angle between $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_i, \ldots, \boldsymbol{v}_{N_\lambda}\}$ and $\{(\boldsymbol{v}_y)_1, \ldots, (\boldsymbol{v}_y)_i, \ldots, (\boldsymbol{v}_y)_{N_\lambda}\}$ is bounded by the following theorem.

Theorem 1 [from Davis and Kahan (1970) Theorem V.4.4]: The angle between $\boldsymbol{v}_i$ and $(\boldsymbol{v}_y)_i$ is given by

$$\sin\left(\angle(\boldsymbol{v}_i,\ (\boldsymbol{v}_y)_i)\right) \leq \frac{\|\boldsymbol{E}\|_2}{\mathrm{gap}\left(i,\ \boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z}\right)}, \tag{31}$$

where $\mathrm{gap}\left(i,\ \boldsymbol{Z}^{\mathsf{T}}\boldsymbol{Z}\right) = \min_{l \neq j} |\boldsymbol{s}_i^2 - (\boldsymbol{s}_y)_i^2|$. Another definition of gap is presented in Nakatsukasa (2018). Since we may reverse the sign of $(\boldsymbol{v}_y)_i$, if necessary, there is a choice of orientation of $(\boldsymbol{v}_y)_i$ for which $(\boldsymbol{v}_y)_i^{\mathsf{T}}\boldsymbol{v}_i \geq 0$. For this choice, we can also deduce that $\|(\boldsymbol{v}_y)_i - \boldsymbol{v}_i\|_2 \leq \sqrt{2}\sin\left(\angle(\boldsymbol{v}_i,\ (\boldsymbol{v}_y)_i)\right)$. In order to illustrate this, we have performed some simulations using the Indian Pines dataset. The Fig. 5 shows the error between first three original and estimated eigenvectors with its respective error bound from Equation (31), which shows that the second term in (17) is actually small.

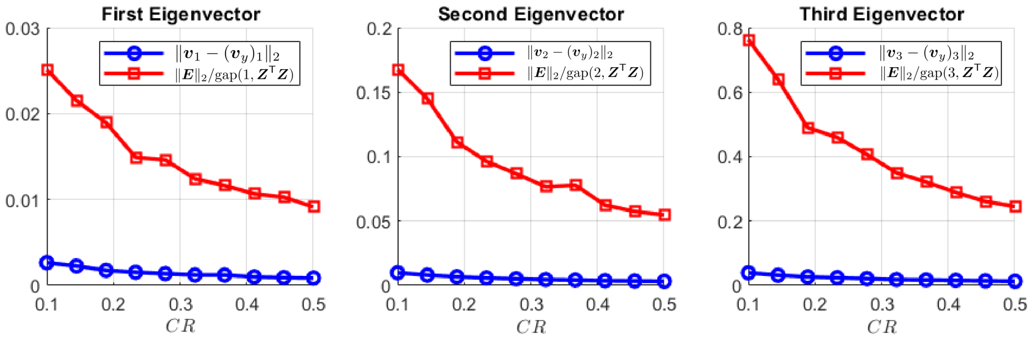


Figure 5. Error between first three original and estimated eigenvectors with its respective error bound.

2.3. Optimization Algorithm

The problem in (28) is solved one matrix at a time, while the other is assumed to be fixed. This procedure is summarized in Algorithm 2.1, where the AO estimator leverages the low-rank constraints by iteratively updating $\boldsymbol{C}$ and $\boldsymbol{Q}$, which has low complexity compared to directly estimating $\boldsymbol{Z}$. To overcome the closed-form expression problem in (28), the ADMM is embedded in each iteration of the AO algorithm.

2.3.1. Optimizing the first variable.

Given a fixed $\boldsymbol{Q}$, the minimization problem in (28) can be solved by introducing auxiliary variables, splitting the objective and the constraints, and using the ADMM method. By introducing the auxiliary variables $\boldsymbol{C}_1 \in \mathbb{R}^{n \times N_r}$ and $\boldsymbol{C}_2 \in \mathbb{R}^{2n \times N_r}$, the optimization problem in (28) with respect to $\boldsymbol{C}$ can be rewritten as:

$$\min_{\boldsymbol{C}_1, \boldsymbol{C}_2} \; f_c(\boldsymbol{C}_1) + g_c(\boldsymbol{C}_2), \quad \text{s.t. } \boldsymbol{D}\boldsymbol{C}_1 - \boldsymbol{C}_2 = \boldsymbol{0}, \tag{32}$$

Algorithm 2.1 Feature extraction based on ℓ_1-ADMM-AO.

```
Input: $\boldsymbol{Y}$, $\boldsymbol{\Phi}_{\text{2D}}$, $N_r$, $\lambda_{\text{tv}}$.
$\boldsymbol{Q}^{(0)} \leftarrow \text{pca}\left(\boldsymbol{Y},\, N_r\right)$ // Initialize $\boldsymbol{Q}$ using Alg.(1.1).
Initialize: $\boldsymbol{C}_2^{(0,0)}$, $\boldsymbol{C}_3^{(0,0)}$
for $t = 1, 2, \ldots$ to stopping rule do
    // Optimize $\boldsymbol{C}$ using ADMM
    for $k = 1, 2, \ldots$ to stopping rule do
        $\boldsymbol{C}_1^{(k)} \in \underset{\boldsymbol{C}_1}{\text{argmin}}\, \mathscr{L}\left(\boldsymbol{C}_1, \boldsymbol{C}_2^{(t-1,k-1)}, \boldsymbol{C}_3^{(t-1,k-1)}\right)$ // (36)
        $\boldsymbol{C}_2^{(t-1,k)} \in \underset{\boldsymbol{C}_2}{\text{argmin}}\, \mathscr{L}\left(\boldsymbol{C}_1^{(k)}, \boldsymbol{C}_2, \boldsymbol{C}_3^{(t-1,k-1)}\right)$
        $\boldsymbol{C}_3^{(t-1,k)} \leftarrow \boldsymbol{C}_3^{(t-1,k-1)} + \boldsymbol{D}\boldsymbol{C}_1^{(k)} - \boldsymbol{C}_2^{(t-1,k)}$
    end for
    $\boldsymbol{C}^{(t)} = \boldsymbol{C}_1^{(k)}$
    // Optimize $\boldsymbol{Q}$ using SVD
    $\left[\boldsymbol{U},\, \boldsymbol{S},\, \boldsymbol{V}^{\top}\right] = \text{svd}\left(\left(\boldsymbol{\Phi}_{\text{2D}}\boldsymbol{C}^{(t)}\right)^{\top}\boldsymbol{Y}\right)$
    $\boldsymbol{Q}^{(t)} = \boldsymbol{U}\boldsymbol{V}^{\top}$
end for
Set $\hat{\boldsymbol{C}} = \boldsymbol{C}^{(t)}$
Output: $\hat{\boldsymbol{C}}$
```

where

$$f(\boldsymbol{C}_1) = \frac{1}{2}\|\boldsymbol{\Phi}_{2\mathrm{D}}\boldsymbol{C}_1\boldsymbol{Q} - \boldsymbol{Y}\|_{\mathsf{F}}^2, \quad g(\boldsymbol{C}_2) = \lambda\sum_{i=1}^{N_r}\|\boldsymbol{C}_{2(:,i)}\|_1.$$

The augmented Lagrangian function is defined as

$$\mathscr{L}(\boldsymbol{C}_1,\boldsymbol{C}_2,\boldsymbol{C}_3) = f(\boldsymbol{C}_1) + g(\boldsymbol{C}_2) + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{C}_1 - \boldsymbol{C}_2 + \boldsymbol{C}_3\|_{\mathsf{F}}^2, \tag{33}$$

where $\boldsymbol{C}_3 \in \mathbb{R}^{2n\times N_r}$ is the scaled dual variable, and $\rho > 0$ is the weighting of the augmented Lagrangian term Boyd et al. (2011). The method to solve $\boldsymbol{C}$ is summarized in Algorithm 2.1, that consists in minimizing $\boldsymbol{C}_1, \boldsymbol{C}_2$ and $\boldsymbol{C}_3$, alternately. More details can be found in Boyd et al. (2011). Forcing the derivative of (33) with respect to $\boldsymbol{C}_1$ to be zero leads to the following linear system

$$\boldsymbol{C}_1 \leftarrow \left(\boldsymbol{\Phi}_{2\mathrm{D}}^{\mathsf{T}}\boldsymbol{\Phi}_{2\mathrm{D}} + \rho\boldsymbol{D}^{\mathsf{T}}\boldsymbol{D}\right)^{-1}\left(\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{Y}\boldsymbol{Q}^{\mathsf{T}} + \rho\boldsymbol{D}^{\mathsf{T}}(\boldsymbol{C}_2 - \boldsymbol{C}_3)\right), \tag{34}$$

where $(\boldsymbol{\Phi}_{2\mathrm{D}}^{\mathsf{T}}\boldsymbol{\Phi}_{2\mathrm{D}} + \rho\boldsymbol{D}^{\mathsf{T}}\boldsymbol{D})$ has dimension $n \times n$. Due to the high dimension, the solution of this system has an extremely heavy computational cost. An alternate strategy can be used to approximately solve the local optimization problem efficiently. Linearized ADMM is a variation of ADMM which is based on approximating the augmented quadratic term to its first order approximation Parikh et al. (2014); Cao and Liu (2018). Let $f_g(\boldsymbol{C}_1) = (\rho/2)\|\boldsymbol{D}\boldsymbol{C}_1 - \boldsymbol{C}_2 + \boldsymbol{C}_3\|_{\mathsf{F}}^2$, and its first order approximation be expressed as

$$f_g(\boldsymbol{C}_1) \approx f_g(\boldsymbol{C}_1^{(t)}) + \left(\nabla f_g(\boldsymbol{C}_1^{(t)})\right)^{\mathsf{T}}\left(\boldsymbol{C}_1 - \boldsymbol{C}_1^{(t)}\right) + \frac{\tau}{2}\left\|\boldsymbol{C}_1 - \boldsymbol{C}_1^{(t)}\right\|_{\mathsf{F}}^2, \tag{35}$$

where $\nabla f_g(\boldsymbol{C}_1^{(t)}) = \rho \boldsymbol{D}^\mathsf{T}(\boldsymbol{D}\boldsymbol{C}_1^{(t)} - \boldsymbol{C}_2 + \boldsymbol{C}_3)$ is the gradient of $f_g(\boldsymbol{C}_1^{(t)})$ at the current point $\boldsymbol{C}_1^{(t)}$ and τ is a positive proximal parameter. Then, the combination of (33) and (35) and forcing its derivative with respect to $\boldsymbol{C}_1$ to be zero leads to the linear system given by

$$\boldsymbol{C}_1 \leftarrow \left(\boldsymbol{\Phi}_{2\mathrm{D}}^\mathsf{T}\boldsymbol{\Phi}_{2\mathrm{D}} + \tau \boldsymbol{I}_n\right)^{-1} \left(\boldsymbol{\Phi}_{2\mathrm{D}}^\mathsf{T}\boldsymbol{Y}\boldsymbol{Q}^\mathsf{T} + \tau \boldsymbol{C}_1^{(t)} - \rho \boldsymbol{D}^\mathsf{T}(\boldsymbol{D}\boldsymbol{C}_1^{(t)} - \boldsymbol{C}_2 + \boldsymbol{C}_3)\right), \tag{36}$$

and decomposing the matrix $\boldsymbol{\Phi}_{2\mathrm{D}}$, the inversion matrix is

$$\begin{aligned}\left(\boldsymbol{\Phi}_{2\mathrm{D}}^\mathsf{T}\boldsymbol{\Phi}_{2\mathrm{D}} + \tau \boldsymbol{I}_n\right)^{-1} &= \left(\boldsymbol{P}^\mathsf{T}\boldsymbol{H}_\mathrm{h}^\mathsf{T}\boldsymbol{R}^\mathsf{T}\boldsymbol{R}\boldsymbol{H}_\mathrm{h}\boldsymbol{P} + \tau \boldsymbol{I}_n\right)^{-1} \\ &= \boldsymbol{P}^\mathsf{T}\boldsymbol{H}_\mathrm{h}^\mathsf{T}(\boldsymbol{R}^\mathsf{T}\boldsymbol{R} + \tau \boldsymbol{I}_n)^{-1}\boldsymbol{H}_\mathrm{h}\boldsymbol{P}\end{aligned}. \tag{37}$$

Note that the matrix $(\boldsymbol{R}^\mathsf{T}\boldsymbol{R} + \eta \boldsymbol{I}_n)^{-1}$ is a diagonal operator, and is thus easily inverted. In general, the requirement on step size τ obeys $0 < \tau \leq \rho/\|\boldsymbol{D}\|_2^2$ (Parikh et al. (2014) Section 4.4.2). This version of the ADMM strategy is advantageous because every substep of the method has a closed-form solution. The optimization problem to solve for $\boldsymbol{C}_2$ is written as

$$\begin{aligned}\boldsymbol{C}_2 &\in \underset{\boldsymbol{C}_2}{\arg\min}\ \lambda \|\boldsymbol{C}_2\|_{1,1} + \frac{\rho}{2}\|\boldsymbol{D}\boldsymbol{C}_1 - \boldsymbol{C}_2 + \boldsymbol{C}_3\|_\mathsf{F}^2 \\ &\leftarrow \mathrm{soft}\left(\boldsymbol{D}\boldsymbol{C}_1 + \boldsymbol{C}_3,\ \lambda/\rho\right)\end{aligned}, \tag{38}$$

where $\mathrm{soft}(\cdot, \lambda)$ denotes the element-wise application of the soft-thresholding operator.

In the linearized ADMM the primal residual is the same as for standard ADMM, however the dual residual changes since the augmented Lagrangian term is linearized. The derivation of the dual residual from Equation (33) is the same as for standard ADMM (Boyd et al., 2011, Section 3.3), however, the term $(\rho/2)\|\boldsymbol{D}\boldsymbol{C}_1-\boldsymbol{C}_2+\boldsymbol{C}_3\|_\mathsf{F}^2$ is here linearized. Denote the optimal variables by $\boldsymbol{C}_1^*$, $\boldsymbol{C}_2^*$, and $\boldsymbol{C}_3^*$. The necessary and sufficient optimality conditions for the ADMM problem in (32) are primal feasibility $\{\boldsymbol{D}\boldsymbol{C}_1^*-\boldsymbol{C}_2^*=\boldsymbol{0}\}$, and dual feasibility $\{0\in\partial f(\boldsymbol{C}_1^*)+\boldsymbol{D}^\top\boldsymbol{C}_3^*, 0\in\partial g(\boldsymbol{C}_2^*)+\boldsymbol{C}_3^*\}$, where $\partial(\cdot)$ denotes the sub-differential of a function. These conditions can be used to derive convergence measures for algorithm iterations $(\boldsymbol{C}_1^{(t)},\boldsymbol{C}_2^{(t)},\boldsymbol{C}_3^{(t)})$. Note that the optimality conditions for the first subproblem (i.e., the subproblem with respect to $\boldsymbol{C}_1$) in (32) are given by

$$0\in\partial f(\boldsymbol{C}_1^{(t)})+\rho\left(\boldsymbol{D}^\top(\boldsymbol{D}\boldsymbol{C}_1^{(t-1)}-\boldsymbol{C}_2^{(t-1)}+\boldsymbol{C}_3^{(t-1)})\right)+\tau(\boldsymbol{C}_1^{(t)}-\boldsymbol{C}_1^{(t-1)}).$$

By using $\boldsymbol{C}_3^{(t)}=\mathrm{C}_3^{(t-1)}+\boldsymbol{D}\boldsymbol{C}_1^{(t)}-\boldsymbol{C}_2^{(t)}$, then

$$0\in\partial f(\boldsymbol{C}_1^{(t)})+\rho\boldsymbol{D}^\top\boldsymbol{C}_3^{(t)}+\tau(\boldsymbol{C}_1^{(t)}-\mathrm{C}_2^{(t-1)})-\rho\boldsymbol{D}^\top\boldsymbol{D}(\boldsymbol{C}_1^{(t)}-\boldsymbol{C}_1^{(t-1)})+\rho\boldsymbol{D}^T(\boldsymbol{C}_2^{(t)}-\boldsymbol{C}_2^{(t-1)}).$$

This means that the quantity

$$\boldsymbol{S}^{(t)}=\rho\boldsymbol{D}^T(\boldsymbol{C}_2^{(t)}-\boldsymbol{C}_2^{(t-1)})+\tau(\boldsymbol{C}_1^{(t)}-\boldsymbol{C}_1^{(t-1)})-\rho\boldsymbol{D}^T\boldsymbol{D}(\boldsymbol{C}_1^{(t)}-\boldsymbol{C}_1^{(t-1)}). \tag{39}$$

2.3.2. Optimizing the second variable. The solution for $\boldsymbol{Q}$ is summarized in Algorithm 2.1, aiming at a more computationally efficient method for solving (28) with respect to $\boldsymbol{Q}$.

By denoting first the indicator function of $\boldsymbol{Q}$ of the set $\mathscr{Q}$ as:

$$i_{\mathscr{Q}}(\boldsymbol{Q}) = \begin{cases} 0 & \text{if } \boldsymbol{Q} \in \mathscr{Q} \\ +\infty & \text{otherwise} \end{cases},$$

and $\boldsymbol{C}$ as an constant. The following constrained optimization problem is solved by computing a low-rank Procrustes rotation (Zou et al., 2006, Theorem 4)

$$\begin{aligned} \boldsymbol{Q} &\in \underset{\boldsymbol{Q}}{\arg\min}\ i_{\mathscr{Q}}(\boldsymbol{Q}) + \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{\Phi}_{\text{2D}}\boldsymbol{C}\boldsymbol{Q}\|_{\mathsf{F}}^2 \\ &\leftarrow \boldsymbol{U}\boldsymbol{V}^{\mathsf{T}}, \end{aligned} \tag{40}$$

where $\boldsymbol{U}$, $\boldsymbol{S}$, $\boldsymbol{V}$ is the SVD of $(\boldsymbol{\Phi}_{\text{2D}}\boldsymbol{C}^{(t)})^{\mathsf{T}}\boldsymbol{Y}$ with $\boldsymbol{U} \in \mathbb{R}^{N_r \times N_r}$, $\boldsymbol{V} \in \mathbb{R}^{N_\lambda \times N_r}$ being orthogonal basis and $\boldsymbol{S} \in \mathbb{R}^{N_r \times N_r}$ a diagonal matrix. This problem has been considered in applications such as linear and nonlinear eigenvalue problems Wen and Yin (2013); Lai and Osher (2014).

2.3.3. Convergence and computational complexity analysis. Consider the non-convex optimization problem in (28) with variables separated into 2 blocks. The proposed AO method cyclically updates $\boldsymbol{C}$ and $\boldsymbol{Q}$ via Algorithm 1. To analyze the convergence of the proposed AO method, this work recalls the convergence criterion for the Block Coordinate Descent (BCD) algorithm stated in Bertsekas (1999).

Theorem 2 (Bertsekas, 1999, Proposition 2.7.1): *Suppose that J is continuously differentiable over the set $\mathscr{Q}$. Suppose also that for each $\{\boldsymbol{C}, \boldsymbol{Q}\}$, $J(\boldsymbol{C}, \boldsymbol{Q})$ viewed as a function of $\boldsymbol{C}$ attains a unique minimum. The similar uniqueness also holds for $\boldsymbol{Q}$. Let $\{\boldsymbol{C}^{(t)}, \boldsymbol{Q}^{(t)}\}$ be the sequence* *generated by the BCD method, then, every limit point of* $\{\boldsymbol{C}^{(t)}, \boldsymbol{Q}^{(t)}\}$ *is a stationary point.*

Note that the generated sequence will monotonically decrease the objective function. If some conditions are satisfied, strong convergence will be obtained (Bertsekas, 1999, Proposition 2.7.1). For example, if each sub-problem in (28) is convex and has a unique solution, then every limit point is a stationary point. The uniqueness of the solution is not required when the number of blocks is two Grippo and Sciandrone (2000). However, the minimization with respect to $\boldsymbol{Q}$ is a non-convex problem, and there is not guarantee that the AO method can reach the global solution of (28). Non-convex problems can get stuck at local solutions by using BCD-based methods. A simple modification of the objective function, consisting in removing the condition $\boldsymbol{Q} \in \mathscr{Q}$ and adding the quadratic term $\lambda_q \|\boldsymbol{Q}\boldsymbol{Q}^{\mathsf{T}} - \boldsymbol{I}_{N_r}\|_{\mathsf{F}}^2$, where $\lambda_q > 0$ is very small, obtaining a convex objective function, enables the use of Theorem 1. Even without including the quadratic term, the convergence of Algorithm 1 can be observed in practice. Furthermore, if $\mathscr{Q}$ is compact, which implies that the sequence generated is bounded, the BCD method is guaranteed to converge to a stationary point Tseng (2001). Then, the stopping criteria in Algorithms 2.1 for optimizing $\boldsymbol{C}$ is the power of primal and dual residual (Boyd et al., 2011, page 19), the value of $\rho = 1$ and $\varepsilon = 10^{-4}$. The stopping rule for Algorithm 2.1 is

$$\frac{\left|J(\boldsymbol{C}^{(t)}, \boldsymbol{Q}^{(t)}) - J(\boldsymbol{C}^{(t-1)}, \boldsymbol{Q}^{(t-1)})\right|}{\left|J(\boldsymbol{C}^{(t)}, \boldsymbol{Q}^{(t)})\right|} \leq 10^{-4},$$

or $t = \lfloor N_\lambda / N_r \rfloor - 1$, where $\lfloor \cdot \rfloor$ is the floor operator. The condition $t = \lfloor N_\lambda / N_r \rfloor - 1$ guarantees that the complexity of Algorithm 1 never reaches the complexity of the traditional reconstruction methods.

In Algorithm 2.1, the computational complexity of the optimization of $\boldsymbol{C}_1$ is $\mathscr{O}(nN_r\log(n))$, and the computational complexity of the optimization of $\boldsymbol{C}_2$ is $\mathscr{O}(n)$. The overall complexity per iteration for solving $\boldsymbol{C}$ in Algorithm 2.1 is given by $\mathscr{O}(nN_r\log(n))$. It should be noted that, in real images, N_λ is very likely to be larger than N_r. Thus, the overall complexity per iteration for solving $\boldsymbol{Q}$ in Algorithm 2.1 is given by $\mathscr{O}(N_r^2N_\lambda)$. Therefore, Algorithm 2.1 has computational complexity $\mathscr{O}(nN_r\log(n))+\mathscr{O}(N_r^2N_\lambda)$.

2.4. Numerical Experiments

This section presents numerical results on the proposed method for supervised classification of every pixel using real datasets. The proposed scheme was implemented in Matlab and all numerical experiments were performed on a computer with an Intel(R) Core(TM) i7 – 4790 CPU@3.60GHz and 32 GB RAM. All experiments follow the data acquisition model given in (24). For each fixed set of parameters (Compression Ratio (CR), N_r, λ, *noise*), the averaged results of 10 realizations of the sensing matrix $\boldsymbol{Y}$ are shown. In the noisy case, for each generated sensing matrix $\boldsymbol{Y}$, the averaged results of 10 independent noise realizations are performed. The classifier used here is the Nearest Neighbor Search (NNS) and it can be written as

$$\text{class}(\boldsymbol{c}_i) = \underset{j=1,\ldots,N_c}{\arg\min}\, \|\boldsymbol{c}_i - \boldsymbol{\theta}_{j,k}\|_2^2, \quad k = 1,\ldots,N_t,$$

where $\boldsymbol{\theta}_{j,k}$ is the k-th training sample belonging to the j-th class, $\boldsymbol{c}_i \in \mathbb{R}^{N_r}$ is the i-th row of $\boldsymbol{C}$. As a summary, the Table 2 shows all matrix dimensions and meaning of the symbols used. In each classification experiment, N_t samples were randomly chosen per class, as training samples and the remaining samples were used for testing. For Indian Pines database, five training pixels of the classes Alfalfa, Grass-pasture-mowed, Oats and Stone-Steel-Towers were chosen randomly.

Table 2
Matrix dimension summary.

Variable	**Description**
n,N_λ	Number of pixels and spectral dimension of the data cube
N_r	Number of features
N_t	Number of training samples per class
N_c	Number of classes

2.4.1. Tuning Parameter Selection. This subsection, explores the effect of selecting the parameter λ and the number of features R on the performance of the proposed method in terms of OA obtained by applying the NNS classifier on the reconstructed features. The experiments are carried out by selecting different numbers of training samples per class, N_t = 5, 25 and 50 and different $CR = 0.05, 0.1, 0.15, 0.2, 0.25$ and 0.3. Figure 6 depicts the behavior of OA for Indian Pines when $0.01 \leq \lambda \leq 0.1$ and the feature number is $5 \leq N_r \leq 25$. It can be seen in Fig. 6 that, for $CR \geq 0.1$, the difference between OAs is not larger than 0.1. Furthermore, the selection of parameter λ offers the potential to improve the performance of OA obtained by the NNS classifier. Note also in Figure 6 that the proposed method produces similar results when the parameter $R > 20$. The selection of the parameter N_r can be done from eigen-decomposition of the matrix $\boldsymbol{Y}^\top\boldsymbol{Y}$ by selecting the number of most representative eigenvalues. Also, note that when, the parameter $N_t < 50$, the OA does not exceed 0.9 for the NNS classifier. In the next experiments the parameters N_r, N_t and λ are fixed to $\lambda = 0.01$, $N_r = 20$ and $N_t = 50$.

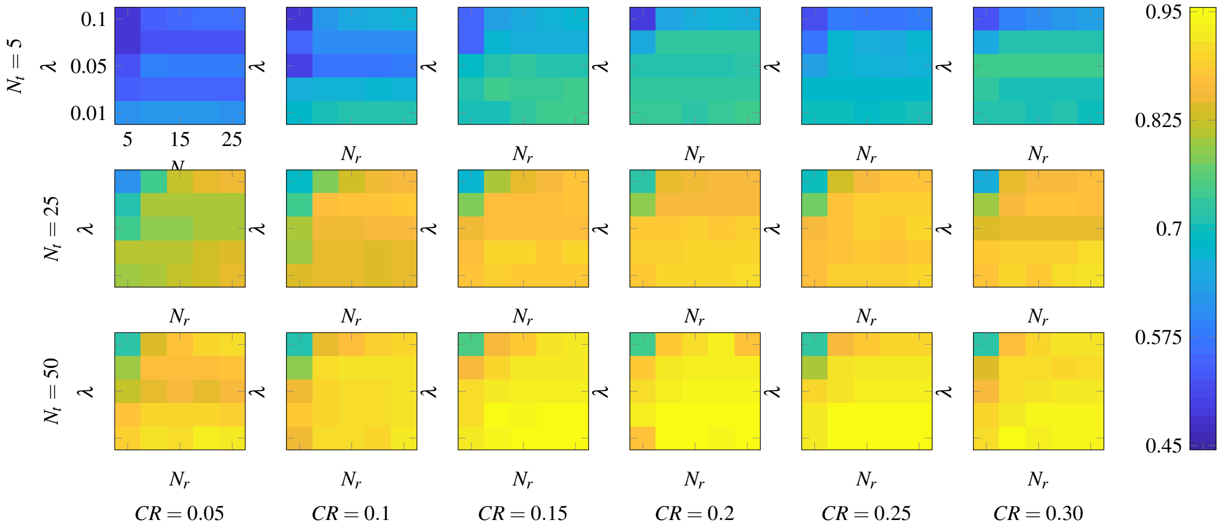


Figure 6. Parameter sensitivity analysis in terms of OA for the NNS classifier on the Indian Pines data base. Different values of CR, number of training samples N_t are used while varying $0.01 \leq \lambda \leq 0.1$ and $5 \leq N_r \leq 25$.

2.4.2. Initialization. There is a need for good initializations for LR matrix factorization given that it is a non-convex problem which has many local minima. A good initialization can improve the speed and accuracy of the algorithms. To illustrate this point, the proposed algorithm is tested in terms of OA and AA when $\boldsymbol{Q}$ is initialized by a random orthogonal matrix labeled as "*Rand*" and when it is initialized by Algorithm 1.1 labeled as "*Prop*". The proposed method is applied on Indian Pines when $CR = 0.1, 0.2$, $\lambda = 0.01$, $N_r = 20$, $N_t = 50$. The extracted features are classified by using the NNS classifier for each iteration of Algorithm 2.1. Figure 7 compares the classification performance when the algorithm is initialized using both approaches cases. It can be seen that initializing with the eigenvectors yields to higher OA and that a random initialization generally leads to a lower value of OA. Therefore, in the next experiments, $\boldsymbol{Q}$ is initialized using

Algorithm 2.1.

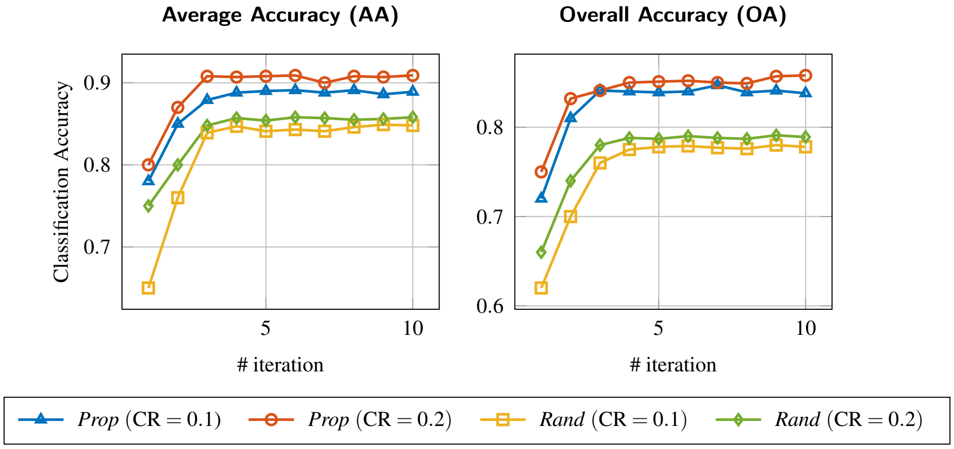


Figure 7. Comparison of the method when it is initialized by using *Prop* with $CR = 0.1$, $CR = 0.2$ and a random orthogonal matrix *Rand* with $CR = 0.1$, $CR = 0.2$ in terms of AA and OA.

2.4.3. Spatial Feature Analysis. When the spatial regularization is used, the resulting reconstructed features are approximately piecewise constant because it promotes piecewise smoothness (homogeneous spatial regions) on the extracted features. In order to show that the proposed method preserves useful information, Figure 8 displays the first three extracted features for both datasets, specifically, false color of the first three spectral features (first row), edge information extracted from first three spectral features (second row) and mean operation on the pixels in each bounded region (third row) with parameters (a) $CR = 1$, $\lambda = 0.001$ (b) $CR = 1$, $\lambda = 0.05$, (c) $CR = 0.2$, $\lambda = 0.05$, (d) $CR = 0.1$, $\lambda = 0.05$, (e) $CR = 1$, $\lambda = 0.001$ (f) $CR = 1$, $\lambda = 0.05$, (g) $CR = 0.2$, $\lambda = 0.05$ and (h) $CR = 0.1$, $\lambda = 0.05$. Note that the extracted features are partitioned into spatial bounded regions which are very similar to the ground truth classification map. Each bounded region belongs to one single structure in the original image, as can be seen in Figure 8 the smallest structures are removed by varying the parameter λ such that only the main structures

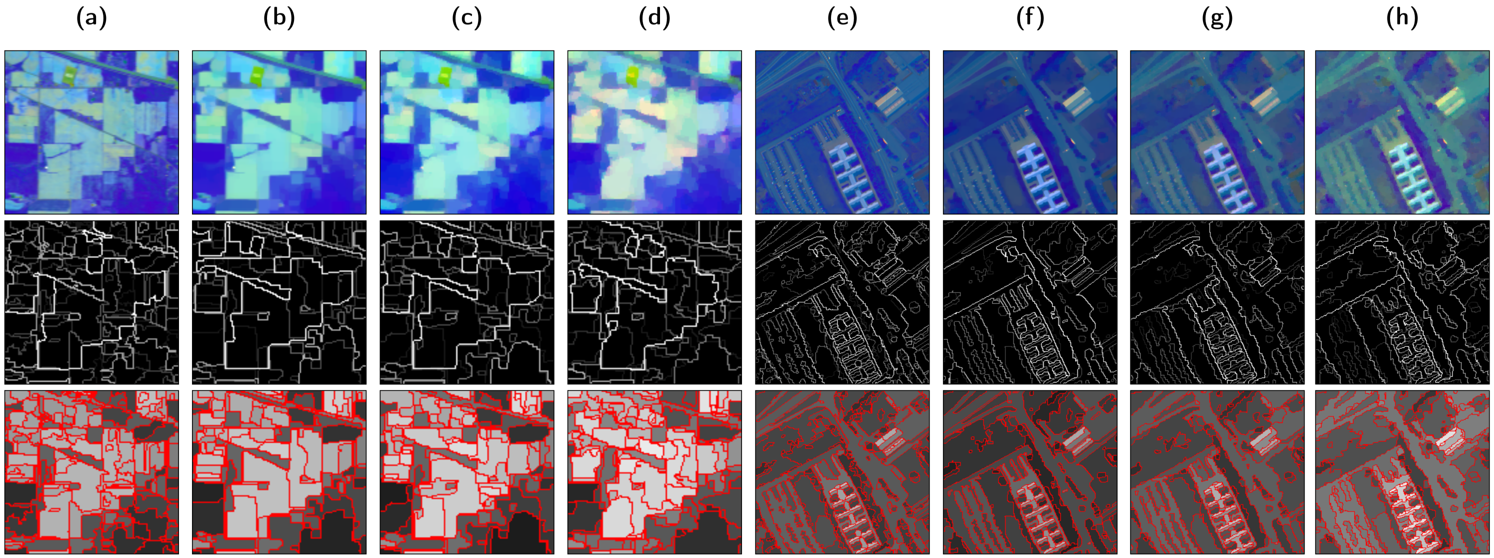


Figure 8. First three extracted features using the proposed method on Indian Pines and Pavia University datasets. **1st row:** False color of the first three spectral features. **2nd row:** Edge information extracted from the first three spectral features. **3rd row:** Mean operation on the pixels in each bounded region.

of interest remain. Furthermore, by visually comparing the features, one can observe that spatial variations of regions are preserved enough for good classification above of $CR = 0.2$. Therefore, the ℓ_1-norm on the differences between adjacent pixels shows better performance in recovering not only edge features but also class structures.

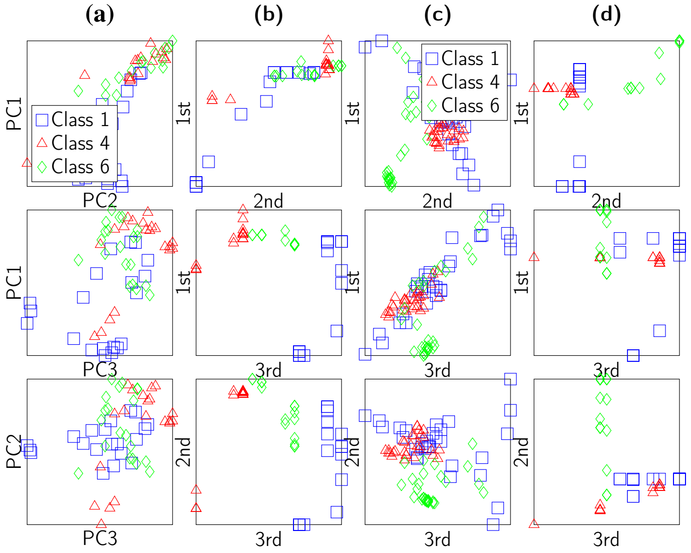


Figure 9. Feature plots of three classes for both datasets. (a) PCA on Indian Pines, (b) *Prop* on Indian Pines with $CR = 0.25$, $\lambda = 0.05$, (c) PCA on Pavia University and (d) *Prop* on Pavia University with $CR = 0.25$, $\lambda = 0.05$.

Additionally, the proposed feature extraction method is compared with PCA in a typical analysis scenario. Particularly, the ability to recover multi-class structure is verified for both Indian Pines and Pavia University datasets. Figure 9 plots 30 samples for each class of the first spectral feature using (a),(c) PCA and (b),(d) the proposed method. The compressed extracted features were estimated with parameters $CR = 0.25$ and $\lambda = 0.05$. In a multi-class scenario, PCA is inundated by many of features when using three classes and cannot display good class structure. In contrast, the proposed method reduces the uncertainty and clearly shows better class structures.

2.4.4. Comparison with other reconstruction methods. In this section we test the robustness of the proposed method labeled as "*Prop*" against the following approaches where the hyperspectral matrix is reconstructed and then it is classified by using the first 20 principal components. By rewriting the acquisition model in (24) as

$$\begin{aligned} \boldsymbol{y} &= (\boldsymbol{I}_{N_\lambda} \otimes \boldsymbol{\Phi}_{\mathrm{2D}})\mathrm{vec}(\boldsymbol{Z}) + \boldsymbol{\eta} \\ &= \boldsymbol{\Phi}\mathrm{vec}(\boldsymbol{Z}) + \boldsymbol{\eta}, \end{aligned}$$

where $\boldsymbol{y} = \mathrm{vec}(\boldsymbol{Y})$, and $\boldsymbol{\eta} = \mathrm{vec}(\boldsymbol{H})$. Two different sparsifying approaches are considered in the reconstruction. The first approach, “*Reco-3DWD*" estimates the data cube by solving the MAP synthesis approach (11) with $\boldsymbol{\Psi} = \boldsymbol{\Psi}_{\mathrm{1D}} \otimes \boldsymbol{\Psi}_{\mathrm{2D}}^{\mathsf{T}}$, where $\boldsymbol{\Psi}_{\mathrm{2D}}$ is the basis formed by a 2-D symmlet wavelet basis and $\boldsymbol{\Psi}_{\mathrm{1D}}$ a 1-D discrete cosine basis Arguello and Arce (2014). This problem can be solved by using Beck and Teboulle (2009) with regularization parameter given by $\lambda = \gamma||(\boldsymbol{\Phi}\boldsymbol{\Psi})^{\mathsf{T}}\boldsymbol{y}||_\infty$, where γ was selected by cross validation over different values between $(0,1)$. The second approach, “*Reco-3DTV*" employs a TV regularization and obtains the data cube by solving the MAP analisys approach (12). This problem can be solved using a Linearized ADMM-based approach Ouyang et al. (2015). Recall that, the problem is formulated as:

$$\min_{\boldsymbol{Z}_1,\boldsymbol{z}_2} \quad f(\boldsymbol{Z}_1) + g(\boldsymbol{z}_2) \quad \text{s.t.} \quad \boldsymbol{D}\mathrm{vec}(\boldsymbol{Z}_1) - \boldsymbol{z}_2 = \boldsymbol{0} \tag{41}$$

where

$$f(\boldsymbol{Z}_1) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\text{vec}(\boldsymbol{Z}_1)\|_2^2, \quad g(\boldsymbol{z}_2) = \lambda\|\boldsymbol{z}_2\|_1 \quad \boldsymbol{D} = \begin{bmatrix} \boldsymbol{I}_{N_\lambda} \otimes \boldsymbol{D}_{\text{h}} \\ \\ \boldsymbol{I}_{N_\lambda} \otimes \boldsymbol{D}_{\text{v}} \\ \\ (\boldsymbol{D}_\lambda)^{\mathsf{T}} \otimes \boldsymbol{I}_n \end{bmatrix}, \quad \boldsymbol{z}_2 = \begin{bmatrix} \boldsymbol{z}_{2,1} \\ \\ \boldsymbol{z}_{2,2} \\ \\ \boldsymbol{z}_{2,3} \end{bmatrix},$$

and $\boldsymbol{z}_{2,1}, \boldsymbol{z}_{2,2}, \boldsymbol{z}_{2,3} \in \mathbb{R}^{nN_\lambda}$ are auxiliary variables. The augmented Lagrangian associated to the optimization can be written as

$$\mathscr{L}(\boldsymbol{Z}_1, \boldsymbol{z}_2, \boldsymbol{z}_3) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{\Phi}\text{vec}(\boldsymbol{Z}_1)\|_2^2 + \lambda\|\boldsymbol{z}_2\|_1 + \frac{\rho}{2}\|\boldsymbol{D}\text{vec}(\boldsymbol{Z}_1) - \boldsymbol{z}_2 + \boldsymbol{z}_3\|_2^2, \tag{42}$$

where $\boldsymbol{z}_3$ is the scaled dual variable, and $\rho > 0$ is the weighting the of augmented Lagrangian term. Note that, the first step is the most expensive, which requires the solution of a quadratic problem. Computing the inverse or pseudoinverse at each iteration is too expensive to implement numerically. In Ouyang et al. (2015), the term $(\rho/2)\|\boldsymbol{D}\text{vec}(\boldsymbol{Z}_1) - \boldsymbol{z}_2 + \boldsymbol{z}_3\|_2^2$ is linearized to solve the quadratic problem iteratively. Each iteration of the linearized method demands updating a linearized parameter. The complexity of this methods is given by four matrix-vector multiplications on computing $\boldsymbol{\Phi}^{\mathsf{T}}\boldsymbol{\Phi}\text{vec}(\boldsymbol{Z})$ and $\boldsymbol{D}^{\mathsf{T}}\boldsymbol{D}\text{vec}(\boldsymbol{Z})$. Since the product of the matrix $\boldsymbol{\Phi}$ by any vector has computational complexity $\mathscr{O}(nN_\lambda \log(nN_\lambda))$ and $\boldsymbol{D}$ can be computed using the Two Dimensional Fast Fourier Transform which leads $\mathscr{O}(nN_\lambda \log(nN_\lambda))$. Then, the overall cost per iteration is $\mathscr{O}(nN_\lambda \log(nN_\lambda))$.

Table 3
Comparison of the computational complexity.

Method	Reconstruction	Classification	Storage
Reco-3DWD	$\mathcal{O}(nN_\lambda \log(nN_\lambda))$	$\mathcal{O}(nN_tN_cN_\lambda)$	$\mathcal{O}(nN_\lambda)$
Reco-3DTV	$\mathcal{O}(nN_\lambda \log(nN_\lambda))$	$\mathcal{O}(nN_tN_cN_\lambda)$	$\mathcal{O}(nN_\lambda)$
Prop	#iter $\times$ $(\mathcal{O}(N_r^2N_\lambda)+\mathcal{O}(nN_r\log(n)))$	$\mathcal{O}(nN_tN_cN_r)$	$\mathcal{O}(nN_r)+\mathcal{O}(N_rN_\lambda)$

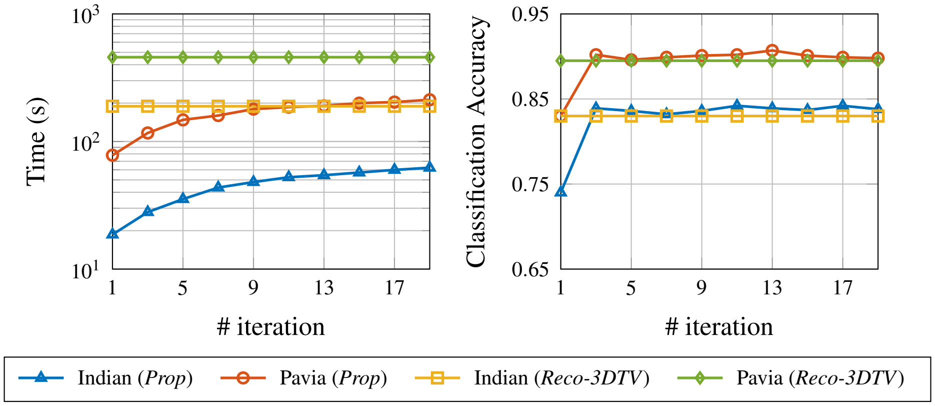


Figure 10. Comparison of *Prop* using Indian and Pavia against *Reco-3DTV* using Indian and Pavia in terms of Time (left) and Classification Accuracy (right).

Table 3 summarizes the computational complexity comparison for *Prop*, *Reco-3DWD* and *Reco-3DTV*. The NNS has a computational complexity $\mathcal{O}(nN_tN_cN_\lambda)$ for the reconstruction method and $\mathcal{O}(nN_tN_cN_r)$ for the proposed method. The computation cost per iteration of *Reco-3DWD* and *Reco-3DTV* are given by $\mathcal{O}(nN_\lambda \log(nN_\lambda))$, because the matrices $\boldsymbol{\Psi}$ and $\boldsymbol{D}$ can be attained via the Fast Fourier Transform.

Figure 10 shows the time in seconds of *Prop* and *Reco-3DTV* with $CR = 0.2$ and $\lambda = 0.01$, for Indian Pines and Pavia datasets. The parameters were fixed to $N_r = 20$, $N_t = 50$ for Indian Pines and $N_r = 15$, $N_t = 100$ for Pavia dataset. Note that, in the first iteration, the CA of *Prop* is lower than the CA of *Reco-3DTV* for both datasets. However, *Prop* yields to better CA than *Reco-3DTV* by using two more iterations. Also, it can be seen that the computing time of *Prop* increases per iteration without reaching that of *Reco-3DTV*.

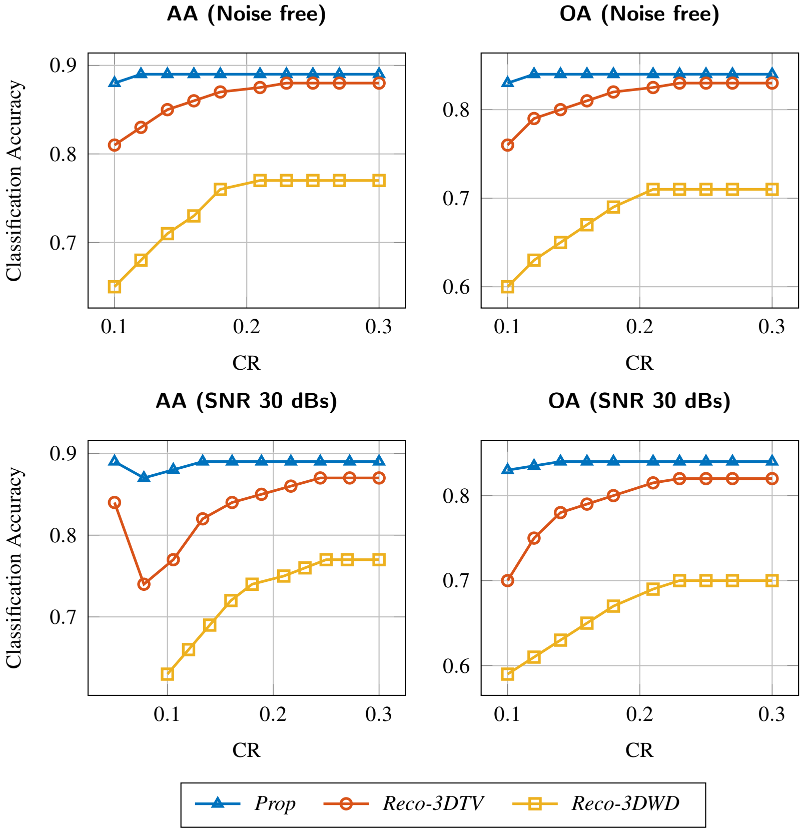


Figure 11. Comparison of the *Prop* against *Reco-3DTV*, *Reco-3DWD* using the NNS classifiers for different CR with and without additive noise.

Figure 11 compares the performance of the *Prop* method against *Reco-3DWD* and *Reco-3DTV* methods in terms of AAs and OAs obtained by applying NNS on the extracted features. The parameters have been fixed to $N_r = 20$, $N_t = 50$, $\lambda = 0.01$ and different compression ratios, with and without additive noise. This test empirically validates the feasibility of *Prop* by showing that the features can be directly extracted from the compressed hyperspectral data by solving the proposed model.

Figure 12 shows the estimated classification maps an binary maps of incorrect classification results obtained by the different methods for Indian Pines and Pavia University datasets. Note that the proposed method provides better classification results than the reconstruction-based methods (*Reco-3DWD*, *Reco-3DTV*). Further, for illustration purposes, Fig. 8 also includes the results for two state-of-the-art feature extraction-based classification methods that work directly on the full hyperspectral data, i.e. the orthogonal total variation component analysis Rasti et al. (2016) (labeled as *OTVCA*), and the extended morphological profiles Dalla Mura et al. (2010) (labeled as *EMP+PCA*). It can be seen that the proposed method provides comparable results to those of the full-data classification. This means that Prop can both, reduce the dimension of the data and well preserve the useful information.

2.5. Conclusion

we have proposed a low rank matrix factorization algorithm based on AO with internal ADMM for compressive feature extraction in order to perform spatial classification. The proposed method considers the spatial information by incorporating ℓ_1-norm prior in the 2D TV domain and spectral information by estimating the subspace from compressive measurements. The experiments indicate that the proposed framework can provide equally competitive classification results when compared to the traditional approach based on the reconstructed images and feature extraction-based classification approaches. Numerical results clearly demonstrate that compressively ac-

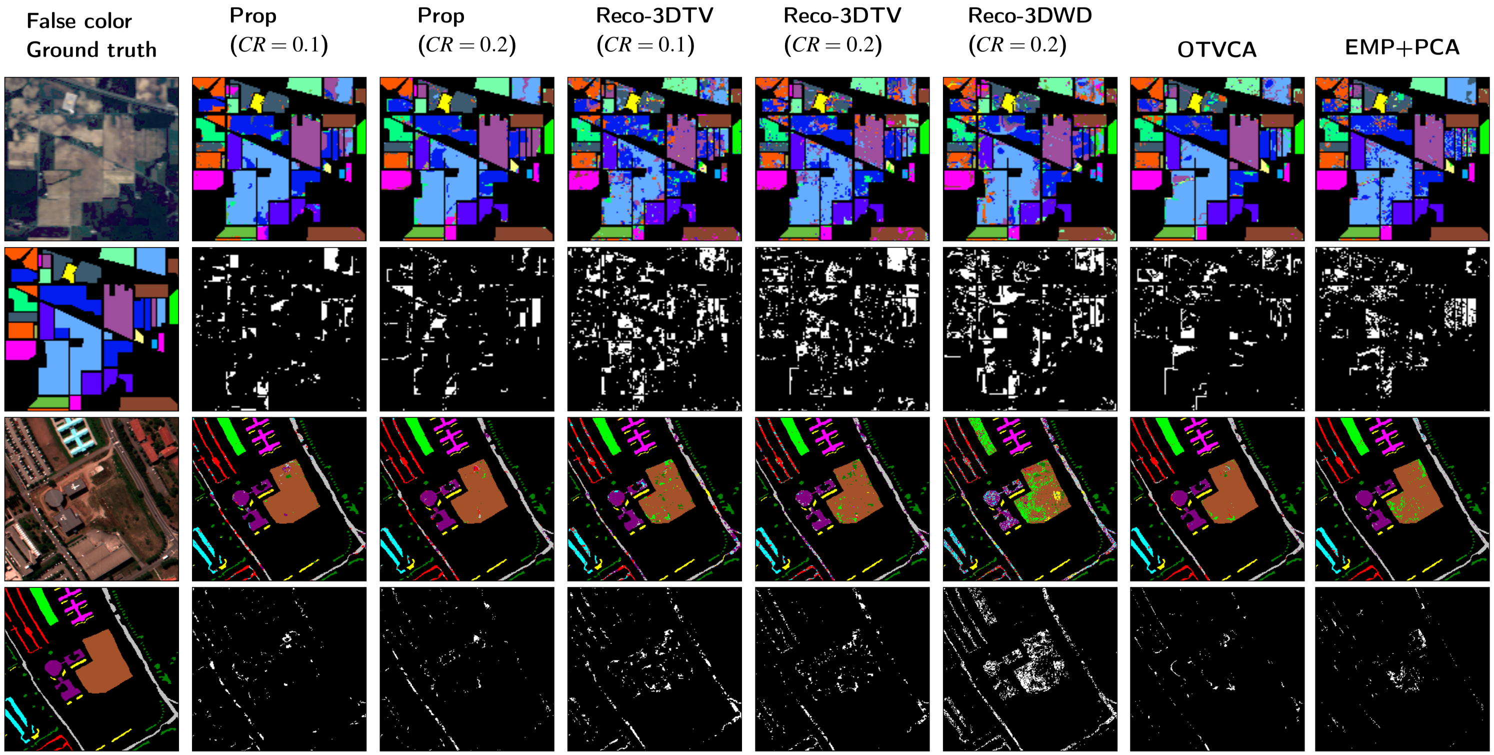


Figure 12. Classification maps obtained using NNS on extracted features from the Indian Pines (**1st row**) and Pavia University (**3rd row**) datasets. Binary maps of incorrectly classification from the Indian Pines (**2nd row**) and Pavia University (**4th row**) datasets.

quired data ranging from 10% to 25% of the full size can produce satisfactory classification results. However, higher compression ratios are the major drawback of the proposed framework. Future work can be focused on feature extraction from multiple compressive spectral sensors as a fusion strategy to perform high classification accuracy from higher compression ratios.

3. Compressive Hyperspectral Image Acquisition with Complementary RGB Sensor, Feature Extraction, and Classification

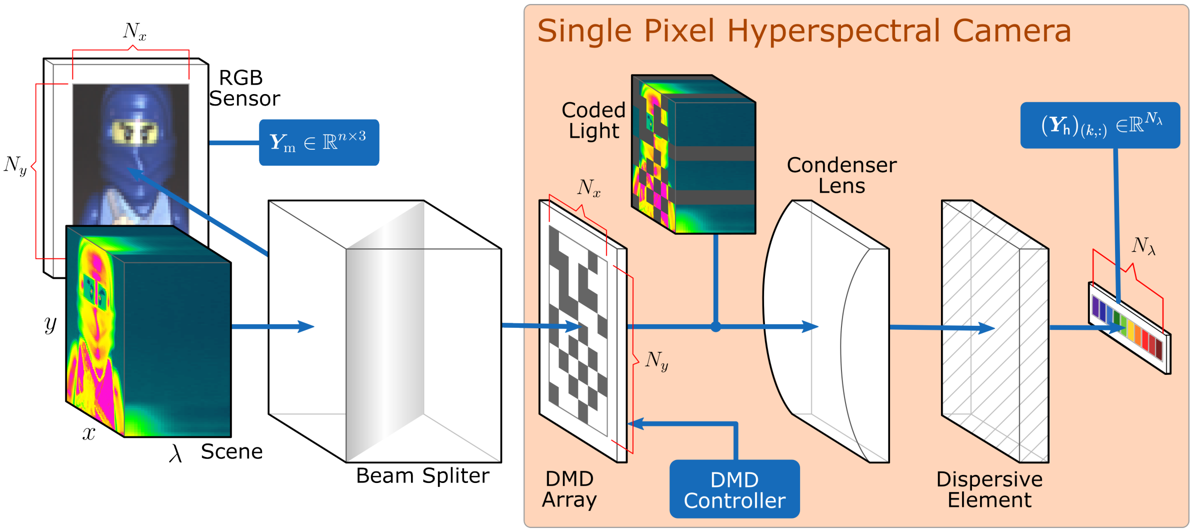

Single Pixel Hyperspectral Camera
N_x
RGB Sensor
N_y
$\boldsymbol{Y}_{\mathrm{m}} \in \mathbb{R}^{n\times 3}$
y
x
λ
Scene
Beam Spliter
Coded Light
N_x
N_y
DMD Array
DMD Controller
Condenser Lens
Dispersive Element
$(\boldsymbol{Y}_{\mathrm{h}})_{(k,:)} \in \mathbb{R}^{N_\lambda}$
N_λ

Figure 13. Schematic of the single pixel hyperspectral camera for compressive high resolution spectral data acquisition with a complementary RGB sensor Garcia et al. (2020); Tao et al. (2021).

Image fusion from different sensors can provide complementary information to improve the performance of classification tasks. Common classification pipelines based on multi-sensor fusion first perform the estimation of full-resolution image, followed by a feature extraction step. However, these features can be extracted directly without recovering of the full-image. Therefore, this work proposes a computational framework to extract features with high-spatial-resolution directly from a multi-sensor system. The multi-sensor setup considered in this work is the single pixel hyperspectral camera with a complementary high-spatial resolution RGB sensor. In this work, we first extracts spatial features from the complementary image using morphological profiles, and we assume that the extracted features and the hyperspectral measurements, lie in a low dimensional subspace. This work developed a optimization scheme to solve the feature fusion problem by integrating the alternating direction method of multipliers with the block coordinate descent method. The alternating optimization method estimates the spatial features in the fusion model by penalizing the ℓ_0-norm of the spatial gradient magnitudes. The quality of extracted features is measured in terms of supervised pixel-based classification methods. The multi-sensor scenario is tested with synthetic experiments by simulating the single pixel hyperspectral camera with a complementary RGB sensor from Pavia University and Houston University datasets. Extensive simulations show that the proposed approach outperforms other state-of-the-art methods in terms of classification accuracy.

3.1. Single Pixel Hyperspectral Camera with Complementary Sensor

The sensing model of the single-pixel hyperspectral imaging system is illustrated in Fig. 13. The data cube is spatially modulated by a series of binary (block-unblock) spatial patterns and the correlated light is detected by an RGB sensor. For clarity and convenience, the system projection is described in the matrix form. Let $\boldsymbol{Y}_{\mathrm{h}} \in \mathbb{R}^{m\times N_\lambda}$ be an observed high-spectral-low-spatial-resolution image with N_λ bands and m compressive samples in each band, and $\boldsymbol{Y}_{\mathrm{m}} \in \mathbb{R}^{n\times 3}$ be an observed low-spectral-high-spatial-resolution image with 3 bands and n pixels in each band ($n = N_xN_y$ represent the numbers of pixels for RGB image). Matrix $\boldsymbol{Z} \in \mathbb{R}^{n\times N_\lambda}$ denotes the high spatial and spectral resolution data to be estimated. With this representation, we model the single pixel hyperspectral measurements as

$$\boldsymbol{Y}_{\mathrm{h}} = \boldsymbol{\Phi}_{\mathrm{h}}\boldsymbol{Z} + \boldsymbol{H}_{\mathrm{h}}, \tag{43}$$

where $\boldsymbol{\Phi}_{\mathrm{h}} \in \mathbb{R}^{m\times n}$ is the single pixel sampling matrix (24), and $\boldsymbol{H}_{\mathrm{h}} \in \mathbb{R}^{m\times N_\lambda}$ represents independent identically distributed (i.i.d.) noise. The assumption that the noise is identically distributed across bands is also made for simplicity. Accommodating statistically independent noise across bands and pixels, but with band-dependent variance, would be straightforward. We model the RGB measurements as

$$\boldsymbol{Y}_{\mathrm{m}} = \boldsymbol{Z}\boldsymbol{\Phi}_{\mathrm{m}} + \boldsymbol{H}_{\mathrm{m}}, \tag{44}$$

where $\boldsymbol{\Phi}_{\mathrm{m}} \in \mathbb{R}^{N_\lambda \times 3}$ represents the spectral response of the high-spatial-resolution RGB sensor Simoes et al. (2014), and $\boldsymbol{H}_{\mathrm{m}} \in \mathbb{R}^{n\times 3}$ represents independent identically distributed (i.i.d.) noise. In general, these degradation models represent the discrete approximation of the components of data acquisition systems (include sensors, filters, signal conditioning, data acquisition hardware, and software applications) for the observed images.

3.2. Problem Formulation

In this section, the common fusion problem is connected with a feature fusion model. The image fusion problem consists of estimating a high-spatial-high-spectral-resolution image $\boldsymbol{Z}$, given the observed images $\boldsymbol{Y}_{\mathrm{h}}$ and $\boldsymbol{Y}_{\mathrm{m}}$ using models described in the equations (43) and (44). For this, state-of-the-art methods assume that spectral bands can be highly correlated, therefore $\boldsymbol{Z}$ usually lives in a subspace whose dimension is much smaller than the number of bands Cawse-Nicholson et al. (2012).

3.2.1. Feature fusion model. In contrast to the image fusion techniques, this work aims to estimate high-spatial-resolution features with appropriate spectral content from observed

$\boldsymbol{Y}_{\mathrm{h}}$ and $\boldsymbol{Y}_{\mathrm{m}}$ images. Hyperspectral data normally have a large correlation between bands, the spectral vectors, of size N_λ, usually live in a subspace of dimension much lower than N_λ. Therefore, we can write

$$\boldsymbol{Z} = \boldsymbol{C}\boldsymbol{Q}_{\mathrm{h}}, \tag{45}$$

where $\boldsymbol{Q}_{\mathrm{h}} \in \mathbb{R}^{N_r \times N_\lambda}$ is a matrix whose N_r columns span the same subspace as the columns of $\boldsymbol{Z}$, and $\boldsymbol{C} \in \mathbb{R}^{n \times N_r}$ are the representation coefficients. Small values of N_r, i.e., $N_r \leq N_\lambda$, translate into a description of the data in a relatively low dimensional space. This decomposition has two advantages. One is that it is computationally more efficient to work in a lower dimensional space than in the original space of $\boldsymbol{Z}$, making algorithms that use these representations comparatively fast. The other advantage is that, since the number of variables to be estimated is significantly reduced, the estimates will normally be more accurate than if we worked in the original dimensionality. Then, we replace (43) with

$$\boldsymbol{Y}_{\mathrm{h}} = \boldsymbol{\Phi}_{\mathrm{h}}\boldsymbol{C}\boldsymbol{Q}_{\mathrm{h}} + \boldsymbol{H}_{\mathrm{h}}, \tag{46}$$

where the error due to the dimensionality reduction has been incorporated into $\boldsymbol{H}_{\mathrm{h}}$. On the other hand, $\boldsymbol{Y}_{\mathrm{m}}$ is assumed be a high-spatial-resolution RGB image, where these limited number of image bands were augmented using morphological profiles (MPs) Fauvel et al. (2012). MPs allow modeling the spatial information of very high-resolution images, expanding its dimensionality to obtain a detailed signature at each pixel. It also allows the application of subspaces-based techniques that integrate both spatial and spectral information. The use of spectral and spatial information simultaneously has become a standard procedure in image classification, especially in high-resolution images Liao et al. (2017). Then, let $\boldsymbol{y}_i \in \mathbb{R}^n$ be the i-th band of the RGB image. Furthermore, let $\{\psi_j, \phi_j\}_{j\in\{1,\dots,N_p\}}$ be a set of opening and closing operators, respectively, where N_p denotes the size of the used filter, the MP of the i-th band can be defined as:

$$\boldsymbol{G}_i = \left[\psi_{N_p}(\boldsymbol{y}_i)\cdots\psi_1(\boldsymbol{y}_i)\;\boldsymbol{y}_i\;\phi_1(\boldsymbol{y}_i)\cdots\phi_{N_p}(\boldsymbol{y}_i)\right],$$

where $\boldsymbol{G}_i \in \mathbb{R}^{n\times(2N_p+1)}$ is the matrix contains the vectorized morphological features Fauvel et al. (2012); Liao et al. (2017). The concatenation of each morphological profile provided a new structure named extended morphological profile (EMP). Specifically, the morphological transformations of the each image are stacked and rearranged along the third dimension forming an extended profile. By staking each MP of the RGB image one over other, e.g. $\boldsymbol{Y}_{\text{mp}} = [\boldsymbol{G}_1\ \boldsymbol{G}_2\ \boldsymbol{G}_3] \in \mathbb{R}^{n\times N_{\text{mp}}}$ with $N_{\text{mp}} = 3(2N_p+1)$, the low-rank matrix decomposition $\boldsymbol{Y}_{\text{mp}}$ can be modeled in lower dimensional space as

$$\boldsymbol{Y}_{\text{mp}} = \boldsymbol{C}\boldsymbol{Q}_{\text{mp}} + \boldsymbol{H}_{\text{mp}}, \tag{47}$$

where $\boldsymbol{Q}_{\text{mp}} \in \mathbb{R}^{N_r\times N_{\text{mp}}}$ is a matrix whose N_r columns span the same subspace as the columns of $\boldsymbol{Z}$, and $\boldsymbol{H}_{\text{mp}}$ is the additive term that include both modeling errors and the sensors noise. The estimation of the above models aims to find the coefficients $\boldsymbol{C}$ that best represent the two image sets $\boldsymbol{Y}_{\text{h}}$ and $\boldsymbol{Y}_{\text{mp}}$ in the subspace spanned by the columns of $\boldsymbol{Q}_{\text{h}}$ and $\boldsymbol{Q}_{\text{mp}}$. In (46) and (47), the low-rank property are enforced using a product of two rank N_r matrices $\boldsymbol{C}$ and $\boldsymbol{Q}_{\text{h}}$ (or $\boldsymbol{Q}_{\text{mp}}$) where $N_r \leq \min(n, N_{\text{h}}, N_{\text{mp}})$. In this work, we observe that both above models can be reduce to following model

$$\boldsymbol{Y}_{\text{h}} = \boldsymbol{\Phi}_{\text{h}}\boldsymbol{C}\boldsymbol{Q}\boldsymbol{P} + \boldsymbol{H}_{\text{h}}, \quad \boldsymbol{Y}_{\text{mp}} = \boldsymbol{C}\boldsymbol{Q}\overline{\boldsymbol{P}} + \boldsymbol{H}_{\text{mp}}, \tag{48}$$

where $\boldsymbol{Q} \in \mathbb{R}^{N_r \times (N_\lambda + N_{\text{mp}})}$ is the union of subspaces. Here, matrices $\boldsymbol{Q}_{\text{h}}$ and $\boldsymbol{Q}_{\text{mp}}$ are replaced with $\boldsymbol{Q}\boldsymbol{P}$ and $\boldsymbol{Q}\overline{\boldsymbol{P}}$, respectively. The matrix $\boldsymbol{P} \in \mathbb{R}^{(N_\lambda + N_{\text{mp}}) \times N_\lambda}$ accounts for a uniform subsampling of the subspace $\boldsymbol{Q}$, whose columns are a subset of the columns of the identity matrix $\boldsymbol{I}_{(N_\lambda + N_{\text{mp}})}$, and $\overline{\boldsymbol{P}} \in \mathbb{R}^{(N_\lambda + N_{\text{mp}}) \times N_{\text{mp}}}$ is the matrix that selects the columns not selected by $\boldsymbol{P}$.

3.2.2. Constrained Optimization. The estimation of the projected high-spatial resolution features $\boldsymbol{C}$ and projection matrix $\boldsymbol{Q}$ from observations $\boldsymbol{Y}_{\text{h}}$ and $\boldsymbol{Y}_{\text{mp}}$ in (48) can performed by solving an inverse problem. In most image estimation problems, the inverse problem is ill-posed, which requires regularization or prior information. With measurements acquired by different sensors, the error matrices in (48) can be assumed statistically independent. Then, the posterior function of $\boldsymbol{C}$ and $\boldsymbol{Q}$ is given by

$$p(\boldsymbol{C},\boldsymbol{Q}|\boldsymbol{Y}_{\text{h}},\boldsymbol{Y}_{\text{mp}}) \propto \; p(\boldsymbol{Y}_{\text{h}}|\boldsymbol{C},\boldsymbol{Q})p(\boldsymbol{Y}_{\text{mp}}|\boldsymbol{C},\boldsymbol{Q})p(\boldsymbol{C})p(\boldsymbol{Q}), \tag{49}$$

where $\propto$ indicates proportionality. The parameter estimation in (49) can be obtained by computing the maximum of the posterior density, i.e., the MAP. The matrix $\boldsymbol{Q}$ is assumed that admit uniform distribution on the Stiefel manifold. Mathematically, the Stiefel manifold is defined as

$$\mathcal{Q}_{n\times m} = \{\boldsymbol{Q} \in \mathbb{R}^{n\times m} \mid \boldsymbol{Q}\boldsymbol{Q}^{\mathsf{T}} = \boldsymbol{I}_n, \; n \leq m\}.$$

The adoption of orthonormal subspace basis simplified the computation of both, the basis updating and coefficients recovery. On the other hand, the error matrices are assumed distributed according to the normal distributions with zero mean and variances σ_{h}^2 and σ_{mp}^2, respectively. By taking the negative logarithm of $p(\boldsymbol{Y}_{\mathrm{h}}|\boldsymbol{C},\boldsymbol{Q})$ and $p(\boldsymbol{Y}_{\mathrm{mp}}|\boldsymbol{C},\boldsymbol{Q})$ in (49), the MAP estimator of $\{\boldsymbol{C},\boldsymbol{Q}\}$ is equivalent to solving to solving the following constrained optimization problem

$$\begin{aligned} \min_{\boldsymbol{C},\boldsymbol{Q}} \quad & J(\boldsymbol{C},\boldsymbol{Q}) = l(\boldsymbol{C},\boldsymbol{Q}) + \lambda \sum_{i=1}^{N_r} r(\boldsymbol{C}_{(:,i)}) \\ \text{s.t} \quad & \boldsymbol{Q}\boldsymbol{Q}^{\mathsf{T}} = \boldsymbol{I}_{N_r}, \end{aligned} \tag{50}$$

where

$$l(\boldsymbol{C},\boldsymbol{Q}) = \frac{1}{2}\|\boldsymbol{\Phi}_{\mathrm{h}}\boldsymbol{C}\boldsymbol{Q}\boldsymbol{P} - \boldsymbol{Y}_{\mathrm{h}}\|_{\mathsf{F}}^2 + \frac{\gamma}{2}\|\boldsymbol{C}\boldsymbol{Q}\overline{\boldsymbol{P}} - \boldsymbol{Y}_{\mathrm{mp}}\|_{\mathsf{F}}^2,$$

$r(\cdot)$ is a penalty ensuring spatial regularization, $\lambda > 0$ is a parameter adjusting the importance of regularization, and $\gamma = \sigma_{\mathrm{mp}}^2/\sigma_{\mathrm{h}}^2$ is the trade-off between the hyperspectral and the complementary image noise parameters.

3.2.3. Spatial Regularization. Based on the observation that the regions in the gradient domain, black pixels usually have nearly uniform intensity values, and the edge pixels follow some distribution, modeling images with a gradient representation has been shown to be very effective in feature fusion Rasti et al. (2017, 2019); Rasti and Ghamisi (2020). Then, the prior of the $\boldsymbol{C}_{(:,i)}$ can be obtained by assuming that in the gradient domain, the pixels are independently and identically distributed following a Laplacian distribution. Note that, Laplace prior is the Bayesian equivalent of ℓ_1-norm regularization. However, this work promotes the ℓ_0-gradient regularization previously used in applications such as image segmentation and super-resolution Xu et al. (2011); Cascarano et al. (2021), to performs a feature level-fusion analysis. The ℓ_0-gradient regularization induces a segmentation-like effect that generates piece-wise constant regions while preserving edges. Then, we consider the $\ell_{2,0}$ mixed pseudo-norm defined as

$$\|\boldsymbol{X}\|_{2,0} = \#\left\{i \mid \left\|\boldsymbol{X}_{(i,:)}\right\|_2 \neq 0\right\},$$

where $\#\{\cdot\}$ denotes the cardinal of the set $\{\cdot\}$. It can be easily shown that using a Bernoulli prior is the Bayesian equivalent of ℓ_0 pseudo-norm. Then, for a vectorized image $\boldsymbol{x} \in \mathbb{R}^n$, we describe the sparsity property with a regularizer in terms of the gradient operator,

$$r(\boldsymbol{x}) = \left\| \left[\boldsymbol{D}_{\mathrm{h}}\boldsymbol{x} \; \boldsymbol{D}_{\mathrm{v}}\boldsymbol{x} \right] \right\|_{2,0}, \tag{51}$$

where the matrices $\boldsymbol{D}_{\mathrm{h}}, \boldsymbol{D}_{\mathrm{v}} \in \mathbb{R}^{n\times n}$ are operators to calculate the first order vertical and horizontal differences, respectively. Assuming periodic boundary condition (BC), matrices $\boldsymbol{D}_{\mathrm{v}}$ and $\boldsymbol{D}_{\mathrm{h}}$ are circulant, thus factors into $\boldsymbol{D}_{\mathrm{v}} = \boldsymbol{F}^{\mathsf{H}}\boldsymbol{\Lambda}_{\mathrm{v}}\boldsymbol{F}$ and $\boldsymbol{D}_{\mathrm{h}} = \boldsymbol{F}^{\mathsf{H}}\boldsymbol{\Lambda}_{\mathrm{h}}\boldsymbol{F}$, where $\boldsymbol{F} \in \mathbb{C}^{n\times n}$ and $\boldsymbol{F}^{\mathsf{H}} \in \mathbb{C}^{n\times n}$ are unitary matrices representing the two dimension discrete fourier transform (2D-DFT) and its inverse, and $\boldsymbol{\Lambda}_{\mathrm{h}}$ and $\boldsymbol{\Lambda}_{\mathrm{v}}$ are diagonal matrices of the 2D-DFT coefficients of the convolution kernel.

3.3. Alternating Optimization Scheme

The problem in (50) is solved one matrix at a time, while the other is assumed to be fixed. This procedure is summarized in Algorithm 3.1, where the AO estimator is adopted to solve effi-

Algorithm 3.1 Feature fusion based on ℓ_0-ADMM-AO.

```
Input: $\boldsymbol{Y}_{\text{h}}$, $\boldsymbol{Y}_{\text{mp}}$, $\boldsymbol{K}$, $N_r$ and $\lambda$.
$\boldsymbol{Q}_{\text{h}} \;= \text{pca}\left(\boldsymbol{Y}_{\text{h}},\, N_r\right)$   // Initialize $\boldsymbol{Q}$ using Alg.(1.1).
$\boldsymbol{Q}_{\text{mp}} = \text{pca}\left(\boldsymbol{Y}_{\text{mp}}, N_r\right)$   // Initialize $\boldsymbol{Q}$ using Alg.(1.1).
$\boldsymbol{Q}^{(0)} = \left[\boldsymbol{Q}_{\text{h}}\ \boldsymbol{Q}_{\text{mp}}\right]$
Initialize: $\boldsymbol{C}_2^{(0,0)}$, $\boldsymbol{C}_3^{(0,0)}$, $\boldsymbol{Q}_2^{(0,0)}$, $\boldsymbol{Q}_3^{(0,0)}$
for $t = 1, 2, \ldots$ to stopping rule do
// Optimize $\boldsymbol{C}$ using ADMM
    for $k_1 = 1, 2, \ldots$ to stopping rule do
        $\boldsymbol{C}_1^{(k_1)} \in \underset{\boldsymbol{C}_1}{\text{argmin}}\, \mathscr{L}_c\left(\boldsymbol{C}_1, \boldsymbol{C}_2^{(t-1,k_1-1)}, \boldsymbol{C}_3^{(t-1,k_1-1)}\right)$
        $\boldsymbol{C}_2^{(t-1,k_1)} \in \underset{\boldsymbol{C}_2}{\text{argmin}}\, \mathscr{L}_c\left(\boldsymbol{C}_1^{(k_1)}, \boldsymbol{C}_2, \boldsymbol{C}_3^{(t-1,k_1-1)}\right)$
        $\boldsymbol{C}_3^{(t-1,k_1)} = \boldsymbol{C}_3^{(t-1,k_1-1)} + \boldsymbol{A}\boldsymbol{C}_1^{(k_1)} + \boldsymbol{B}\boldsymbol{C}_2^{(t-1,k_1)}$
    end for
    Set $\boldsymbol{C}^{(t)} = \boldsymbol{C}_1^{(k_1)}$
// Optimize $\boldsymbol{Q}$ using ADMM
    for $k_2 = 1, 2, \ldots$ to stopping rule do
        $\boldsymbol{Q}_1^{(k_2)} \in \underset{\boldsymbol{Q}_1}{\text{argmin}}\, \mathscr{L}_q\left(\boldsymbol{Q}_1, \boldsymbol{Q}_2^{(t-1,k_2-1)}, \boldsymbol{Q}_3^{(t-1,k_2-1)}\right)$
        $\boldsymbol{Q}_2^{(t-1,k_2)} \in \underset{\boldsymbol{Q}_2}{\text{argmin}}\, \mathscr{L}_q\left(\boldsymbol{Q}_1^{(k_2)}, \boldsymbol{Q}_2, \boldsymbol{Q}_3^{(t-1,k_2-1)}\right)$
        $\boldsymbol{Q}_3^{(t-1,k_2)} = \boldsymbol{Q}_3^{(t-1,k_2-1)} + \boldsymbol{Q}_1^{(k_2)} - \boldsymbol{Q}_2^{(t-1,k_2)}$
    end for
    Set $\boldsymbol{Q}^{(t)} = \boldsymbol{Q}_1^{(k_2)}$
end for
Set $\hat{\boldsymbol{C}} = \boldsymbol{C}^{(t)}$
Output: $\hat{\boldsymbol{C}}$
```

ciently $\boldsymbol{C}$, and $\boldsymbol{Q}$ iteratively. To overcome the closed-form expression problem in (50), the ADMM is embedded in each iteration of the AO algorithm.

3.3.1. Optimization with respect to the matrix C (Q fixed). Given a fixed $\boldsymbol{Q}$, the minimization problem in (50) with respect to $\boldsymbol{C}$ can be solved by converting it into ADMM form. By introducing the auxiliary variables $\boldsymbol{C}_1 \in \mathbb{R}^{n\times N_r}$ and $\boldsymbol{C}_2 \in \mathbb{R}^{2n\times N_r}$, the optimization problem in (50) with respect to $\boldsymbol{C}$ can be rewritten as:

$$\min_{\boldsymbol{C}_1,\boldsymbol{C}_2} \quad f_c(\boldsymbol{C}_1)+g_c(\boldsymbol{C}_2), \quad \text{s.t.} \quad \boldsymbol{A}\boldsymbol{C}_1+\boldsymbol{B}\boldsymbol{C}_2=\boldsymbol{0}, \tag{52}$$

where

$$\begin{aligned} f_c(\boldsymbol{C}_1) &= \frac{1}{2}\|\boldsymbol{\Phi}_\text{h}\boldsymbol{C}_1\boldsymbol{Q}\boldsymbol{P}-\boldsymbol{Y}_\text{h}\|_\mathsf{F}^2+\frac{\gamma}{2}\|\boldsymbol{C}_1\boldsymbol{Q}\overline{\boldsymbol{P}}-\boldsymbol{Y}_\text{mp}\|_\mathsf{F}^2, \\ g_c(\boldsymbol{C}_2) &= g_c(\boldsymbol{C}_{21},\boldsymbol{C}_{22}) = \lambda\sum_{i=1}^{N_r}\left\|\begin{bmatrix}\boldsymbol{C}_{21(:,i)} & \boldsymbol{C}_{22(:,i)}\end{bmatrix}\right\|_{2,0}, \end{aligned}$$

and

$$\boldsymbol{A}=\begin{bmatrix}\boldsymbol{D}_\text{h}\\ \\ \boldsymbol{D}_\text{v}\end{bmatrix}, \quad \boldsymbol{B}=\begin{bmatrix}-\boldsymbol{I}_n & \boldsymbol{0}\\ \\ \boldsymbol{0} & -\boldsymbol{I}_n\end{bmatrix}, \quad \boldsymbol{C}_2=\begin{bmatrix}\boldsymbol{C}_{21}\\ \\ \boldsymbol{C}_{22}\end{bmatrix},$$

with $\boldsymbol{C}_{21},\boldsymbol{C}_{22} \in \mathbb{R}^{n\times N_r}$. The iterative procedure to solve the formulation in (52) is shown in Algorithm 3.1. The augmented Lagrangian function is defined as

$$\mathscr{L}_c(\boldsymbol{C}_1,\boldsymbol{C}_2,\boldsymbol{C}_3) = f_c(\boldsymbol{C}_1)+g_c(\boldsymbol{C}_2)+\frac{\rho_c}{2}\|\boldsymbol{A}\boldsymbol{C}_1+\boldsymbol{B}\boldsymbol{C}_2+\boldsymbol{C}_3\|_\mathsf{F}^2, \tag{53}$$

where $\rho_c > 0$ is the Lagrange penalty parameter Boyd et al. (2011), and $\boldsymbol{C}_3 = [\boldsymbol{C}_{31};\boldsymbol{C}_{32}]$ with $\boldsymbol{C}_{31},\boldsymbol{C}_{32} \in \mathbb{R}^{n\times N_r}$ denotes the scaled Lagrange multipliers related to the constraint $\boldsymbol{A}\boldsymbol{C}_1 + \boldsymbol{B}\boldsymbol{C}_2 = \boldsymbol{0}$. Then, $\mathscr{L}_c(\boldsymbol{C}_1,\boldsymbol{C}_2,\boldsymbol{C}_3)$ is minimized with respect to $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$, and update $\boldsymbol{C}_3$ as in Algorithm 3.1.

The sub-problem $\boldsymbol{C}_1$ is solved in step 7 of Algorithm 3.1 by forcing the derivative of (53) with respect to $\boldsymbol{C}_1$. It leads the following linear system:

$$\boldsymbol{C}_1 = \left(\boldsymbol{\Phi}_{\mathrm{h}}^{\top}\boldsymbol{\Phi}_{\mathrm{h}} + \boldsymbol{E}_2\right)^{-1}\boldsymbol{E}_1, \tag{54}$$

where

$$\begin{aligned}
\boldsymbol{E}_2 &= \gamma\boldsymbol{I}_n + \rho_c\left(\boldsymbol{D}_{\mathrm{h}}^{\top}\boldsymbol{D}_{\mathrm{h}} + \boldsymbol{D}_{\mathrm{v}}^{\top}\boldsymbol{D}_{\mathrm{v}}\right),\\
\boldsymbol{E}_1 &= \boldsymbol{\Phi}_{\mathrm{h}}^{\top}\boldsymbol{Y}_{\mathrm{h}}\boldsymbol{P}^{\top}\boldsymbol{Q}^{\top} + \gamma\left(\boldsymbol{Y}_{\mathrm{mp}}\overline{\boldsymbol{P}}^{\top}\boldsymbol{Q}^{\top}\right) + \cdots\\
&\qquad \rho_c\left(\boldsymbol{D}_{\mathrm{h}}^{\top}(\boldsymbol{C}_{21} - \boldsymbol{C}_{31}) + \boldsymbol{D}_{\mathrm{v}}^{\top}(\boldsymbol{C}_{22} - \boldsymbol{C}_{32})\right).
\end{aligned}$$

The first order optimality conditions lead to the solution of large-size linear systems. To solve them efficiently, we make use of conjugate gradient (CG) algorithm with a warm-start initialisation at every iteration. However, under suitable assumptions, the problem admits faster solution.

Compressive hyperspectral imaging: In compressive sensing the compressive HS single pixel camera is implemented as $\boldsymbol{\Phi}_{\mathrm{h}} = \boldsymbol{M}\boldsymbol{H}_{\mathrm{d}}$, where $\boldsymbol{H}_{\mathrm{d}} \in \mathbb{R}^{n\times n}$ is the Walsh-Hadamard transform, and $\boldsymbol{M} \in \{0,1\}^{m\times n}$ is a random down-sampling operator Vargas and Arguello (2019); Garcia et al. (2020). Consequently, the sensing matrix is an identity matrix when $\boldsymbol{M}\boldsymbol{H}_{\mathrm{d}}\boldsymbol{H}_{\mathrm{d}}^{\top}\boldsymbol{M}^{\top} = \boldsymbol{I}_m$. The coefficient $\sqrt{(n/m)}$ normalizes the transform so that the energy of the measurement vector is almost similar to that of the input signal vector. An alternative to solve efficiently the inversion problem presented in 54 from the masking operator is to use the method of the Reeves-Sorel Technique Almeida and Figueiredo (2013). Following Almeida and Figueiredo (2013), notice that

$$\boldsymbol{H}_{\mathrm{d}} = \boldsymbol{R}\begin{bmatrix}\boldsymbol{M}\boldsymbol{H}_{\mathrm{d}}\\ \\ \overline{\boldsymbol{M}}\boldsymbol{H}_{\mathrm{d}}\end{bmatrix},$$

where $\overline{\boldsymbol{M}}$ is the matrix that selects the rows not selected by $\boldsymbol{M}$ and $\boldsymbol{R}$ is a permutation matrix that puts these missing rows in their original positions in $\boldsymbol{H}_{\mathrm{d}}$. Then,

$$\begin{aligned}\boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\boldsymbol{H}_{\mathrm{d}} &= \begin{bmatrix}\boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\boldsymbol{M}^{\mathsf{T}} & \boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\overline{\boldsymbol{M}}^{\mathsf{T}}\end{bmatrix}\boldsymbol{R}^{\mathsf{T}}\boldsymbol{R}\begin{bmatrix}\boldsymbol{M}\boldsymbol{H}_{\mathrm{d}}\\ \\ \overline{\boldsymbol{M}}\boldsymbol{H}_{\mathrm{d}}\end{bmatrix},\\ &= \boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\boldsymbol{M}^{\mathsf{T}}\boldsymbol{M}\boldsymbol{H}_{\mathrm{d}} + \boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\overline{\boldsymbol{M}}^{\mathsf{T}}\overline{\boldsymbol{M}}\boldsymbol{H}_{\mathrm{d}},\end{aligned}$$

($\boldsymbol{R}$ is a permutation matrix, thus $\boldsymbol{R}^{\mathbf{T}}\boldsymbol{R} = \boldsymbol{I}_n$), the inverse of (36) can be written as

$$\begin{aligned}&\left(\boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\boldsymbol{M}^{\mathsf{T}}\boldsymbol{M}\boldsymbol{H}_{\mathrm{d}} + \gamma\boldsymbol{I}_n + \rho_c\left(\boldsymbol{D}_{\mathrm{h}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{h}} + \boldsymbol{D}_{\mathrm{v}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{v}}\right)\right)^{-1}\\ &= \left((1+\gamma)\boldsymbol{I}_n + \rho_c\left(\boldsymbol{D}_{\mathrm{h}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{h}} + \boldsymbol{D}_{\mathrm{v}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{v}}\right) - \boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\overline{\boldsymbol{M}}^{\mathsf{T}}\overline{\boldsymbol{M}}\boldsymbol{H}_{\mathrm{d}}\right)^{-1}\\ &= \boldsymbol{E}_3^{-1} - \boldsymbol{E}_3^{-1}\boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\overline{\boldsymbol{M}}^{\mathsf{T}}\left(\overline{\boldsymbol{M}}\boldsymbol{H}_{\mathrm{d}}\boldsymbol{E}_3^{-1}\boldsymbol{H}_{\mathrm{d}}^{\mathsf{T}}\overline{\boldsymbol{M}}^{\mathsf{T}} - \boldsymbol{I}_m\right)^{-1}\overline{\boldsymbol{M}}\boldsymbol{H}_{\mathrm{d}}\boldsymbol{E}_3^{-1}\end{aligned}$$

where the second equality results from using the Sherman Morrison–Woodbury matrix inversion identity, after defining $\boldsymbol{E}_3 = (1+\gamma)\boldsymbol{I}_n + \rho_c\left(\boldsymbol{D}_{\mathrm{h}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{h}} + \boldsymbol{D}_{\mathrm{v}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{v}}\right)$. We use the CG algorithm to solve the above inversion; we confirmed experimentally that taking only one CG iteration yields the fastest convergence, without degrading the final result.

Super-resolution: In image super-resolution $\boldsymbol{\Phi}_{\mathrm{h}} = \boldsymbol{S}\boldsymbol{K}$, where $\boldsymbol{K} \in \mathbb{R}^{n\times n}$ is the matrix representation of the cyclic convolution operator, i.e., $\boldsymbol{K}$ is a block circulant matrix with circulant blocks, and $\boldsymbol{S} \in \mathbb{R}^{m\times n}$ is a down-sampling operator, while its transpose $\boldsymbol{S}^{\mathsf{T}}$ interpolates the decimated image with zeros. The constant m represents the number of samples with $m = M_x M_y$ and d represents the scaling factor with $n = d^2 m$. Since $\boldsymbol{K}$ is block circulant with circulant blocks, it can be factored into $\boldsymbol{K} = \boldsymbol{F}^{\mathsf{H}}\boldsymbol{\Lambda}\boldsymbol{F}$. Consequently, the down-sampling matrix is an identity matrix when $\boldsymbol{S}\boldsymbol{S}^{\mathsf{T}} = \boldsymbol{I}_m$, and is a binary diagonal matrix with ones at the observed positions and zeros elsewhere when $\boldsymbol{S}^{\mathsf{T}}\boldsymbol{S} \in \mathbb{R}^{n\times n}$. Then, the inversion in (54) can be computed in closed form following Zhao et al. (2016). From (54), note that

$$\begin{aligned}&\left(\boldsymbol{K}^{\mathsf{T}}\boldsymbol{S}^{\mathsf{T}}\boldsymbol{S}\boldsymbol{K} + \gamma\boldsymbol{I}_n + \rho_c\left(\boldsymbol{D}_{\mathrm{h}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{h}} + \boldsymbol{D}_{\mathrm{v}}^{\mathsf{T}}\boldsymbol{D}_{\mathrm{v}}\right)\right)^{-1}\\ &= \boldsymbol{F}^{\mathsf{H}}\left(\boldsymbol{\Lambda}^{\mathsf{H}}\left(\boldsymbol{F}\boldsymbol{S}^{\mathsf{T}}\boldsymbol{S}\boldsymbol{F}^{\mathsf{H}}\right)\boldsymbol{\Lambda} + \gamma\boldsymbol{I}_n + \rho_c\left(\boldsymbol{\Lambda}_{\mathrm{h}}^2 + \boldsymbol{\Lambda}_{\mathrm{v}}^2\right)\right)^{-1}\boldsymbol{F}.\end{aligned}$$

The theoretical result presented in Zhao et al. (2016) for 2D images, allows the following decomposition:

$$\begin{aligned}\boldsymbol{F}\boldsymbol{S}^{\mathsf{T}}\boldsymbol{S}\boldsymbol{F}^{\mathsf{H}} &= \frac{1}{d^2}\left(\left(\mathbf{1}_d\mathbf{1}_d^{\mathsf{T}} \otimes \boldsymbol{I}_{M_y}\right) \otimes \left(\mathbf{1}_d\mathbf{1}_d^{\mathsf{T}} \otimes \boldsymbol{I}_{M_x}\right)\right),\\ &= \frac{1}{d^2}\left(\left(\mathbf{1}_d \otimes \boldsymbol{I}_{M_y}\right) \otimes \left(\mathbf{1}_d \otimes \boldsymbol{I}_{M_x}\right)\right)\left(\left(\mathbf{1}_d^{\mathsf{T}} \otimes \boldsymbol{I}_{M_y}\right) \otimes \left(\mathbf{1}_d^{\mathsf{T}} \otimes \boldsymbol{I}_{M_x}\right)\right).\end{aligned}$$

Then, further simplification can be achieved, i.e.,

$$\begin{aligned}
&\boldsymbol{\Lambda}^{\mathsf{H}}\left(\boldsymbol{F}\boldsymbol{S}^{\mathsf{T}}\boldsymbol{S}\boldsymbol{F}^{\mathsf{H}}\right)\boldsymbol{\Lambda}\\
&=\boldsymbol{\Lambda}^{\mathsf{H}}\left(\frac{1}{d^2}\left((\mathbf{1}_d\otimes\boldsymbol{I}_{M_y})\otimes(\mathbf{1}_d\otimes\boldsymbol{I}_{M_x})\right)\left(\left(\mathbf{1}_d^{\mathsf{T}}\otimes\boldsymbol{I}_{M_y}\right)\otimes\left(\mathbf{1}_d^{\mathsf{T}}\otimes\boldsymbol{I}_{M_x}\right)\right)\right)\boldsymbol{\Lambda},\\
&=\frac{1}{d^2}\left(\underline{\boldsymbol{\Lambda}}^{\mathsf{H}}\underline{\boldsymbol{\Lambda}}\right),
\end{aligned}$$

where $\underline{\boldsymbol{\Lambda}}=\left(\left(\mathbf{1}_d^{\mathsf{T}}\otimes\boldsymbol{I}_{M_y}\right)\otimes\left(\mathbf{1}_d^{\mathsf{T}}\otimes\boldsymbol{I}_{M_x}\right)\right)\boldsymbol{\Lambda}$. Then applying the Woodbury matrix identity to obtain

$$\left(\frac{1}{d^2}\left(\underline{\boldsymbol{\Lambda}}^{\mathsf{H}}\underline{\boldsymbol{\Lambda}}\right)+\boldsymbol{E}_3\right)^{-1}=\boldsymbol{E}_3^{-1}-\boldsymbol{E}_3^{-1}\underline{\boldsymbol{\Lambda}}^{\mathsf{H}}\left(d^2\boldsymbol{I}_m+\underline{\boldsymbol{\Lambda}}\boldsymbol{E}_3^{-1}\underline{\boldsymbol{\Lambda}}^{\mathsf{H}}\right)^{-1}\underline{\boldsymbol{\Lambda}}\boldsymbol{E}_3^{-1},$$

where $\boldsymbol{E}_3=\gamma\boldsymbol{I}_n+\rho_c\left(\boldsymbol{\Lambda}_{\mathrm{h}}^2+\boldsymbol{\Lambda}_{\mathrm{v}}^2\right)$. Note that $\boldsymbol{E}_3^{-1}$ involves a diagonal inversion, with $\mathscr{O}(n)$ cost. Also note the term $\underline{\boldsymbol{\Lambda}}\boldsymbol{E}_3^{-1}\underline{\boldsymbol{\Lambda}}^{\mathsf{H}}$ is diagonal matrix then the above expression involves a diagonal inversion with $\mathscr{O}(m)$ cost.

The sub-problem $\boldsymbol{C}_2$ is decouple into two variables $\boldsymbol{C}_{21}$ and $\boldsymbol{C}_{22}$ in step 8 of Algorithm 3.1. Due to decomposability of the $\ell_{2,0}$-mixed pseudo norm, solving for $\boldsymbol{C}_{21}$ and $\boldsymbol{C}_{22}$ corresponds to solve a $\ell_{2,0}$-ℓ_2 problem

$$\{\boldsymbol{C}_{21},\boldsymbol{C}_{22}\}\in\underset{\boldsymbol{C}_{21},\boldsymbol{C}_{22}}{\operatorname{argmin}}\quad g_c(\boldsymbol{C}_{21},\boldsymbol{C}_{22})+\frac{\rho_c}{2}\left\|\boldsymbol{A}\boldsymbol{C}_1+\boldsymbol{B}\begin{bmatrix}\boldsymbol{C}_{21}\\ \\ \boldsymbol{C}_{22}\end{bmatrix}+\begin{bmatrix}\boldsymbol{C}_{31}\\ \\ \boldsymbol{C}_{32}\end{bmatrix}\right\|_{\mathsf{F}}^2,$$

whose solution is given by a row-wise vector-hard threshold function Xu et al. (2011),

$$\left\{\boldsymbol{C}_{21(:,i)}, \boldsymbol{C}_{22(:,i)}\right\} = \mathrm{rhard}\left(\left\{\boldsymbol{E}_{4(:,i)}, \boldsymbol{E}_{5(:,i)}\right\}, \lambda/\rho_c\right), \tag{55}$$

where $\boldsymbol{E}_4 = \boldsymbol{D}_{\mathrm{h}}\boldsymbol{C}_1 + \boldsymbol{C}_{31}$, $\boldsymbol{E}_5 = \boldsymbol{D}_{\mathrm{v}}\boldsymbol{C}_1 + \boldsymbol{C}_{32}$. The derivation of the Algorithm 3.2 from the functional (51) can be reviewed from Xu et al. (2011).

Algorithm 3.2 Row-hard-threshold algorithm.

```
function RHARD($\boldsymbol{X}$, $\tau$)
// Let be $\boldsymbol{X} \in \mathbb{R}^{n\times m}$ and $\tau \geq 0$, then
    for $i = 1,2,\ldots,n$ do
        if $\left\|\boldsymbol{X}_{(i,:)}\right\|_2^2 \leq \tau$ then
            $\boldsymbol{X}_{(i,:)} = \mathbf{0}$        // where $\mathbf{0} \in \mathbb{R}^m$
        end if
    end for
    return $\boldsymbol{X}$
end function
```

3.3.2. Optimization with respect to the matrix Q (C Fixed).

The solution for $\boldsymbol{Q}$ is summarized in Algorithm 3.1. Aiming at a more computationally efficient method for solving (50) with respect to $\boldsymbol{Q}$, ADMM introduces auxiliary variables to split the orthogonality constraints, which leads to another formulation of (50). In order to minimize with respect to $\boldsymbol{Q}$, the following constrained optimization problem is solved when $\boldsymbol{C}$ is assumed constant

$$\min_{\boldsymbol{Q}_1,\boldsymbol{Q}_2} \quad f_q(\boldsymbol{Q}_1) + g_q(\boldsymbol{Q}_2) \quad \text{s.t.} \quad \boldsymbol{Q}_1 - \boldsymbol{Q}_2 = \mathbf{0}, \tag{56}$$

where

$$g_q(\boldsymbol{Q}_2) = \begin{cases} 0 & \text{if } \boldsymbol{Q}_1 \in \mathscr{Q} \\ +\infty & \text{otherwise} \end{cases}, \qquad ,$$

$$f_q(\boldsymbol{Q}_1) = \frac{1}{2}\|\boldsymbol{\Phi}_{\mathrm{h}}\boldsymbol{C}\boldsymbol{Q}_1\boldsymbol{P} - \boldsymbol{Y}_{\mathrm{h}}\|_{\mathsf{F}}^2 + \frac{\gamma}{2}\|\boldsymbol{C}\boldsymbol{Q}_1\overline{\boldsymbol{P}} - \boldsymbol{Y}_{\mathrm{mp}}\|_{\mathsf{F}}^2$$

and $\boldsymbol{Q}_1, \boldsymbol{Q}_2 \in \mathbb{R}^{N_r \times (N_\lambda + N_{\mathrm{mp}})}$ are auxiliary variables. This problem has been considered in applications such as linear and nonlinear eigenvalue problems Wen and Yin (2013). The Lagrangian function associated to the optimization of $\boldsymbol{Q}$ can be written as

$$\mathscr{L}_q(\boldsymbol{Q}_1, \boldsymbol{Q}_2, \boldsymbol{Q}_3) = f_q(\boldsymbol{Q}_1) + g_q(\boldsymbol{Q}_2) + \frac{\rho_q}{2}\|\boldsymbol{Q}_1 - \boldsymbol{Q}_2 + \boldsymbol{Q}_3\|_{\mathsf{F}}^2, \tag{57}$$

where $\boldsymbol{Q}_3 \in \mathbb{R}^{N_r \times (N_\lambda + N_{\mathrm{mp}})}$ is the scaled dual variable. The optimization of $\mathscr{L}(\boldsymbol{Q}_1, \boldsymbol{Q}_2, \boldsymbol{Q}_3)$ consists in updating $\boldsymbol{Q}_1$, $\boldsymbol{Q}_2$, and $\boldsymbol{Q}_3$ iteratively as summarized in Algorithm 3.1.

From (57), computing the $\boldsymbol{Q}_1$-update requires to solve the linear system. We can take advantage of the masking matrix $\boldsymbol{P}$ to separate $\boldsymbol{Q}_1$ into $\boldsymbol{Q}_1\boldsymbol{P}$ and $\boldsymbol{Q}_1\overline{\boldsymbol{P}}$, where $\overline{\boldsymbol{P}}$ is the matrix that selects the pixels not selected by $\boldsymbol{P}$. We then have

$$\begin{aligned} \boldsymbol{Q}_1\boldsymbol{P} &= \left(\boldsymbol{C}^{\mathsf{T}}\boldsymbol{\Phi}_{\mathrm{h}}^{\mathsf{T}}\boldsymbol{\Phi}_{\mathrm{h}}\boldsymbol{C} + \rho_q\boldsymbol{I}_{N_r}\right)^{-1}\left(\boldsymbol{C}^{\mathsf{T}}\boldsymbol{\Phi}_{\mathrm{h}}^{\mathsf{T}}\boldsymbol{Y}_{\mathrm{h}}\boldsymbol{P}^{\mathsf{T}} + \rho_q(\boldsymbol{Q}_2 - \boldsymbol{Q}_3)\right)\boldsymbol{P}, \\ \boldsymbol{Q}_1\overline{\boldsymbol{P}} &= \left(\boldsymbol{C}^{\mathsf{T}}\boldsymbol{C} + \rho_q\boldsymbol{I}_{N_r}\right)^{-1}\left(\boldsymbol{C}^{\mathsf{T}}\boldsymbol{Y}_{\mathrm{mp}}\overline{\boldsymbol{P}}^{\mathsf{T}} + \rho_q(\boldsymbol{Q}_2 - \boldsymbol{Q}_3)\right)\overline{\boldsymbol{P}}. \end{aligned} \tag{58}$$

Note that $(\boldsymbol{C}^{\mathsf{T}}\boldsymbol{\Phi}_{\mathrm{h}}^{\mathsf{T}}\boldsymbol{\Phi}_{\mathrm{h}}\boldsymbol{C} + \rho_q\boldsymbol{I}_{N_r})$ and $(\boldsymbol{C}^{\mathsf{T}}\boldsymbol{C} + \rho_q\boldsymbol{I}_{N_r})$ have dimensions $N_r \times N_r$ and therefore can be precomputed.

From (57), the update of $\boldsymbol{Q}_2$ can be calculated by simply computing the Euclidean projection of $(\boldsymbol{Q}_1+\boldsymbol{Q}_3)$ onto the set $\mathscr{Q}$, also known as the nearest orthogonal matrix problem (Lai and Osher, 2014, Theorem 1). Suppose we have the SVD decomposition of the matrix $(\boldsymbol{Q}_1+\boldsymbol{Q}_3)$ as $\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^{\mathsf{T}}$ where $\boldsymbol{U}\in\mathbb{R}^{N_r\times N_r}$, $\boldsymbol{V}\in\mathbb{R}^{(N_\lambda+N_{\mathrm{mp}})\times N_r}$ and $\boldsymbol{S}\in\mathbb{R}^{N_r\times N_r}$, then the global solution is

$$\begin{aligned}\boldsymbol{Q}_2 &\in \underset{\boldsymbol{Q}_2}{\operatorname{argmin}}\; g_q(\boldsymbol{Q}_2)+\frac{\rho_c}{2}\left\|\boldsymbol{Q}_1-\boldsymbol{Q}_2+\boldsymbol{Q}_3\right\|_{\mathsf{F}}^2,\\ &\leftarrow \boldsymbol{U}\boldsymbol{V}^{\mathsf{T}}.\end{aligned} \tag{59}$$

When $(\boldsymbol{Q}_1+\boldsymbol{Q}_3)$ has full row rank, the solution also is unique.

3.3.3. Convergence and computational complexity analysis.

Note that, if each sub-problem in (50) is convex and has a unique solution, then every limit point is a stationary point (Tseng, 2001, Theorem 4.1). The uniqueness of the solution is not required when the number of blocks is two Grippo and Sciandrone (2000). However, the convergence results of our proposed algorithm do not apply due to the non-convexity of the function $r(\cdot)$ and the minimization concerning $\boldsymbol{Q}$. Therefore, those functionals are analyzed as follows.

- The minimization with respect to $\boldsymbol{C}$ belongs to the class of multiplier algorithms that can be considered as a '*Generalized Lasso*' problem with ℓ_0 regularization rather than the ℓ_1-norm (Boyd et al., 2011, Sec. 6.4.1). A convergence results for this ADMM version is the recently developed theory by Cascarano et al. (2021), which is applicable to non convex and non-differentiable functions $g_c(\cdot)$. Following the suggestion in Cascarano et al. (2021) to ensure convergence, the penalty sequences are adjusted heuristically as $\rho^{(t)} = t(1+\varepsilon)^t$ with $\varepsilon = 10^{-4}$.

- The minimization with respect to $\boldsymbol{Q}$ is a non-convex problem, and there is not guarantee that the AO method can reach the global solution of (50). However, the convergence of Algorithm 3.1 can be observed in practice. Furthermore, if $\mathscr{Q}$ is compact, which implies that the sequence generated is bounded, the BCD method is guaranteed to converge to a stationary point Tseng (2001).

The stopping rule for Algorithm 3.1 is

$$\frac{\left|J(\boldsymbol{C}^{(t)}, \boldsymbol{Q}^{(t)}) - J(\boldsymbol{C}^{(t-1)}, \boldsymbol{Q}^{(t-1)})\right|}{\left|J(\boldsymbol{C}^{(t)}, \boldsymbol{Q}^{(t)})\right|} \leq 10^{-4}.$$

As a final observation we mention that the most computationally expensive part for the proposed method is the solution of $\boldsymbol{C}$ in Algorithm 3.1. Following the proposed solution, the order of computation complexity decreased significantly from $\mathscr{O}(n^3 N_r)$ to $\mathscr{O}(nN_r \log(n))$, which allows the analytical solution (36) to be computed efficiently. Then, in Algorithm 3.1 the computational complexity of the optimization of $\boldsymbol{C}$ is $\mathscr{O}(nN_r \log(n))$, and the computational complexity of the optimization of $\boldsymbol{Q}$ is $\mathscr{O}(nN_r)$. The overall complexity per iteration in Algorithm 3 is given by $\mathscr{O}(nN_r \log(n)) + \mathscr{O}(nN_r)$.

3.4. Numerical Experiments

In this section, we assess the performance of the proposed AO-ADMM in terms of classification accuracy and computational efficiency on synthetic and real datasets. The proposed scheme was implemented in Matlab and all numerical experiments were performed on a computer with an Intel(R) Core(TM) $i7-4790$ CPU@3.60GHz and 32 GB RAM. In each classification experiment, N_t samples were randomly chosen per class, as training samples and the remaining samples were used for testing.

3.4.1. Pavia University dataset. For this dataset, we obtain the measurements of the HS image by simulating the single pixel hyperspectral camera Li et al. (2012) with a compression rat $\text{CR} = 25.00\%$. Figure 21(b) shows an image projection obtained by the compressive acquisition system. Unless stated otherwise, the RGB image and the compressive measurement set are contaminated with additive white Gaussian noise whose SNR is fixed to 30 dB. Finally, the ground truth map that contains nine distinct classes is illustrated in Fig. 21(c), where every class labels a different material in the urban cover. All experiments follow the data acquisition model given in (24). For each fixed set of parameters (Compression Ratio (CR), N_r, λ, *noise*), the averaged results of 10 realizations of the sensing matrix $\mathbf{\Phi}_{\text{h}}$ are shown. In the noisy case, for each generated sensing matrix $\mathbf{\Phi}_{\text{h}}$, the averaged results of 10 independent noise realizations are performed.

First, to observe the effects of the ℓ_0-norm regularization term on the classification features, Fig. 14 displays four feature bands yielded by the proposed method for three different values of the regularization parameter, i.e. $\lambda = 0$, $\lambda = 5 \times 10^{-5}$, and $\lambda = 5 \times 10^{-4}$. For this experiment, we set the number of feature bands to $N_r = 20$. In addition, we use a disk-shaped element for the morphological opening and closing functions with radius $(5, 10, 20, 50)$. As can be seen in this figure, the proposed fusion approach yields feature bands with smoother piecewise regions as the value of λ increases. In addition, as λ increases, it can be seen that the proposed fusion approach minimizes the influence of the image noise and the spatial structure of the objects is preserved.

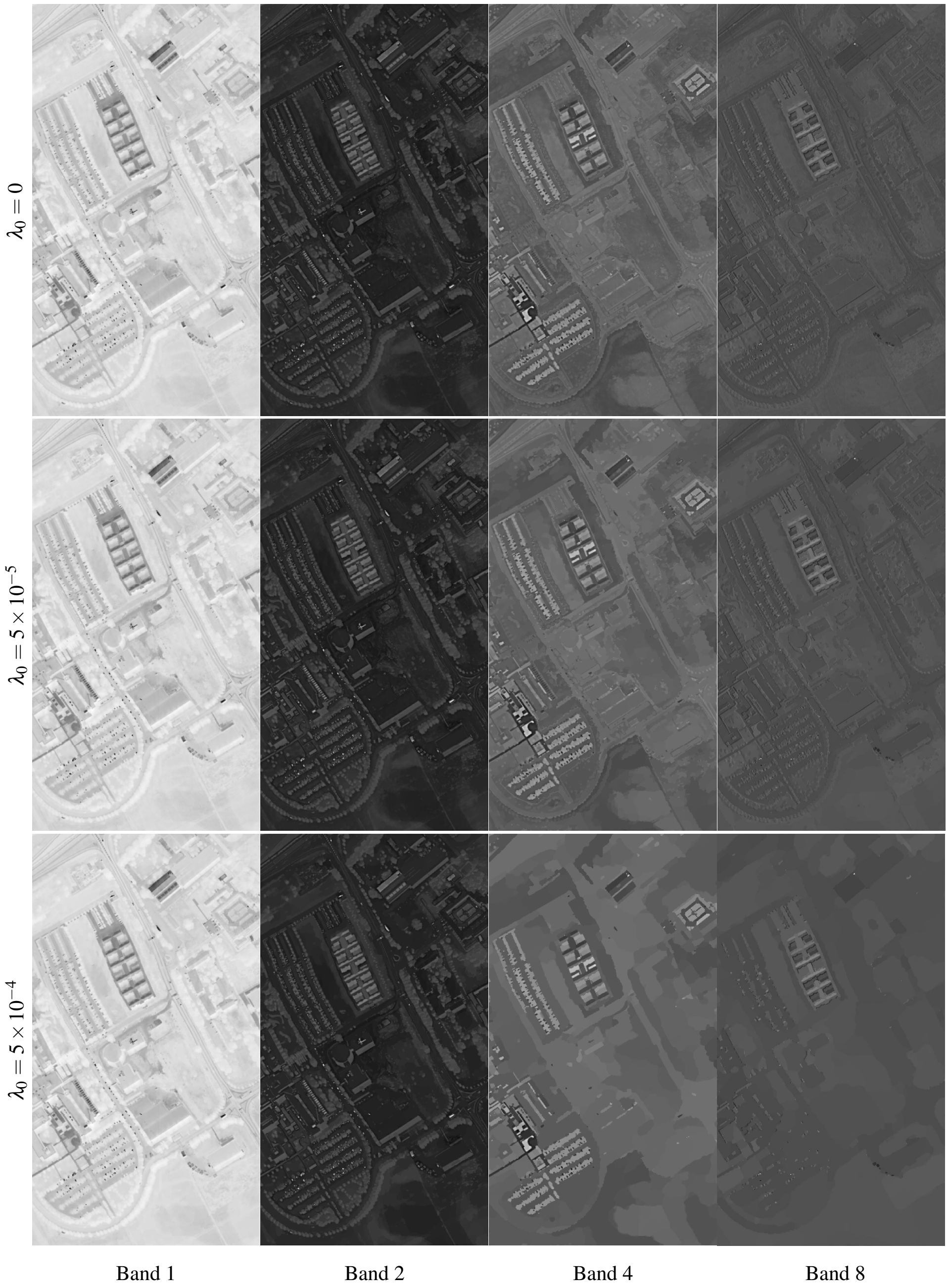

$\lambda_0 = 0$
$\lambda_0 = 5 \times 10^{-5}$
$\lambda_0 = 5 \times 10^{-4}$
Band 1
Band 2
Band 4
Band 8

Figure 14. Pavia University dataset. Feature bands obtained by the proposed usion technique for (upper) $\lambda = 0$, (middle) $\lambda = 5 \times 10^{-5}$, and (bottom) $\lambda = 5 \times 10^{-4}$.

In addition, we evaluate the performance of the proposed feature fusion in terms of the accuracy yielded by different supervised methods for pixel-based image classification. Table 4 shows the accuracy values obtained using different supervised classifiers: a feedforward neural network (FFNN) Ramirez et al. (2021), the nearest neighbor (1NN) classifier, a support vector machine with a polynomial kernel (SVM-PLY) Camps-Valls and Bruzzone (2005), and a random forest (RF) classifier Gislason et al. (2006).

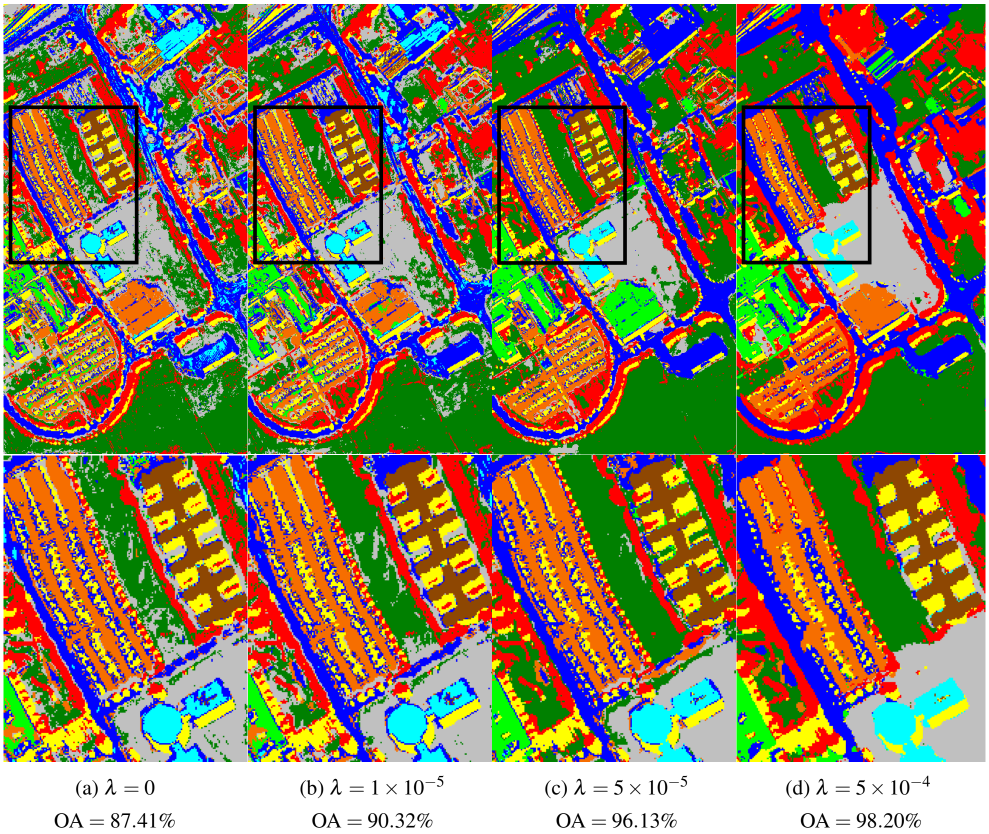
Table 4
Performance of the proposed feature fusion approach for different supervised classification methods.

Classes	# Samples		FFNN	1NN	SVM-PLY	RF
	Train	Test				
Asphalt	50	6439	82.56 ± 7.10	96.47 ± 1.98	97.48 ± 1.89	**97.63 ± 1.11**
Meadows	50	18242	93.29 ± 2.98	97.23 ± 1.64	96.35 ± 2.19	**97.97 ± 0.88**
Gravel	50	1930	83.78 ± 25.76	**99.27 ± 0.41**	99.12 ± 0.55	99.23 ± 0.91
Trees	50	2984	89.53 ± 3.12	90.59 ± 1.96	94.48 ± 1.39	**94.73 ± 1.28**
Metal	50	1295	98.24 ± 1.14	99.89 ± 0.07	98.83 ± 0.64	**99.92 ± 0.04**
Soil	50	4979	97.18 ± 2.28	99.94 ± 0.05	99.91 ± 0.06	**99.97 ± 0.03**
Bitumen	50	1280	99.20 ± 0.47	99.75 ± 0.16	99.45 ± 0.41	**99.93 ± 0.12**
Bricks	50	3632	89.53 ± 9.03	97.40 ± 0.97	97.94 ± 0.95	**98.69 ± 1.12**
Shadows	50	897	95.50 ± 2.60	96.37 ± 1.69	94.52 ± 2.53	**98.45 ± 1.69**
Overall accuracy (%)			91.44 ± 2.91	97.21 ± 0.86	97.22 ± 1.19	**98.17 ± 0.46**
Average accuracy (%)			92.09 ± 3.93	97.43 ± 0.31	97.57 ± 0.54	**98.50 ± 0.33**
Kappa Statistic (κ)			0.888 ± 0.038	0.963 ± 0.011	0.963 ± 0.015	**0.976 ± 0.006**

As can be seen in this table, we randomly select 50 training samples for each class and the remaining pixels are used to test the corresponding machine learning model. For this experiment, the proposed feature fusion algorithm is executed using $\lambda = 5 \times 10^{-4}$ and $N_r = 20$. Note that the FFNN is built with 10 hidden layers whose training stage is performed using the Levenberg-Marquardt algorithm. In addition, the parameters of the multiclass SVM-PLY model of order 3 are set to $\sigma = 1$ and $C = 1$. The number of trees of the RF classifier is fixed to 200. Every value of this table is obtained by averaging 10 realizations of the respective experiment and at each trial, a different realization of additive noise is generated and a different set of training samples is selected. It can be observed that the best accuracy values are in bold font. Furthermore, the last three rows of Table 4 includes the overall accuracy (OA), the average accuracy (AA), and the Cohen's kappa statistic (κ). As can be seen in this table, the RF classifier outperforms the other supervised methods in terms of accuracy. For the remaining experiments, we shall use the RF classifier with 200 trees.

To evaluate the influence of the ℓ_0-norm regularization term on the classification performance, Fig. 15 displays the labeling maps yielded by the proposed feature fusion approach for different values of the penalty parameter, i.e. $\lambda = 0$, $\lambda = 1 \times 10^{-5}$, $\lambda = 5 \times 10^{-5}$, and $\lambda = 5 \times 10^{-4}$. Furthermore, a zoomed version of the corresponding labeling map is included for visual comparison and the overall accuracy of each classification map is shown in the figure caption. As can be seen in this figure, the classification noise is reduced as the λ increases leading to more homogeneous labeling regions. Note also that the classification accuracy improves as λ increases for the evaluation interval.

Parameter analysis displays the classification performance of the subspace-based feature fusion method on the Pavia University data set for different values of the regularization parameter and numbers of feature bands. Furthermore, this analysis is obtained by capturing the compressive measurements of the HS image with two different compression rates: Fig. 16(left): CR $= 12.50\%$ and Fig. 16(right): CR $= 25.00\%$. More precisely, every value is estimated by averaging the overall accuracy of 50 realizations of the corresponding experiment, where at each trial a different


(a) $\lambda = 0$
OA = 87.41%

(b) $\lambda = 1 \times 10^{-5}$
OA = 90.32%

(c) $\lambda = 5 \times 10^{-5}$
OA = 96.13%

(d) $\lambda = 5 \times 10^{-4}$
OA = 98.20%

Figure 15. Classification maps yielded by the fused features for (a) $\lambda = 0$, (b) $\lambda = 1 \times 10^{-5}$, (c) $\lambda = 5 \times 10^{-5}$, and (d) $\lambda = 5 \times 10^{-4}$.

training set is randomly selected. As can be seen in these figures, the proposed feature fusion method improves the classification accuracy as the number of features N_r increases. In addition, the regularization parameter λ should be carefully selected to obtain an outstanding classification performance. Note that for small values of λ, image details are preserved in the feature bands, however, the piece-wise regions are not properly smoothed leading to classification noise in the labeling maps. Rather, for large values of λ, the feature bands progressively lose the image edges affecting the class separability, degrading in turn, the classification performance.

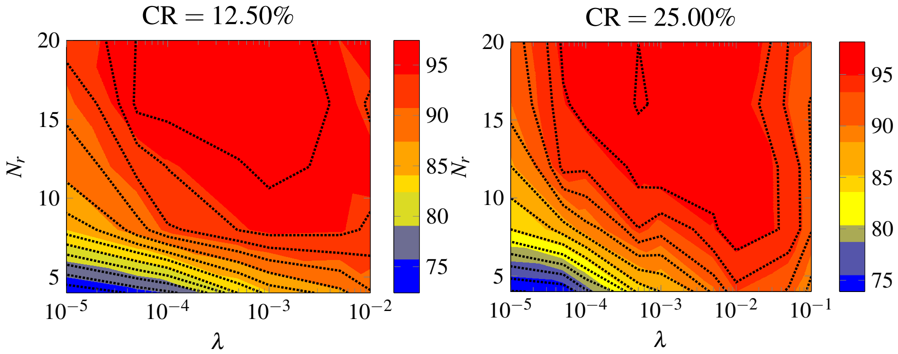


Figure 16. Effects of the number of feature bands and the value of λ on the overall accuracy for (left) CR = 12.50% and (right) CR = 25.00%

3.4.2. Houston University 2013 dataset. For comparison purposes, Fig. 17 illustrates the classification maps obtained from different kinds of features. Specifically, Figs 17(a)-(c) display the labeling maps yielded by the RGB image, the HSI, and the stacking of the RGB image with an interpolated version of the HSI (RGB + HSI). Furthermore, Fig. 17(d) depicts the classification maps obtained by the subspace sensor fusion (SubFus) method Rasti and Ghamisi (2020). Notice that SubFus method obtains a set of fused features directly from the RGB image and the HSI. The classification maps obtained by the proposed approach is shown in Fig. 17(e). For this dataset, the parameter setting of the proposed feature fusion approach is fixed to $\lambda = 1 \times 10^{-3}$ and $N_r = 20$. Furthermore, the morphological opening and closing operations are performed using a disk-shape with radius (20, 50, 100, 200). Table 5 displays the accuracy values generated by the approaches under test. Each value is obtained by averaging the results of 10 realizations of the respective experiment, and at each trial, the procedure generates a different realization of the additive noise with SNR at 30 dB. As can be observed in Table 5, the proposed approach exhibits a competitive performance with respect to the state-of-the-art feature fusion method. Indeed, our method outperforms other approaches in terms of OA, AA, and κ.

3.4.3. Houston University 2018 dataset. To test the proposed method, an RGB composite is obtained by projecting the HS image using the IKONOS sensor response in the visible wavelength interval (0.38 - 0.86 μm). Fig. 22(a) displays the RGB composite of the Houston University 2018 dataset. Moreover, the hyperspectral compressive measurements are obtained by simulating the single pixel hyperspectral camera using a compression ratio $\rho = 25.00\%$. A projection captured by the compressive hyperspectral imaging system is illustrated in Fig. 22(b). Furthermore, this dataset includes fifteen different classes, where each class label corresponds to a distinct structure in the urban cover. Finally, the training sample set and the test sample set are shown in Fig. 23(c) and Fig. 23(d), respectively.

We compare the performance of the feature fusion proposed method with respect to other classification approaches. In this sense, we first obtain the labeling maps from single sensor data. Figs 18(d) and 18(e) display the classification maps obtained from the RGB data and an interpolated version of the HS image, respectively. In order to consider a set of features obtained by

Table 5
Classification accuracies yielded by the different feature fusion approaches for the Houston University 2013 dataset.

Classes	# Samples		RGB	HSI	CHSI	SubFus	TV-SFF	Our
	Train	Test			+ RGB	Rasti and Ghamisi (2020)	Ramírez et al. (2021)	
Healthy grass	198	1053	80.56	82.60	81.65	71.13	66.60	80.92
Stressed grass	190	1064	75.80	83.38	70.17	82.79	83.24	82.02
Synthetic grass	192	505	94.26	97.78	97.11	100.00	100.00	100.00
Trees	188	1056	85.94	91.49	84.22	95.11	73.66	89.20
Soil	186	1056	90.84	96.80	95.72	97.97	96.85	97.92
Water	182	143	93.29	99.16	90.98	95.38	99.58	95.38
Residential	196	1072	62.54	74.96	86.08	79.53	73.61	81.54
Commercial	191	1053	30.46	32.90	42.17	41.40	61.32	52.27
Road	193	1059	60.66	68.40	73.36	74.58	79.57	88.22
Highway	191	1036	36.81	43.54	33.93	61.20	61.33	63.12
Railway	181	1054	57.32	70.18	61.00	76.86	80.46	92.97
Parking lot 1	192	1041	41.86	55.06	71.41	81.81	93.92	86.61
Parking lot 2	184	285	48.67	60.46	60.00	80.07	70.74	74.25
Tennis court	181	247	93.93	99.15	97.09	99.19	100.00	99.55
Running track	187	473	96.58	97.55	95.98	99.64	100.00	98.92
Overall accuracy (%)			65.70	72.95	72.75	78.95	79.49	**83.31**
Average accuracy (%)			69.97	76.89	76.06	82.44	82.72	**85.53**
Kappa statistic (κ)			0.630	0.710	0.707	0.772	0.777	**0.820**

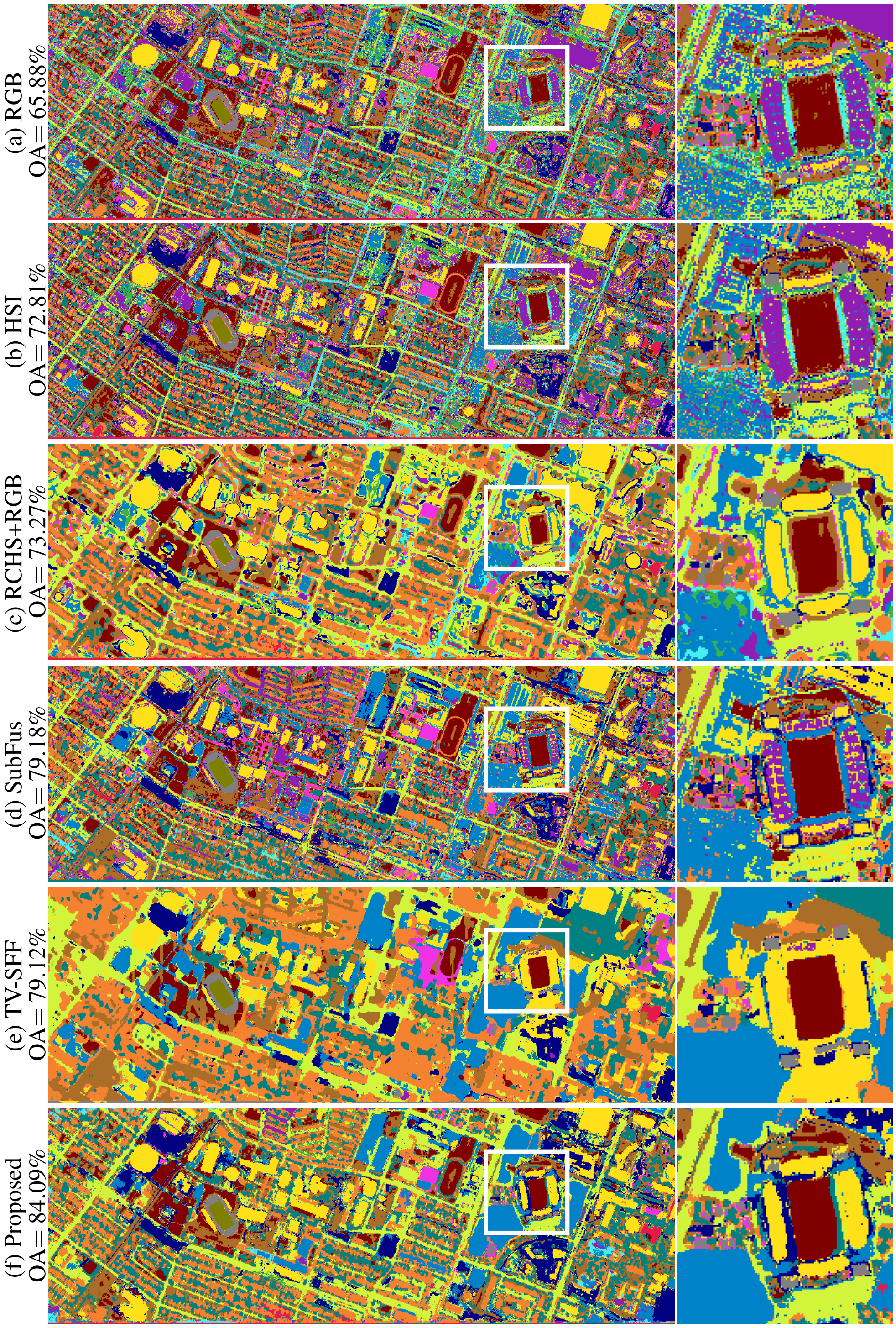
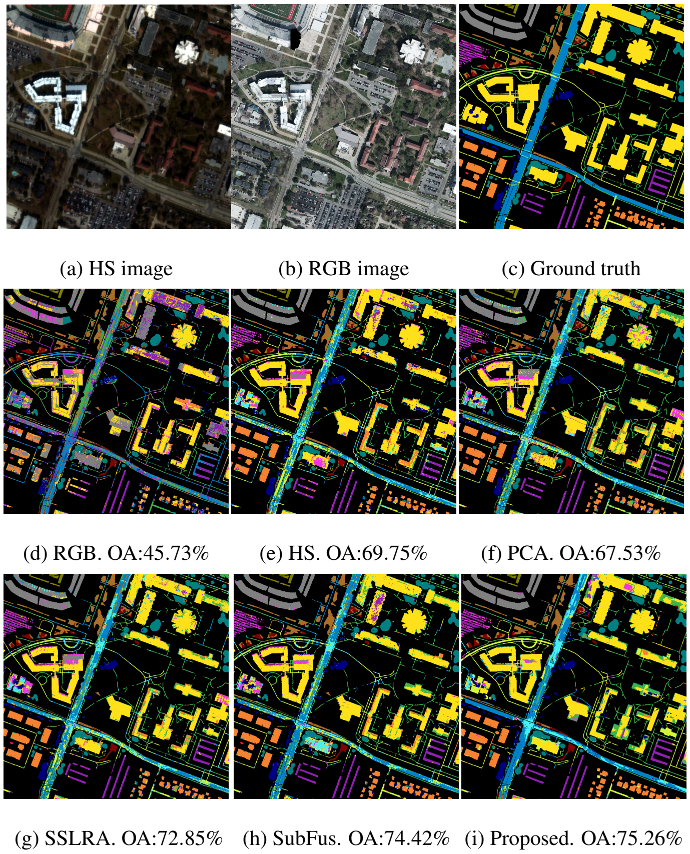


Figure 17. Houston University 2013. Classification maps yielded by (a) the RGB image, (b) the HSI, (c) the stacking of the RGB image with an interpolated verion of the HSI (RGB+HSI), (d) the SubFus method, (e) the proposed approach.

fusing information from multi-sensor data, the PCA obtained from stacking the RGB image and an interpolated version of the HS image is determined. Then, a set of $N_r = 16$ PCA bands is selected to evaluate the classification performance. Fig. 18(e) shows the labeling map yielded by the PCA. We also apply the SSLRA method Rasti et al. (2019) to the stacked data set whose classification map is illustrated in Fig. 18(f). Furthermore, Fig. 18(g) shows the labeling map yielded by the SubFus method Rasti and Ghamisi (2020) from HS and RGB images. Finally, the labeling map obtained by the proposed feature fusion method is illustrated in Fig. 18(i). Parameter setting of the proposed feature fusion approach is fixed to $N_r = 16$ and $\lambda = 0.05$.


(a) HS image (b) RGB image (c) Ground truth

(d) RGB. OA:45.73% (e) HS. OA:69.75% (f) PCA. OA:67.53%

(g) SSLRA. OA:72.85% (h) SubFus. OA:74.42% (i) Proposed. OA:75.26%

Figure 18. Houston data set. (a) RGB image, (b) the RGB composite of the HS image, and (c) the ground truth map. (d)-(i) Labeling maps obtained by various the methods with their respective OA.

Color	Classes	# Samples		PCA	SSLRA	SubFus	Proposed
		Train	Test		Rasti et al. (2019)	Rasti and Ghamisi (2020)	
🟥	Healthy grass	75	5729	90.08	90.89	91.35	**91.39**
🟫	Stressed grass	75	20055	85.06	**86.64**	85.00	85.91
🟩	Synthetic grass	75	2639	99.51	99.57	99.51	**100.00**
🟩	Evergreen trees	75	25387	94.45	**94.78**	93.80	93.90
🟦	Deciduous trees	75	11652	93.76	**94.62**	94.27	94.33
🟥	Water	75	55	98.18	99.55	**99.64**	99.27
🟧	Residential	75	31853	83.74	87.49	91.59	**96.42**
🟨	Commercial	75	191549	65.62	72.29	**72.59**	71.31
🟩	Road	75	27971	47.56	52.87	54.67	**67.45**
🟩	Sidewalk	75	49354	52.06	57.79	54.39	**59.51**
🟦	Crosswalk	75	1754	69.82	76.60	78.02	**85.16**
🟦	Major thoroughfares	75	71007	50.95	55.32	58.90	**65.32**
🟪	Paved Parking	75	16374	94.58	95.37	94.91	**96.88**
🟪	Cars	75	4392	80.20	86.52	90.20	**94.84**
⬜	Seats	75	26959	88.80	92.41	**98.33**	98.12
Overall accuracy (%)				68.11	72.93	73.82	**76.00**
				±1.52	±1.41	±1.64	**±1.24**
Average accuracy (%)				79.63	82.85	83.81	**86.65**
				±0.50	±0.51	±0.35	**±0.46**
Kappa Statistic				0.624	0.678	0.688	**0.713**
				0.016	0.015	0.017	**±0.014**

Table 6
Labeling accuracies yielded by the different feature extraction methods.

To quantitatively evaluate the performance of the proposed method, Table 6 shows the classification accuracy obtained by the various feature fusion approaches. Specifically, every accuracy value is obtained by averaging 20 realizations of the respective experiment and at each trial, a different set of training samples is randomly selected. Furthermore, the overall accuracy, the average accuracy, and the Kappa statistic are shown in the last three rows of Table 6 for the various feature fusion techniques. Notice that the best accuracy values are shown in bold font. As can be observed in this table, the proposed method exhibit a competitive performance compared to the other feature fusion methods.

Table 7 displays the labeling accuracies using different feature fusion approaches for the

Houston University 2018 dataset. More precisely, we include the results yielded by the HS image, the RHSI+RGB approach, the feature fusion based on the orthogonal total variation component analysis (OTVCA_Fus) Lorenz et al. (2019), the SubFus method Rasti and Ghamisi (2020), and our approach. For this table, we select 10% of the ground truth pixels as training samples and the remaining 90% as testing samples. The parameters of the proposed approach are set to $\lambda = 1 \times 10^{-4}$ and $N_r = 12$. In addition, the morphological operations use disk shape masks with sizes (20, 50, 100, 200). Each accuracy value is obtained by averaging 10 realizations of the respective experiment, where every trial randomly generates a new set of training samples. As can be seen in Table 7, the proposed method exhibits a competitive performance with respect to other state-of-the-art approaches. Even, the proposed approach provides better results than the other approaches in terms of OA, AA, and κ. Finally, Fig. 19 shows the overall accuracy (OA) obtained by various feature fusion approaches for different training sample ratios. As a result, the proposed approach outperforms other feature fusion methods for the entire evaluation interval with an accuracy gain of at least 4%.

3.5. Conclusions

A feature fusion method was proposed using the subspace-based approach and an ℓ_0-norm regularization for spectral image classification from HSI observations and RGB images. More precisely, the proposed feature fusion method was developed under the assumption that the morphological profiles (MP) of the RGB image and the HSI measurements can be described as a high-resolution feature matrix lying in different subspaces. In contrast to previous works, the proposed method has been developed as a joint optimization framework that included the ADMM approach

Table 7
Classification accuracies yielded from various feature fusion approaches for the Houston University 2018 dataset.

Classes	# Samples		HSI	CHSI	OTVCA_Fus	SubFus	TV-SFF	Our
	Train	Test		+ RGB	Lorenz et al. (2019)	Rasti and Ghamisi (2020)	Ramírez et al. (2021)	
Healthy grass	3290	35276	86.39 ± 3.08	86.53 ± 3.31	84.27 ± 2.09	68.30 ± 4.87	77.29 ± 0.83	**88.49 ± 0.52**
Stressed grass	13001	117007	93.88 ± 0.46	93.93 ± 0.64	93.07 ± 0.84	85.68 ± 3.18	91.45 ± 0.20	**95.32 ± 0.20**
Artificial turf	274	2462	70.69 ± 24.05	70.90 ± 23.48	86.83 ± 10.33	99.80 ± 0.10	**99.84 ± 0.09**	99.77 ± 0.07
Evergreen trees	5432	48890	92.31 ± 0.65	92.42 ± 0.85	91.57 ± 0.96	67.19 ± 7.85	87.20 ± 0.61	**94.69 ± 0.29**
Deciduous trees	2017	18155	37.61 ± 6.53	39.72 ± 6.35	37.41 ± 6.47	39.54 ± 11.00	48.96 ± 3.34	**79.95 ± 1.44**
Bare earth	1806	16258	60.95 ± 13.18	67.60 ± 13.58	76.37 ± 7.82	76.17 ± 12.93	84.58 ± 2.08	**97.44 ± 0.33**
Water	106	958	85.73 ± 5.86	86.68 ± 4.33	91.29 ± 4.01	87.84 ± 4.75	88.36 ± 2.02	**95.23 ± 0.87**
Residential	15900	143095	70.14 ± 3.56	71.21 ± 2.69	71.07 ± 2.21	91.11 ± 0.99	89.33 ± 0.60	**94.19 ± 0.41**
Non-residential	89477	805292	91.41 ± 0.58	92.75 ± 0.61	92.41 ± 1.10	90.88 ± 1.39	94.98 ± 0.25	**97.58 ± 0.13**
Roads	18328	164955	38.19 ± 4.95	39.83 ± 5.02	40.46 ± 4.20	46.95 ± 5.77	59.61 ± 2.37	**80.29 ± 1.14**
Sidewalks	13604	122431	35.77 ± 3.99	41.23 ± 3.97	44.51 ± 3.87	33.65 ± 8.04	47.18 ± 1.98	**71.03 ± 1.31**
Crosswalks	606	5453	0.16 ± 0.21	0.16 ± 0.23	0.21 ± 0.23	0.97 ± 1.68	1.53 ± 0.53	**16.55 ± 2.05**
Thoroughfares	18544	166894	48.31 ± 4.88	49.82 ± 5.07	51.32 ± 4.97	65.91 ± 5.21	73.27 ± 1.57	**89.17 ± 0.76**
Highways	3944	35494	32.84 ± 12.20	36.39 ± 11.35	43.35 ± 6.13	54.82 ± 6.13	65.71 ± 2.02	**88.98 ± 0.96**
Railways	2775	24973	59.10 ± 14.75	60.66 ± 13.69	61.56 ± 8.12	94.53 ± 3.37	93.19 ± 0.60	**99.02 ± 0.18**
Paved parking	4593	41339	14.28 ± 16.08	23.67 ± 16.39	38.16 ± 0.12	52.92 ± 11.42	74.85 ± 3.78	**95.14 ± 0.72**
Unpaved parking	59	528	11.74 ± 17.14	13.41 ± 20.73	25.36 ± 0.26	26.33 ± 28.60	64.19 ± 7.43	**99.22 ± 0.99**
Cars	2629	23660	1.11 ± 1.55	40.23 ± 4.10	14.34 ± 6.47	38.27 ± 6.27	40.48 ± 2.01	**53.58 ± 1.23**
Trains	2148	19331	12.63 ± 10.54	14.23 ± 11.87	23.73 ± 7.01	**83.09 ± 3.96**	69.95 ± 0.98	81.29 ± 1.52
Stadium seats	2730	24566	50.90 ± 11.27	54.87 ± 10.82	54.42 ± 7.41	85.56 ± 9.89	79.92 ± 1.41	**91.37 ± 0.55**
Overall accuracy (%)			70.24 ± 3.08	72.08 ± 3.10	73.00 ± 2.80	76.07 ± 3.47	82.26 ± 0.92	**91.33 ± 0.45**
Average accuracy (%)			49.71 ± 7.50	52.00 ± 7.64	56.09 ± 6.03	64.47 ± 6.14	71.59 ± 1.56	**85.42 ± 0.60**
Kappa statistic (κ)			0.594 ± 0.047	0.619 ± 0.047	0.636 ± 0.040	0.680 ± 0.048	0.763 ± 0.013	**0.886 ± 0.006**

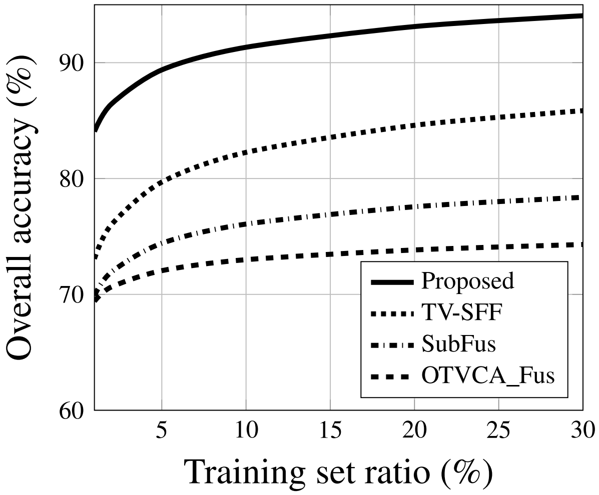


Figure 19. OA versus the training set ratio for different feature fusion approaches.

and the block coordinate descent technique. Furthermore, to exploit the rich spatial information embedded in RGB images, the ADMM optimization stage focused on minimizing a cost function regularized by the ℓ_0-norm of the feature band gradient. The proposed approach was evaluated in the context of land cover classification using two scenarios. First, we tested the feature fusion method by simulating both compressive HSI data and an RGB projection. In addition, the proposed method was evaluated on a real dataset. The numerical results shown an outstanding performance in terms of classification accuracy for different parameter settings. Additionally, we illustrated that the proposed feature fusion method outperforms other state-of-the-art multi-sensor feature fusion approaches on simulated and real datasets. In future works, we are interested in considering other multimodal data such as light detection and ranging (LiDAR) and synthetic aperture radar (SAR). Furthermore, as further research lines, we are also interested in developing unsupervised methods for extracting the image degradation models that enable an accurate problem formulation.

Discussion, Conclusions and Future Works

Discussion: The utilization of compressive classification techniques in conjunction with multi-channel images has brought about a significant paradigm shift in the realm of image analysis Vargas et al. (2018). This innovative approach has profoundly impacted new research by offering enhanced efficiency, reduced data storage requirements, and improved accuracy object classification. By compressing multi-channel images data while preserving essential spectral information, researchers can expedite data transmission and storage processes, enabling quicker access to vital information. This advancement not only optimizes resource allocation but also facilitates the exploration of large-scale and time-sensitive applications. Some results of these investigations have motivated the following works Kwan et al. (2019); Machidon and Pejovic (2021); Lucena et al. (2021). On the other hand, solving regularization-based problems using the Alternating Direction Method of Multipliers (ADMM) is an important and widely used approach in optimization and signal processing Vargas and Arguello (2019). ADMM is a powerful optimization technique that can effectively address a variety of problems, including those involving ℓ_1 regularization, and it has significant implications for many fields Jurdana et al. (2021); Wang et al. (2022).

On the other hand, it is essential to acknowledge certain limitations associated with the proposed methodology. The primary limitation pertains to the determination of the optimal number of samples required to maintain classification accuracy. While there are existing theoretical studies that address the minimization of measurements needed to reconstruct the full data cube, the literature lacks insight into the selection of the minimal number of measurements necessary for feature extraction while preserving classification accuracy. The second limitation is linked to real-world implementation. The single-pixel hyperspectral camera requires long acquisition times due to the integration periods of the hyperspectral sensor. This acquisition period poses challenges in scenarios involving remote sampling, especially when aircraft are in constant motion, leading to potential misalignment issues over time. The third aspect, equally significant, pertains to the integration of alternative modalities. This study primarily emphasized the utilization of a high-resolution RGB camera, chosen for its capacity to extract intricate details at high resolutions while maintaining low sensor noise. When contemplating the inclusion of other modalities like light detection and ranging (LiDAR) and synthetic aperture radar (SAR), it introduces complexities to the proposed methodology due to the inherent noise models associated with these types of images.

Conclusions: Hyperspectral imaging systems offer much more information than conventional digital cameras and have received increasing attention in a wide range of applications. However, the challenges faced when implementing new technology have unfortunately held back many new developments due to high costs and computational times. Fortunately, recent advances in signal processing have suggested partial alternatives that alleviate these problems. The main objective of this work was to investigate these alternatives and their integration with the compressive sensing theory, in hyperspectral systems. The contributions of this thesis are framed in the three main components: Feature extraction model for hyperspectral images through compressive sensors, reconstruction of these features from their compressive measurements, and detailed analysis of the model performance and proposed estimation method. The achievements and conclusions can be summarized as follows:

- We found that such multiplexing strategies have a significant impact on the performance of many existing methods that exploit the high-dimensional structures presenting data types. This approach allows for the acquisition of hyperspectral data with reduced data volume, making it more efficient in terms of storage and transmission. This advancement is crucial for applications where data size is a concern, such as remote sensing and unmanned aerial vehicles (UAVs). Also, Chapter 2 investigated a strategy that exploits the low-rank structure available for hyperspectral capture and produces a feature extraction method, which uses subspace modeling and spatial regularization. This innovative approach effectively bypasses established reconstruction techniques and has demonstrated comparable performance to methods using 100% of the data in terms of classification accuracy metrics.

- We found that combining the single-pixel hyperespectral architecture with a high-resolution RGB sensor enables the acquisition of hyperspectral data with both high spatial and spectral resolution. The RGB sensor captures detailed spatial information, while the hyperspectral component provides rich spectral data. This combination is advantageous in applications such as remote sensing, where the identification of specific materials or objects depends on the spatial resolution of the scene. In Chapter 3, we present a multisensor-based model that independently combines spatial and spectral property preservation applied to feature extraction. In addition, an optimization problem for feature extraction was investigated, which takes advantage of these two desired properties by minimizing the ℓ_0 gradient regularization norm, and the ℓ_2-norm. The efficacy of this method was demonstrated through algorithmic implementation using ADMM, and an improvement in classification accuracy metric over existing techniques was achieved and demonstrated through experimentation.

- We found that the Alternating Direction Method of Multipliers (ADMM) to solve the compressive feature extraction problem has distinct advantages. ADMM is a versatile optimization technique that can be applied to a wide range of feature extraction problems. It allows for the incorporation of various constraints and regularization terms to tailor the feature extraction process to specific requirements. Additionally, ADMM is known for its strong convergence guarantees, which ensure that the optimization process converges to a solution, even for non-convex problems. This reliability is essential in feature extraction, where finding a global minimum is often challenging.

In conclusion, the benefits of extracting features from compressed measurements in hyperspectral imaging and combining a single-pixel architecture with a high-resolution RGB sensor are numerous. These advantages encompass efficiency, cost-effectiveness, improved data quality, and enhanced capabilities for various applications, ultimately enabling more effective hyperspectral data acquisition and analysis.

Future Works: The research and study carried out in this dissertation can be continued and extended in several directions. Some interesting ideas and future works are listed below.

- The thesis was focused on hyperspectral images and RGB, we are interested in considering other multimodal data such as multispectral, light detection and ranging (LiDAR) and synthetic aperture radar (SAR).

- In the models used in the thesis, the noise of the compressive hyperspectral image is assumed to be zero-mean additive Gaussian. Although, the performance of the techniques are satisfactory, a future work is to take into consideration the other types of noise such as heteroskedastic noise modeling.

- Applying different types of penalties (for example Plug-and-Play with convolutional neural networks scheme as a regularizer) rather than ℓ_0 gradient regularization for feature extraction can be the next step for further study.

- Identifying the subspace dimension is a crucial step in most hyperspectral algorithms. As future work, it is possible to investigate the problem of determining the subspace dimension of the hyperspectral image using the method based on Stein's unbiased risk estimator (SURE).

On the other hand, future research could explore different options for the sampling operator that allow other types of decompositions, such as spectral unmixing, while maintaining the properties that enable an efficient solution to the compressive feature extraction problem. Furthermore, when analyzing the classifications obtained by the algorithms, additional work is required to obtain higher-quality pixel labeling from the extracted features. The proposed algorithms can be applied as a preprocessing step for deep learning-based classification methods and could also be used to enhance the training and verification of neural networks for hyperspectral image classification.

Bibliography

Almeida, M. S. and Figueiredo, M. (2013). Deconvolving images with unknown boundaries using the alternating direction method of multipliers. *IEEE Transactions on Image processing*, 22(8):3074–3086.

Arguello, H. and Arce, G. R. (2011). Code aperture optimization for spectrally agile compressive imaging. *JOSA A*, 28(11):2400–2413.

Arguello, H. and Arce, G. R. (2013). Rank minimization code aperture design for spectrally selective compressive imaging. *IEEE Transactions on Image Processing*, 22(3):941–954.

Arguello, H. and Arce, G. R. (2014). Colored coded aperture design by concentration of measure in compressive spectral imaging. *IEEE Transactions on Image Processing*, 23(4):1896–1908.

Bacca, J., Correa, C. V., and Arguello, H. (2019). Noniterative hyperspectral image reconstruction from compressive fused measurements. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Bardsley, J. M. (2012). Laplace-distributed increments, the laplace prior, and edge-preserving regularization. *Journal of Inverse and Ill-Posed Problems*, 20(3):271–285.

Báscones, D., González, C., and Mozos, D. (2018). Hyperspectral image compression using vector quantization, pca and jpeg2000. *Remote sensing*, 10(6):907.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202.

Bertsekas, D. (1999). *Nonlinear programming*. Athena Scientific optimization and computation series. Athena Scientific.

Bioucas-Dias, J. M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N., and Chanussot, J. (2013). Hyperspectral remote sensing data analysis and future challenges. *IEEE Geoscience and remote sensing magazine*, 1(2):6–36.

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. *IEEE journal of selected topics in applied earth observations and remote sensing*, 5(2):354–379.

Boutsidis, C. and Gittens, A. (2013). Improved matrix algorithms via the subsampled randomized hadamard transform. *SIAM Journal on Matrix Analysis and Applications*, 34(3):1301–1340.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122.

Brady, D. J. (2009). *Optical imaging and spectroscopy*. John Wiley & Sons.

Breuer, M. and Albertz, J. (2000). Geometric correction of airborne whiskbroom scanner imagery using hybrid auxiliary data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1; PART 3):93–100.

Camps-Valls, G. and Bruzzone, L. (2005). Kernel-based methods for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 43(6):1351–1362.

Candes, E. and Romberg, J. (2007). Sparsity and incoherence in compressive sampling. *Inverse problems*, 23(3):969.

Candes, E. J. and Tao, T. (2006). Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425.

Cao, X. and Liu, K. R. (2018). Distributed linearized admm for network cost minimization. *IEEE Transactions on Signal and Information Processing over Networks*, 4(3):626–638.

Cascarano, P., Calatroni, L., and Piccolomini, E. L. (2021). Efficient ℓ_0 gradient-based super-resolution for simplified image segmentation. *IEEE Transactions on Computational Imaging*, 7:399–408.

Cawse-Nicholson, K., Damelin, S. B., Robin, A., and Sears, M. (2012). Determining the intrinsic dimension of a hyperspectral image using random matrix theory. *IEEE Transactions on Image Processing*, 22(4):1301–1310.

Chen, C., Li, W., Tramel, E. W., and Fowler, J. E. (2014). Reconstruction of hyperspectral imagery from random projections using multihypothesis prediction. *IEEE Transactions on Geoscience and Remote Sensing*, 52(1):365–374.

Chen, Y., Nasrabadi, N. M., and Tran, T. D. (2011a). Hyperspectral image classification using dictionary-based sparse representation. *IEEE Transactions on Geoscience and Remote Sensing*, 49(10):3973–3985.

Chen, Y., Nasrabadi, N. M., and Tran, T. D. (2011b). Sparse representation for target detection in hyperspectral imagery. *IEEE Journal of Selected Topics in Signal Processing*, 5(3):629–640.

Dale, L. M., Thewis, A., Boudry, C., Rotar, I., Dardenne, P., Baeten, V., and Pierna, J. A. F. (2013). Hyperspectral imaging applications in agriculture and agro-food product quality and safety control: A review. *Applied Spectroscopy Reviews*, 48(2):142–159.

Dalla Mura, M., Atli Benediktsson, J., Waske, B., and Bruzzone, L. (2010). Extended profiles with morphological attribute filters for the analysis of hyperspectral data. *International Journal of Remote Sensing*, 31(22):5975–5991.

Davis, C. and Kahan, W. M. (1970). The rotation of eigenvectors by a perturbation. iii. *SIAM Journal on Numerical Analysis*, 7(1):1–46.

Debes, C., Merentitis, A., Heremans, R., Hahn, J., Frangiadakis, N., van Kasteren, T., Liao, W., Bellens, R., Pižurica, A., Gautama, S., et al. (2014). Hyperspectral and lidar data fusion: Outcome of the 2013 grss data fusion contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6):2405–2418.

Do, T. T., Gan, L., Nguyen, N. H., and Tran, T. D. (2012). Fast and efficient compressive sensing using structurally random matrices. *IEEE Transactions on Signal Processing*, 60(1):139–154.

Duarte, M. F., Davenport, M. A., Takhar, D., Laska, J. N., Sun, T., Kelly, K. E., Baraniuk, R. G., et al. (2008). Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):83.

Elad, M., Milanfar, P., and Rubinstein, R. (2007). Analysis versus synthesis in signal priors. *Inverse problems*, 23(3):947.

Fauvel, M., Tarabalka, Y., Benediktsson, J. A., Chanussot, J., and Tilton, J. C. (2012). Advances in spectral-spatial classification of hyperspectral images. *Proceedings of the IEEE*, 101(3):652–675.

Fowler, J. E. (2014). Compressive pushbroom and whiskbroom sensing for hyperspectral remote-sensing imaging. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 684–688. IEEE.

Garcia, H., Correa, C. V., and Arguello, H. (2018). Multi-resolution compressive spectral imaging reconstruction from single pixel measurements. *IEEE Transactions on Image Processing*, 27(12):6174–6184.

Garcia, H., Correa, C. V., and Arguello, H. (2020). Optimized sensing matrix for single pixel multi-resolution compressive spectral imaging. *IEEE Transactions on Image Processing*, 29:4243–4253.

Gat, N. (2000). Imaging spectroscopy using tunable filters: a review. In *AeroSense 2000*, pages 50–64. International Society for Optics and Photonics.

Gehm, M., Kim, M., Fernandez, C., and Brady, D. (2008). High-throughput, multiplexed pushbroom hyperspectral microscopy. *Optics express*, 16(15):11032–11043.

Gislason, P. O., Benediktsson, J. A., and Sveinsson, J. R. (2006). Random forests for land cover classification. *Pattern recognition letters*, 27(4):294–300.

Golay, M. J. (1949). Multi-slit spectrometry. *JOSA*, 39(6):437–444.

Golbabaee, M., Arberet, S., and Vandergheynst, P. (2013). Compressive source separation: Theory and methods for hyperspectral imaging. *IEEE Transactions on Image Processing*, 22(12):5096–5110.

Grippo, L. and Sciandrone, M. (2000). On the convergence of the block nonlinear gauss–seidel method under convex constraints. *Operations research letters*, 26(3):127–136.

Grupo de Inteligencia Computacional (2008). Hyper Remote Sensing Scenes. http://www.ehu.eus/.

Guan, Y., Guo, S., Xue, Y., Liu, J., and Zhang, X. (2004). Application of airborne hyperspectral data for precise agriculture. In *Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International*, volume 6, pages 4195–4198. Ieee.

Jurdana, V., Volaric, I., and Sucic, V. (2021). Sparse time-frequency distribution reconstruction based on the 2d rényi entropy shrinkage algorithm. *Digital Signal Processing*, 118:103225.

Keshava, N. and Mustard, J. F. (2002). Spectral unmixing. *IEEE signal processing magazine*, 19(1):44–57.

Kwan, C., Gribben, D., and Tran, T. (2019). Multiple human objects tracking and classification directly in compressive measurement domain for long range infrared videos. In *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pages 0469–0475. IEEE.

Lai, R. and Osher, S. (2014). A splitting method for orthogonality constrained problems. *Journal of Scientific Computing*, 58(2):431–449.

Li, C., Sun, T., Kelly, K. F., and Zhang, Y. (2012). A compressive sensing and unmixing scheme for hyperspectral data processing. *IEEE Transactions on Image Processing*, 21(3):1200–1210.

Liao, W., Chanussot, J., Dalla Mura, M., Huang, X., Bellens, R., Gautama, S., and Philips, W. (2017). Taking optimal advantage of fine spatial resolution: Promoting partial image reconstruction for the morphological analysis of very-high-resolution images. *IEEE geoscience and remote sensing magazine*, 5(2):8–28.

Lin, X., Liu, Y., Wu, J., and Dai, Q. (2014). Spatial-spectral encoded compressive hyperspectral imaging. *ACM Transactions on Graphics (TOG)*, 33(6):1–11.

Lopez, S., Vladimirova, T., Gonzalez, C., Resano, J., Mozos, D., and Plaza, A. (2013). The promise of reconfigurable computing for hyperspectral imaging onboard systems: A review and trends. *Proceedings of the IEEE*, 101(3):698–722.

Lorente, D., Aleixos, N., Gómez-Sanchis, J., Cubero, S., García-Navarrete, O. L., and Blasco,

J. (2012). Recent advances and applications of hyperspectral imaging for fruit and vegetable quality assessment. *Food and Bioprocess Technology*, 5(4):1121–1142.

Lorenz, S., Seidel, P., Ghamisi, P., Zimmermann, R., Tusa, L., Khodadadzadeh, M., Contreras, I. C., and Gloaguen, R. (2019). Multi-sensor spectral imaging of geological samples: A data fusion approach using spatio-spectral feature extraction. *Sensors*, 19(12):2787.

Lucena, L. V., Correa, C. V., and Arguello, H. (2021). Automatic motion segmentation of spectral videos in the compressed domain using a fully convolutional network. In *2021 XXIII Symposium on Image, Signal Processing and Artificial Vision (STSIVA)*, pages 1–6. IEEE.

Machidon, A. L. and Pejovic, V. (2021). Deep learning techniques for compressive sensing-based reconstruction and inference–a ubiquitous systems perspective. *arXiv preprint arXiv:2105.13191*.

Martín, G. and Bioucas-Dias, J. M. (2016). Hyperspectral blind reconstruction from random spectral projections. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9(6):2390–2399.

Martín, G., Bioucas-Dias, J. M., and Plaza, A. (2015). Hyca: A new technique for hyperspectral compressive sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 53(5):2819–2831.

Nakatsukasa, Y. (2018). Sharp error bounds for ritz vectors and approximate singular vectors. *arXiv preprint arXiv:1810.02532*.

Nascimento, J. M., Véstias, M. P., and Martín, G. (2020). Hyperspectral compressive sensing with a system-on-chip fpga. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:3701–3710.

Ouyang, Y., Chen, Y., Lan, G., and Pasiliao Jr, E. (2015). An accelerated linearized alternating direction method of multipliers. *SIAM Journal on Imaging Sciences*, 8(1):644–681.

Parikh, N., Boyd, S., et al. (2014). Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239.

Plaza, A., Benediktsson, J. A., Boardman, J. W., Brazile, J., Bruzzone, L., Camps-Valls, G., Chanussot, J., Fauvel, M., Gamba, P., Gualtieri, A., et al. (2009). Recent advances in techniques for hyperspectral image processing. *Remote sensing of environment*, 113:S110–S122.

Ramírez, J., Vargas, H., Martínez, J. I., and Arguello, H. (2021). Subspace-based feature fusion from hyperspectral and multispectral images for land cover classification. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 3003–3006. IEEE.

Ramirez, J. M. and Arguello, H. (2019). Multiresolution compressive feature fusion for spectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(12):9900–9911.

Ramirez, J. M. and Arguello, H. (2019). Spectral image classification from multi-sensor compressive measurements. *IEEE Transactions on Geoscience and Remote Sensing*, 58(1):626–636.

Ramirez, J. M., Martínez, J. T. I., and Arguello, H. (2021). Feature fusion via dual-resolution compressive measurement matrix analysis for spectral image classification. *Signal Processing: Image Communication*, 90:116014.

Rasti, B. and Ghamisi, P. (2020). Remote sensing image classification using subspace sensor fusion. *Information Fusion*, 64:121–130.

Rasti, B., Ghamisi, P., and Gloaguen, R. (2017). Hyperspectral and lidar fusion using extinction profiles and total variation component analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3997–4007.

Rasti, B., Ghamisi, P., and Ulfarsson, M. O. (2019). Hyperspectral feature extraction using sparse and smooth low-rank analysis. *Remote Sensing*, 11(2):121.

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.

Rasti, B., Ulfarsson, M. O., and Sveinsson, J. R. (2016). Hyperspectral feature extraction using total variation component analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 54(12):6976–6985.

Romberg, J. (2008). Imaging via compressive sampling [introduction to compressive sampling and recovery via convex programming]. *IEEE Signal Processing Magazine*, 25(2):14–20.

Simoes, M., Bioucas-Dias, J., Almeida, L. B., and Chanussot, J. (2014). A convex formulation for hyperspectral image superresolution via subspace-based regularization. *IEEE Transactions on Geoscience and Remote Sensing*, 53(6):3373–3388.

Smith, W. L., Zhou, D. K., Harrison, F. W., Revercomb, H. E., Larar, A. M., Huang, H.-L., and Huang, B. (2001). Hyperspectral remote sensing of atmospheric profiles from satellites and aircraft. In *Second International Asia-Pacific Symposium on Remote Sensing of the Atmosphere, Environment, and Space*, pages 94–102. International Society for Optics and Photonics.

Tao, C., Zhu, H., Wang, X., Zheng, S., Xie, Q., Wang, C., Wu, R., and Zheng, Z. (2021). Compressive single-pixel hyperspectral imaging using rgb sensors. *Optics express*, 29(7):11207–11220.

Tropp, J. A., Wakin, M. B., Duarte, M. F., Baron, D., and Baraniuk, R. G. (2006). Random filters for compressive sampling and reconstruction. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 3, pages III–III. IEEE.

Tseng, P. (2001). Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494.

Vaidyanathan, P. (2001). Generalizations of the sampling theorem: Seven decades after nyquist. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 48(9):1094–1109.

Vargas, H. and Arguello, H. (2019). A low-rank model for compressive spectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(12):9888–9899.

Vargas, H., Fonseca, Y., and Arguello, H. (2018). Object detection on compressive measurements using correlation filters and sparse representation. In *2018 26th European signal processing conference (EUSIPCO)*, pages 1960–1964. IEEE.

Wang, J., Xie, F., Nie, F., and Li, X. (2022). Robust supervised and semisupervised least squares regression using l2,p-norm minimization. *IEEE Transactions on Neural Networks and Learning Systems*.

Wang, L., Feng, Y., Gao, Y., Wang, Z., and He, M. (2018). Compressed sensing reconstruction of hyperspectral images based on spectral unmixing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(4):1266–1284.

Wang, M., Wang, Q., Chanussot, J., and Hong, D. (2021). l_0-l_1 hybrid total variation regularization and its applications on hyperspectral image mixed noise removal and compressed sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 59(9):7695–7710.

Wei, Q., Dobigeon, N., and Tourneret, J.-Y. (2015). Fast fusion of multi-band images based on solving a sylvester equation. *IEEE Transactions on Image Processing*, 24(11):4109–4121.

Wen, Z. and Yin, W. (2013). A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434.

Xu, L., Lu, C., Xu, Y., and Jia, J. (2011). Image smoothing via l 0 gradient minimization. In *Proceedings of the 2011 SIGGRAPH Asia conference*, pages 1–12.

Xu, Y., Du, B., Zhang, L., Cerra, D., Pato, M., Carmona, E., Prasad, S., Yokoya, N., Hänsch, R., and Le Saux, B. (2019). Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 ieee grss data fusion contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(6):1709–1724.

Yang, M., de Hoog, F., Fan, Y., and Hu, W. (2015a). Compressive hyperspectral imaging via adaptive sampling and dictionary learning. *arXiv preprint arXiv:1512.00901*.

Yang, S., Jin, H., Wang, M., Ren, Y., and Jiao, L. (2014). Data-driven compressive sampling and learning sparse coding for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 11(2):479–483.

Yang, S., Ma, Y., Wang, M., Xie, D., Wu, Y., and Jiao, L. (2013). Compressive feature and kernel sparse coding-based radar target recognition. *IET Radar, Sonar & Navigation*, 7(7):755–763.

Yang, S., Wang, M., Li, P., Jin, L., Wu, B., and Jiao, L. (2015b). Compressive hyperspectral imaging via sparse tensor and nonlinear compressed sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 53(11):5943–5957.

Zhao, N., Wei, Q., Basarab, A., Dobigeon, N., Kouamé, D., and Tourneret, J.-Y. (2016). Fast single image super-resolution using a new analytical solution for $\ell_2 - \ell_2$ problems. *IEEE Transactions on Image Processing*, 25(8):3683–3697.

Zhu, L., Suomalainen, J., Liu, J., Hyyppä, J., Kaartinen, H., Haggren, H., et al. (2018). A review: Remote sensing sensors. *Multi-purposeful application of geospatial data*, pages 19–42.

Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286.

Apeendices

Indian Pines and Pavia University Data sets

The first data set is the Indian Pines from AVIRIS sensor that generates 220 bands across the spectral range from 0.2 to 2.4 μm. In the experiments, the number of bands is reduced to 200 by removing 20 water absorption bands. This image has spatial resolution of 20m per pixel and spatial dimension of 128×128 pixels. The second hyperspectral image used in this work, the University of Pavia, is an urban image acquired by the Reflective Optics System Imaging Spectrometer (ROSIS). The ROSIS sensor generates 115 spectral bands ranging from 0.43 to 0.86 μm and it has a spatial resolution of 1.3 m per pixel. The University of Pavia image consists of 256×256 pixels, and 103 bands. In Table 8, the number of classes, total samples available for each class and color label for both hyperspectral datasets are shown and Figure 20 represents the classification maps. The black pixels from the ground truth maps are unlabeled regions which are not taken into account in the classification results.

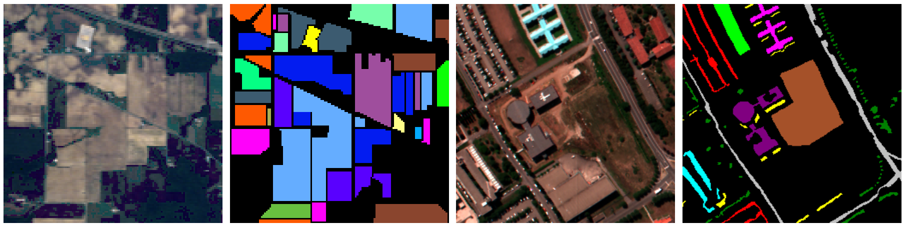


Figure 20. False color image and ground truth classification maps for Indian Pines (**1st** and **2nd**) and Pavia University (**3rd** and **4th**) datasets. Black pixels represent unlabeled regions.

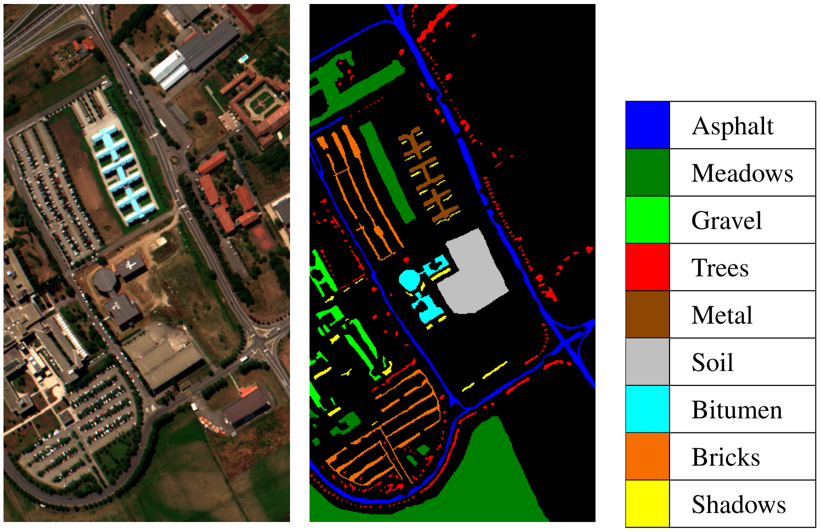
Table 8

Total labeled samples for classification of the Indian Pines and Pavia University data sets.

Indian			Pavia		
#	**Class name**	**Samp.**	**#**	**Class name**	**Samp.**
1	Alfalfa	46	1	Asphalt	6631
2	Corn-notill	1428	2	Meadows	18649
3	Corn-mintill	830	3	Gravel	2099
4	Corn	237	4	Trees	3064
5	Grass-pasture	483	5	Painted-metal-sheets	1345
6	Grass-tress	730	6	Bare Soil	5029
7	Grass-pasture-mowed	20	7	Bitumen	1330
8	Hay-windrowed	478	8	Self-Blocking Bricks	3682
9	Oats	20	9	Shadows	947
10	Soybean-notill	972			
11	Soybean-mintill	2455			
12	Soybean-clean	593			
13	Wheat	205			
14	Woods	1265			
15	Buildings-Grass Tress-Drives	386			
16	Stone-Steel-Towers	93			

Pavia University Data sets (full)

This dataset was collected by the Reflective Optics Imaging Spectrometer (ROSIS-03) over an urban area of the University of Pavia, Italy. In particular, the acquisition of the HS image was managed by the German Aerospace Agency and it was funded by the HySens Project. The captured dataset exhibits a high-spatial-resolution (1.3 m per pixel) with 610×340 pixels and 103 spectral channels that cover the wavelength interval from 0.43 to 0.84 μm Grupo de Inteligencia Computacional (2008). To evaluate the performance of the proposed feature fusion method, an RGB image is built by projecting the HS image using the IKONOS sensor color response Wei et al. (2015). Figure 21(a) displays the synthetic RGB image.



(a) RGB image (b) Ground truth (c) Labels

Figure 21. Pavia University dataset. (a) the RGB image, (b) a compressive HS projection, (c) and (d) ground truth data.

Houston University Data sets 2013

This spectral image was obtained in 2012 by the Compact Airborne Spectrographic Imager (CASI) over an urban area of the University of Houston, USA. This dataset was distributed for the 2013 IEEE Geoscience and Remote Sensing Society Data Fusion Contest (GRSS_DFC_2013) Debes et al. (2014). More precisely, this image exhibits dimensions of $344 \times 1,904$ pixels and 144 spectral bands in the wavelength range from 0.38 to 1.05 μm. To test the proposed method, an RGB composite is obtained by projecting the HSI using the IKONOS sensor response in the visible wavelength interval (0.38 - 0.86 μm). Fig. 22(a) displays the RGB composite of the Houston University 2013 dataset. Moreover, the HSI is obtained by downscaling each band of the the original HSI with a spatial decimation ratio of 4:1. In consequence, the HSI exhibits dimensions of 172×952 pixels and 144 image bands. A grayscale image of a single spectral band of the HSI is illustrated in Fig. 22(b). Furthermore, this dataset includes fifteen different classes, where each class label corresponds to a distinct structure in the urban cover. Finally, the training sample set and the test sample set are shown in Fig. 22(b) and Fig. 22(c), respectively.

Houston University Data sets 2018

In this case, the hyperspectral image was collected by the ITRES CASI 1500 camera over an urban area of the University of Houston Xu et al. (2019). Specifically, this image exhibits a spatial resolution of 1m per pixel with dimensions of 601×2384 pixels and 48 spectral bands in the wavelength interval from 0.38 to 1.05 μm. Fig. 22(a) displays the RGB composite of the HS image. Moreover, this database includes an RGB image that was captured by the DIMAC ULTRALiGHT+ sensor. This image exhibits dimensions of 8984×6732 pixels with a spatial resolution of 5 cm per pixel (Fig. 23(b)). In addition, Fig. 23(c) illustrates the ground truth map included in the database representing 20 classes in the urban cover.

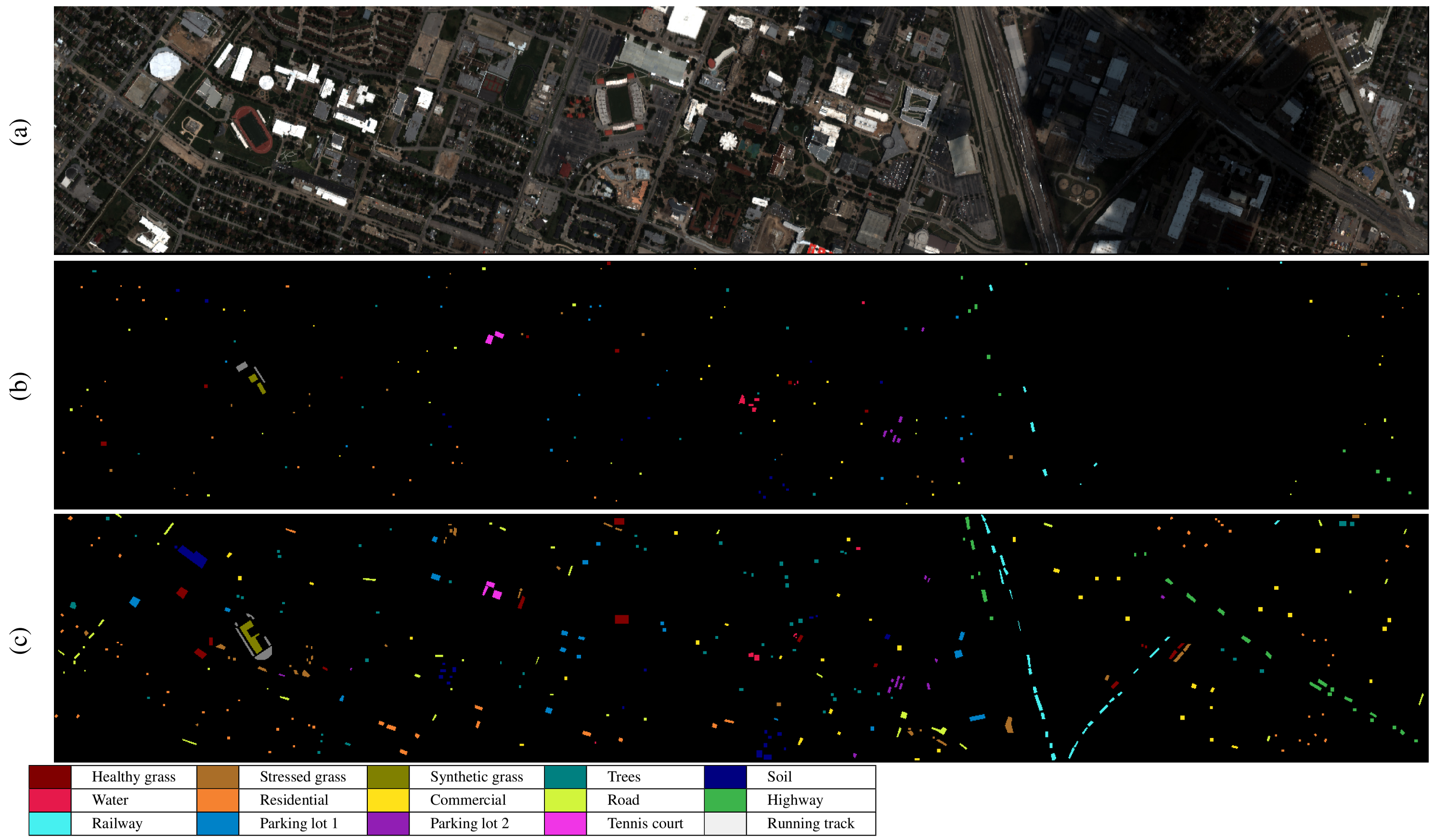


Figure 22. Houston University 2013 dataset. (a) the RGB composite, (b) a hyperspectral image, (c) training samples, and (d) test samples.

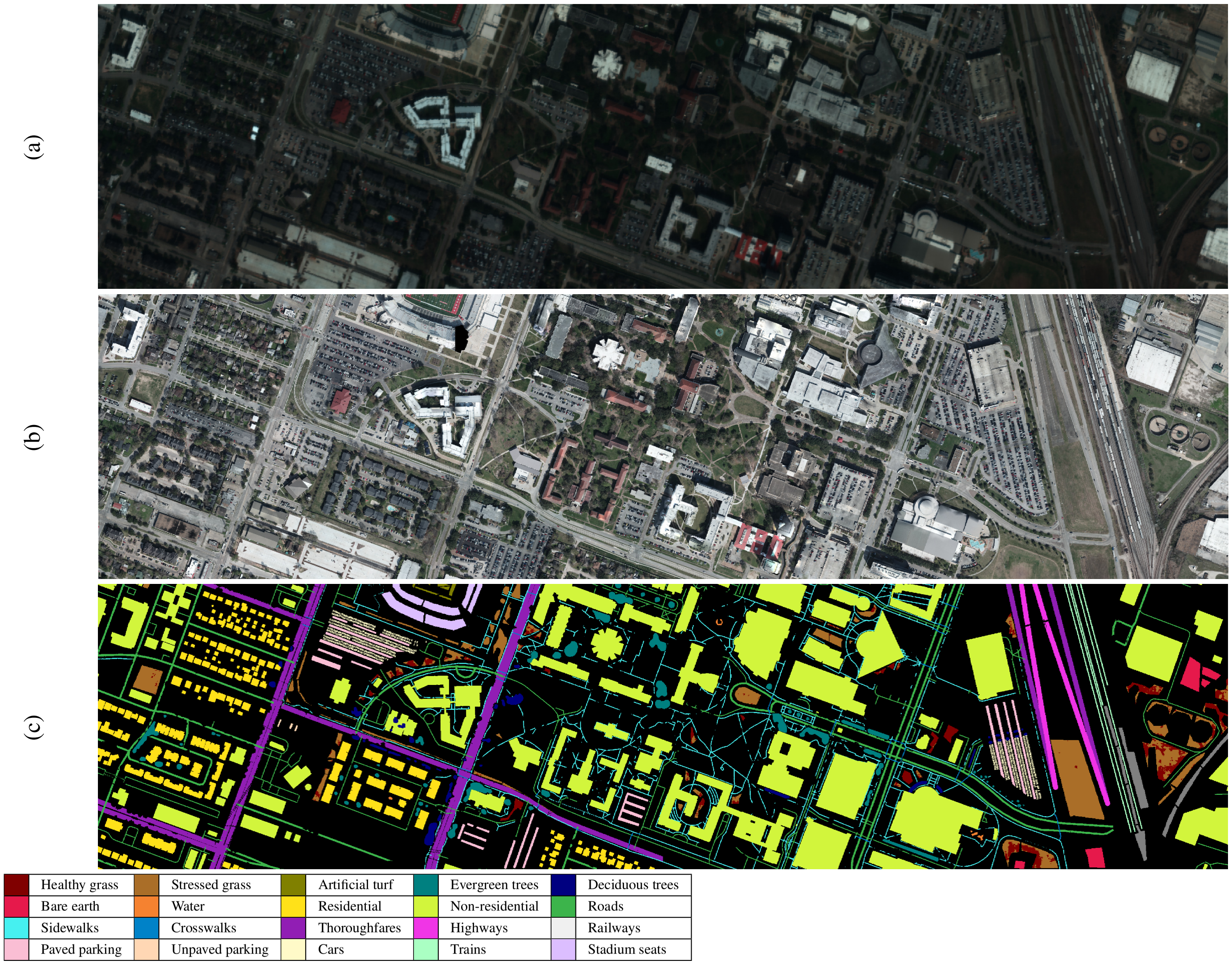


Figure 23. Houston University 2018 dataset. (a) the RGB composition of the HS image, (b) the RGB image, and (c) the ground truth map.

Kronecker product

Given two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ with dimensions $N_i \times N_j$ and $N_k \times N_l$, respectively, their Kronecker product $\boldsymbol{A} \otimes \boldsymbol{B}$ with resultant dimensions $N_i N_k \times N_j N_l$ is defined by

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{1,1}\boldsymbol{B} & a_{1,2}\boldsymbol{B} & \cdots & a_{1,N_j}\boldsymbol{B} \\ a_{2,1}\boldsymbol{B} & a_{2,2}\boldsymbol{B} & \cdots & a_{2,N_j}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N_i,1}\boldsymbol{B} & a_{N_i,2}\boldsymbol{B} & \cdots & a_{N_i,N_j}\boldsymbol{B} \end{bmatrix}.$$

A basic property is that $\mathrm{vec}(\boldsymbol{B}\boldsymbol{X}\boldsymbol{A}^\top) = (\boldsymbol{A} \otimes \boldsymbol{B})\mathrm{vec}(\boldsymbol{X})$ where $\boldsymbol{X}$ has dimensions $N_l \times N_j$.

Woodbury matrix identity

The Woodbury matrix identity is

$$(\boldsymbol{A}+\boldsymbol{B}\boldsymbol{C}\boldsymbol{D})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{B}\left(\boldsymbol{C}^{-1}+\boldsymbol{D}\boldsymbol{A}^{-1}\boldsymbol{B}\right)^{-1}\boldsymbol{D}\boldsymbol{A}^{-1},$$

where $\boldsymbol{A}$ is an $N_a \times N_a$ matrix, $\boldsymbol{B}$ is an $N_{b1} \times N_{b2}$ matrix, $\boldsymbol{C}$ is an $N_c \times N_c$ matrix, and $\boldsymbol{D}$ is an $N_{d1} \times N_{d2}$ matrix. It is assumed than the matrices $\boldsymbol{A}$ and $\boldsymbol{C}$ are invertible.

Vertical and horizontal differences operator

Calculation of the matrix operators for the vertical and horizontal differences to apply on a vectorized image can be defined as follow. Assume that we have an $N_x \times N_y$ image $\boldsymbol{X}$. Now, apply a vertical difference matrix on $\boldsymbol{X}$, i.e., $\boldsymbol{D}_x\boldsymbol{X}$, where $\boldsymbol{D}_x$ is an $N_x \times N_x$ matrix given by

$$\boldsymbol{D}_x = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & -1 \\ -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix},$$

where $\boldsymbol{D}_x$ is the circular convolution matrix of the kernel $\boldsymbol{k} = [1, -1]$. Now vectorize $\boldsymbol{D}_x\boldsymbol{X}$, i.e.,

$$\operatorname{vec}(\boldsymbol{D}_x\boldsymbol{X}) = \left(\boldsymbol{I}_{N_y} \otimes \boldsymbol{D}_x^{\mathsf{T}}\right) \operatorname{vec}(\boldsymbol{X}) = \boldsymbol{D}_{\mathrm{v}}\boldsymbol{x},$$

where $\boldsymbol{x}$ is the vectorized image of length $n = N_xN_y$. This shows that $\boldsymbol{D}_{\mathrm{v}}\boldsymbol{x}$ contains a vertical difference of an image $\boldsymbol{X}$. Moreover, with a similar argument, $\boldsymbol{D}_{\mathrm{h}}\boldsymbol{x}$ contains a horizontal difference of an image $\boldsymbol{X}$.

Notation of vector norm

The p-norm (also ℓ_p-norm) of a vector $\boldsymbol{x} \in \mathbb{R}^n$ is defined as

$$\|\boldsymbol{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}.$$

Note that for

- $p = \infty$, $\|\boldsymbol{x}\|_\infty = \max_{i\in\{1,\ldots,n\}} (|x_i|)$.

- $p = 0$, $\|\boldsymbol{x}\|_0 = \#\{i \mid x_i \neq 0\}$ is a pseudo-norm.

- $p \geq 1$, is a convex function.

- $p \in [0, 1)$, is a non-convex function.

Notation of matrix norm

The mixed (p, q)-norm (also $\ell_{p,q}$-mixed norm) of a matrix $\boldsymbol{X} \in \mathbb{R}^{N_i \times N_j}$ is defined as

$$\|\boldsymbol{X}\|_{p,q} = \left(\sum_{i=1}^{N_i} \left(\sum_{j=1}^{N_j} \left| x_{i,j} \right|^p \right)^{q/p} \right)^{1/q}.$$

In particular, for $p = 2$, $q = 0$ the pseudo-norm

$$\|\boldsymbol{X}\|_{2,0} = \#\left\{ i \;\middle|\; \left\| \boldsymbol{X}_{(i,:)} \right\|_2 \neq 0 \right\},$$

where $\#\{\cdot\}$ denotes the cardinal of the set $\{\cdot\}$.

Reconstruction quality metrics

PSNR is calculated in dB as

$$\text{PSNR} = 10 \log_{10} \left(\frac{(\max(\boldsymbol{x}))^2}{\text{MSE}(\boldsymbol{x}, \hat{\boldsymbol{x}})} \right)$$

where $\boldsymbol{x}$ is the vectorized original image of length n, $\hat{\boldsymbol{x}}$ is the vectorized estimated image, $\max(\boldsymbol{x})$ is the maximum value of vector $\boldsymbol{x}$ and

$$\mathrm{MSE}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{1}{n}\|\boldsymbol{x}-\hat{\boldsymbol{x}}\|_2^2.$$

Then, the average PSNR is the band-based PSNR averaged over all bands of the hyperspectral dataset.

Classification quality metrics

The basis of the classification is the confusion matrix given by

$$\boldsymbol{C}_{\mathrm{m}} = \begin{bmatrix} (c_{\mathrm{m}})_{1,1} & (c_{\mathrm{m}})_{1,2} & \cdots & (c_{\mathrm{m}})_{1,N_c} \\ (c_{\mathrm{m}})_{2,1} & (c_{\mathrm{m}})_{2,2} & \cdots & (c_{\mathrm{m}})_{2,N_c} \\ \vdots & \vdots & \ddots & \vdots \\ (c_{\mathrm{m}})_{N_c,1} & (c_{\mathrm{m}})_{N_c,2} & \cdots & (c_{\mathrm{m}})_{N_c,N_c} \end{bmatrix}$$

where $(c_{\mathrm{m}})_{i,j}$ indicates the number of pixels that belong to the i-th class sample in the experimental area and are assigned to the j-th class, and N_c is the number of classes. According to the classification confusion matrix the CA is the proportion of correctly classified pixels for each class. Then, the Overall Accuracy (OA) and Average Accuracy (AA) are calculated as

$$\mathrm{OA} = \frac{\sum_{i=1}^{N_c}(c_{\mathrm{m}})_{i,i}}{\sum_{i,j=1}^{N_c}(c_{\mathrm{m}})_{i,j}}, \quad \mathrm{AA} = \frac{1}{N_c}\left(\frac{\sum_{i=1}^{N_c}(c_{\mathrm{m}})_{i,i}}{\sum_{j=1}^{N_c}(c_{\mathrm{m}})_{i,j}}\right).$$

Kappa coefficient is a statistical measurement of agreement and is given by

$$\kappa = \frac{\mathrm{OA} - \mathrm{P}}{1 - \mathrm{P}},$$

where

$$\mathrm{P} = \frac{\left(\boldsymbol{C}_{\mathrm{m}} \mathbf{1}_{N_c}\right)^{\mathsf{T}} \left(\boldsymbol{C}_{\mathrm{m}} \mathbf{1}_{N_c}\right)}{\left(\sum_{i,j} \boldsymbol{C}_{\mathrm{m}(i,j)}\right)^2}.$$