Spectral Mapping using Simple Sensors

Srinivasan Vijayarangan

6 May 2024
CMU-RI-TR-24-20

School of Computer Science
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:

David Wettergreen *(chair)*
George Kantor
Michael Kaess
Roger Clark (Planetary Science Institute)

Submitted in partial fulfillment of the requirements for the degree of PhD in Robotics

திரைகடல் ஓடியும் திரவியம் தேடு

300 BCE

Venture across the oceans and seek knowledge

To my father for his steadfast belief in me,

my mother for her selfless love,

and both my parents for their endless sacrifices.

Abstract

Spectral mapping holds significant importance in many exploration endeavors as it facilitates a deeper comprehension of material composition within a surveyed area. While imaging spectrometers excel in recording reflectance spectra into spectral maps, their large physical footprint, substantial power requirements, and operational intricacies render them unsuitable for integration into small rovers or resource-constrained missions. Conversely, RGB cameras offer advantages in terms of portability, energy efficiency, and widespread applicability for fundamental tasks like hazard identification and terrain analysis but do not measure many of the most diagnostic wavelengths.

In our study, we introduce a method to leverage data from RGB cameras for spectral mapping purposes. This is achieved by representing broad spectra in a latent space and learning a transform from RGB image to this latent space. Despite the inherent lower dimensionality of RGB images, characterized by only three channels compared to the hundreds of channels present in hyperspectral images, we illustrate the feasibility of constructing a regression framework using a sparse set of hyperspectral measurements for training. Our approach employs deep learning methods to encode regional remote sensing spectra and Gaussian methods to learn the transform between the latent representation and image data. By harnessing texture and spatial information from RGB imagery, we observe enhancements in reconstruction accuracy. We demonstrate that the system can be trained using spectra sourced from a broad general spectral catalog, thus eliminating the need for specialized pre-training and enabling deployment flexibility across diverse locations.

The efficacy of our system is evaluated through mineral prediction tasks, yielding prediction accuracies of up to 80% against ground truth. No other method has achieved comparable levels of accuracy in mineral prediction utilizing images. Our experimental evaluation reveals the efficacy of the method at three distinct mineral-rich sites, and establishes its potential for enhancing spectral mapping for exploration.

Acknowledgements

First, I would like to thank my advisor and mentor, David Wettergreen for patiently listening to my ideas for all these years and gradually teaching me how to transition from being an engineer to a researcher. There were numerous occasions when I would be drawn to the next shiny thing, yet he consistently provided guidance that kept me grounded and directed me towards the right path. I am thankful for his wise counsel on all occasions. Next, I would like to thank George Kantor for being a great supervisor and mentor. His open, no-nonsense advice has helped me tremendously to progress in my career.

I would like to thank Michael Kaess for agreeing to be on my committee. Throughout, I have enjoyed Michael's distinct style of gentle and subtle guidance. I extend my gratitude to Roger Clark for engaging in extensive discussions about spectroscopy and mineralogy. These conversations have greatly contributed to the advancement of my research. Your profound knowledge of spectroscopy and wealth of experience spanning decades are invaluable for anyone in this field. I am deeply honoured to have you on my committee.

Next, I would like to thank Maggie Hansen and Abby Breitfeld for being awesome team mates in all these field experiments. You both always stepped up to whatever the challenges were with a cheerful attitude. It is a pleasure working with you both. A special thanks to Alberto Candela for formulating the regression framework which served as a foundational concept that influenced my research trajectory. I would like to thank Chuck Whittaker in getting our robots safely to the field every single time. Chuck, I also really appreciate you checking with me about my family and kids making me feel at home. I want to express my gratitude to Red Whittaker for involving me in FRC projects 16 years ago, which eventually paved way for all these. Your boundless energy and visionary leadership continue to inspire me.

None of these would have been possible without the constant support of my family. My parents have made enormous sacrifices to enable me to get to where I am today. Thank you, both. I would like to thank my brother for taking care of my parents in India, so I can peacefully pursue my dreams here in the US. Brother, I deeply appreciate your support. My wife played a crucial role in my degree completion. Without her support, I'm uncertain how many additional years it would have taken. She managed all aspects of our life, allowing me to focus solely on my research. Thank you deeply, Deepi. Lastly, I want to express gratitude to my daughters for their unwavering love, despite the time I missed playing with them due to my research focus. I am truly thankful to my incredible family.

Funding Acknowledgements

I would like to thank the following PIs and their respective projects for funding this research.

Project	PI	Agency	Grant number
Toolbox for Research and Exploration	Amanda Hendrix	NASA SSERVI	80ARC017M0005
MoonRanger	Red Whittaker	LSITP	80MSFC20C0008
Co-Exploration using Science Hypothesis Maps	David Wettergreen	NSF/NRI	1526667
Deep Learning applied to Detecting Salient Features and Building Better 3D Models		NASA SBIR	80NSSC18P2133
SafeForest	George Kantor	PSTF	1031323
AI Institute for Resilient Agriculture		NSF/Iowa State	024408B

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Planetary exploration by surface rover entails a thorough examination of an environment, with a specific emphasis on the geological characteristics. This necessitates an understanding of the composition and formation processes underlying the observed physical features. Mineralogy is a crucial aspect in this endeavor, through the identification of various minerals and their combinations. By discerning the mineral composition, hypotheses regarding the geological processes at play can be formulated and tested, ultimately elucidating the geological history and origin of the region. This process thus proceeds from observed or inferred spectra to identified minerals to an exploration of geologic origin (Figure 1.1).

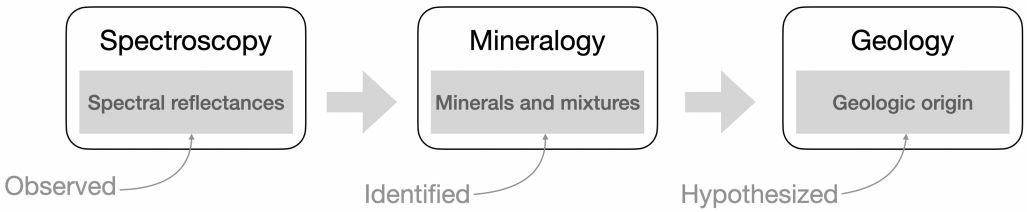


Figure 1.1: Spectroscopy, mineralogy and geology

Imaging spectroscopy is pivotal in advancing mineralogical inference by meticulously measuring electromagnetic radiation interactions with materials. This technique involves filtering, measuring, and investigating the reflected radiation, wherein diverse minerals exhibit distinct absorption characteristics, manifesting in unique spectral signatures within the reflectance spectra. Notably, imaging spectroscopy is favored for its passive observation approach, devoid of any physical or chemical interactions, thereby offering a non-invasive means of characterizing mineral compositions.

1.1 Spectral Mapping

Spectral mapping entails the collection of spectral data, followed by the creation of a spatial map depicting the distribution of the spectral features across the region. These

maps not only aid in comprehending the planetary body's history and evolution but also provide insights into phenomena such as volcanic activity (Achieng et al., 2017) and the presence and impact of water (Li and Milliken, 2017).

Spectral maps serve as a basis for generating mineral maps, enabling researchers to conduct thorough analyses and understand the geologic history of study regions. By integrating spectral mapping techniques into planetary exploration, researchers can infer the mineralogy of distant planetary surfaces, leading to significant discoveries and advancements.

1.1.1 Spectrometers

Spectrometers are scientific instruments utilized for measuring the radiance or reflectance of materials across different wavelengths of the electromagnetic spectrum. They typically incorporate a grating or dispersive element that separates light into its constituent wavelengths, allowing the intensity of light at different wavelengths to be measured (De Sanctis et al., 2012; Scheeline, 2017). Radiance refers to the amount of electromagnetic radiation emitted or reflected from a surface representing the inherent brightness of an object influenced by the object's temperature and emissivity. Reflectance, on the other hand, refers to the ratio of the radiant flux reflected by a surface to the incident flux upon it. It is a unit-less quantity often expressed as a percentage. Reflectance describes how much light is reflected by a surface compared to how much light it receives. It is influenced by the surface properties, such as its color, texture, and composition.

Spectrometers capture the spectral signature of objects or surfaces to derive information about the material composition, chemical properties, and other characteristics of the objects being observed. A spectral signature is a unique characteristic pattern of electromagnetic radiation emitted, reflected, or absorbed by a material across different wavelengths of the electromagnetic spectrum. It represents the distinctive response of a material to incident light and is often used to identify and differentiate between materials based on their spectral properties. These signatures can vary widely depending on factors such as the material's composition, structure, and surface properties.

Imaging spectrometers simultaneously acquire spectral information from multiple locations often by an array of detectors across a surface either by scanning or by moving (i.e. aerially or orbitally).

1.1.2 Spectral Cubes

Spectral maps are commonly depicted as cubes, structured as three-dimensional datasets (Meier, 2012). The first two dimensions represent the spatial coordinates, while the third

dimension represent the reflectance information as shown in Figure 1.2. Each layer within the cube corresponds to a distinct wavelength band, creating a stack of two-dimensional images that portray the spectral data across the imaged area. A spectrum, in this context, is a one-dimensional signal that characterizes the intensity or radiance values across different wavelength ranges or bands within the electromagnetic spectrum. Each element of the spectrum represents a specific wavelength band, providing a detailed spectral signature for a given spatial location within the image.

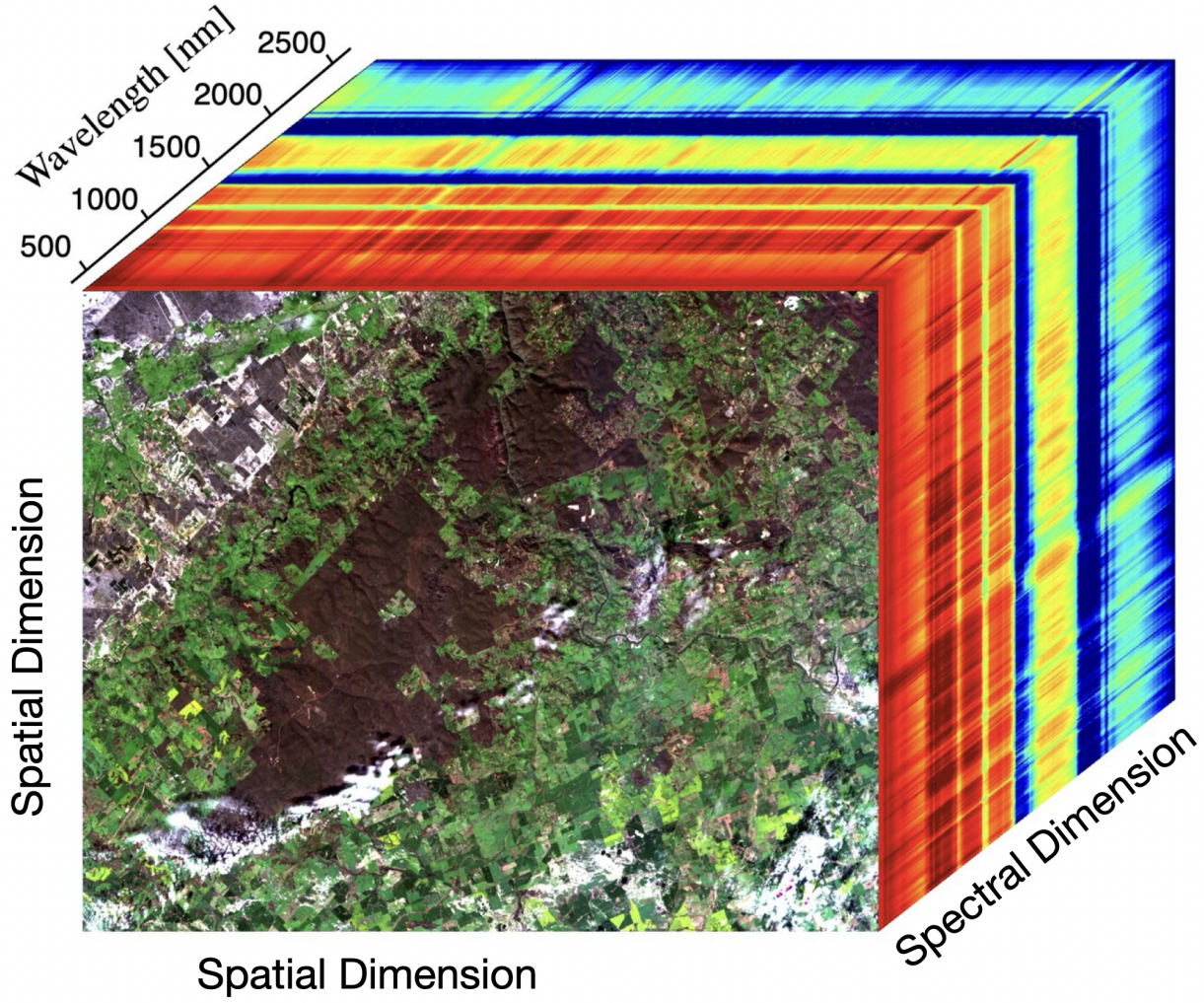


Figure 1.2: Spectral cube generated using EMIT, an imaging spectrometer, in which layers record irradiance in a narrow band of wavelengths. Top image is the composite of all visible bands. The rainbow colors extending through the main part of the cube are the spectral fingerprints from corresponding spots in the top image (Image credit: NASA/JPL-Caltech)

A spectral cube representing RGB data comprises three bands, corresponding to red, green, and blue wavelengths. In contrast, multispectral data cube includes additional bands spaced apart by tens or hundreds of nanometers. Hyperspectral data cube comprises continuous spectral bands with minimal intervals, typically 5 or 10 nm apart as

illustrated in Figure 1.3.

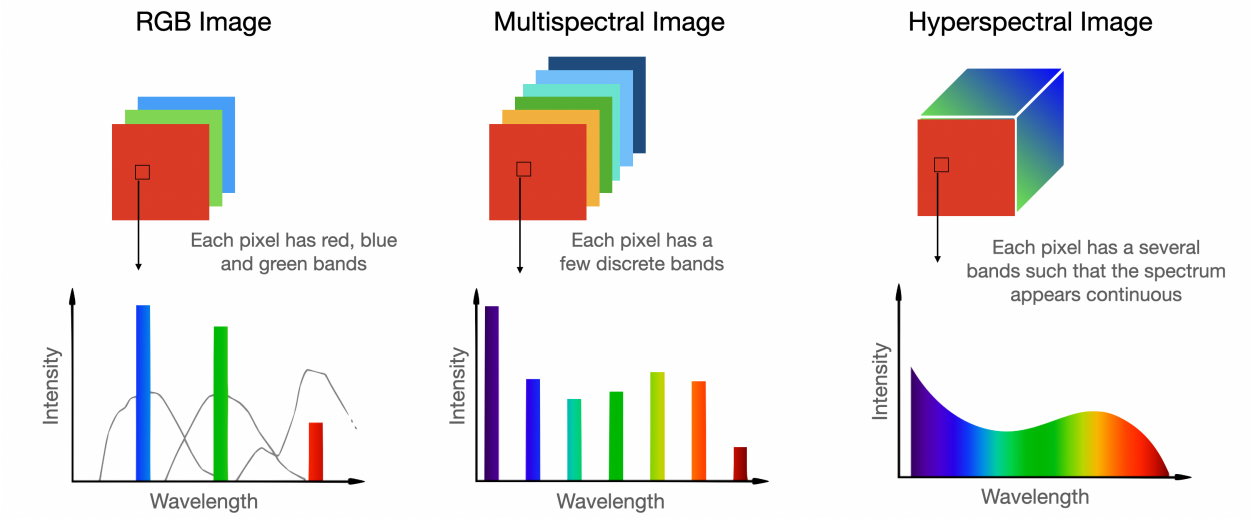


Figure 1.3: Distinguishing RGB, multispectral, and hyperspectral data in terms of spectral band coverage

1.1.3 Spectral Mapping at Orbital Scale

Imaging spectrometers are deployed in space or aboard aircraft, conduct surveys over expansive areas from elevated altitudes. The Figure 1.4 showcases imaging spectrometers utilized in space missions. These instruments capture a broad spectrum spanning 0.3μm to 3μm. Operating at orbital scale offers the benefit of extensive coverage, but it comes at the cost of lower spatial resolution. Typical orbital spectrometers image on the order of 30 meters to 120 meters per pixel due to their high altitude. As a result, this presents challenges for research endeavors that necessitate high spatial resolution or strategic planning for surface rovers, as the resolution falls short of adequately characterizing surface features.

1.1.4 Spectral Mapping at Rover Scale

To function effectively on the scale of rovers, smaller, more portable spectrometers are developed and installed on ground robots like the SuperCam (Fouchet et al., 2022) aboard the Perseverance Rover, capable of capturing wavelengths from 1.3μm to 2.6μm, or the ASD FieldSpec Pro (Młynarczyk, 2020), which captures wavelengths from 0.35μm to 2.5μm, on research platforms like Zoë (depicted in Figure 1.5) (Vijayarangan et al., 2018). Spectrometers on ground robots offer distinct advantages over orbital instruments because of their proximity to the surface, facilitating detailed and targeted observations of specific regions. This close-range enables higher spatial resolution and precise measurements enhancing the rover's ability to characterize its immediate surroundings.

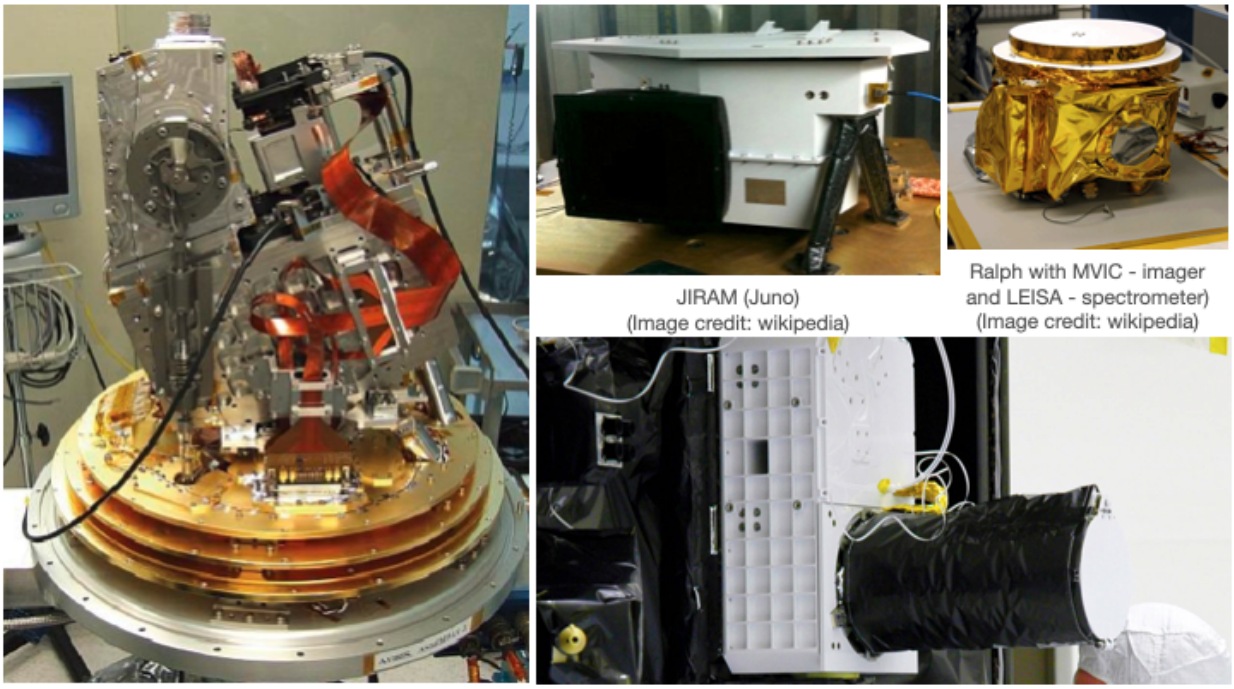


Figure 1.4: Imaging spectrometers, with wide spectrum coverage, are intricate and bulky instruments

Figure 1.5: Zoë is a solar powered field robot with an onboard spectrometer (ASD FieldSpec Pro) and can measure wavelengths in visible, NIR and SWIR bands ranging up to 3000 nano meters. It uses a one-degree fore-optic mounted on a pan/tilt to measure surrounding terrain

However, operating spectrometers on ground robots introduces its unique challenges. Compact spectrometers, functioning as point instruments, record one pixel at a time, significantly delaying spectral coverage even for small areas. Their frequent calibration to counteract atmospheric effects extends the time required to acquire spectra. This delays data collection and reduces the daily spectral data volume, making exhaustive pixel-by-pixel spectral mapping impractical for rover-scale operations.

1.2 Thesis Research

This research aims to utilize images captured by cameras to infer spectral measurements for mineralogical analysis. Cameras generate high spatial resolution RGB images which can compensate for the limited spectral data volume in rover-scale operations, effectively trading low spatial resolution across many wavelengths for high spatial resolution at fewer wavelengths. In contrast to imaging spectrometers, cameras are simpler, more affordable, operate faster, and offer wider fields of view.

The proposed solution entails encoding spectra to an intermediate representation, called a latent space, and training a system to use images to decode from that space. This allows spectra to be generated from cameras; to reconstruct spectra from RGB images. The high spatial resolution of image data offers additional information such as texture, which enhances accuracy of these spectral reconstructions.

It must be acknowledged that the proposed method is especially challenging in situations where the spectral diversity is not expressed in RGB images, particularly in mineral-rich with minimal visual distinction (i.e. two minerals are alike in color and texture but compositionally different). However, in realistic exploration scenarios, such regions are relatively uncommon or fine distinction is inconsequential, with the majority of the environment being spectrally homogeneous. Nonetheless, the effectiveness of this method is validated in mineralogically diverse field sites.

1.2.1 Application Motivation: Micro-Aerial Vehicles

Recent advancements in consumer electronics have fueled a notable proliferation of micro-aerial vehicles, commonly known as drones, across various sectors, including aerial photography, agriculture, and environmental monitoring. Moreover, the successful demonstration of the Mars helicopter (Balaram et al., 2018) (depicted in Figure 1.6) represents a groundbreaking achievement, affirming the viability of aerial vehicles on other planetary bodies. Despite their proficiency in capturing high-resolution images and videos, the scientific potential of these drones remains largely underutilized. One promising avenue to maximize the scientific value of micro-aerial vehicles is by employing them to construct spectral maps using our proposed method.

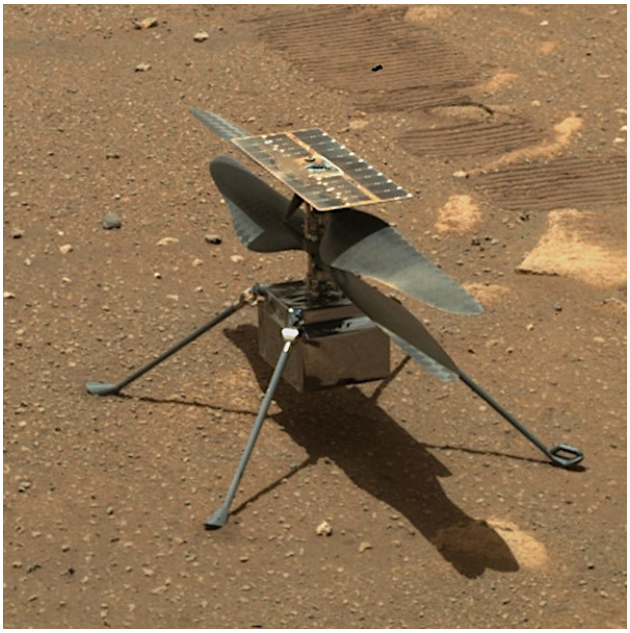


Figure 1.6: Ingenuity Helicopter on Mars (Image credit: wikipedia)

Integrating aerial platforms into planetary exploration endeavors allows researchers to leverage their capability to cover extensive areas and swiftly capture detailed terrain imagery from an elevated viewpoint. Aerial projections generated from images collected by a micro-aerial vehicle can exhibit exceptional spatial resolution, providing comprehensive coverage of the target area, as shown in Figure 1.7. In summary, micro-aerial vehicles offers a promising opportunity to enhance science returns, through the construction of spectral maps, advancing our understanding of planetary surfaces.

1.2.2 Approach

Camera images are employed to perform spectral mapping using the simple-sensor-spectral-model approach (Figure 1.8). It is a two step process:

- Learn a low dimension latent representation of the spectra
- Learn to predict the latent representation using camera images

The *spectral encoder* condenses spectra into a lower-dimensional latent representation, while the *spectral decoder* restores the spectra to its original form based on this latent representation. Together, they comprise the *spectral compressor* (SC). The *latent encoder* (LE) establishes connections between the latent representation of spectra and camera images, generating latent predictions which are then translated into spectra using the

Figure 1.7: Ortho image from a field dataset spanning an area of 900x700 meters, showcasing its high spatial resolution of 3 cm/pixel

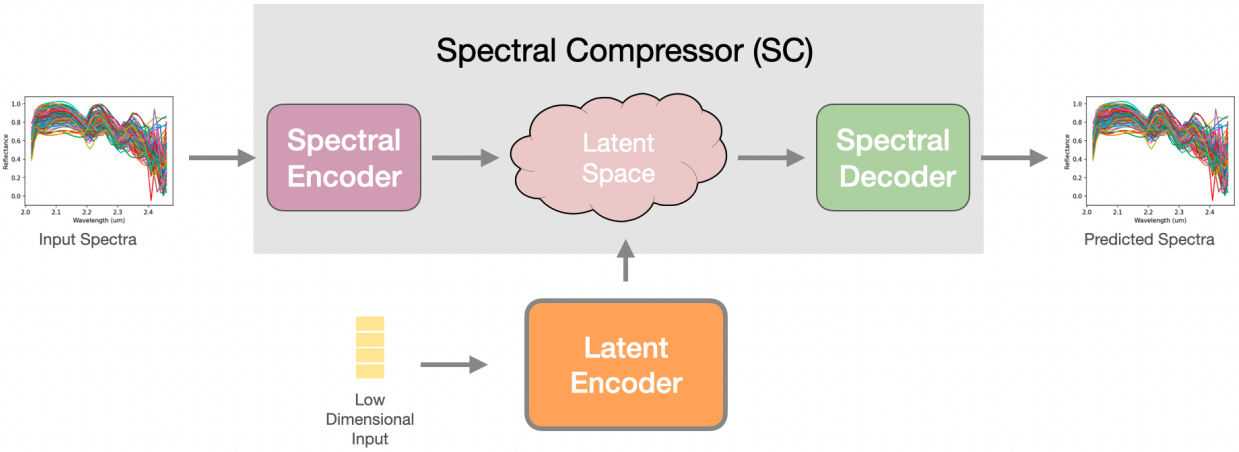


Figure 1.8: Simple sensor spectral model which enables simple sensors like cameras to participate in spectral mapping process

spectral decoder. The *spectral decoder* functions regardless of the source of the latent input, allowing for the utilization of camera images in predicting spectra.

In a broader sense, the simple-sensor-spectral-model enables a simple sensor (like cameras) producing low dimensional data (images) to produce high dimensional data (spectra) usually produced using complex sensors (like imaging spectrometers).

1.2.3 Thesis Statement

Simple sensors can operate in a latent space of complex, information-rich sensors and generate similar high dimensional data.

Simple sensors are defined as those producing low-dimensional spectral data, such as RGB or multispectral cameras. The term 'simplicity' relates to the spectral dimension of the data. Simplicity also implies sensors that are lightweight, compact, and require minimal calibration. In contrast, *information-rich sensors* capture continuous wavelengths across the electromagnetic spectrum, providing a high-dimensional spectral dataset. The term 'high dimensionality' specifically pertains to the spectral dimension. The *latent space*, meanwhile, denotes the transformed representation of the hyperspectral signal generated by a learned model, capturing essential features in a more compact form.

1.2.4 Contributions

This thesis makes contributions by enabling planetary science investigation through the use of simple sensors. The proposed method involves employing cameras to gain evidence of mineralogy, which represents a novel approach to traditional methods that rely on specialized instruments such as imaging spectrometers. Additionally, this thesis proposes a novel strategy for employing simple sensors within the latent space of complex and information-rich sensors, even in scenarios with limited training examples. By utilizing this strategy, simple sensors can extend their capabilities and map low-dimensional data to high-dimensional data previously produced only by complex sensors. This innovative approach is not restricted to a particular field and can be applied in various domains, potentially revolutionizing the use of simple sensors in diverse compositional analysis applications.

1.3 Thesis Overview

The thesis is structured as follows: Chapter 2 outlines the spectral compression process, exploring various network architectures to identify an efficient system for spectral compression and reconstruction. Chapter 3 elaborates on spectral maps, detailing the mapping process and introducing the three field datasets utilized in this study.

Chapter 4 introduces the main concept of the research: latent encoders. It explains how these encoders enable the use of RGB data for spectral mapping, followed by the introduction of multispectral latent encoders, facilitating the utilization of multispectral

sensors for spectral mapping. Chapter 5 leverages the high spatial resolution of RGB images to extract complementary features like textures, enhancing spectral mapping capabilities.

Chapter 6 ties together the discussions from previous chapters, demonstrating the viability and usefulness of this work through a mineral prediction application. Finally, Chapter 7 presents a summary and conclusion, followed by directions for future research exploration.

Chapter 2

Spectral Compression

The first step, in this approach, to facilitate spectral mapping using camera images involves encoding the spectral data into a lower-dimensional latent space. This operation is managed by the spectral compressor (SC) in the simple sensor spectral model, as shown in Figure 2.1. This chapter starts by reviewing the literature for prior work on compression, categories based on the approach used. Next, the discussion will delve into the implementation of spectral compressors, introducing three variations and conducting experiments to assess their efficacy. To evaluate their performance, quantitative metrics will be introduced. Lastly, an experiment will be conducted to determine the optimal number of latent dimensions to use, in this investigation.

2.1 Prior Work: Compression

Spectral images can have hundreds of spectral bands per pixel and often benefit from compression or feature extraction as it reduces the dimensionality while retaining the important information. Compression algorithms can be broadly classified into transform-based, prediction-based, learning-based and vector quantization based methods.

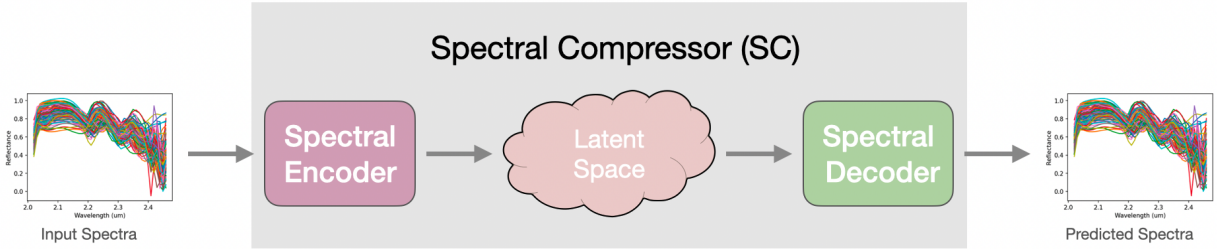


Figure 2.1: In the simple-sensor-spectral model, the spectral compressor learns a low-dimensional representation of the spectral signal through the spectral encoder. The spectral decoder then reconstructs the spectra from this low-dimensional representation

2.1.1 Transform-Based Methods

Transform-based methods apply a function to convert high dimensional spectra to a low dimensional latent space (Dua et al., 2020). The following are the major categories of transform-based methods found in literature for spectral compression.

- *Principal Component Analysis (PCA)* (Greenacre et al., 2022) projects high dimensional spectra on a low dimensional hyperplane, minimizing the variance (Gandhi, 2020). This is accomplished by identifying a new set of orthogonal axes, called principal components (PCs), that explain the maximum variance in the data. Spectral data points are then projected onto these new axes, resulting in a compressed latent representation. PCA is a linear transformation technique that can also be applied to the frequency domain of the spectral signal (Wang et al., 2018; Bajpai et al., 2019). PCA is rarely effective alone (Mantripragada et al., 2022) and is used in combination with other techniques for spectral compression. In PCA information tends to concentrate on the initial features or latent dimensions (Candela Garza, 2021). This is detrimental to the simple-sensor-spectral model as it expects each latent dimension to carry optimal amount of information for effective operation.
- *Manifold Learning* identifies the intrinsic low-dimensional manifold on which the high-dimensional spectral points lie. Compared to PCA, manifold learning applies a non-linear transformation preserving geometric relationships while reducing the dimensionality. Lunga et al. (2014) showed spectral compression using manifold-learning by relying on the idea that spectral data lie on sparse, non-linear manifolds. Expanding on this Huang et al. (2020) incorporated spatial-spectral relationships within spectral cubes to improve the manifold reconstruction accuracy.
- *Linear Discriminant Analysis (LDA)* (Tharwat et al., 2017) achieves compression through feature extraction. LDA finds a linear combination of features that maximizes the separation between different classes in the spectral data while minimizing the variance within each class. This linear combination is achieved by projecting the spectral data onto a lower-dimensional subspace. Unlike PCA, which focuses on maximizing overall variance, LDA specifically considers class separability. While non-linear variations like Kernel Discriminant Analysis (KDA) (Roth and Steinhage, 1999) exist, they are prone to overfitting. However, LDA is inherently suited for spectral classification problems (Bandos et al., 2009).
- *Non-Negative Matrix Factorization (NMF)* (Pauca et al., 2006) factorizes spectral cubes into two non-negative matrices, one representing the spectral signatures or end-members, and the other representing the abundances of these end-members. The non-negativity constraint ensures that the decomposed representations retain physical meaning, as negative values would lack interpretation in the context of representing end-members or abundances. NMF is widely used for spectral unmixing operations (Feng et al., 2022).

The simplicity of transform-based methods stands out as one of their key advantages. However, the size of the low dimensional latent space changes depending on the spectral data used for compression. This variability poses an issue for systems that use the low dimensional latent space for other functions.

2.1.2 Prediction-Based Methods

Prediction-based methods exploit the inherent redundancy and correlation present in the spectral bands. These techniques work by predicting the value of each pixel in a spectral band based on the values of neighboring pixels within the same band and across adjacent bands. A residual image is calculated by finding the difference between the predicted and actual values of the pixel. Compression is achieved by quantization, reducing the bit depth of the residuals.

- *Linear Prediction* (Pizzolante and Carpentieri, 2016) models the spectral values of a band as a linear combination of the values of its neighboring bands. The prediction coefficients are estimated using techniques such as least squares estimation or autocorrelation analysis.
- *Autoregressive Models* exploit the temporal correlation of spectral data in addition to the spatial and spectral correlation. Shen et al. (2018) use correntropy least mean squares algorithm for prediction of pixels on the basis of already predicted spectral and temporal information achieving relatively high compression performance, primarily contributed by the addition of the temporal component.
- *Adaptive Prediction* methods dynamically adjust the prediction model based on the characteristics of the spectral data. These methods may employ techniques such as adaptive filtering or recursive estimation to update the prediction model parameters as new spectral bands are processed (Xu et al., 2018). By adapting to changes in the spectral data, adaptive prediction methods can achieve more accurate predictions and higher compression ratios.

One of the strengths of the prediction-based methods is that they rely heavily on the redundancy and correlation in the spectral, spatial or temporal data to achieve better compression. This leads to a spectrum having different latent representation in different spectral cubes. However, this variability poses a challenge for the simple-sensor-spectral model, which requires the latent representation of spectra to be consistent and reproducible.

2.1.3 Vector Quantizations Methods

In vector quantization methods, initially, a dictionary is constructed from a training dataset, with spectral features encoded as codewords within this dictionary. Then, during standard operations, features are extracted from the spectra and compared against the dictionary and a histogram of the codewords in the dictionary is used as the compressed representation.

- *K-Means Clustering* partitions the spectral data into k clusters, where each cluster is represented by its centroid (Wenbin et al., 2018). The centroids serve as codewords that approximate the spectral data. Spectral vectors are assigned to the nearest centroid, and the compression is achieved by storing the indices of the centroids instead of the original spectral vectors.
- *Sparse Coding and Dictionary Learning* (Chen et al., 2011; Li et al., 2016; Fu et al., 2017) represents each spectra in the high-dimensional space as a linear combination of a small number of basis vectors from a dictionary. The key idea is to find a sparse representation where most coefficients are zero, indicating that only a few basis vectors contribute significantly to the representation of the spectra. The number of coefficients in this representation depends on the sparsity level chosen and can vary for each spectra.
- *Hierarchical Vector Quantization (HVQ)* (Kamano et al., 2001) organizes the spectral data into a hierarchical tree structure, where each node represents a cluster of spectral vectors. The tree structure allows for efficient encoding and decoding of the data at different levels of granularity, enabling adaptive compression based on the desired level of detail.
- *Gaussian Mixture Models* (Yuan et al., 2015) represents the spectral data as a mixture of Gaussian distributions. Each Gaussian component represents a cluster, and the parameters of the Gaussians (mean and covariance) serve as codewords. Spectral vectors are assigned to the Gaussian component with the highest probability, and compression is achieved by storing the parameters of the Gaussian components.

Quantization introduces noise, which can be reduced by utilizing a larger dictionary. However, a larger dictionary increases the size of the low dimensional latent space and thereby diluting the information encoded in the latent space.

2.1.4 Learning Based Methods

Learning-based methods use self-supervised methods to compress spectra.

- *Auto-Encoders* are neural network models crafted for self-supervised learning, with a goal of acquiring efficient representations of input data. Webster and Sheppard (2022) demonstrate using an auto-encoder to compress spectra. They comprise an encoder network compressing input into a latent space representation, and a decoder network reconstructing the input from this representation, aiding in dimensionality reduction. While auto-encoders excel with clean input data, treating it as discrete points for reproduction, they may falter when confronted with noisy input, as they lack robustness to handle such perturbations.

- *Variational Auto-encoders VAEs* (Kingma and Welling, 2013) are a variant of autoencoders that learn probabilistic latent representations of spectral data (Portillo et al., 2020). A VAE encodes the input as a distribution over the latent space which makes them robust to noise. They are generative models and can produce data not seen in the training set. This means, in the right conditions, they can identify new spectral mixtures not trained on.

- *Generative Adversarial Networks (GANs)* (Creswell et al., 2018) consist of two neural networks, the generator and the discriminator, which are trained simultaneously in a competitive manner. The generator aims to produce data indistinguishable from real data, while the discriminator strives to accurately distinguish between real and generated data. While auto-encoders are prone to noise, GANs are particularly sensitive to hyper-parameters, which can lead to instability during training (Becker et al., 2022). Additionally, the lack of an explicit loss function in GANs complicates progress measurement.

While auto-encoders are sensitive to noise in the input data, GANs are sensitive to hyperparameters. VAEs, on the other hand, are resilient to noise in the input data and are less sensitive to hyper-parameters compared to GANs, facilitating more stable training and model behavior.

Network Architectures

The selection of network architectures for learning-based methods depends on the characteristics of the data used for training.

- *Multilayer Perceptron Network (MLP)* (Delashmit et al., 2005) is a class of feedforward artificial neural network that consists of multiple layers of neurons, each layer fully connected to the next, widely used for a variety of pattern recognition and machine learning tasks. Candela et al. (2018) demonstrated the efficacy of MLP network to compress and reconstruct spectra.

- *Convolutional Neural Networks (CNNs)* are specialized deep learning architectures that use convolution layers to capture specific features such as edges, textures, and shapes in spatial data like images. The convolution filters produce feature maps that summarize the presence of detected features in the input enabling the network to learn increasingly complex patterns as more layers are added. Numerous studies in the literature (Li and Liu, 2019; Guan et al., 2021; Dua et al., 2021) have employed 2D convolutions, akin to those utilized in image processing, for spectral compression. However, Jiang et al. (2020) suggested an alternative approach using 1D convolutions with varying kernel sizes, which demonstrated more effective capture of spectral features compared to employing convolutional networks designed for image data.
- *Recurrent Neural Networks (RNNs)* (Salehinejad et al., 2018), tailored for sequence prediction tasks like time series analysis or natural language processing, feature feedback connections within their hidden layers. LSTM (Long Short-Term Memory) (Hochreiter and Schmidhuber, 1997), a specialized form of RNN, excels at capturing temporal dependencies in sequential prediction tasks. Webster and Sheppard (2022) applied an LSTM-based architecture for spectral compression within the context of an auto-encoder framework, demonstrating its efficacy in handling spectral data sequences.

This research uses a VAE similar to the one outlined by Portillo et al. (2020) for compressing the spectral data. During our assessment of network architectures, all three designs mentioned above are examined, with the sequence-based LSTMs exhibiting superior performance.

Next, an examination of how VAEs are used to perform spectral compression is presented.

2.2 Spectral Compressor

The spectral compressor (SC) in the simple-sensor-spectral model comprises of the spectral encoder and decoder modules. The spectral encoder transforms the input spectra into a latent space, while the spectral decoder reconstructs the original spectra based on this latent representation. VAEs are employed for spectral compression, experimenting with four variations of the spectral compressor and comparing their performance. Initially, auto-encoder is implemented and contrasted with its probabilistic variational counterpart, VAE.

An auto-encoder (see Figure 2.2) converts high dimensional observation $H \subset \mathbb{R}^n$ into a low dimensional latent space $L \subset \mathbb{R}^l$ and $n > l$ where n and l are the number of bands in the spectra and the size of the latent representations respectively. There is no probablistic aspect.

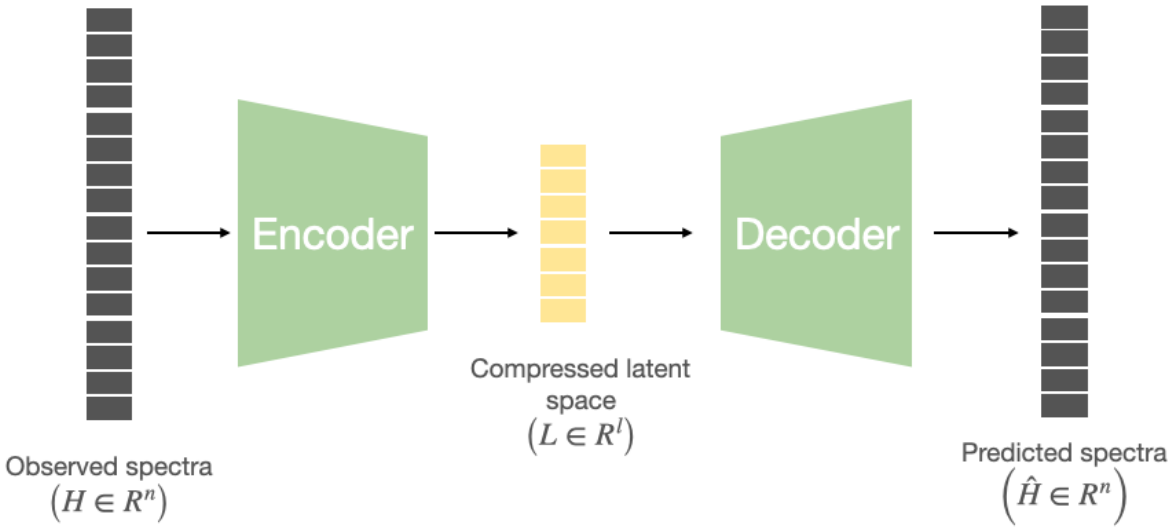


Figure 2.2: Auto-encoder generates a latent representation which is a compressed form of the input signal. Ideally, after training, the predicted spectra is identical to the observed spectra, meaning, the spectra is reconstructed accurately without any losses

2.2.1 Variational Auto-Encoders

VAEs (see Figure 2.3) elevate auto-encoders by modeling the input as a distribution, fostering more robust representations by including variance in the encoder predictions. Similar to an auto-encoder, VAEs compress high dimensional input data, $H \subset \mathbb{R}^n$, into a latent representation, $L \subset \mathbb{R}^l$, and then reconstruct it into the original data space, $\hat{H} \subset \mathbb{R}^n$ where $n > l$ and n and l are the number of bands in the spectra and the latent space respectively. Instead of directly predicting the latent space, the encoder predicts a mean (L_μ) and variance (L_σ), which are then utilized to sample from a normal distribution. The latent predictions are given by,

$$L = L_\mu + L_\sigma \sim \epsilon$$

where $\epsilon \sim \mathcal{N}(0,1)$ is the zero mean and unit variance normal distribution.

To compare the performance of VAE against an auto-encoder, an experiment was setup to calculate the error between the spectra processed through the VAE and auto-encoder with the original spectra and accumulated the results (see Table 2.1). Both models use the same encoder-decoder architecture and are trained on the same field datasets. The VAE shows superior performance, evident in its lower reconstruction error, which can be largely attributed to its robustness to noise often found in field data. Although the difference in error values are small, a paired t-test with a t-statistic of 77 confirms that this difference is statistically significant, highlighting the VAE's effectiveness.

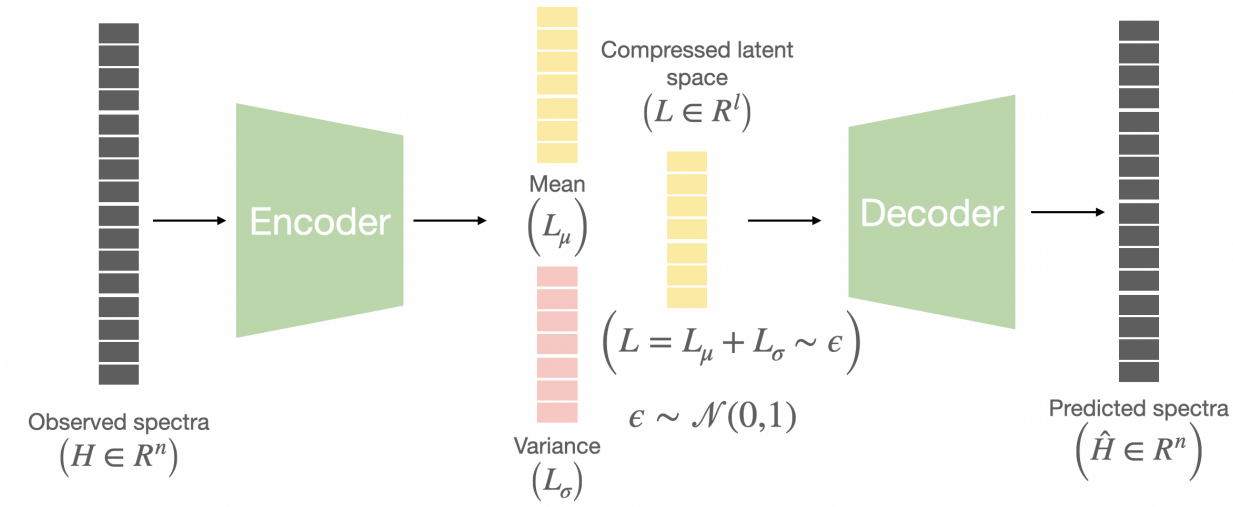


Figure 2.3: VAE compresses the high dimensional spectra to a low dimensional latent vector by modeling the input as a distribution

Type	Root Mean Square Error	Mean Spectral Angle Error
Auto-Encoder	0.0178	1.388
Variational Auto-Encoder	*0.0166*	*1.327*

Table 2.1: Reconstruction error between an auto-encoder and a variational auto-encoder with identical encoder-decoder networks

2.2.2 Loss Function

While the autoencoder is trained solely on the reconstruction loss, the VAE's loss function incorporates an additional regularization term, defined as:

$$Loss^{VAE} = Loss_{Reconstruction} + Loss_{Regularization}$$

While the reconstruction loss evaluates how accurately the input signal is reconstructed, the regularization loss ensures that the VAE encoder learns a distribution and produces latent vectors that conform to the distribution it is learning by incorporating a regularization factor that penalizes for deviation from the distribution. Typically a normal distribution is chosen as the encoder can be trained to return its mean and variance as output. The regularization loss uses the Kullback-Leibler (KL) divergence metric which measures the differences between two distributions (Sharma, 2021), in this case between the trained distribution and zero mean normal distribution (to ensure continuity and completeness). The regularization loss is defined as:

$$Loss_{Regularization} = KL\left[\mathcal{N}(\mu_L, \sigma_L^2), \mathcal{N}(0,1)\right]$$

Using a normal distribution also ensures that the VAE guarantees continuity, slight changes in the latent space result in corresponding minor variations in the target spectra and completeness which ensures that any point sampled from the learned distributions will produce a valid spectra, similar to the ones used during training.

When the VAE operates on spectra, the conventional reconstruction loss that employs l_2 error between the input and the predicted signals may not be the best choice because l_2 error measures the magnitude of individual points in the spectra instead of considering the relationship to the neighboring points. Instead, the spectral angle (Kruse et al., 1993) error is used as the metric for reconstruction loss. The spectral angle between spectra is a measure of similarity between two spectra and is based on the concept of the angle between vectors in Euclidean space. It is calculated as the angle between two n-dimensional vectors representing the spectral signatures, where n is the number of spectral bands. A smaller spectral angle indicates a higher degree of similarity between the two spectra, while a larger spectral angle indicates greater dissimilarity. This is a common and widely accepted metric to measure similarity between spectra (Dennison et al., 2004).

Before delving further, some metrics suitable for the qualitative analysis of spectral predictions are introduced.

2.3 Evaluation Metrics: Quantitative Analysis

In this section, quantitative metrics are examined to assess the alignment of predicted spectra with the original spectra. Rather than comparing individual spectral predictions, let us evaluate several predictions together as a cumulative measure. Let H denote a spectrum with b spectral bands and each spectral band denoted as h_i as shown in Equation 2.1 .

$$H = \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_b \end{bmatrix} \tag{2.1}$$

Let $H_m \in R^{b \times m}$ denote m spectrum stacked to form a matrix.

2.3.1 Root Mean Squared Error

Root Mean Squared Error (RMSE) is a widely used metric which measures l_2 error between spectra. It provides a comprehensive understanding of the prediction performance by quantifying the differences between predicted and actual values across all wavelengths.

$$RMSE(H_m^p, H_m^t) \quad = \quad \frac{||H_m^p - H_m^t||_F}{\sqrt{m \times n}} \tag{2.2}$$

where $||X||_F = \sqrt{trace\,(X^T X)}$ is the Frobenius norm of X. The superscript p and t in Equation 2.2 denote the predicted and ground-truth spectra respectively. RMSE is a global metric as it measures the average error over all the spectral predictions m. When the spectra are identical the RMSE will be zero. Since RMSE measures the error between spectral reflectances, it is a unit-less quantity.

2.3.2 Cross-correlation

Cross-correlation (CC) measures the degree of similarity and alignment between predictions and is calculated as shown in Equation 2.2,

$$CC(H_m^p, H_m^t) \quad = \quad \frac{1}{n} \sum_{i=1}^{m} CCB_i(H_i^p, H_i^t) \tag{2.3}$$

where CCB_i is the cross-correlation for a single-band image as shown in Equation 2.4.

$$CCB_i(A, B) = \frac{\sum_{j=1}^{m}(A_j - \mu_A)(B_j - \mu_B)}{\sqrt{\sum_{j=1}^{m}(A_j - \mu_A)^2 \sum_{j=1}^{m}(B_j - \mu_B)^2}} \tag{2.4}$$

where $\mu_A = (1/n) \sum_{j=1}^{m} A_j$ is the sample mean of A.

CC is a spatial measure which measures the geometric distortion between the spectral predictions (Loncan et al., 2015). When the spectral signals are identical, the CC value is 1. CC is a unit-less quantity that ranges between 0 an 1.

2.3.3 Mean Spectral Angle Error

Spectral angle measure (SAM) (Kruse et al., 1993) calculates the angle between spectral signals using trigonometric functions, providing a measure of similarity irrespective of differences in intensity or scale. Lower SAM values indicate greater spectral similarity, while higher values suggest greater dissimilarity.

The SAM between two spectral signals is calculated as shown in Equation 2.5.

$$SAM(H^p, H^t) = \arccos\left(\frac{\langle H^p, H^t \rangle}{||H^p||\,||H^t||}\right) \tag{2.5}$$

MSAE (refer to Equation 2.6) is the mean of SAM between the predicted and target spectral signals.

$$MSAE(H_m^p, H_m^t) = \frac{\sum_{i=1}^{m} SAM(H_i^p, H_i^t)}{m} \tag{2.6}$$

For identical spectra, MSAE is 0. MSAE is expressed in radians.

2.3.4 Mean Latent Error

Latent error quantifies the l_2 distance between two latent vectors and is calculated as shown in Equation 2.7.

$$Latent\ Error\ (L^p, L^t) \quad = \quad \frac{\sqrt{\sum_{i=1}^{l} \left(l_i^p - l_i^t\right)^2}}{l} \tag{2.7}$$

where L^p snd L^t represent a latent vector for the predicted and target spectra. l_i^p is the i^{th} element in the latent vector which has l latent dimensions. The mean latent error (MLE) (refer to Equation 2.8) between a vector of latent predictions is a simple mean of the latent error along the latent dimensions.

$$MLE(L_m^p, L_m^t) \quad = \quad \frac{\sum_{i=1}^{m} Latent\ Error(L_i^p, L_i^t)}{m} \tag{2.8}$$

For identical spectral predictions, MLE is 0. Since the latent error measures the error in the feature space, it does not represent any physical quantity and hence it is unit-less.

2.3.5 Mineral Prediction Accuracy

Mineral Prediction Accuracy (MPA) (refer to Equation 2.9) quantifies the proportion of correct mineral predictions relative to the total predictions made, expressed as a percentage. The mineral prediction for the original spectra acts as ground truth, against mineral predictions from predicted spectra, including those generated by the spectral compressor.

$$MPA = \frac{\sum_{i=1}^{m} \left[Tet\left(H_i^t\right) = Tet\left(H_i^p\right)\right]}{m} \times 100 \tag{2.9}$$

where $Tet(.)$ is the mineral prediction function.

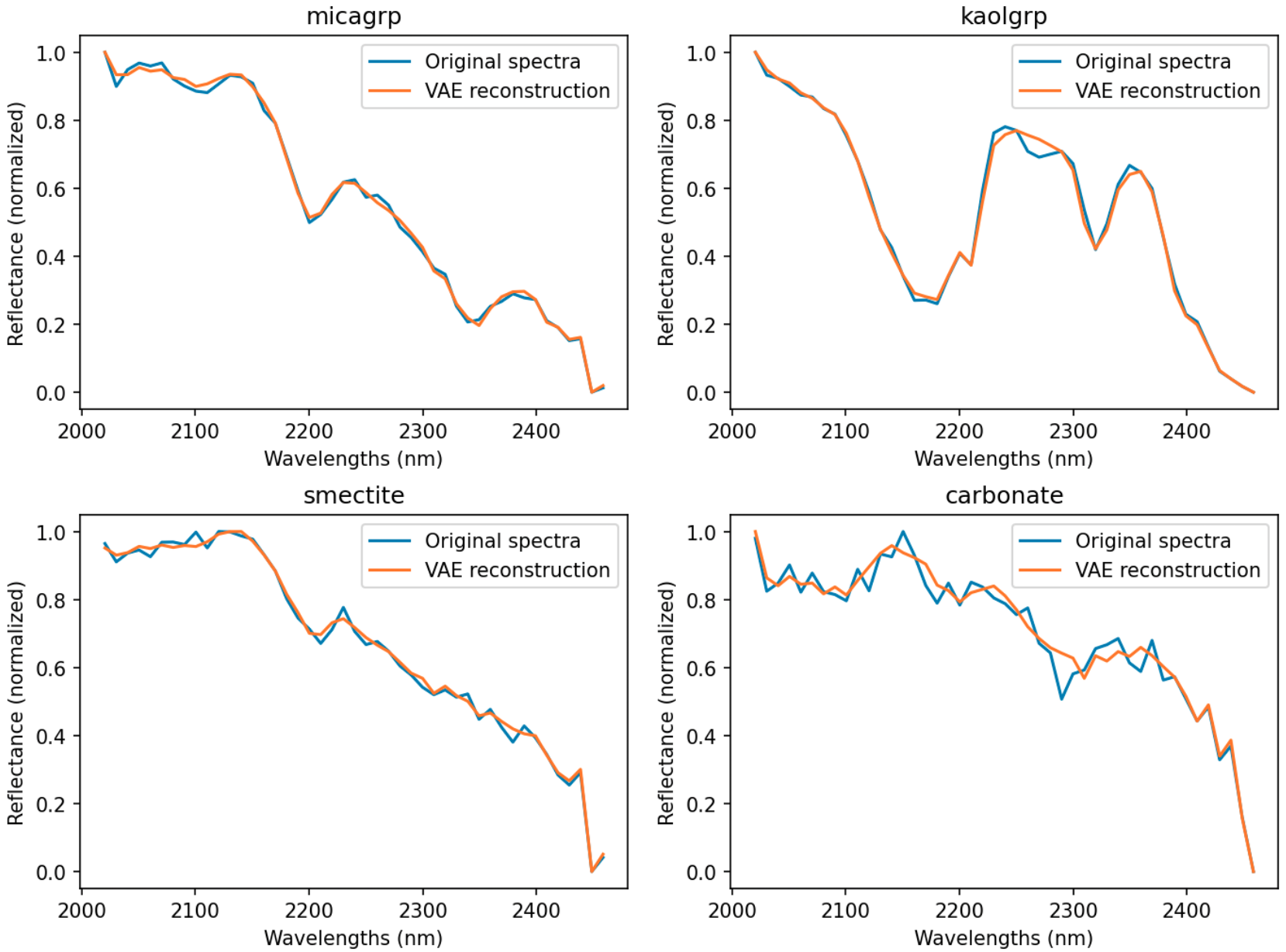


Figure 2.4: Qualitative results of predictions from the VAE compared to the original signal for a few minerals. Since the reconstruction error minimizes the spectral angle error instead of the l_2 error, the reconstructed signal captures the shape of the spectra but smoothes out some of the high frequency components

2.3.6 Analysis

In summary, RMSE quantifies the l_2 error, CC evaluates the similarity, and MSAE measures the mean of angle between the predicted and original spectra. These metrics allow for quantitative but only comparative assessments of spectral predictions; they cannot provide absolute judgments. Conversely, the MPA is an absolute metric assessing mineral prediction accuracy of spectra which will be utilized in a later chapter to validate our findings.

Rather than focusing on errors in the predicted spectra, MLE targets errors in the latent dimension. MLE is used to compare the latent encoders (in the simple-sensor-spectral model) against the VAE.

2.4 Spectral Reconstruction: Examples

A VAE is trained using hyperspectral data from Cuprite Hills, one of the field datasets introduced in Chapter 3 (in Section 3.3). The network architecture used for the VAE will be detailed in Section 2.5. The 2000nm to 2500nm wavelengths bands from the hyperspectral dataset is used for training and testing the VAE. A qualitative comparison of the spectral predictions generated by the VAE, juxtaposed with the original spectral signal show that the reconstructed signal captures the shape of the spectra but does not achieve complete fidelity (as shown in Figure 2.4) due to the minimization of spectral angle error rather than l_2 error during the VAE training process. Additionally, these results highlight VAE's capability to compress various spectral signatures representing different minerals.

Before delving further, let us introduce some metrics suitable for the qualitative analysis of spectral predictions.

2.5 Network Architectures

Architectures for the encoder-decoder network play a crucial role in determining the information retention capacity of the spectral compressor, influencing the quality of the reconstructed output. Specifically, the depth, width, and types of layers significantly influence how well essential features are captured and information loss is minimized during encoding and decoding. Three variations of spectral compressors: FCNN-SC, MCConv-SC, and SeqMem-SC, are explored and their information retention capability accessed.

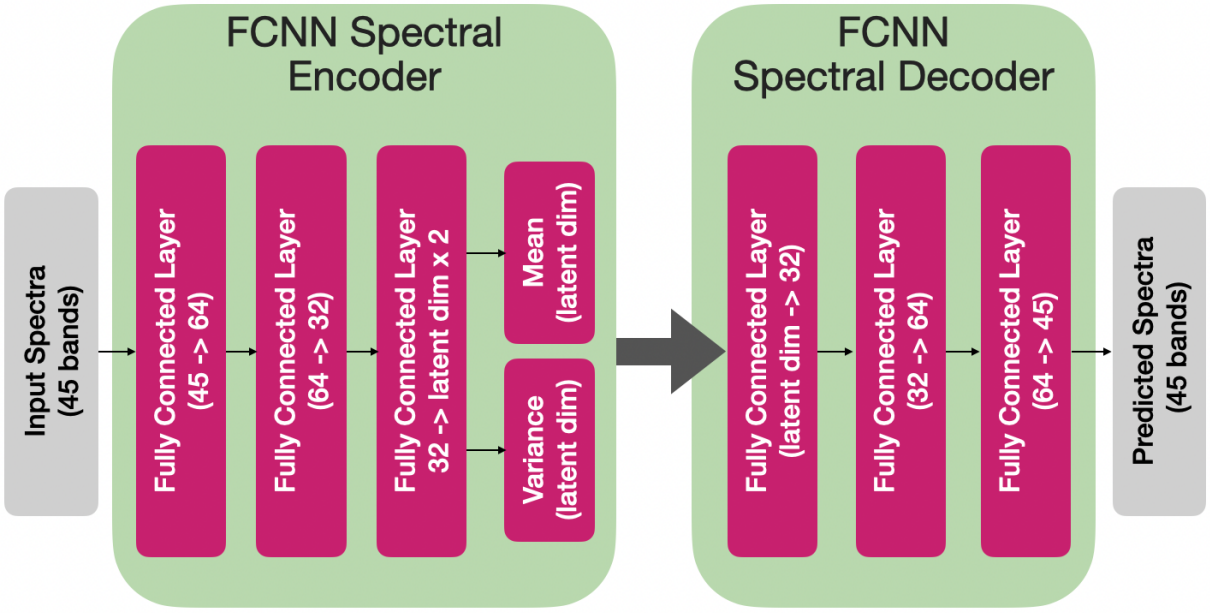


Figure 2.5: FCNN-SC, the simplest of the 3 SCs, with 3 hidden layers in both the encoder and decoder

2.5.1 Fully-Connected Neural Network - SC

The Fully Connected Neural Network (FCNN) SC employs a fully-connected multi-layer neural network. Similar to the architecture proposed by Candela et al. (2018), our version features additional neurons and an extra hidden layer, illustrated in Figure 2.5. In the FCNN-SC, spectral signals pass through three fully connected layers before mean and variance are extracted for VAE calculations. The main advantages of using a FCNN include its straightforward structure, relatively low computational requirements, and ease of training.

2.5.2 Multi-Coded-Conv - SC

The Multi-Coded-Conv (MCConv) SC is inspired by Jiang et al. (2020)'s work, which introduced a deep learning network for spectral signals, demonstrating state-of-the-art performance. Jiang et al. (2020) employed 1D convolutions with varying kernel sizes in the input stage before incorporating regular 2D convolutional layers. The rationale behind using 1D convolutions lies in their capability to capture features in the 1D spectral signals at different scales. MCConv-SC adopts a hybrid approach with 1D convolutions in the input stage followed by fully connected layers, similar to FCNN-SC. Moreover, MCConv-SC incorporates a larger fully connected layer compared to FCNN-SC, enhancing its learning capacity and reducing spectral signal reconstruction error.

The encoder and decoder network architectures are illustrated in Figure 2.6 and Figure 2.7, respectively.

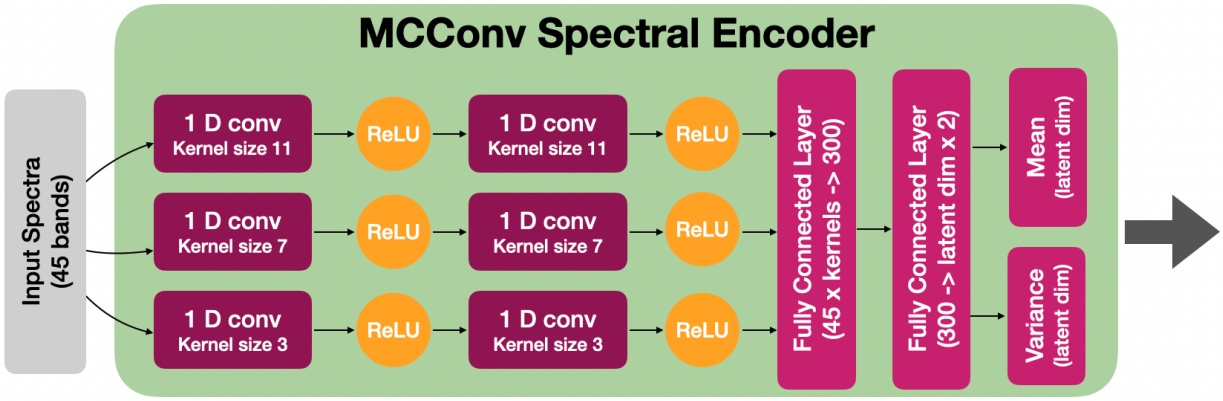


Figure 2.6: MCConv Spectral Encoder: The spectral signal is passed through 1D convolutions of different kernel sizes to capture the spectral features at different scales

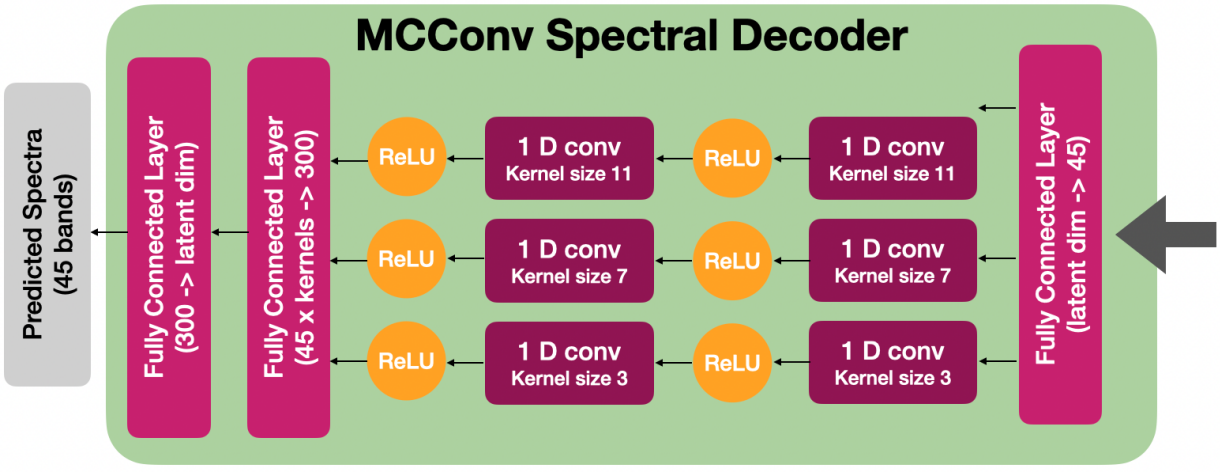


Figure 2.7: MCConv Spectral Decoder: The latent dimensions are expanded to spectral dimension and then passed through 1D convolutions similar to the encoder network before passing through the fully connected layers

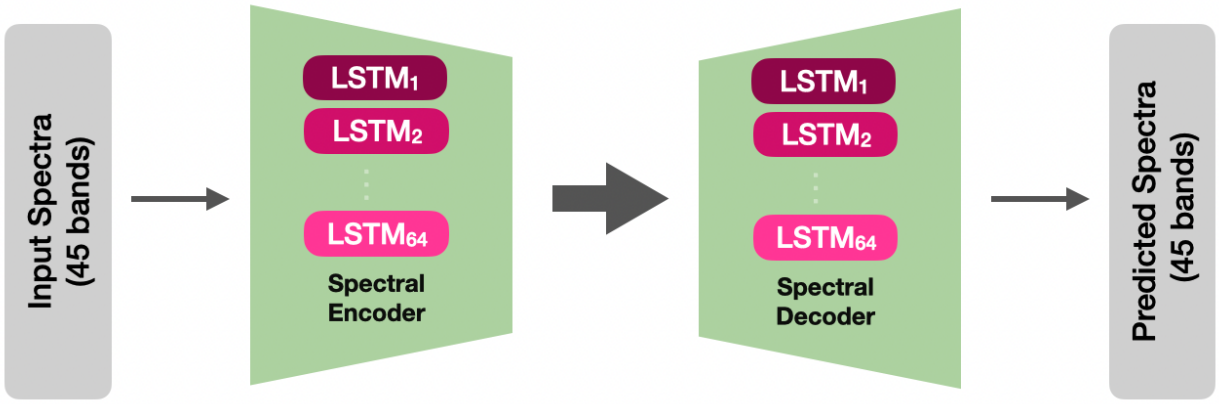


Figure 2.8: Seq-Mem SC: The LSTM-based spectral compressor is configured with a single layer, consisting of 64 LSTM units in both the encoder and decoder modules

2.5.3 Seq-Mem - SC

Long Short-Term Memory (LSTM) networks, are recurrent neural networks capable of capturing dependencies within sequential data, utilize explicit memory components to significantly enhance learning. The Seq-Mem SC employs LSTM modules to process spectral signals as continuous sequences, akin to how Large Language Models (Bansal, 2018) handle word sequences. This capability is especially relevant given the correlation observed between adjacent bands in spectral signals, aligning with the sequential nature inherent to LSTM architectures. In our current investigation, a single-layer configuration with 64 LSTM units are employed for both the encoder and decoder modules.

2.5.4 Performance Comparisons

All three spectral compressor variations are constructed and trained using hyperspectral data from Cuprite Hills dataset, introduced in Chapter 3 (in Section 3.3). Similar to Section 2.4, the 2000nm to 2500nm wavelength regions, comprising of 45 bands, are used for training the spectral compressor variations. Cuprite Hills dataset has a total of 600k hyperspectral measurements. From those, 120k (20%) hyperspectral measurements are randomly chosen and processed through the spectral compressors. The error metrics between the original and predicted spectra are calculated for the three SC variations and shown in Table 2.2. A histogram of the l_1 error between the predicted and original spectra for five different wavelength ranges from 2μm to 2.45μm is shown in Figure 2.9. Seq-Mem SC exhibits superior performance compared to both FCNN and MCConv SCs, likely due to its treatment of the spectral signal as a continuous signal, which preserves the dependency between spectral bands. MCConv follows closely, benefiting from the 1D convolutions. FCNN demonstrates the poorest performance among the three models.

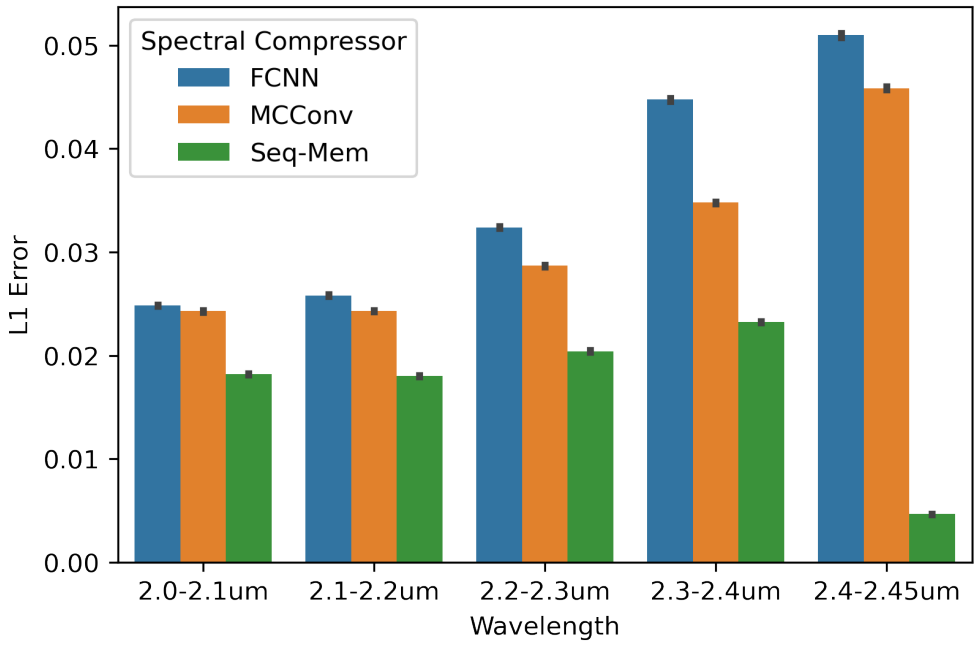


Figure 2.9: Reconstruction errors are computed for spectra randomly chosen from Cuprite Hills data using all three SCs, with the L1 error calculated between the original and reconstructed spectra. The plot depicts the reconstruction error across different wavelengths. Among the three, Seq-Mem-SC exhibits the best performance.

Spectral Compressor	Cross-correlation	RMSE	MSAE
FCNN	0.8268	0.0509	0.0677
MCConv	0.8694	0.0435	0.0578
Seq-Mem	*0.9474*	*0.0277*	*0.0358*

Table 2.2: Quantitative comparison between the different SC networks. Overall Seq-Mem SC performs better with lesser error.

MCConv demonstrates a 14.6% improvement over FCNN in mean spectral angle error, while Seq-Mem exhibits a more significant improvement of 47.1% (Table 2.2). Given the superior performance of Seq-Mem SC, it is used as the spectral compressor for the rest of this study. The spectral predictions shown in Figure 2.4 are generated using the Seq-Mem spectral compressor.

2.6 Latent Vector Size

The size of the latent vector produced by the spectral encoder plays a crucial role in the compression process. The smaller the latent vector, the higher the compression and greater the information loss and vice-versa. The latent vector serves as the bottleneck in the compression process influencing the accuracy of the spectral compressor.

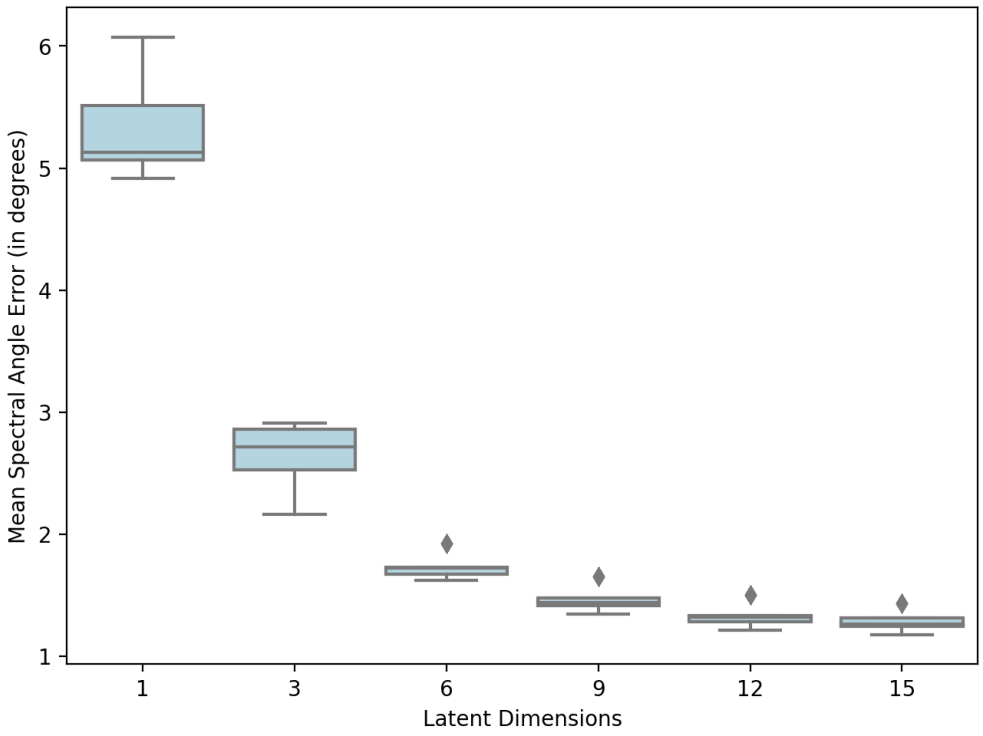


Figure 2.10: Plot showing the mean spectral angle error as a function of latent dimension of a spectral compressor

Several experiments were setup with SCs with different latent dimensions, recognizing that an larger vector may hinder compression effectiveness, while a smaller vector can limit the network's expressiveness. Multiple Seq-Mem SCs were trained with latent vector sizes ranging from 1 to 15, and their mean spectral angle error are assessed. As the latent vector size increases, the mean spectral angle error decreases. However, beyond a latent vector size of 12, the rate of improvement diminishes (refer to Figure 2.10). Consequently, a latent vector size of 12 is opted for the remainder of this study.

2.7 Summary

In this chapter, spectral compressors were explored as an effective tool to compress spectra. Our exploration involved a comparison between auto-encoder and VAE, with the conclusion that VAEs demonstrate superior performance. The difference between the loss function of auto-encoder and VAE was discussed and how it impacts the reconstructed spectra. Several quantitative evaluation metrics to compare the predicted and the original spectra were introduced. Furthermore, comparative analysis of three network variations of spectral encoder and decoder modules were conducted, revealing that Seq-Mem SCs outperform the others significantly, prompting their selection for

subsequent experiments. Additionally, experiments were conducted to determine the optimal latent dimension of 12 for the SC.

Chapter 3

Spectral Mapping

In this chapter, existing literature on spectral mapping is reviewed. Next, the process to build a spectral map using a trained model is looked at. To validate these methods, three field datasets used in this research are introduced. Finally, spectral compressors with varying levels of generalizability are examined and compared assessing their spectral reconstruction accuracy.

3.1 Prior Work

Spectral mapping involves associating spectral measurements to spatial location. It is conducted on both orbital and rover scales, utilizing distinct techniques and serving various purposes.

3.1.1 Airborne to Orbital Scale

Spectral mapping can be achieved, in a orbital scale, through comprehensive point observations using orbital (Goetz et al., 1983; Hubbard and Crowley, 2005) or aerial (Chapman et al., 2019) instruments. The primary goal of spectral mapping is assisting scientific and technological applications like remote sensing, resource prospecting, crop monitoring, environment monitoring, etc. Instruments such as Landsat (Williams et al., 2006), one of the longest-running satellite programs, have significantly evolved to become pivotal in spectral mapping on a global scale. Through this program, notable breakthroughs in handling digital satellite imagery have set a precedent for future missions, enhancing various scientific applications (Wulder et al., 2022). Other orbital instruments like the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) (Fujisada, 1995) have diversified applications from modeling urban heat (Mallick et al., 2013; Nath et al., 2021), supporting mining activities (Charou et al., 2010) and disaster management (Duda and Abrams, 2012) and continue to enhance the spectral mapping capabilities. Hyperspectral instruments like the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) (Green et al., 1998) with finer coverage of the electromagnetic spectrum, has

pushed the boundaries further in the identification of minerals (Kruse, 2002) and precision in spectral imaging (Swayze et al., 1992). These advancements have revolutionized spectral mapping, enabling precise analysis and broadened applications.

3.1.2 Rover Scale

Spectral mapping on a rover scale is crucial for autonomous exploration, aiding navigation decisions (Thompson, 2008). However, the spatial resolution of orbital spectra often falls short of the requirements for such navigation planning tasks. There are several works in literature that attempt to improve the resolution of orbital spectra. Thompson et al. (2015) developed an objective function to extract pure spectral signatures to refine orbital measurements. Vargas et al. (2019) enhanced spatial resolution of spectral data through a fusion model used for spectral unmixing. Bodrito et al. (2021) advanced this further by using handcrafted image priors and sparse coding to train a deep learning model, improving the resolution of orbital spectra. These methods view spectra as collections of pure materials, called end-members, and focus on quantifying their proportions through a process known as spectral unmixing, which can be modeled through linear combinations Quintano et al. (2012) or non-linear approaches Heylen et al. (2014).

Candela et al. (2020) introduced a model that synthesizes the spectral signal within a regression framework, without dividing it into separate elements. Operating in a continuous space avoids the errors associated with quantization in any discrete setup. Candela et al. (2020)'s forumulation learned the spectral model employing Gaussian processes (GPs), leveraging their non-parametric nature and flexibility in predicting unknown functions (Rasmussen and Williams, 2006). Numerous studies (Kumar et al., 2019; Binney and Sukhatme, 2012; Binney et al., 2010; Popović et al., 2020) utilize GPs to model real-world phenomena as scalar field due to their ability to capture the continuous nature of the world. While Akhtar and Mian (2020) employed Gaussian processes for spectral modeling, our approach differs by modeling the latent space, akin to the work of Candela et al. (2020). Additionally, this latent space is used to enable camera images to make spectral predictions.

Unlike exhaustive observations made by orbital instruments, our approach focuses on building a model using GPs, that learns the association between location and spectra using a few training examples. Next, let us look at the steps involved in creating a spectral map.

3.2 Spectral Mapping Process

Spectral mapping involves creating a map where each pixel represents a spectral measurement. A model is trained and used to make comprehensive predictions for each pixel of the spectral map.

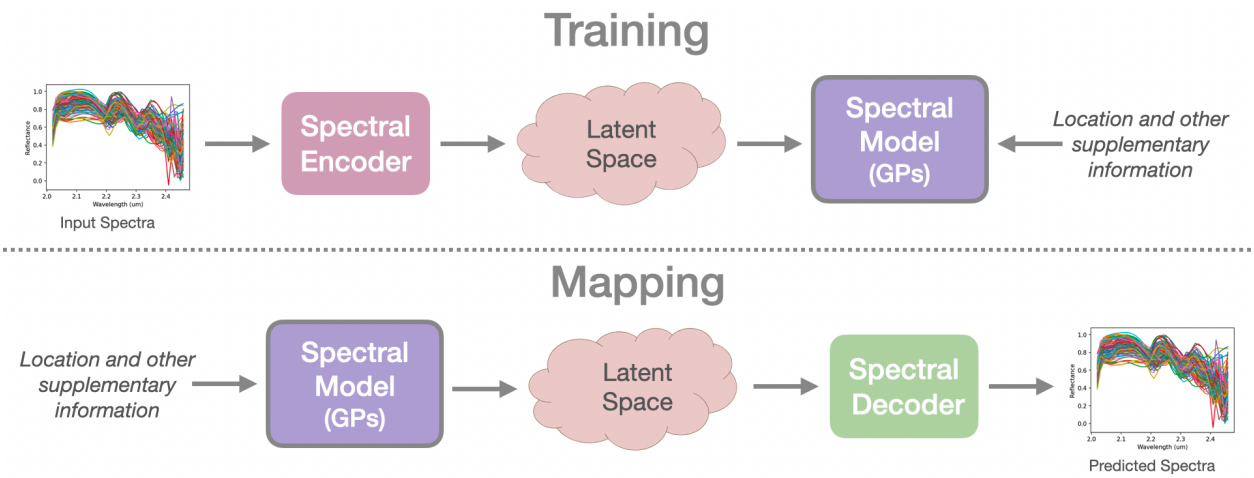


Figure 3.1: Training and mapping steps of the simple-sensor-spectral model. During the training step, the spectral model learns the association between the latent space and positional information. During the mapping step, given location information, the spectral model produces latent values which are then reconstructed as spectra by the spectral decoder.

This process consists of two main steps: in the *training phase*, the spectra are compressed into the latent space using the spectral encoder, allowing the spectral model to learn the association between the latent representation of the spectra and their respective locations. This model is constructed using GPs, chosen for their effectiveness in learning from limited training examples. In the *mapping phase*, the spectral model generates a latent prediction for each pixel on the map, which is subsequently decoded by the latent decoder to reconstruct the uncompressed spectra as shown in Figure 3.1. GPs allow mapping at any desired resolution. This is because GPs use kernel functions to define the covariance between any two points in the input space. The kernel function effectively measures similarity and ensures smoothness in the function modeled by the GP. The kernel function specifies how changes in the input space influence correlations between outputs, leading to a continuous surface or function that smoothly interpolates between observed data points.

In addition to the location information other supplementary information like multispectral priors from orbital data can be used. However it should be noted that this supplemental information should be available both during training and the mapping phases.

Next let us explore the datasets used in this research.

3.3 Field Datasets

The concepts proposed in this study require validation using real-world data, and for this purpose, data from Cuprite Hills region in Nevada is primarily used for most of the

Details/Site	Cuprite Hills	Hopi Buttes	Yellow Cat
Location	37.53 N 117.23 W	36.20 N 110.90 W	38.85 N 109.54 W
State	Nevada	Arizona	Utah
AVIRIS	Yes	Yes	Yes
AVIRIS-NG	Yes	No	No
ASTER	Yes	Yes	Yes
Landsat	Yes	Yes	Yes
Dronemaps	6	19	21

Table 3.1: Data products available from the three field sites

analysis. Cuprite Hills, spanning a 10-square-kilometer area (refer to Figure 3.2), boasts a diverse range of minerals and geological features. Notably, it is abundant in clays (Montmorillonite, Kaolinite, Illite), carbonates (including calcite and various mixtures), sulfates (such as Gypsum), as well as other minerals like Chlorites, Serpentines, and Zeolites. This region has been extensively studied by geologists over decades (Swayze et al., 2014) and serves as a benchmark for calibrating new orbital instruments.

Achieving high performance on a specific dataset does not necessarily guarantee effectiveness on other datasets with different characteristics. Therefore, evaluating the model's performance across multiple datasets exhibiting diverse mineralogical compositions and varying spectral characteristics is essential to validate its generalizability. Furthermore, testing the model on datasets collected using different sensors, platforms, and resolutions can help assess its robustness and identify potential limitations. To enable this and to validate the generalizability of our approach, data from two other mineralogically diverse sites are utilzed. Hopi Buttes, a monogenetic volcanic field situated on the Colorado Plateau in northeastern Arizona, and Yellow Cat, an old uranium mining site rich in minerals located in Utah. The summary of the data products used from these sites is listed in Table 3.1.

3.3.1 Advanced Spaceborne Thermal Emission and Reflection Radiometer

The Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) [1] (Fujisada, 1995), a multispectral instrument launched in December 1998, measures reflectances in Visible and Near-Infrared (VNIR) (520nm to 860nm) and Shortwave Infrared (SWIR) (1600nm to 2430nm) bands and Thermal InfraRed (TIR) (8125nm to 11650nm) bands. With fourteen bands, ASTER provides detailed maps of surface temperature,

[1] ASTER and Landsat datasets are publicly available from USGS's EarthExplorer (earthexplorer.usgs.gov) tool

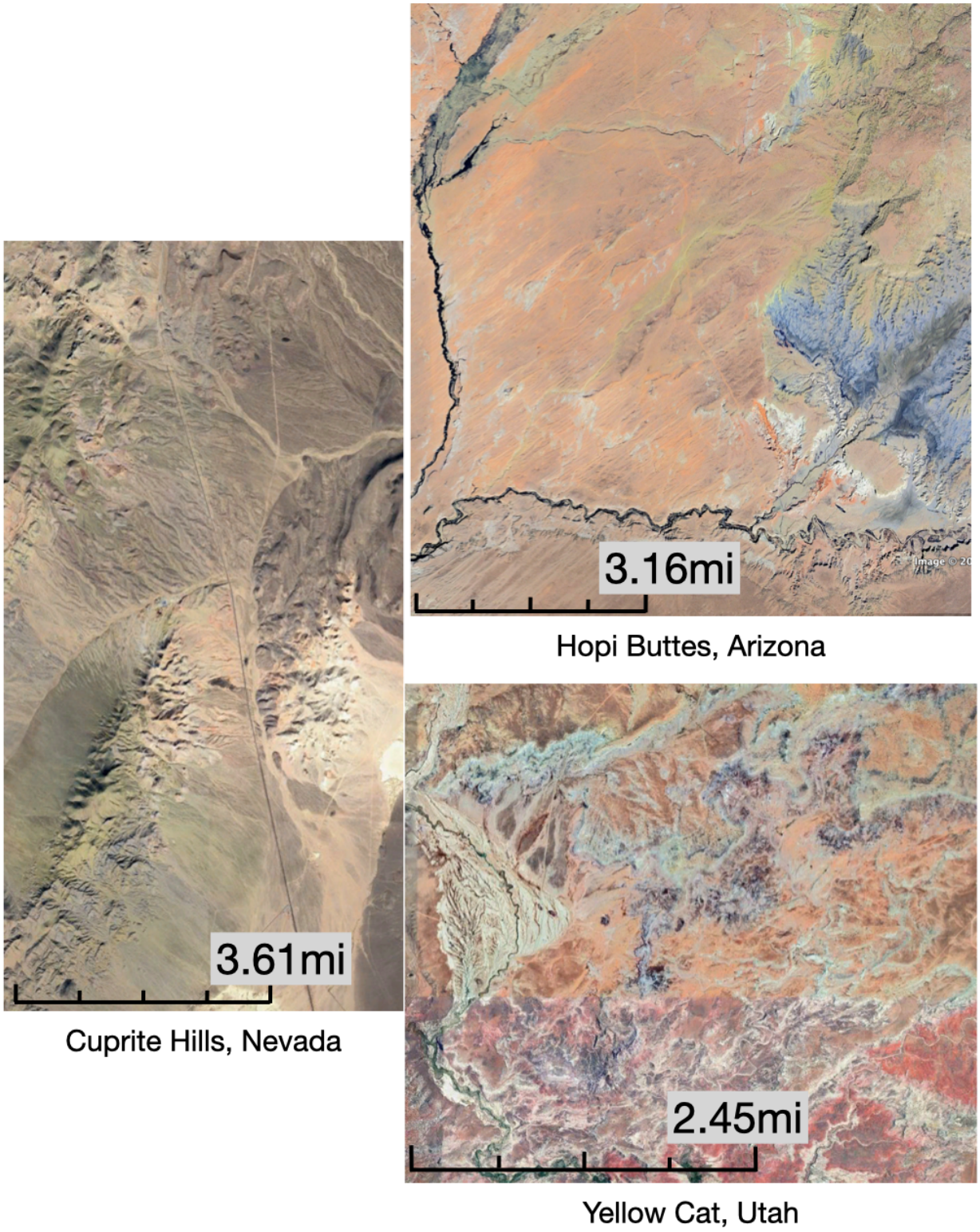


Figure 3.2: Three fields sites - Cuprite Hills in Nevada, Hopi Buttes in Arizona and Yellow Cat in Utah spanning a few kilometers each

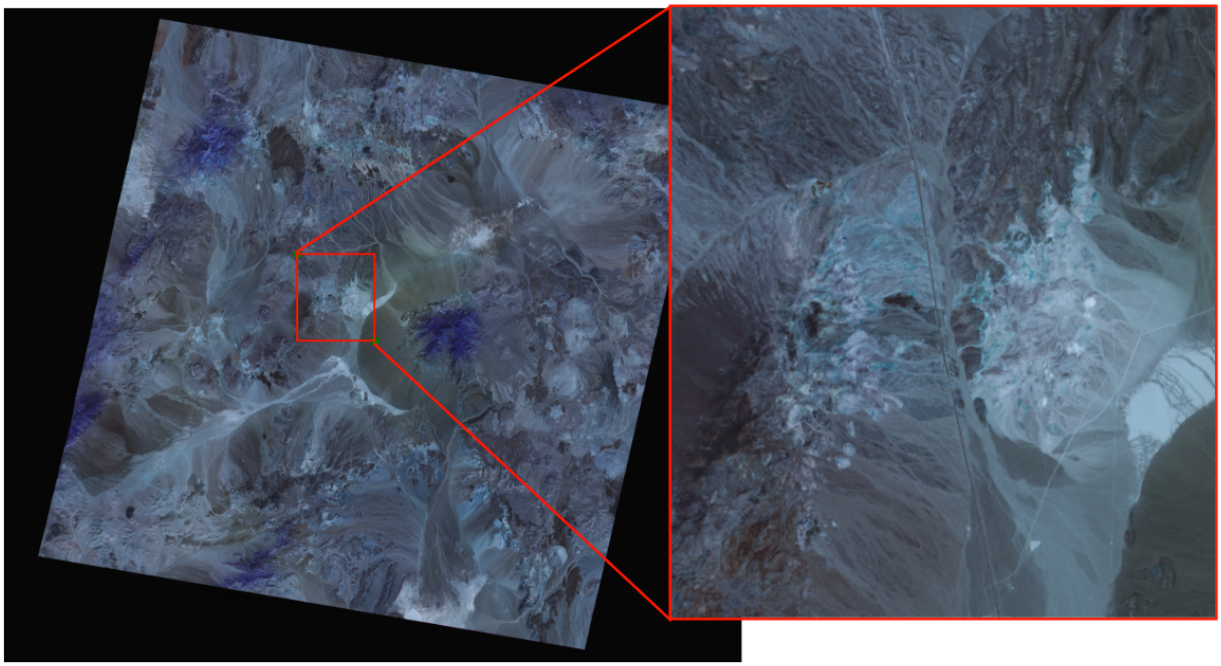


Figure 3.3: First 3 bands of ASTER dataset (AST_L1T_00304282007183857_20150519072639_69421) with Cuprite Hills region zoomed in.

emissivity, reflectance, and elevation, offering pixel resolutions of 15m for the first three bands, 30m for the next six SWIR bands, and 90m for the remaining five TIR bands. For our purposes, only the VNIR and SWIR bands are utilized, disregarding the TIR bands due to their lower resolution and higher sensitivity to noise. ASTER data is encoded with 8-bit precision. The RGB bands of the ASTER dataset from Cuprite Hills is shown in Figure 3.3.

3.3.2 Landsat

Landsat data is obtained from the latest Landsat 9 (Markham et al., 2016) instrument, equipped with an imager (OLI-2) and a thermal infrared sensor (TIRS-2), covering visible (bands 1 to 4), VNIR (band 5), SWIR (bands 6 and 7), and thermal infrared radiation (bands 10 and 11) from Earth's surface. The pixel resolutions for OLI-2 and TIRS-2 bands are 30m and 100m, respectively, while Band 8 provides panchromatic imagery with a 15m spatial resolution, and Band 9 measures thin high-altitude clouds. Similar to ASTER, only seven bands from the OLI-2 imager are used. These are level 2 surface reflectance data corrected for atmospheric effects (Ihlen, 2019). Unlike ASTER's 8-bit precision, Landsat 9's data is encoded with 14-bit precision, offering increased resolution that contributes to improved spectral predictions. The RGB bands of the Landsat 9 dataset for Cuprite Hills are shown in Figure 3.4.

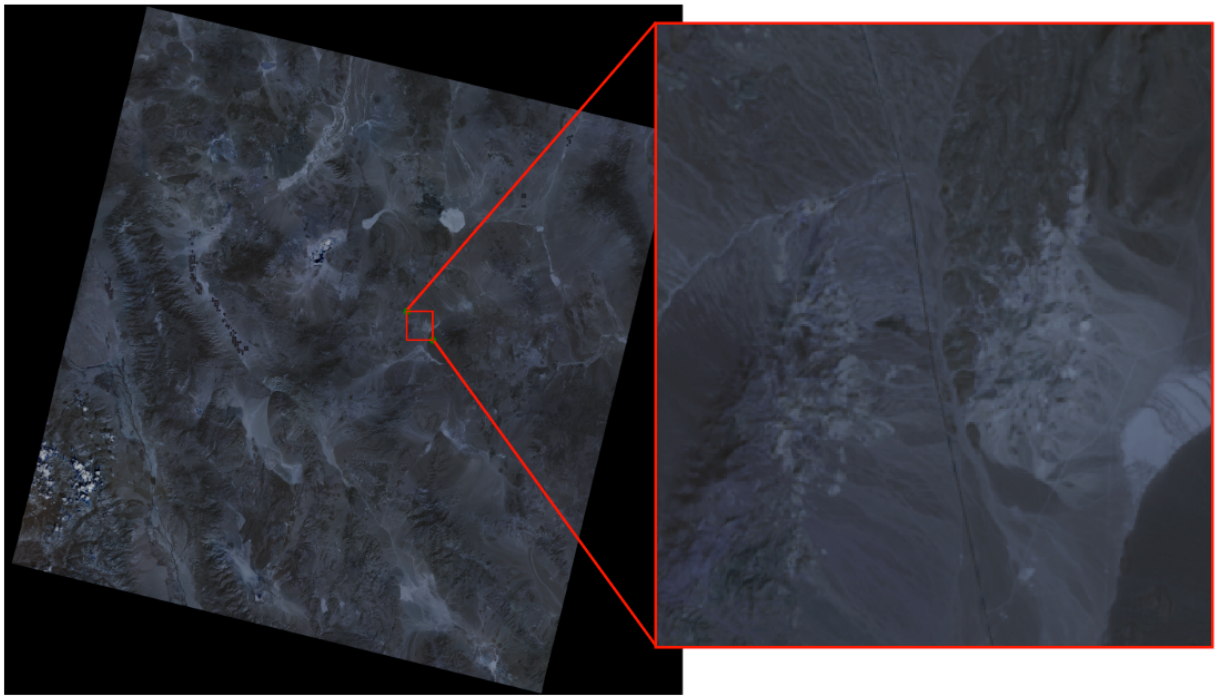


Figure 3.4: First 3 bands of LANDSAT dataset (LC09_L2SP_041034_20220919_20220921_02_T1) with Cuprite Hills region zoomed in.

3.3.3 Airborne Visible InfraRed Imaging Spectrometer - Classic

The Airborne Visible InfraRed Imaging Spectrometer (AVIRIS) (Vane et al., 1993) has been operational since 1986, capturing hyperspectral data by measuring reflectances in VNIR and SWIR bands. Unlike ASTER and Landsat 9, AVIRIS is not a satellite-based instrument but is instead mounted on aircraft to achieve the desired spatial resolution. The dataset utilized in our research has a spatial resolution of 20 meters per pixel, with wavelength measurements spanning from 380nm to 2510nm at 10nm intervals. Specifically, our focus is on the 2μm to 2.5μm spectral bands as hyperspectral ground truth for all experiments. Cuprite Hills and Yellow Cat datasets are calibrated to surface reflectance using Radiative Transfer Ground Calibrated (RTGC) method (Clark et al., 2002). Hopi Buttes dataset is radiometrically corrected and Atmosphere Removal Algorithm (ATREM) (Gao et al., 2000) is applied to get atmospherically corrected surface reflectances (L2 data products) (Lundeen, 2022). For each spectrum, the 2μm to 2.5μm wavelength region is extracted (45 bands) and scaled between 0 and 1 and used as reflectance. The region covered by AVIRIS in our experiments is depicted in Figure 3.2, while Figure 3.5 illustrates the mean of all AVIRIS spectra from each site, showcasing the diversity of minerals present in these regions.

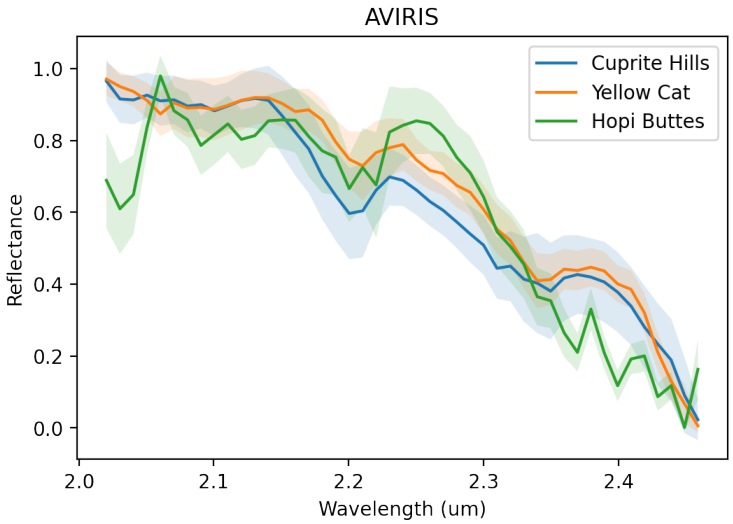


Figure 3.5: Mean (darker shade) and variance (lighter shade) of all the AVIRIS spectra in 2µm-2.5µm bands from the three field sites

3.3.4 Airborne Visible InfraRed Imaging Spectrometer - Next Generation

The Airborne Visible InfraRed Imaging Spectrometer - Next Generation (AVIRIS-NG) (Hamlin et al., 2011; Chapman et al., 2019) measures reflectances in the same wavelength range from 380nm to 2510nm as AVIRIS but with even finer sampling at 5nm intervals. Like AVIRIS, AVIRIS-NG is an airborne instrument mounted on aircraft. In our study, data from five AVIRIS-NG flights over the Cuprite region which have been stitched together to cover an area of 10 square kilometers is used. The visible wavelength mosaic as seen by AVIRIS-NG sensor from these flights are shown in Figure 3.6 (Thompson et al., 2018a). This dataset offers a significantly finer spatial resolution of 3.5 meters per pixel. Unfortunately, AVIRIS-NG data for the Yellow Cat or Hopi Buttes regions is not available.

3.3.5 Dronemaps

During our explorations (Cuprite Hills in 2019, Hopi Buttes in 2021 and Yellow Cat in 2022), multiple drone flights are conducted to capture footage of the studied regions. The collected images are later stitched together to create ortho-projections with a resolution ranging from 3 to 5 centimeters per pixel, referred to as dronemaps. Each dronemap is built from images collected from a drone flight lasting 20 to 25 minutes (see Figure 3.7 - Figure 3.11 for the individual dronemaps). When covering larger regions, the drone is flown at higher altitudes, resulting in lower resolution dronemap. Despite efforts to capture footage during mid-day to avoid shadows, some flights inadvertently produced

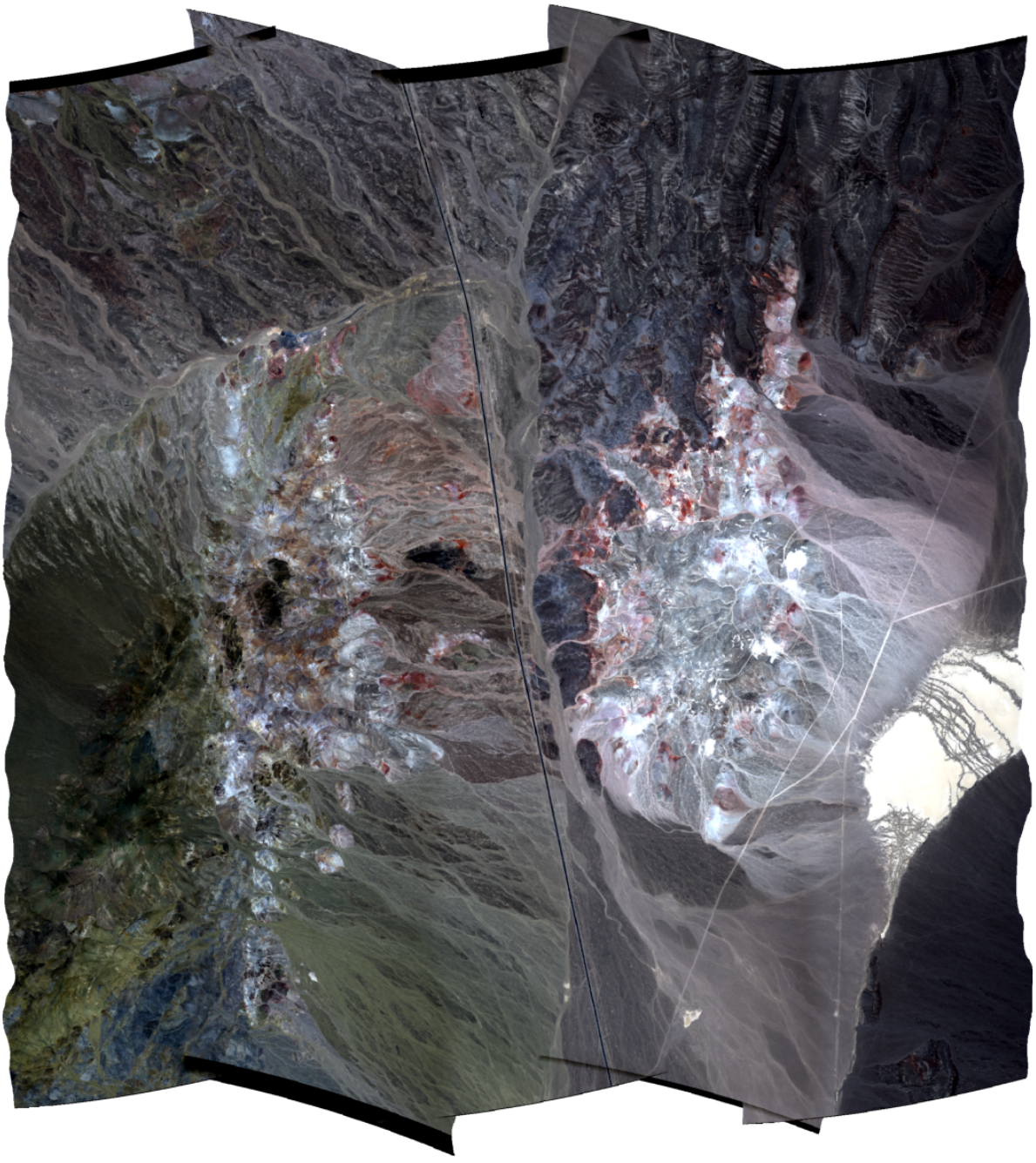


Figure 3.6: Visible wavelength mosaic of Cuprite Hills, Nevada, as seen by the AVIRIS-NG sensor during an overflight in June 2014

dronemaps with shadows, either due to low Sun angles or shadows cast by nearby terrain features (H02 and H06 in the Hopi Buttes dataset - refer Figure 3.8; Y01, Y02 and y07 in Yellow Cat dataset - refer to Figure 3.10). The dronemaps are numbered to correspond with the datasets used in subsequent chapters' experiments.

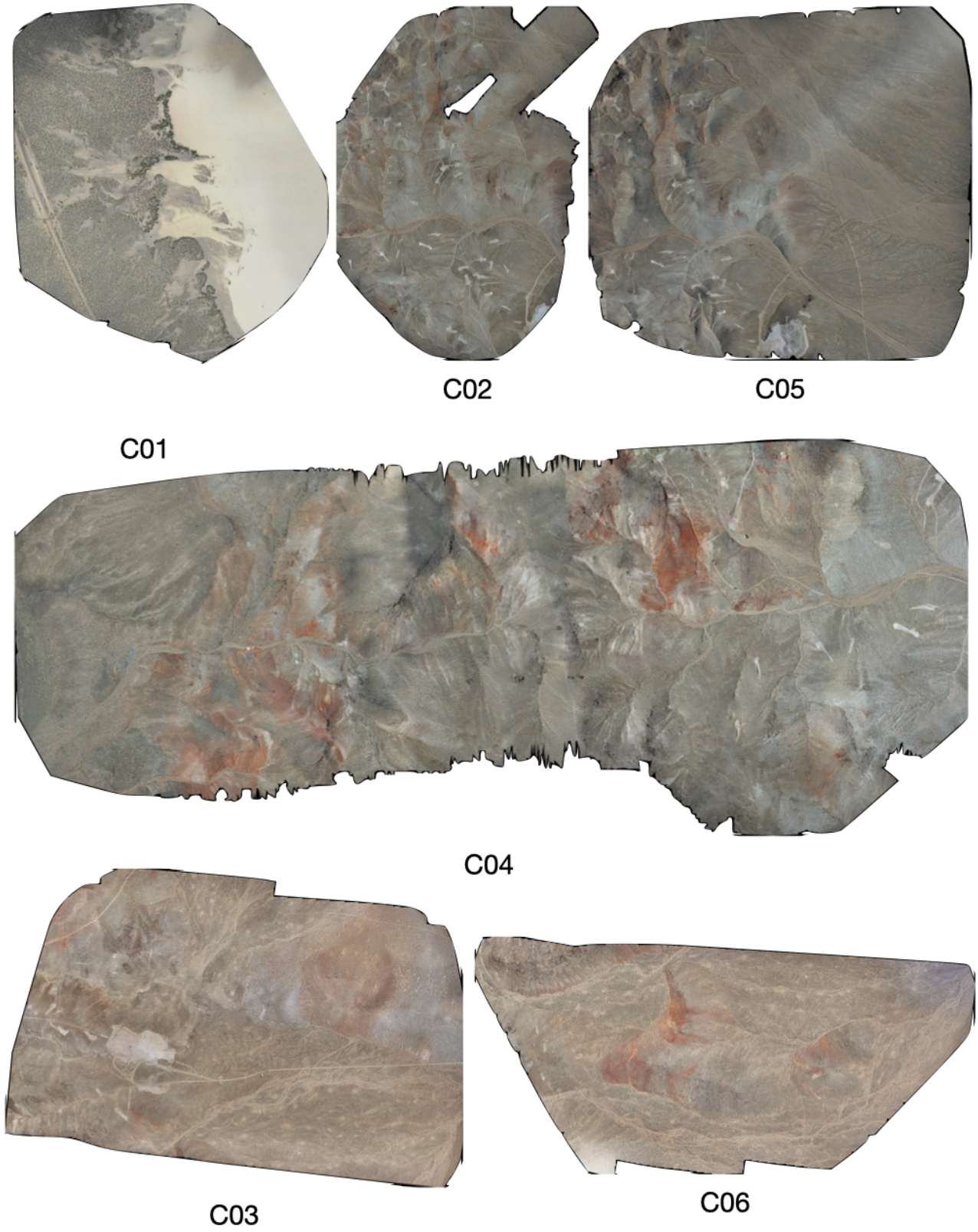


Figure 3.7: Dronemaps (C01-C06) from Cuprite Hills, Nevada

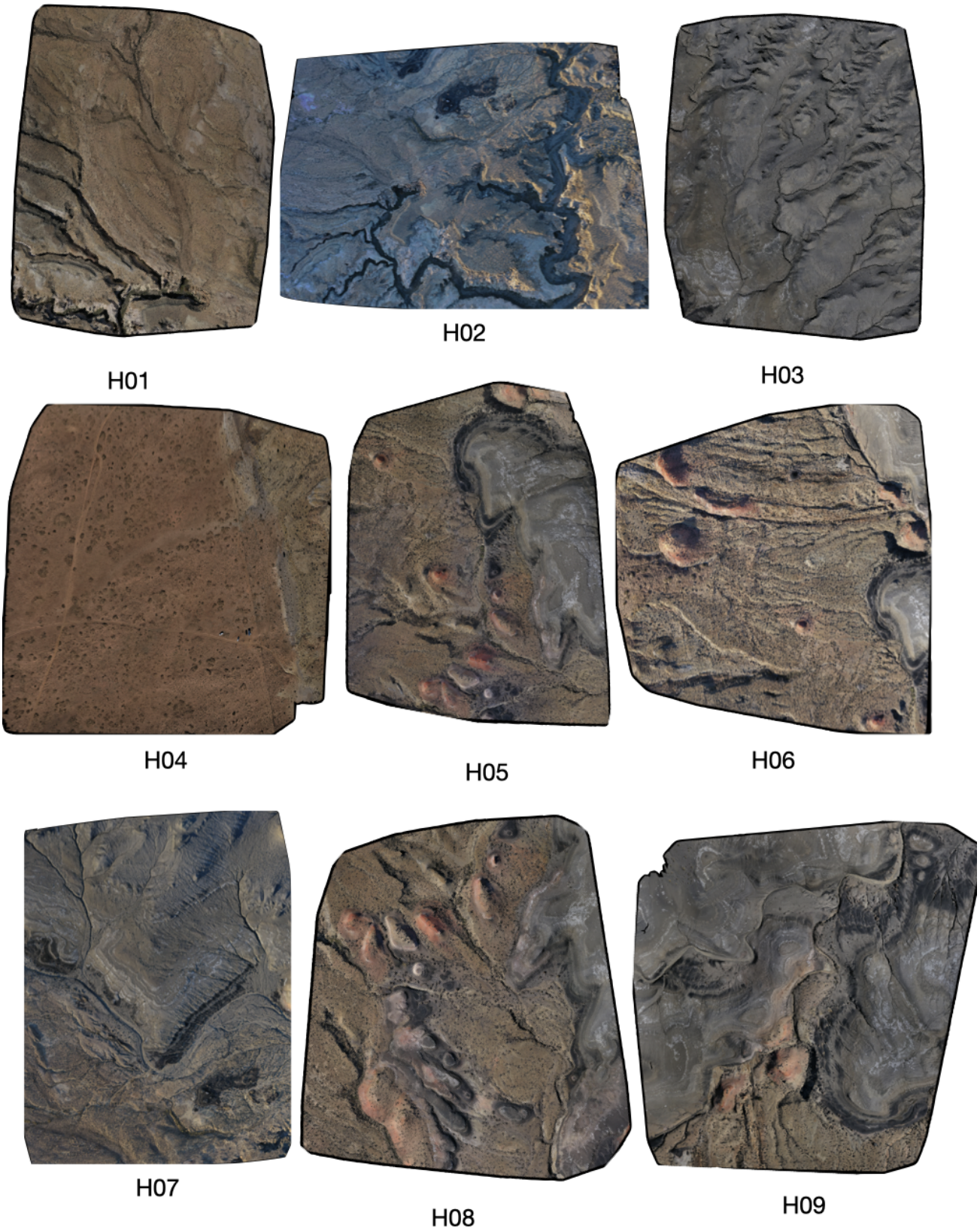


Figure 3.8: Dronemaps(H01-H09) from Hopi Buttes, Arizona

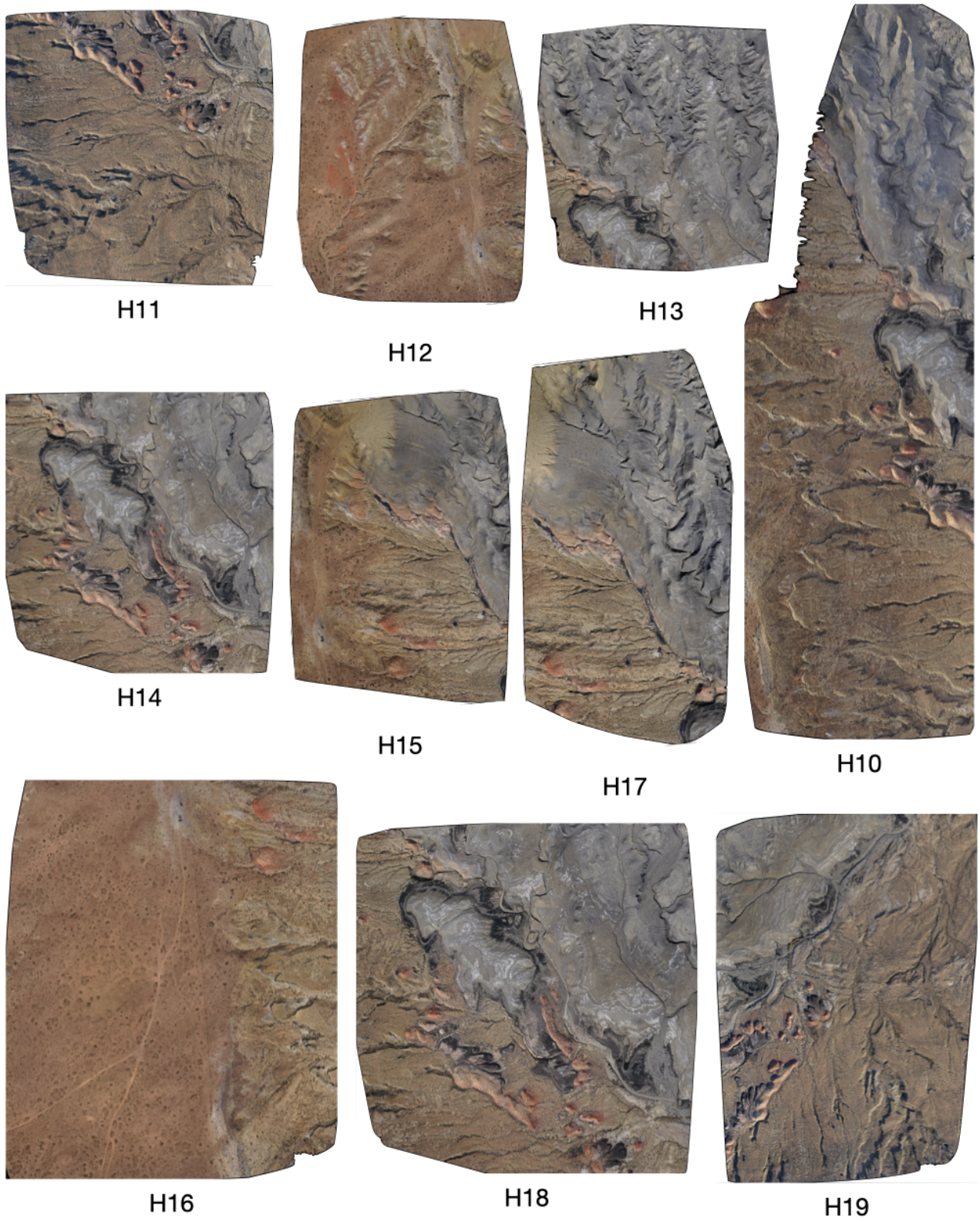


Figure 3.9: Dronemaps (H10-H19) from Hopi Buttes, Arizona

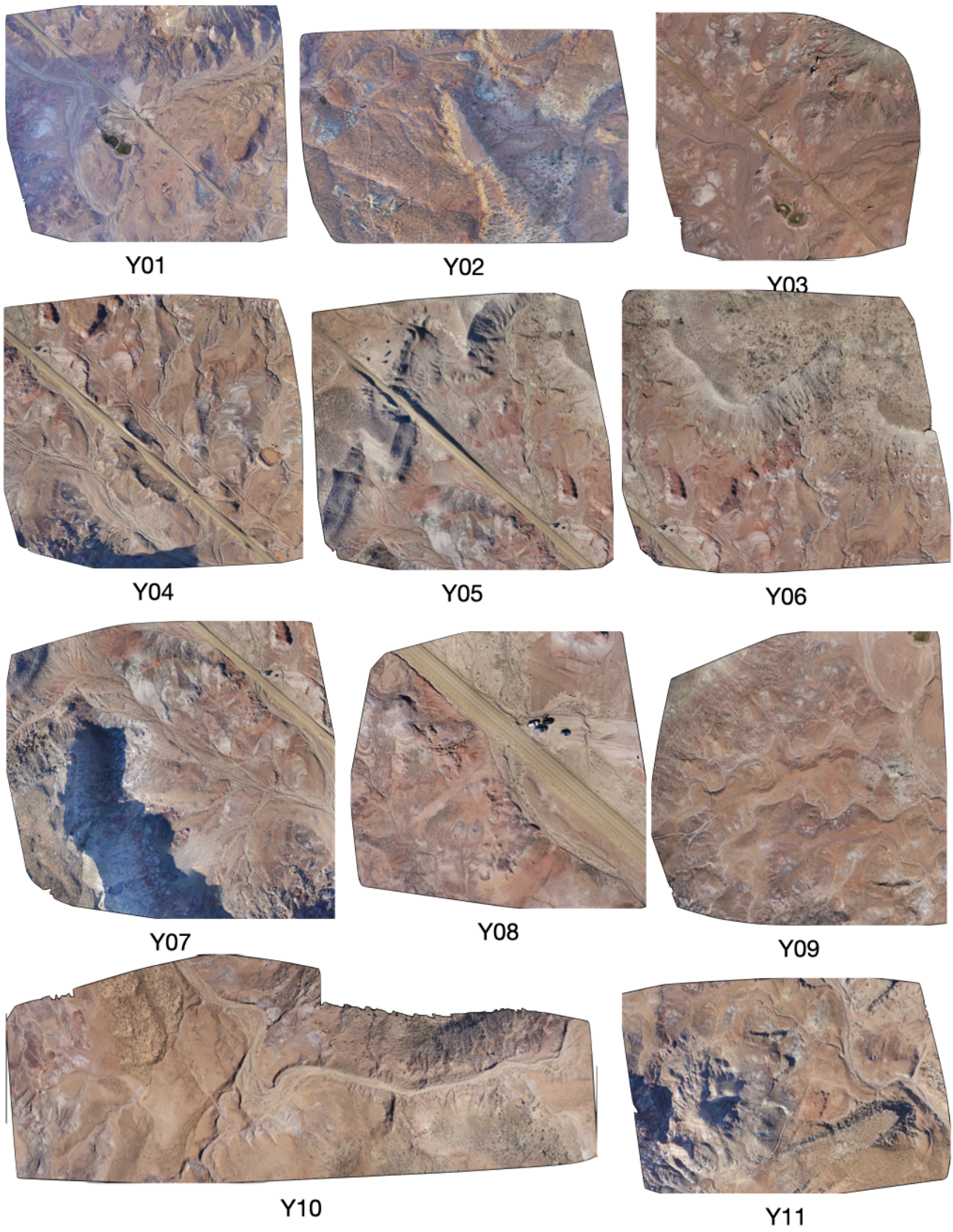


Figure 3.10: Dronemaps (Y01-Y11) from Yellow Cat, Utah

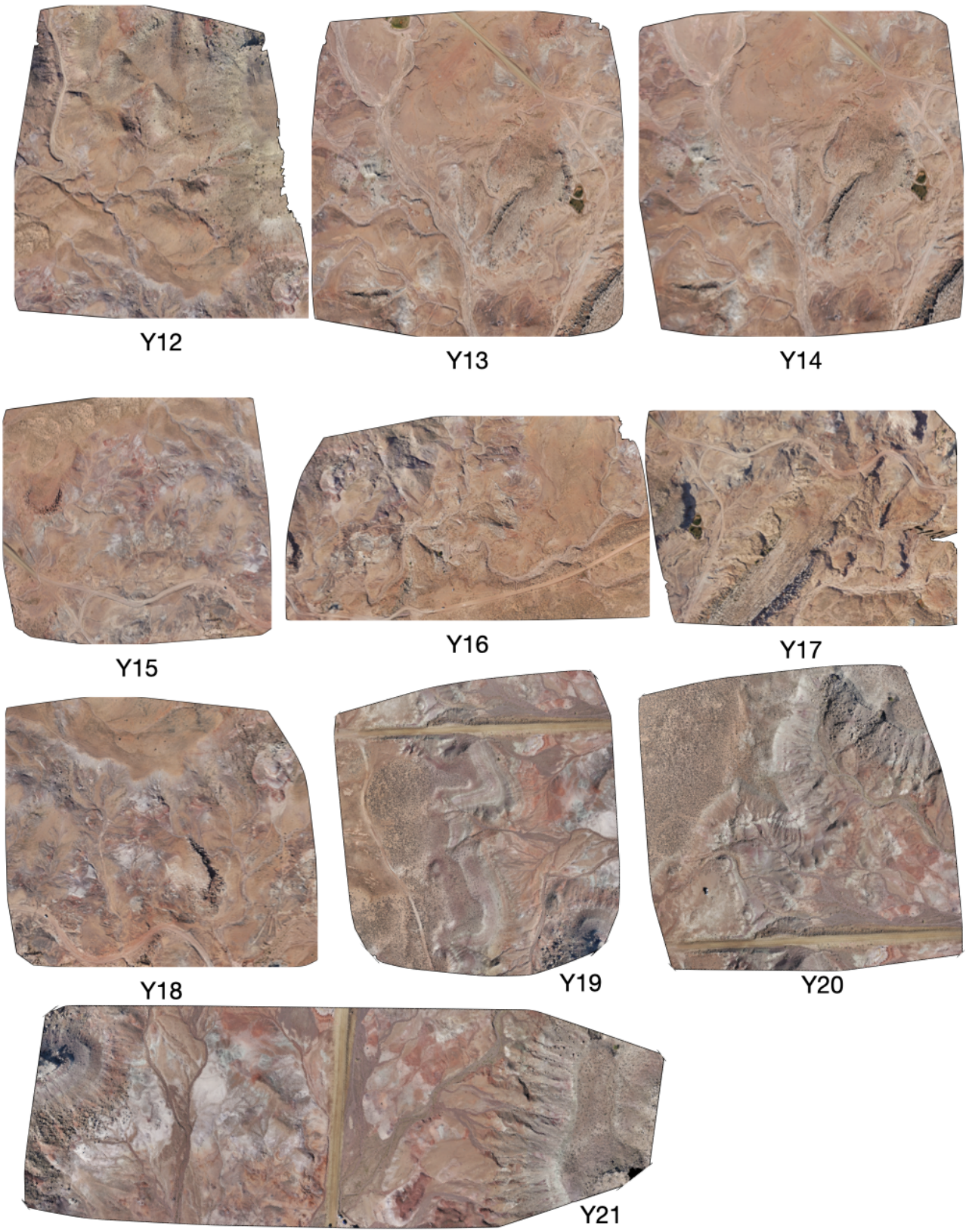


Figure 3.11: Dronemaps (Y12-Y21) from Yellow Cat, Utah

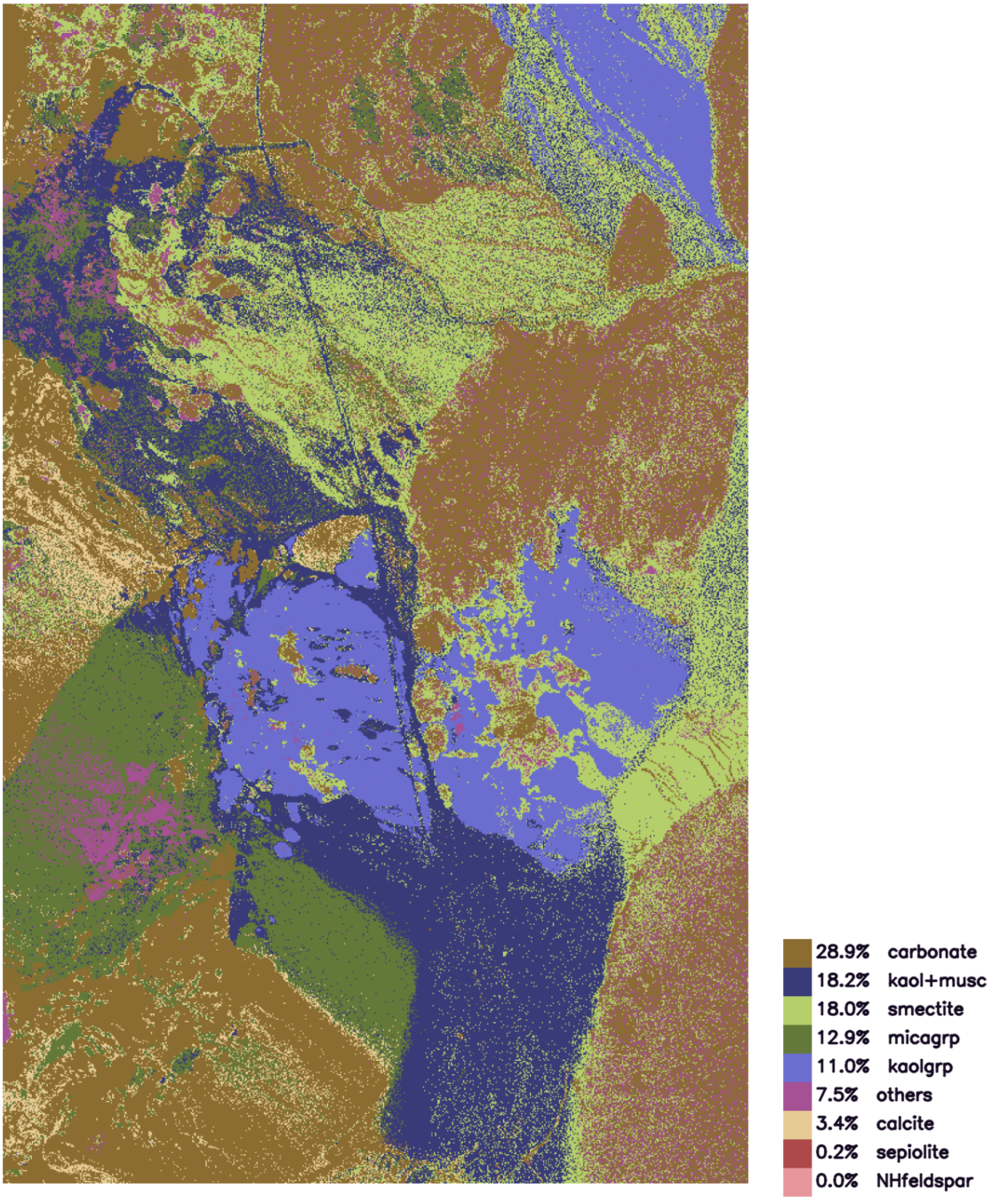


Figure 3.12: Mineral map of Cuprite Hills and the abundance for each mineral category

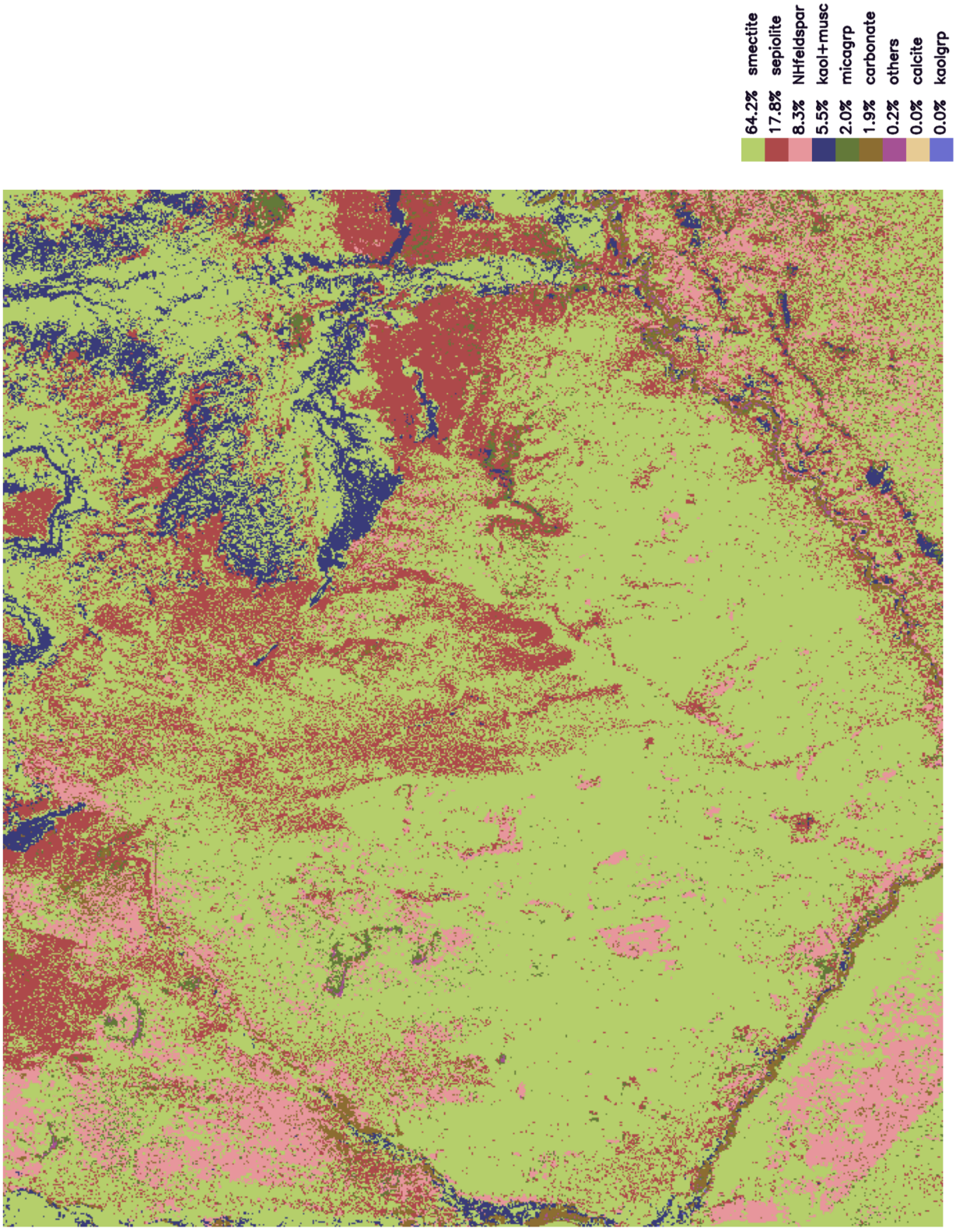


Figure 3.13: Mineral map of Hopi Buttes and the abundance for each mineral category

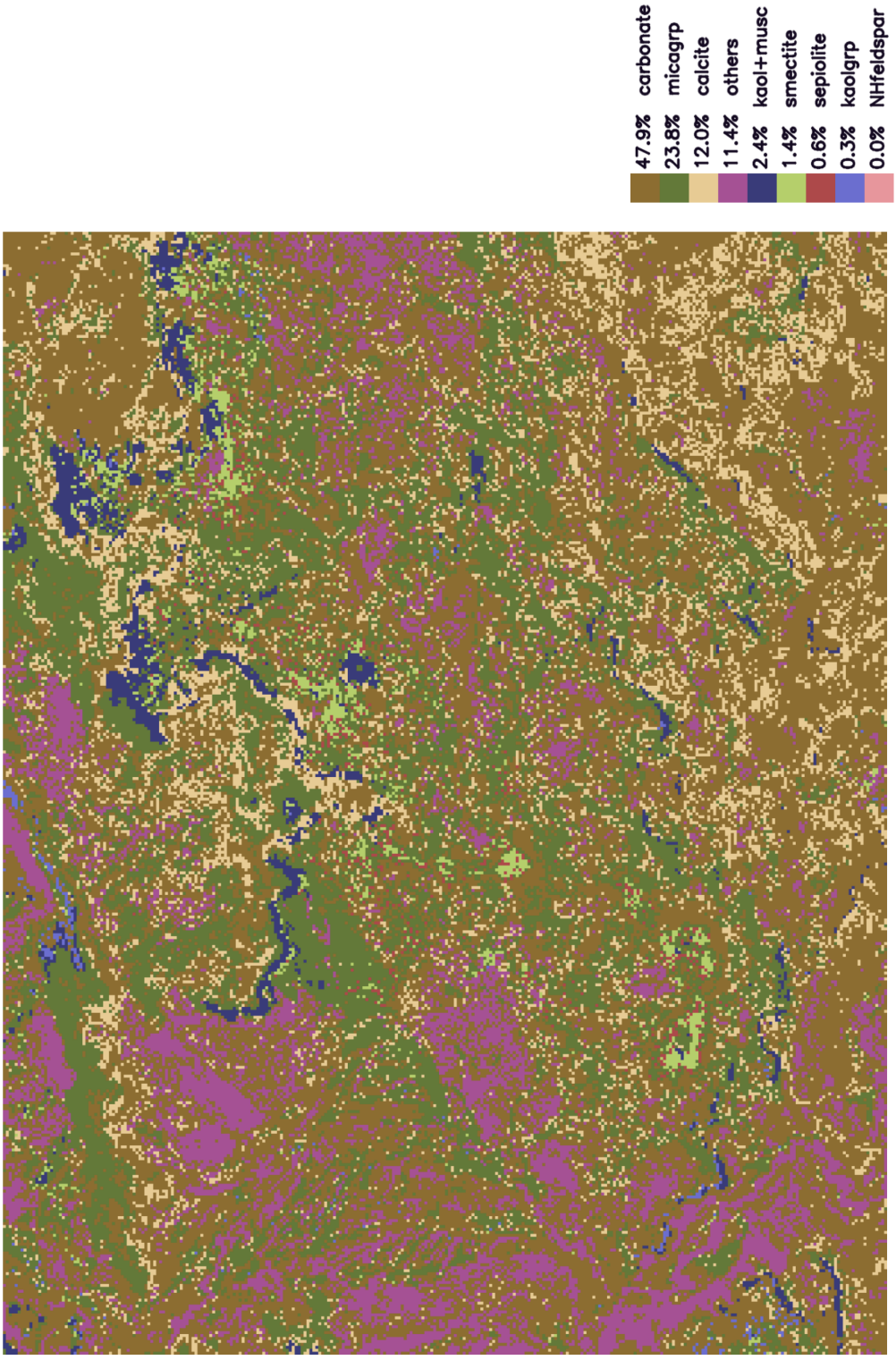


Figure 3.14: Mineral map of Yellow Cat and the abundance for each mineral category

3.3.6 Mineral maps

The mineralogical maps for the three sites are generated using version 5.27c (@f84e78b) of the Tetracorder tool [2] (Clark et al., 2003, 2024). The mineral maps are shown in Figures 3.12,3.13 and 3.14. The 2.0μm to 2.5μm wavelength bands from AVIRIS are used. For these bands, Teracorder identifies minerals using a total of 227 reference spectra which is then grouped into 9 mineral categories. The mineral categories are: kaolinite group (*kaolgrp*), kaolinite-muscovite (*kaol+musc*), mica group (*micagrp*), *smectite, carbonate, calcite, sepiolite, NHfeldspar*, and *others*.

Cuprite Hills features a mineralogically diverse environment with *carbonate* being the most prevalent mineral category, comprising less than 30% of the area. Additionally, the site includes significant portions of *kaolinite+muscovite* category at 18%, *smectite* at 18%, *micagrp* at 13%, and other minerals making up the remaining 20%. In contrast, Hopi Buttes displays minimal mineral diversity, with *smectite* accounting for 64% of its composition. *Sepiolite* and *NHfeldspar* category collectively represent another 26%, leaving just 10% for other mineral categories. Yellow Cat has a moderate level of diversity, where *carbonate* and *micagrp* categories dominate 72% of the site. *Calcite* and *other* categories fill the next 23%, with the final 5% comprising other mineral categories.

3.4 Training Input for SC

The spectral compressor (SC) can be trained using a variety of data sources, each impacting the reconstruction accuracy of the spectra. Three variations are examined, each designed to progressively enhance the spectral compressor's generalizability (see Figure 3.15.

3.4.1 SiteSpecific and Common SC

In the Site-Specific SC, training data is exclusively obtained from the specific site under consideration. For instance, for Cuprite SC, AVIRIS spectra from Cuprite Hills alone are utilized. This approach enables fine-tuning of the network and reduces prediction errors. However, since each site has diverse mineralogy, relying solely on site-specific data may introduce bias towards the minerals present in that region.

To address this bias, a Common SC is introduced. This approach accumulates spectra from all three sites to train a single SC. By leveraging data from multiple sites, the Common SC is expected to generalize better across different sites, without being constrained by the specific mineralogy present in a single site.

[2]Tetracoder (tetracorder.psi.edu) is an open-source tool to analyze spectra and identify minerals

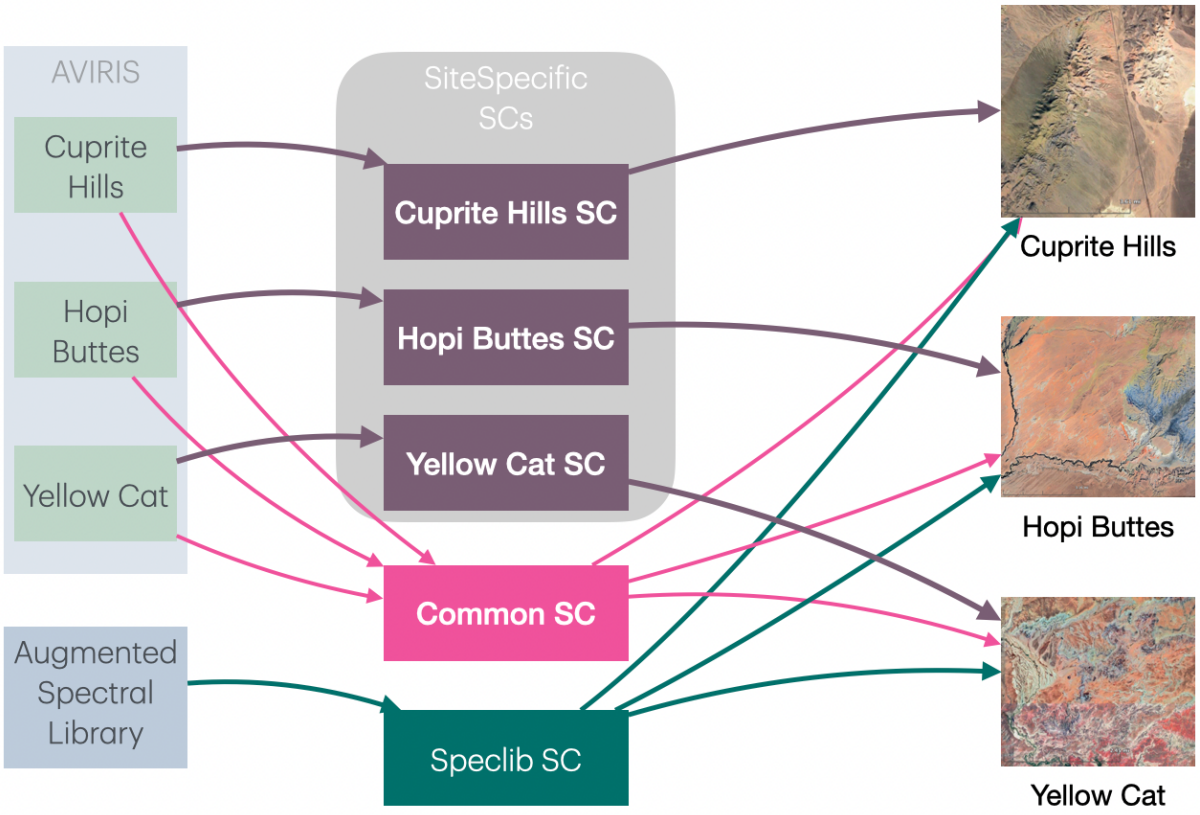


Figure 3.15: Spectral compressors differentiated based on the training input

Category	Number of spectra
Minerals	1275
Soils and mixtures	209
Artificial materials	290
Coatings	11
Liquids	23
Vegetation	286
Organic compounds	260

Table 3.2: Spectral categories in USGS spectral library

3.4.2 SpecLib SC

While the Common SC offers improved generalizability compared to the Site-Specific SC, it remains constrained by the data from the sites on which it is trained on. To achieve even greater generalizability, a different approach is opted for which utilizes data from a comprehensive spectral library instead of data from specific sites. The USGS spectral library (Kokaly et al., 2017), which contains a vast collection of spectra obtained from various sources including laboratory, field, and airborne spectrometers is chosen for this purpose. These spectra cover a wide range of materials, categorized into seven categories including pure minerals, soils and mixtures, etc. as noted in Table 3.2. This study utilizes version 7 (splib07) of the USGS spectral library, an expansion of the prior version, splib06, which serves as the basis for the mineral identification tool (Tetracorder).

However, the spectra in the spectral library may not fully capture the noise and interferences typically present in field observations along with the possible mixtures encountered in the field. To address these limitations, the spectral library dataset is augmented by introducing random mixtures of different spectra from the library itself. This augmentation increases the dataset size and incorporates various mixtures not initially seen in the spectral library into the training set. Let L denote the spectra from the USGS spectral library.

$$L \rightarrow S_1^l, S_2^l, \ldots, S_n^l$$

Where the subscript denote the i^{th}element in the set and the superscript denote the set it belongs to.

Sample m random spectra from L to form the random set, R

$$R \rightarrow S_1^r, S_2^r, \ldots, S_m^r$$

To create mixture ratios, sample p random values between 0 and 1.

$$A \rightarrow \alpha_1, \alpha_2, \ldots, \alpha_p \mid 0 < \alpha < 1$$

The augmented spectral dataset, M, is created by mixing spectra from the sets L and R with α weights

$$M \rightarrow \alpha_k S_i^r + (1 - \alpha_k) S_j^l \mid \forall_i \in \{1, 2, \ldots, m\}, \forall_j \in \{1, 2, \ldots, n\} \forall_k \in \{1, 2, \ldots, p\}$$

3.4.3 Comparative Analysis

To compare the spectral compressors, AVIRIS spectra for every point on all dronemaps are processed through the different spectral compressors and results are shown in Figure 3.16. Both the SiteSpecific SC and Common SC demonstrate comparable performance, indicating that training a SC using data from different sites does not significantly affect its efficacy compared to training on a specific site. This implies that training separate spectral compressors for each site may not be necessary and a single common version can be trained without noticeable impact on the reconstruction accuracy.

The Speclib SC exhibits higher error compared to the other two SCs. The error can be attributed to the lack of any prior information from the field, in the training set. Particularly in the Hopi Buttes dataset, the Speclib SC shows notably poorer performance, likely due to the lower mineral diversity in this region. While the specific minerals and mixtures found in Hopi Buttes are included in the training sets of SiteSpecific and Common SCs, the likelihood of these specific mineral mixtures being included in the augmented Speclib dataset is lower, resulting in higher errors. Despite these large errors, the mineral prediction capability of the Speclib SC remains relatively robust across all three sites (shown in Chapter 6).

Two conclusions can be drawn from these observations. First, creating a single, Common SC trained on data from a large dataset, that performs well on multitude of sites, is a viable option. The prediction accuracies might be minimally affected, if at all. Second, when no prior information about a site is available, these methods can still be applied using the Speclib SC, albeit with decreased mineral prediction accuracy.

3.5 Summary

In this chapter, existing work on spectral mapping in orbital and rover scale were discussed. Next, the steps involved in training a model and later using the model to create spectral maps were detailed. This is followed by three field datasets used in this research were introduced and the influence of different training data sources on the accuracy of spectral prediction achieved by the spectral compressor were examined.

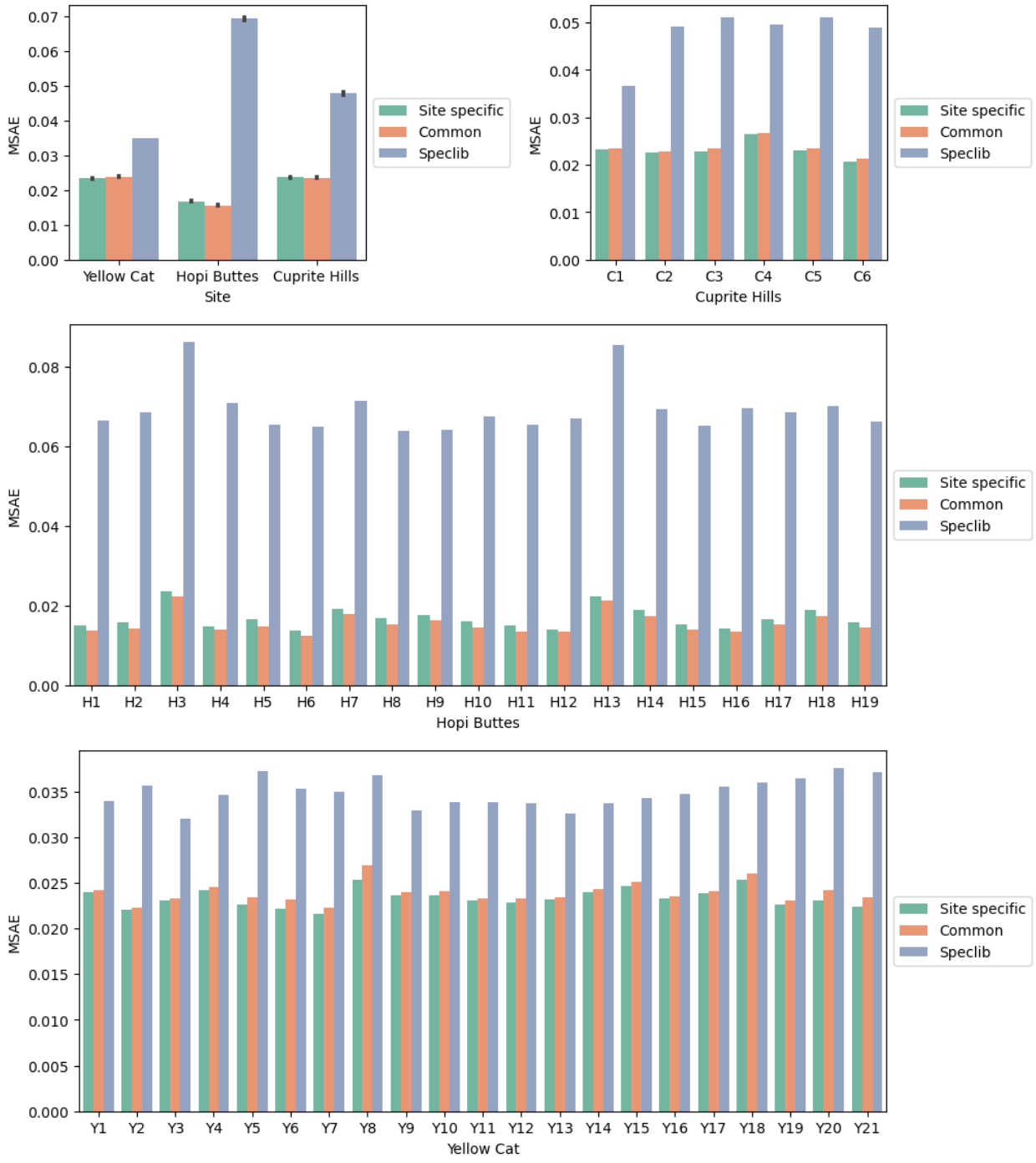


Figure 3.16: Performance comparison of SiteSpecific SC, Common SC and Speclib SC. Smaller bars indicate better performance. SiteSpecific SC and Common SC show comparable performance. Speclib SC shows the largest error, particularly, for Hopi Buttes it is relatively larger. One possible explanation is that the prediction accuracy of the most abundant mineral could have dropped causing the error to increase dramatically.

Chapter 4

Latent Encoders

The latent encoder, a pivotal element of the simple-sensor-spectral model, is presented in this chapter, which facilitates spectral mapping using camera images. Following this, the training methodology employed for the latent encoder is examined, and its behavior for various input scenarios is explored. Subsequently, qualitative and quantitative evaluations of the spectral predictions made by the latent encoder are conducted, utilizing metrics outlined in Chapter 2. Lastly, the multispectral latent encoder is introduced, and experiments are set up to compare multispectral latent encoders using ASTER and Landsat datasets.

4.1 Prior work

4.1.1 Sensor Options for Spectral Mapping

For spectral mapping in orbital scale, imaging spectrometers deployed as orbital instruments are used. Remote sensing is the primary use-case for these data which includes mapping agricultural fields (Ramakrishnan and Bharti, 2015; Whiting et al., 2006) or studying mineralogy (Thompson et al., 2018b; Swayze et al., 2014).

For rover scale operations, spectrometers like SuperCam (Royer et al., 2022) on Perseverance rover (Bell et al., 2021) or ASD FieldSpec Pro (Młynarczyk, 2020) on prototype rover Zoë (Vijayarangan et al., 2018) gives hyperspectral measurements. Thompson et al. (2015); Candela et al. (2018) show spectral mapping using *in situ* data from these spectrometers.

Recently, compact hyperspectral (XIMEA; TELOPS) and multispectral (SpectralDevices; SILIOS) imaging devices have become available. Capable of sensing NIR and SWIR (which are diagnostic for mineralogy) wavelengths, these devices are increasingly deployed on aerial vehicles for spectral mapping purposes (Heincke et al., 2019; Kedia et al., 2021), expanding the accessibility of spectral mapping across various applications and platforms.

4.1.2 Pan-Sharpening

The process of correlating RGB data with spectra for each pixel in a map closely resembles pan-sharpening techniques in existing literature. Pan-sharpening uses a panchromatic image which has higher spatial resolution to increase the resolution of low spatial, narrow spectral band images (Rebillard and Nguyen, 1983; Zhou et al., 1998; Xu et al., 2014). In pan-sharpening, the multispectral image is first resampled to match the higher spatial resolution of the panchromatic image. The two images are then merged using a fusion algorithm that blends the high spectral information from the multispectral image with the high spatial resolution of the panchromatic image. The result is a new image with enhanced spatial and spectral detail, which can be used for analysis and interpretation. Various methods exist for fusing these images.

- *Component Substitution (CS):* splits the images into spatial and spectral components using intensity-hue-saturation (Tu et al., 2004; Choi, 2006) or principal component analysis (Ghassemian, 2016). The spatial component of the panchromatic image is substituted to the multispectral image retaining the spectral component. These algorithms are generally simple and fast and demand minimal computational resources (Dadrass Javan et al., 2021).

- *Multi-Resolution Analysis (MRA):* decomposes the panchromatic and the multispectral images into different resolution scales using techniques like Laplacian pyramids, wavelets, contourlets or curvelet transforms (Javan et al., 2018). The coarse resolution components of the multispectral image are enhanced by replacing or merging them with the corresponding higher-resolution components from the panchromatic image, thereby improving spatial details while maintaining the spectral information (Vivone et al., 2021).

- *Variational Optimization (VO) methods:* frame pan-sharpening as a functional with fidelity, smoothness, regularization and other priors as function inputs to optimize. For instance, Bayesian data fusion (BDF) (Fasbender et al., 2008), is a VO approach that utilizes a probabilistic model to statistically deduce high-resolution multispectral data. This method estimates the conditional probability distribution of multispectral pixels based on observed panchromatic pixels, thereby merging the spatial resolution of the panchromatic image with the spectral details of the multispectral image. Other VO methods incorporate techniques like sparse representation (Zhu and Bamler, 2013; Yang et al., 2018), regularization (Aly and Sharma, 2014), or model optimization to enhance pan-sharpening results.

- *Deep learning methods:* try to extend image based convolutional neural networks architectures like super-resolution (Scarpa et al., 2018), de-noising auto-encoder (Azarang et al., 2019), deep residual network (Wei and Yuan, 2017) and generative adversarial network (Liu et al., 2018) for pan-sharpening. Many of these methods

are single image techniques, using just the multispectral image instead of exploiting the spatial resolution of the panchromatic image leading to subpar results in the spatial dimensions. To mitigate this, some approaches infuse ideas from CS and MRA into these deep learning models and achieve enhanced pan-sharpening performance (He et al., 2019a).

Pan-sharpening typically involves enhancing multispectral images using panchromatic images that cover the same spectral bands (Padwick et al., 2010). Hyperspectral pan-sharpening involves improving both the spatial and spectral resolution simultaneously as the hyperspectral image often extend beyond the spectral range of the panchromatic image. The mismatch in spectral range, coupled with the requirement to detail numerous continuous narrow bands, renders hyperspectral pan-sharpening susceptible to spectral distortion and spatial blurring. Loncan et al. (2015) provide a comprehensive review of various hyperspectral pan-sharpening methods, detailing the advancements and adaptations needed to handle the extended spectral characteristics. Recent advancement in hyperspectral pan-sharpening involve employing deep learning priors (Xie et al., 2020; Bandara et al., 2022) and specialized predictive structures (He et al., 2019b) focussing the spectral layers, yielding improved results.

The pan-sharpening process typically require having both complete panchromatic and hyperspectral images in advance. This is feasible in remote sensing applications where comprehensive measurements of an entire region are collected. However, in rover-based scenarios where only limited hyperspectral data is available, traditional pan-sharpening techniques are not suitable.

4.1.3 Predicting Spectra using Camera Images

There are several works in literature that estimate hyperspectral data from camera images. Zhang et al. (2022) recently published a survey that categorizes spectral reconstruction methods into two main groups: data-driven methods and prior-based methods.

Data-driven approaches typically utilize deep learning models and are further classified based on network architecture. Linear CNNs (Xiong et al., 2017; Han et al., 2018) employ simple deep-learning networks without any residual feedback. Since these methods are from computer vision domains, they pay more attention towards Euclidean loss instead of focussing on the loss in the spectral domain. UNet models (Stiebel et al., 2018; Yan et al., 2018; Fubara et al., 2020) employ an encoder/decoder network which enables it to extract features at different scales from the RGB image. Most of these methods merely focus on the spatial information and ignore spectral features or treat them as another spatial dimension. Hence the spectral reconstruction accuracy is not guaranteed. Attention networks (Li et al., 2020; Zhao et al., 2020; Peng et al., 2020) try to address this deficiency by adding attention mechanisms that allow the flexibility to focus

on important features for spectral reconstruction. While data-driven methods generally outperform prior-based methods, their effectiveness often relies on large amounts of data, particularly labeled data. These methods typically train in a supervised setting where the label explicitly informs the learning algorithm what the output should be for a given input. Despite the availability of numerous spectral datasets in the public domain, such as the data from the Mars CRISM instrument (Seelos et al., 2023) or the Landsat instrument on Earth (Irons et al., 2012), only a limited number of datasets are labeled.

Methods that use a prior, operate under the assumption that a spectrum is a composition of basis spectra, referred to as end-members, and aim to identify their proportions, known as abundances. Statistical analyses have revealed that hyperspectral images exhibit sparsity both in space and spectrum. The dictionary learning method (Thompson et al., 2018a; Arad and Ben-Shahar, 2016) leverages this property by learning a dictionary of basis spectra. These methods operate at the pixel level but do not account for spatial similarity. On the other hand, manifold learning methods (Jia et al., 2017) represent hyperspectral signals in a lower-dimensional manifold. While this approach bears similarity to our work, it diverges by employing variational auto-encoders instead of manifolds. Although this presents a plausible alternative, a comprehensive investigation is necessary to ascertain its validity.

We adopt a hybrid approach, harnessing the strengths of both data-driven and prior-based methodologies. Our strategy integrates deep learning models to acquire spectral representations in a self-supervised manner, while employing Gaussian priors configured to function as a regression framework, rather than predicting discrete mixtures.

Next, let us look at what a latent encoder is and how it enables camera images to make spectral predictions in the simple-sensor-spectral model.

4.2 Latent Encoder: Generalize the Feature Space

A latent encoder learns to associate its input to a latent representation similar to those generated by the spectral encoder. A *RGB latent encoder* receives RGB data from the camera as input. Similar to the spectral model introduced in Chapter 3, the latent encoder operates in two phases as shown in Figure 4.1.

In the training phase, first, the latent representation of a spectra is generated using the spectral encoder. Next, RGB values corresponding to that spectral data is extracted. The RGB latent encoder is then trained using both the latent representation of the spectra and its RGB value. The training continues for a fixed number of points (in our case, 120 points) and the RGB latent encoder learns to correlate these two values. During the mapping phase, given only the RGB data, the RGB latent encoder predicts latent values

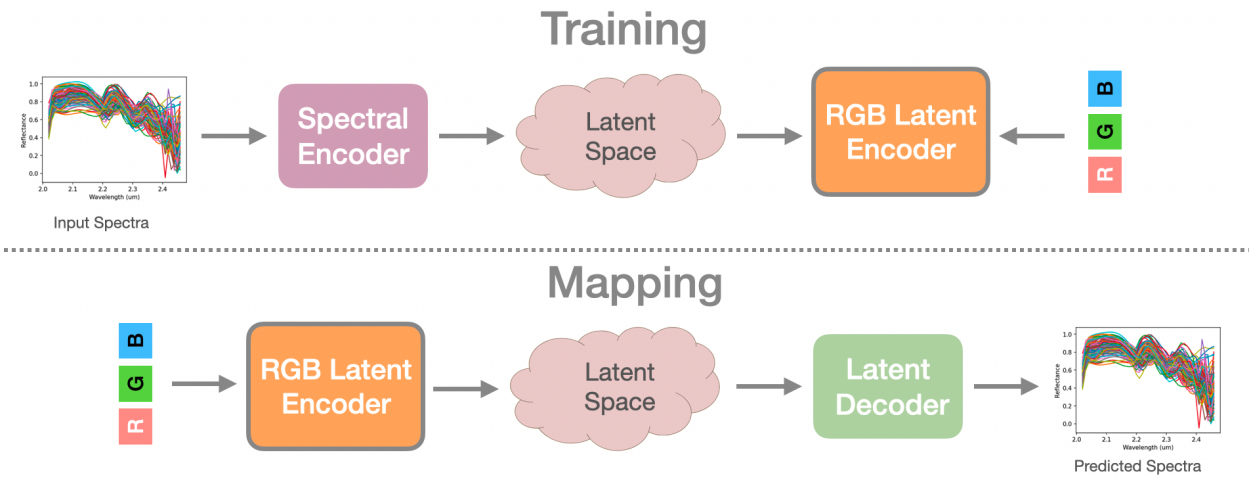


Figure 4.1: Steps to train a RGB latent encoder and employing it for spectral mapping

which are then reconstructed as spectra by the latent decoder. This procedure repeats for every pixel within the mapped area.

The latent encoder is not restricted to RGB input alone, it can operate on any type of vectorized input. In a later section, multispectral latent encoders are introduced which use multispectral data as input along with some experimental results proving the validity of this method.

The latent encoder closely resembles the spectral model introduced in Chapter 3. There are two main differences. First, the RGB latent encoder does not use positional information as the spectral model does. Second, the RGB latent encoder expects RGB data is available for every pixel within the mapped area which is not a requirement for the spectral model.

By collecting hyperspectral signatures for a representative sample of minerals in a local area, they can be associated with their corresponding RGB signatures for accurate identification. However, attempting to replace hyperspectral data with lower-dimensional RGB, for spectral mapping, poses its own challenges. Fundamentally the low dimensional RGB data lacks information present in the hyperspectral data. It could manifest in situations where different minerals having distinct spectral signatures, discernible only through hyperspectral sensing, exhibit similar visual appearance in RGB space. This poses a challenging situation to the RGB latent encoder as it may predict an incorrect spectra. However, within a localized region, the diversity of minerals with conflicting RGB signatures is limited.

4.3 Latent Encoder Model

The latent encoder (see Figure 4.2) comprises of n independent Gaussian Processes (GPs). Similar to the spectral model, discussed in Chapter 3, GPs are chosen to represent the

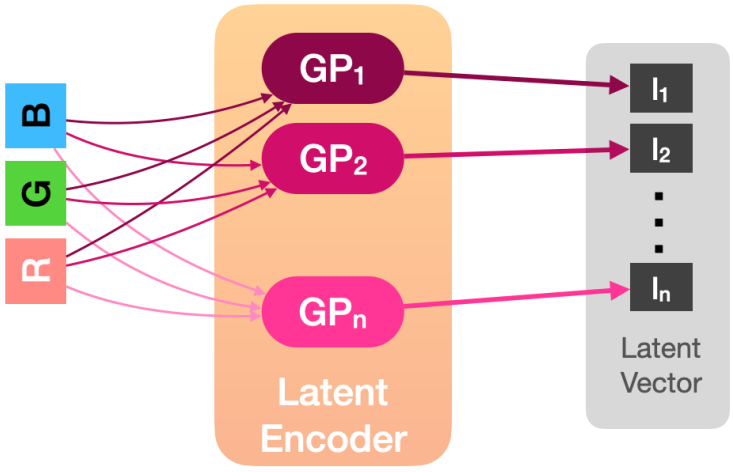


Figure 4.2: Latent encoder with n independent Gaussian processes each producing output to one dimension of the latent vector.

latent encoder because GPs excel in limited training sample scenarios as they leverage the kernel trick to automatically learn rich representations from training input. This implicit high-dimensional feature mapping allows GPs to capture relevant patterns even with fewer training examples (Pryzant). The size of the latent dimension, n dictates the number of GPs in the latent encoder. As noted in Chapter 2 (in Sec. 2.6) a latent dimension of 12 is found as optimal and is being used here. The VAE dimensions are uncorrelated and hence independent. This allows us to use independent GPs to model the latent encoder.

Formally, let $H \in \mathbb{R}^h$ and $C \in \mathbb{R}^3$ denote the hyperspectral and RGB data (from camera images) respectively, where h represents the number of wavelength bands in the hyperspectral signal and $h >> 3$. Let us assume there exists a relationship between the hyperspectral data, H and camera image, C for a given location, meaning

$$f : C \mapsto H$$

Let $L \in \mathbb{R}^n$ denote the latent space of the spectral compressor noted in Chapter 2 (Sec. 2.2.1) and $h > n$. The spectral compressor, learns the encoder function $f_e(.)$ and decoder function $f_d(.)$, such that given a hyperspectral signal H, the spectral encoder predicts the latent representation L, as

$$L = f_e(H)$$

and given a latent representation, L the spectral decoder predicts the original hyperspectral signal, H, as

$$H = f_d(L)$$

During the *training phase*, the latent encoder learns the association between the RGB pixels (C_{RGB}) and the latent representation L of the corresponding hyperspectral data, H, obtained from the spectral encoder. The latent encoder learns a proxy function f_l that maps RGB data to the same latent space given by the spectral encoder,

$$L = f_l(C_{RGB})$$

With just a few training samples, the GPs in the latent encoder learn to map the RGB data to the latent dimension of the spectral signal. During the *mapping phase*, given the RGB pixels (C_{RGB}), its latent representation L is generated using the function f_l, learned during the training phase.

As noted before, this process can be extended to include multispectral input. In this case, the multispectral latent encoder learns the function f_{lm}, such that, it predicts the latent representation given the multispectral data, M, as shown below,

$$L = f_{lm}(M)$$

mapping multispectral data to the latent space.

Next, the steps to train a latent encoder and utilize it to create a spectral map of a dataset will be examined.

4.4 Training and Mapping

The steps to train a RGB latent encoder using Cuprite Hills dataset and generate a spectral map of the region is detailed below (refer Figure 4.1).

- *Training the latent encoder* - m points ($m = 100$) are chosen in random from the Cuprite Hills region. For each point, the AVIRIS-NG spectra along with the RGB values are extracted. The AVIRIS-NG spectra is processed through the spectral encoder to yield the latent values which are then used for training the latent encoder.
- *Mapping* - Cuprite Hills region is uniformly sampled at 30 meter interval and RGB values are extracted. The RGB data is processed through the latent encoder followed by the spectral decoder to form the RGB predicted spectra.

- *Evaluation* - For the uniformly sampled points in the previous step, AVIRIS-NG spectra are extracted. The AVIRIS-NG spectra is compressed and reconstructed using the spectral compressor to get SC predicted spectra. The evaluation metrics (cross-correlation, root mean squared error, mean spectral angle error and mean latent error) are calculated using original AVIRIS-NG spectra and predictions from SC and the RGB latent encoder.

Next, let us validate the latent encoder by studying its response for different types of input.

4.5 Latent Response Map

A latent response map is a visualization of the latent space produced by a latent encoder. The latent response map, $\left(L_{map} \in R^{p \times q}\right)$ is the mean ($l = \frac{\sum_{i=1}^{n} L_i}{n}$) of latent values ($L \in R^n$) arranged as a $p \times q$ pixel grid.

To perform a sanity check on the latent encoder, two separate latent encoders are trained using the steps mentioned in Section 4.4. However, instead of using RGB values from the site as input, the first latent encoder uses random values as input, while the second is trained using a constant value, specifically zero, as its input.

Next, the mapping step is followed, but the latent response from the latent encoder is used to create the latent response map (instead of processing through the spectral decoder). The latent encoder trained on random input is expected to produce random predictions in the latent space, as evidenced by random values on the latent response map. Conversely, the encoder trained on constant input should consistently output the same constant value across all points on the latent response map. This behavior is also confirmed as shown in Figure 4.3.

When the latent encoder is trained with RGB points, the latent response map captures the structural features of the dataset as seen in Figure 4.4.

It's important to highlight that the latent encoder operates without any positional input, both during training and mapping phases. Yet, its ability to generate a latent response map that accurately captures the structural features of the training dataset underscores its effectiveness and affirms its practical value.

Next, let us assess the spectral prediction capabilities of the latent encoder. To obtain a qualitative understanding, let us first examine the responses of the latent encoder for single pixels.

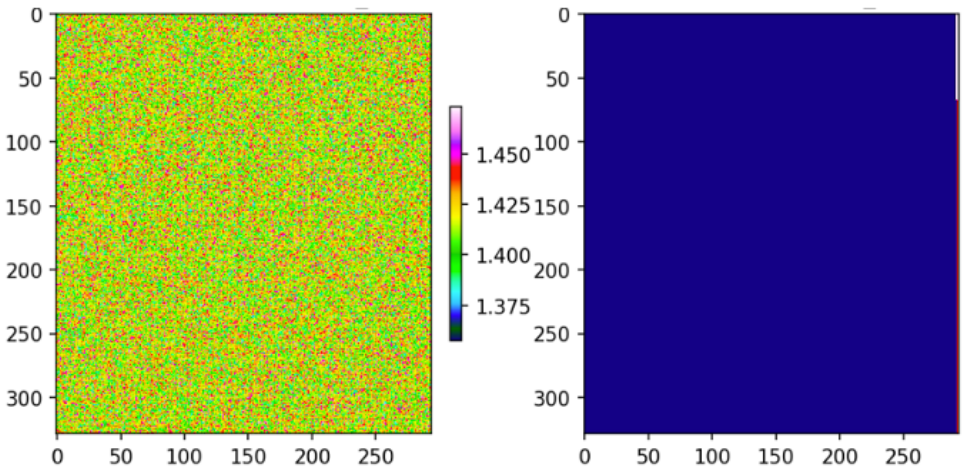


Figure 4.3: The response from the latent encoder trained on random (left) and constant (right) inputs. As the latent encoder does not encode positional information, random and constant input results in random and constant output, respectively

4.6 Evaluating Single Pixels

First, the spectral predictions of the RGB latent encoder are compared with the original spectra, along with the predictions from the spectral compressor for three distinct minerals (refer to Figure 4.5). Both RGB latent encoder and spectral compressor generally mirror the shape of the AVIRIS signal. Differences in shape between spectral compressor and AVIRIS signals indicate information loss during the spectral compression process. Similarly, disparities between the shapes from RGB latent encoder and spectral compressor highlight errors in the latent encoder's emulation of the spectral encoder, emphasizing the drawbacks of using the RGB signal instead of the original spectra.

4.7 Evaluating Spectral Error Maps

A spectral error map $\left(S_{map} \in R^{p \times q}\right)$ is similar to the latent response map where the value of each pixel is calculated by evaluating the spectral angle error, $(SAM(.))$ between the original spectra, (H^t) and the predicted spectra, (H^p) denoted by $(SAM(H^p, H^t))$. The spectral error maps for spectra from spectral compressor and the RGB latent encoder is shown in Figure 4.6. The errors in spectral error map for spectral compressor are significantly lower than those observed for the latent encoder. This is expected because the latent encoder uses the values from only 3 bands (red, blue and green) and trained on a handful of hyperspectral data points compared to the spectral compressor which is pre-trained on a larger volume of data. The spectral error map also highlights the regions where using RGB data leads to larger error.

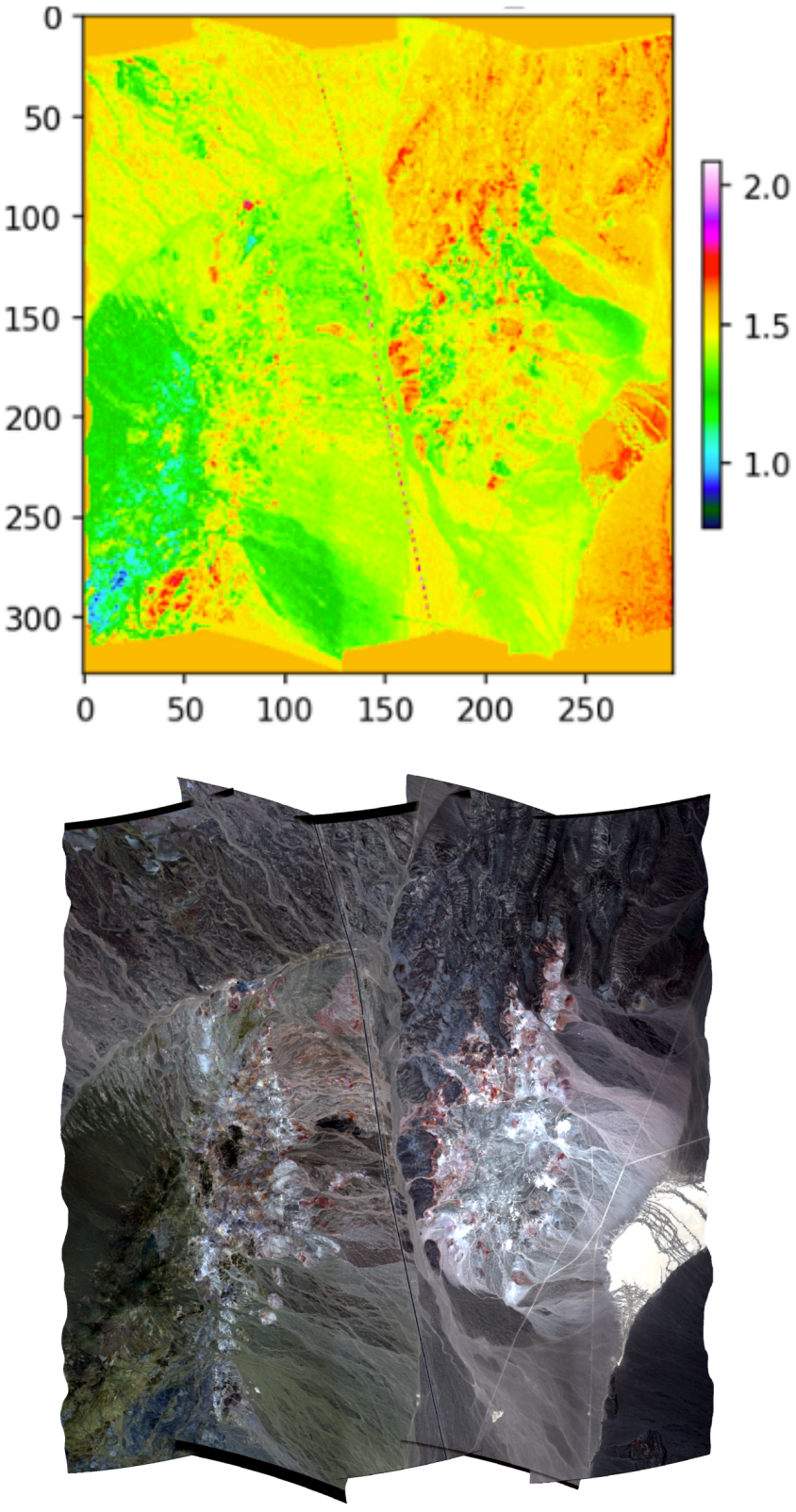


Figure 4.4: Responses from a RGB latent encoder (top) trained on RGB data from Cuprite Hills dataset. The Cuprite Hills RGB map is shown (bottom) for reference. Despite lacking positional information, the latent encoder's response intriguingly captures the structure in the environment

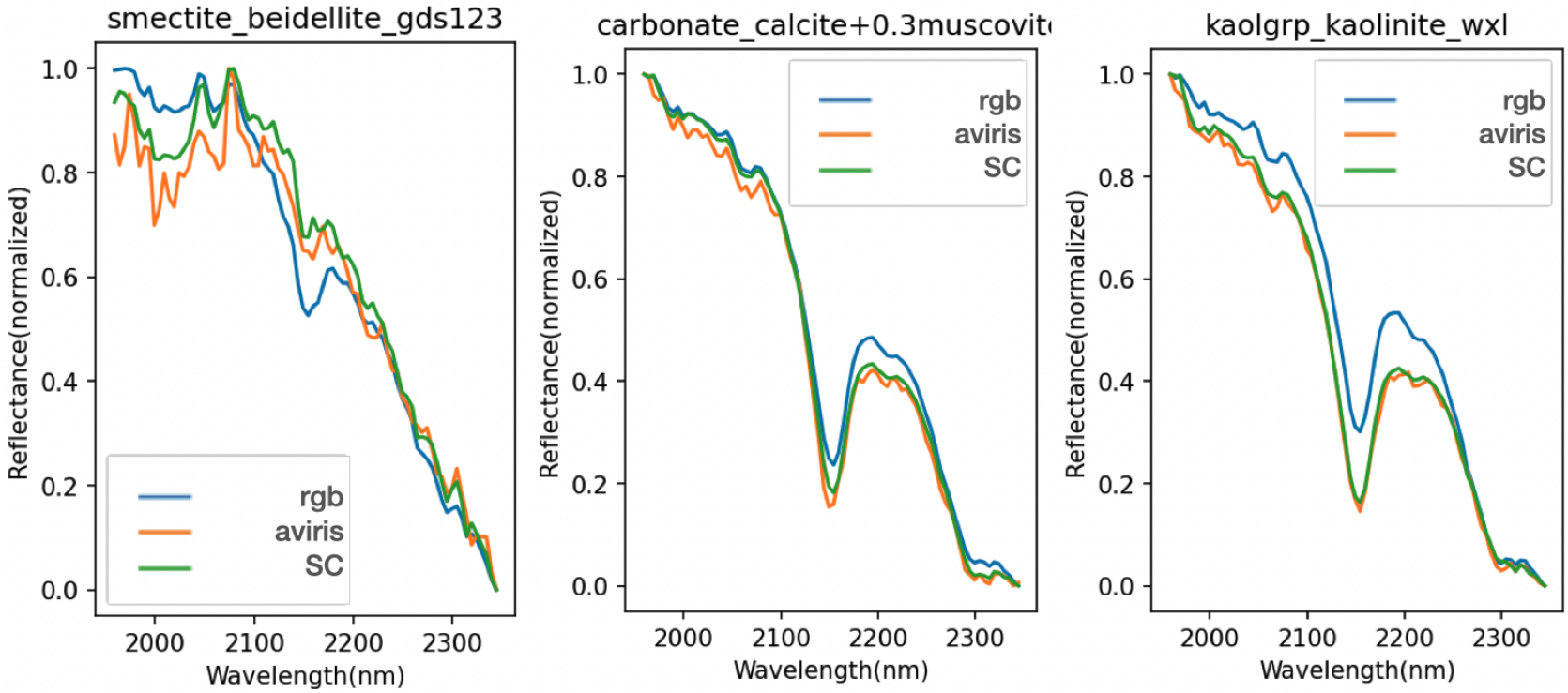


Figure 4.5: Qualitative results of the spectral predictions for three abundant minerals with distinct spectral signatures. The horizontal axis shows the wavelengths in nanometers and the vertical axis shows the reflectance normalized to 0 and 1. The aviris *graphs show the original spectra from AVIRIS-NG dataset. The* SC *graphs show the prediction after processing the spectra through the spectral compressor.* rgb *graphs show the predictions from the latent encoder which is trained using RGB data.*

Type	CC	RMSE	MSAE	MLE
Spectral compressor	0.437	0.024	0.017	-
RGB latent encoder	0.266	0.221	0.068	1.161
Random latent encoder	0.017	0.226	0.092	1.555

Table 4.1: Quantitative evaluation of the spectral prediction accuracy of the spectral compressor, RGB and random latent encoder

Next let us carry on a quantitative analysis summarizing predictions for a group of pixels.

4.8 Quantitative Evaluation

For a quantitative assessment, an experiment is setup following the steps detailed in Section 4.4. For statistical significance, the steps are repeated 100 times, each time selecting a different set of training points, and the cumulative results are presented in Table 4.1 along with the evaluation metrics discussed in Chapter 2. Cross-correlation (CC) is unit-less and ranges between 0 and 1. Root Mean Squared Error (RMSE) measures the error in the spectral reflectances and is unit-less. Mean Spectral Angle Error (MSAE) measures the angle between the spectral vectors and is expressed in radians. Mean Latent Error (MLE) measures the error in the latent space and is unit-less. In the ideal case when the predicted spectra is identical to the original spectra, CC will be 1, RMSE, MSAE and MLE will be 0.

The errors from the spectral compressor stem from the data loss that occurs during compression and reconstruction steps. The error from the latent encoder is higher than the error from the spectral compressor due to the lack of information in the RGB channels. The errors from the latent encoders will at best be equal to those from the spectral compressor, as the latent encoders attempt to mimic the spectral compressor in generating the latent values.

It is interesting to note that random latent encoder, despite being trained on random input, seems to perform surprisingly well. This is because the spectral decoder has learned the mean of the spectra for the Cuprite Hills region during its training. This leads to spectral predictions which looks like a spectral signal, but does not correspond to real spectra.

The fact that the RGB latent encoder performs better compared to the random latent encoder demonstrates that the latent encoder is adding meaningful information for spectral predictions.

In the next section, let us introduce multispectral latent encoder and study two variations trained using Landsat and ASTER datasets.

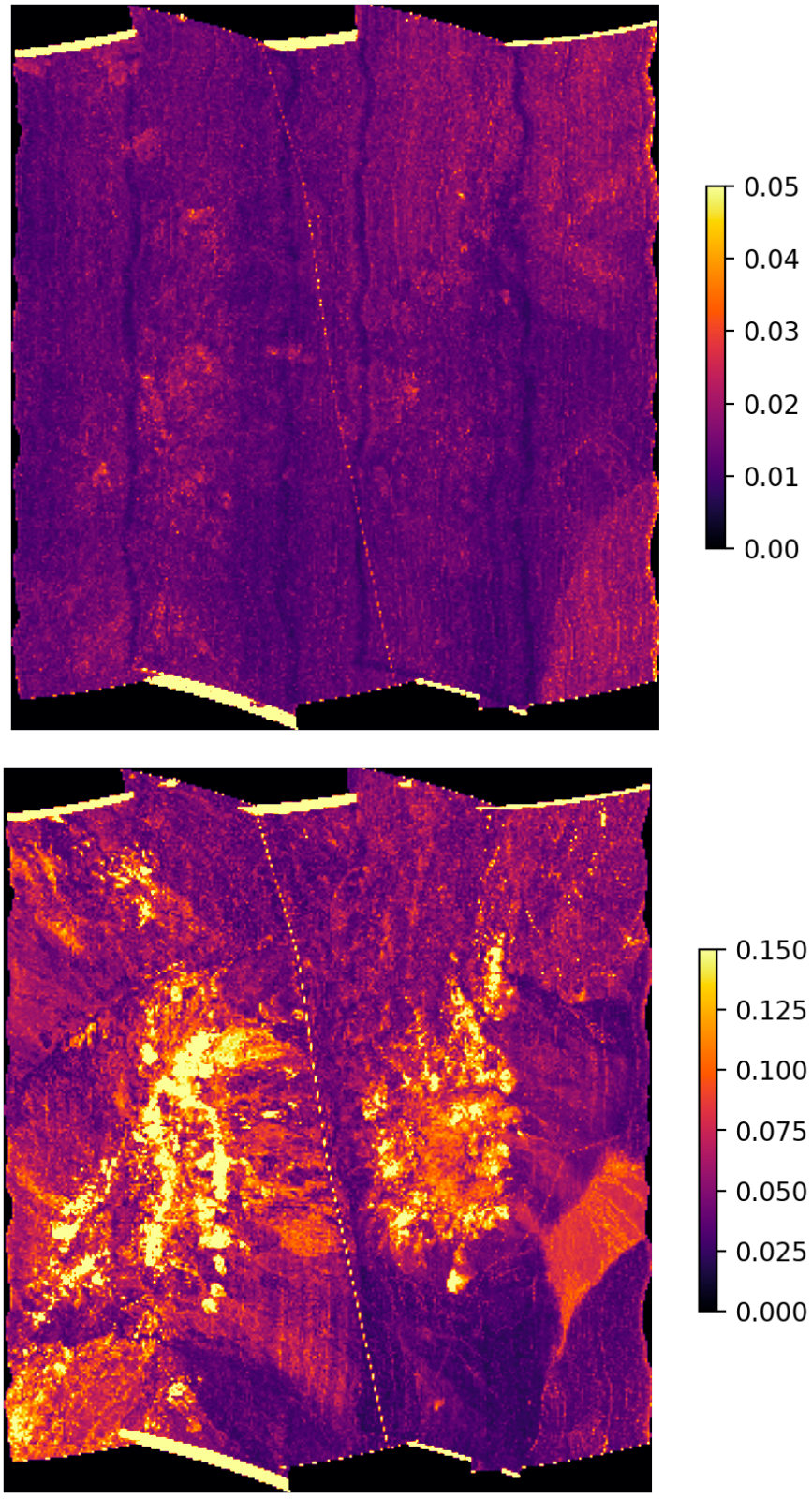


Figure 4.6: Spectral angle error between the original spectra and predicted spectra using SC (top) and RGB latent encoder (bottom). Notice the different scales, the SC error is much lower than error from RGB latent encoder. Compared to the more uniform error distribution of SC's spectral error map, RGB latent encoder's spectral error map shows error concentrated on certain areas. This is due to the same color mapping to different minerals in those regions

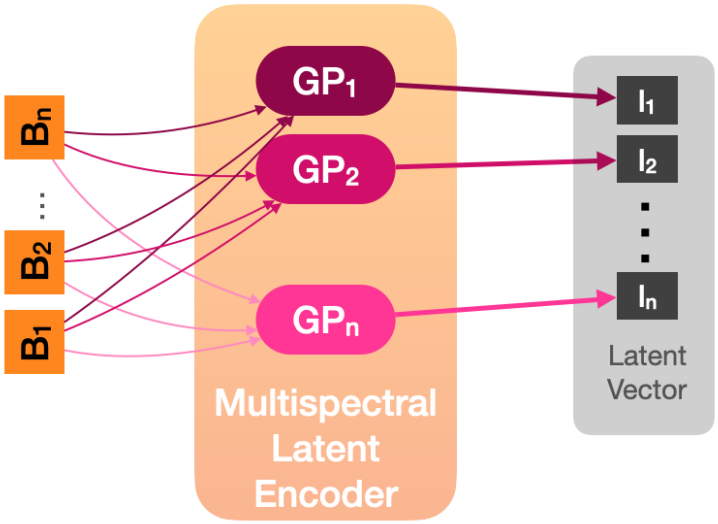


Figure 4.7: Multispectral latent encoder with n GPs as the latent dimensions

Latent Encoder	CC	RMSE	MSAE	MLE
landsat	0.405	0.053	0.060	0.220
aster	0.259	0.061	0.069	0.251
rgb	0.194	0.064	0.073	0.264

Table 4.2: Quantitative evaluation of the spectral prediction accuracy of the multispectral and rgb latent encoders

4.9 Multispectral Latent Encoder

The multispectral latent encoder extends the concept of the RGB latent encoder. Unlike the RGB latent encoder, the multispectral-LE accommodates as many inputs as the size of the multispectral bands. For instance, in the case of ASTER, it comprises 9 inputs, while in the case of Landsat, it consists of 7 inputs. Each input corresponds to a GP in the latent encoder, as illustrated in Figure 4.7.

To validate multispectral latent encoder, an experiment is setup involving the two multispectral LEs: one utilizing ASTER data and the other using Landsat data. The latent encoders are tested on the dronemaps from the three field sites and the experiment is repeated 100 times for statistical significance.

The cumulative results show that landsat-LE performs slightly better than aster-LE (refer Table 4.2). Despite ASTER having two extra bands compared to Landsat, its lower precision (8-bits) (refer to Table 4.3) contributes to the dip in performance. Both landsat-LE and aster-LE outperform RGB-LE, owing to the inclusion of SWIR wavelengths, which

Type	Channels	Spatial Resolution	Precision
ASTER	9	15-30 meters/pixel	8-bits
Landsat	7	30 meters/pixel	14-bits

Table 4.3: Differences between Landsat and ASTER datasets

aids the latent encoder in distinguishing differences in the spectral signal's latent representation. This underscores the importance of adding additional diagnostic bands to enhance spectral reconstruction accuracy.

The mean spectral angle error for the three sites is generated along with the individual breakdown for each site as illustrated in Figure 4.8. Notably, RGB-LE shows the worst performance in Cuprite Hills, likely due to the region's higher mineralogical diversity, which poses challenges for RGB in producing accurate results. Upon examining the individual graphs, it is observed that landsat-LE performs better than aster-LE in Yellow Cat and Hopi Buttes. However, for Cuprite Hills, aster-LE performs slightly better in some cases. This could be attributed to the fact that the bands covered by ASTER may have a better response to the particular minerals found in those regions.

4.10 Summary

In this chapter, the latent encoders were introduced as a method to conduct spectral mapping using RGB or other low dimensional data. The latent encoder's response to different types of input were explored, and its spectral predictions compared against the spectra produced by the spectral compressor. Additionally, multispectral latent encoders were introduced, utilizing data from ASTER and Landsat instruments. Finally experiments were conducted using the dronemap dataset which revealed that incorporating additional spectral bands enhances spectral reconstruction accuracy.

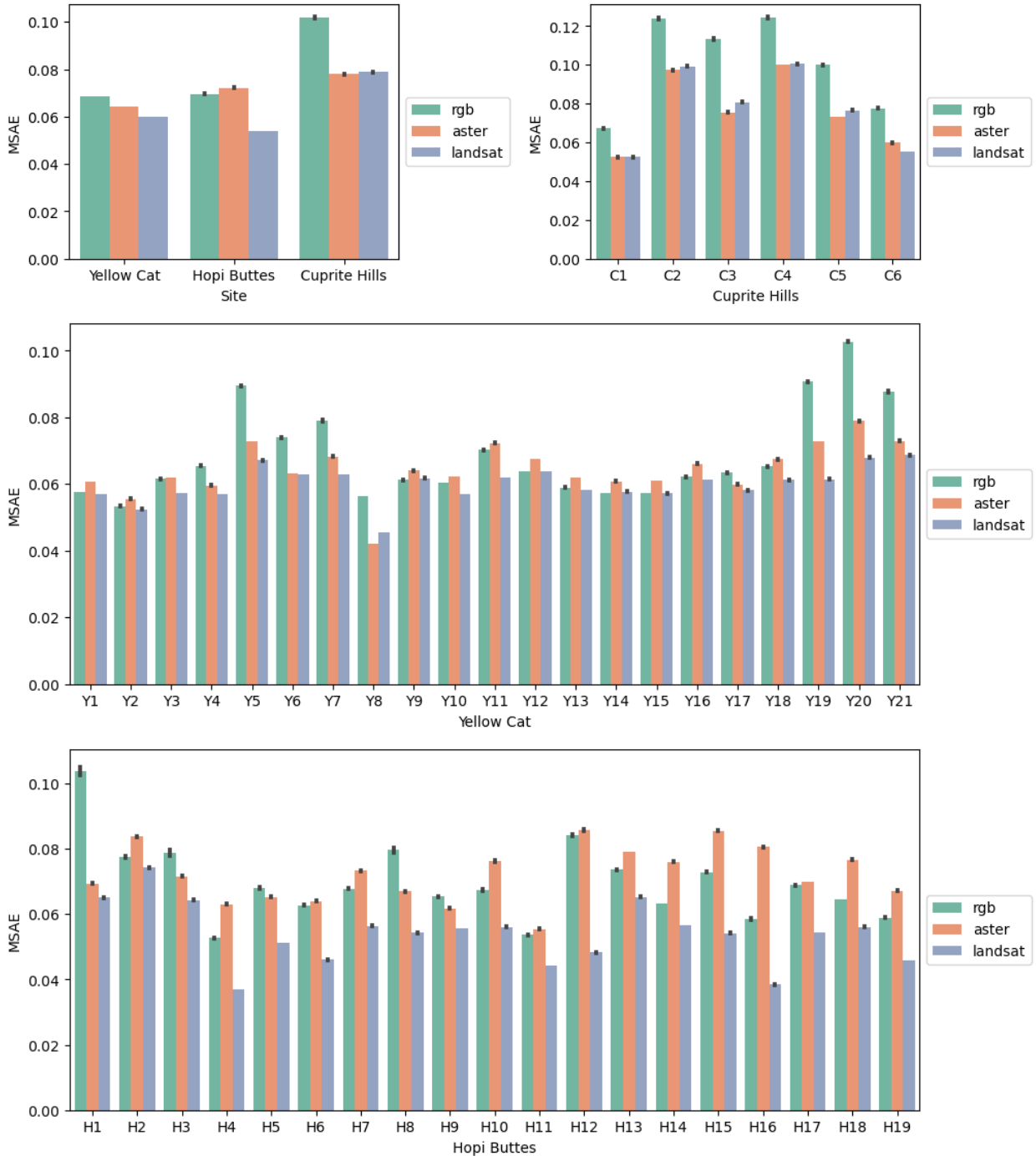


Figure 4.8: Results of the experiment testing multispectral latent encoders on dronemaps from the field sites. In these graphs, shorter bars indicate better performance. Overall landsat-LE performs better than ASTER-LE. RGB-LE performs the worst as it lacks the spectral information necessary to make better predictions

Chapter 5

Latent Encoders with Textures

Latent encoders introduced in the previous chapter treated RGB as a single pixel with three channels corresponding to red, green and blue wavelength bands. However, RGB data typically possesses higher spatial resolution which can be exploited by extracting complimentary information like textures. In this chapter, a variety of techniques to extract texture from RGB data are explored. Next, texture latent encoder is introduced which incorporates the information from textures into the simple sensor spectral model. Following this, the performance of different texture techniques are assessed through experiments, measuring their impact on spectral reconstruction, while identifying the most effective texture.

5.1 Prior Work

Processing texture from images has been studied for decades Coggins, James Michael (1982). Different methods to extract and analyze texture from images can be broadly classified into structural, statistical, transform-based, model-based or learning-based approaches (Humeau-Heurtier, 2019; Fekri-Ershad, 2019).

- *Structural approaches* analyze images by decomposing them into primitives and describe the relationships between these elements to capture texture and structural patterns. Most key-point based feature detectors like Scale-Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF) and Oriented FAST and Rotated BRIEF (ORB) (Lowe, 2004; Tyagi, 2020; Rublee et al., 2011), in computer vision literature fall under this category. These methods detect key points or features in the image and describe them in a way that is invariant to scaling, rotation, and illumination changes, making them highly effective symbolic description tools for images.
- *Statistical approaches* analyze texture by examining the spatial relationships and distributions of pixel intensities within an image. Local Binary Pattern (LBP) (Nanni

et al., 2010) focuses on comparing each pixel with its neighborhood to generate a binary code that captures local texture patterns, while Grey-Level Co-occurence Matrix (GLCM) (Nouri et al., 2018; Barburiceanu et al., 2021) assesses how often different combinations of pixel brightness values (gray levels) occur in an image, providing a measure of textural contrast and uniformity. In contrast, Histogram of Oriented Gradients (HOG) (Routray et al., 2017) computes the occurrences of gradient orientation in localized portions of an image, capturing edge directions and intensity gradients that are indicative of texture.

- *Transform-based methods* analyze the textural features by transforming them into a different domain, such as the frequency or wavelet domain, where the characteristics of texture can be more easily analyzed and described. While Gabor filters target specific frequency components, Wavelet transforms operate at different scales. Shearlet transforms and Contourlet based methods offer better handling of direction using shearing operations (Manjunath and Ma, 1996; Kamarainen et al., 2006; Vácha and Haindl, 2024).

- *Model-based methods* use the statistical dependencies and structural arrangements of pixels, allowing for an analytical description of texture. Fractal based methods model complex patterns that appear similar at various scales (Xu et al., 1993; Costa et al., 2012) and random fields describe textures through probabilistic distributions of pixel values, capturing the inherent randomness and spatial relationships within the texture (Cross and Jain, 1983).

- *Learning based methods* typically incorporate textural feature like GLCM or LBP into existing learning models (Pandey and Jha, 2020; Kalra, 2023) or modify existing deep learning architectures to focus on texture features like using orderless pooling of intermediate layers to exclude the overall shape analysis (Andrearczyk, 2017).

While each class of algorithms possesses its strengths and weaknesses, our goal is to assess whether incorporating texture - thus leveraging the higher spatial resolution of RGB images - can enhance spectral prediction accuracies and partially compensate for the lack of spectral information. To achieve this goal, a variety of texture-based methods from different approaches are integrated into our system, and their performance is evaluated.

5.2 Features from RGB: Textures

Our investigation includes texture algorithms from statistical, structural, transform-based and model-based approaches as noted in Table 5.1.

Each texture algorithm generates a feature vector that summarizes the high-resolution RGB patch as shown in Figure 5.1. This vector serves as the input for the texture latent

Approach	Descriptor	Feature Vector
Structural	SIFT	128
	ORB	32
Statistical	GLCM	6
	LBP	26
	MB-LBP	1
	HOG filters	512
Transform-based	MOSAIC features	1024
	Gabor filters	64
Model-based	Fractal Dimension	1

Table 5.1: Feature descriptors used to summarize texture information from high resolution RGB patches

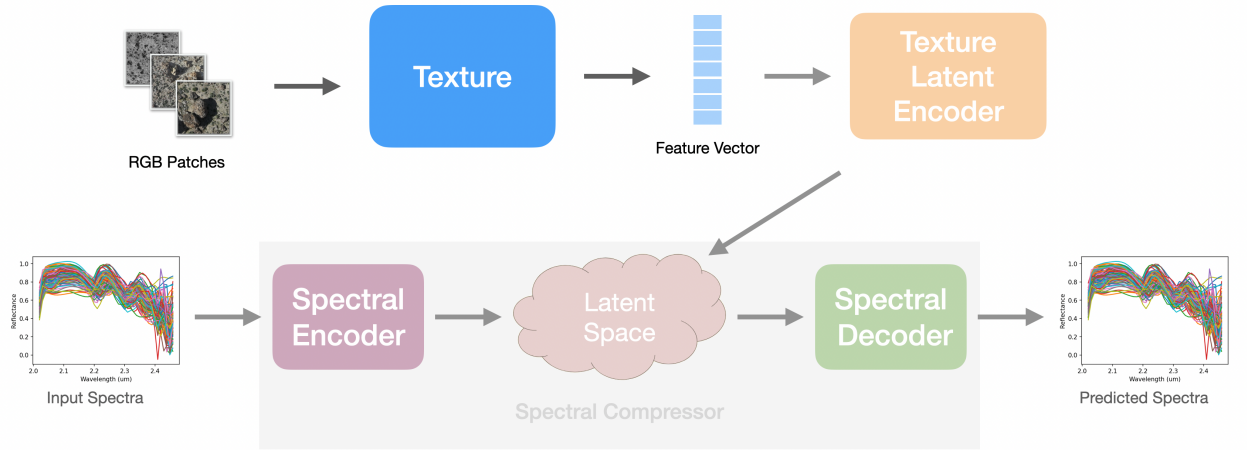


Figure 5.1: RGB patches are summarized into a feature vector, which serves as input for the latent encoder

encoder, aiding in learning correlations between the RGB data and the spectral data's latent representation.

Next, the specifics of each texture algorithm, focusing on how they operate and the processes for generating feature vectors are explored. SIFT and ORB, which are structural methods used for texture extraction, will be examined first.

5.2.1 Scale-Invariant Feature Transform

Scale-Invariant Feature Transform (SIFT) is widely regarded as one of the most robust and versatile feature descriptors in the computer vision domain. The feature extraction process comprises of four major steps detailed below (see Figure 5.2).

- Key-point Detection: This step involves calculating the difference of Gaussian of all image patches at various scales and identifying corner pixels in each of them.

- Key-point Localization: Once key-points are detected, their locations are determined with sub-pixel accuracy using quadratic functions.
- Orientation Assignment: Histograms of gradients are computed to identify the major orientation of each key-point.
- Feature Descriptor Generation: Feature descriptors are generated for each key-point based on the scale and orientation determined in the previous steps. This involves dividing the neighborhood of each key-point into different bins and computing histograms of gradients for each bin. These bins are then concatenated to form a feature descriptor vector of size 128.

Depending on the number of key-points detected, the number of rows in the descriptor vector will vary (refer to Figure 5.2). However, the texture latent encoder expects a fixed-size vector to operate on. This issue is common for all structural texture approaches.

To address this issue, two approaches are considered. The first approach uses a naive method to create the feature vector while the second approach uses a method from the computer vision literature known as Bag-of-Visual-Words (Shekhar and Jawahar, 2012) to address this issue.

The naive approach involves computing the mean of the descriptors along the descriptor dimension. Let $D^k = d_1^k, d_2^k, \ldots, d_{128}^k$ be the k^{th} descriptor. Then the feature vector for the naive approach is calculated as follows:

$$Naive\ feature\ vector = \frac{1}{N}\sum_{k=1}^{N} D^k$$

5.2.1.1 Bag-of-Visual-Words

The Bag of Visual Words (BoVW) technique is akin to describing pictures by focusing on recognizable elements, much like describing a sentence by highlighting its key words. This method breaks down images into smaller parts, identifies specific patterns or features within these parts (such as shapes or colors), and then counts the frequency of each pattern. By doing so, images can be represented in a simplified manner that is easily understandable and comparable for computers. It is a two-step process.

In the first step, a codebook or dictionary is built (refer to Figure 5.3). Feature descriptors are extracted from all the images, followed by K-means clustering operating on the n-dimensional space of the feature descriptors (in the case of SIFT, n is 128). The number of cluster centers determine the number of words in the codebook.

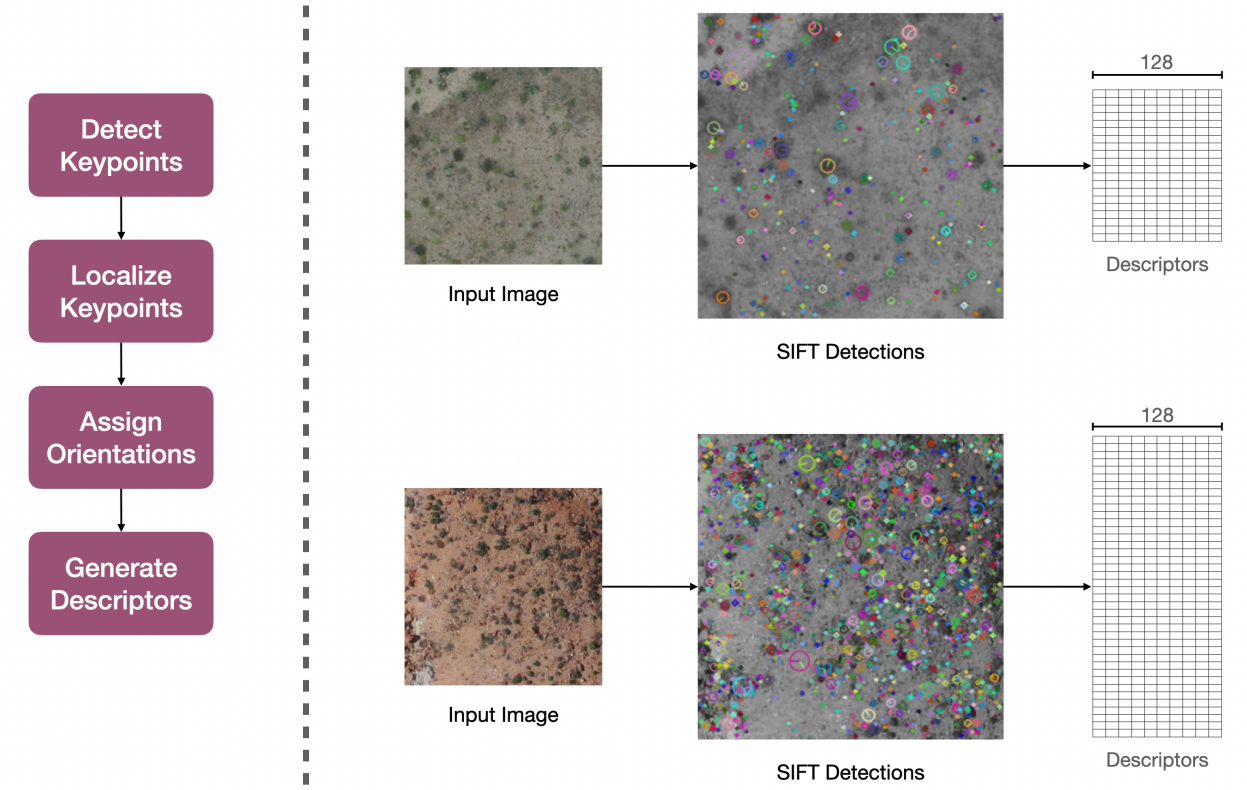


Figure 5.2: (Left) SIFT pipeline (Right) Two image patches with SIFT features detected on them. The circles denote the feature location, the orientation and the size of the circle shows the orientation of the descriptor and the scale at which the descriptor is detected respectively

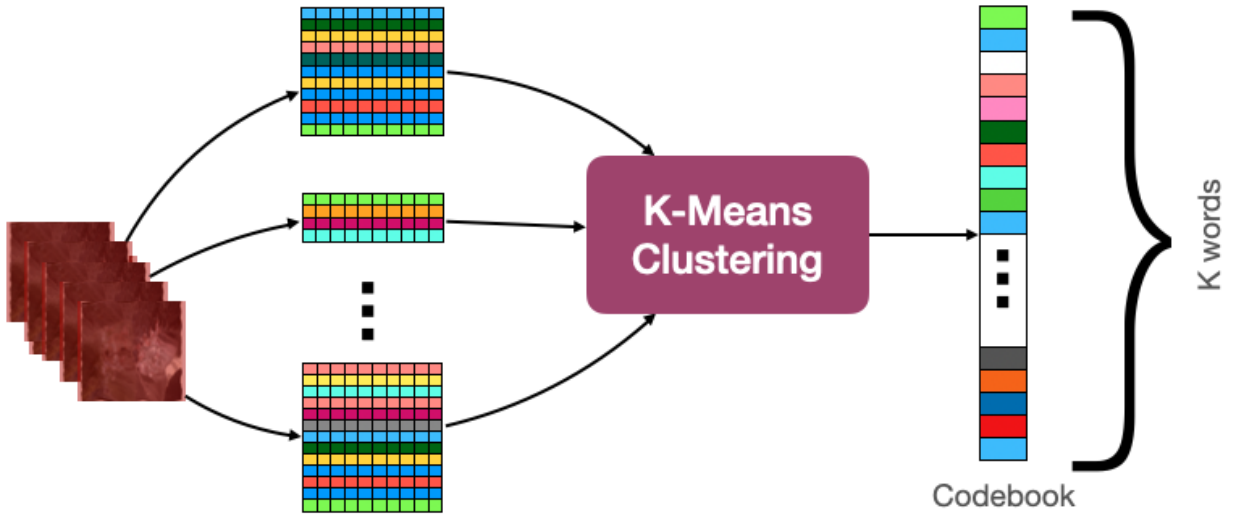


Figure 5.3: To create the codebook for Bag-of-Visual-Words, features descriptors are calculated for a subset of images in the dataset and K-means clustering is performed along the descriptor dimension. The cluster centers form the words for the codebook

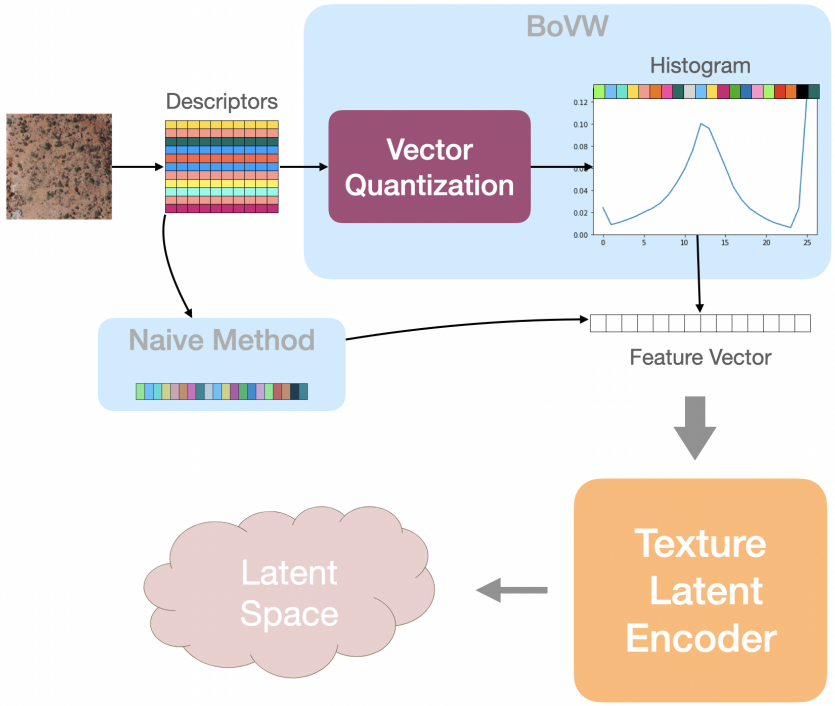


Figure 5.4: Approach to integrate SIFT features into the texture lantent encoder. The texture latent encoder operates on the feature vector which is created using either the BoVW method or the naive method

In the second step, descriptors are extracted from the image of interest (see Figure 5.4). Each descriptor is compared to the codebook to find which word it is closest to in the n-dimensional feature space. A histogram of the detected words is then calculated which serves as the feature vector.

A simple experiment is conduced using one of the Cuprite Hills dronemap. Initially, the codebook is constructed for this dronemap and two latent encoders are trained. The $\text{SIFT}_{\text{BoVW}}$ latent encoder is trained using features from BoVW feature vector and $\text{SIFT}_{\text{Naive}}$ latent encoder is trained using the feature vector from the naive method. The comparison results between these two methods are listed in Table 5.2. While the $\text{SIFT}_{\text{BoVW}}$ method exhibits a slightly better RMSE, both methods demonstrate identical MSAE. Consequently, the naive method is opted for, over the BoVW method, due to its simplicity and reduced overhead, as BoVW necessitates pre-building and maintaining the codebook.

SIFT is scale and rotation invariant and robust to affine transformation making it highly effective for analyzing textures in varied orientations and scales. However its main disadvantages are its comparatively high computational demands and its sensitivity to illumination changes.

Latent Encoder	RMSE	MSAE
$SIFT_{BoVW}$	0.0728	0.092
$SIFT_{Naive}$	0.0736	0.092

Table 5.2: Comparison of reconstruction error between $SIFT_{BoVW}$ and $SIFT_{Naive}$ latent encoders on a dronemap from Cuprite Hills dataset

5.2.2 Oriented FAST and Rotated BRIEF

Oriented FAST and Rotated BRIEF (ORB) is a robust feature descriptor designed for speed and efficiency. The steps for extracting ORB descriptor are detailed below:

- Key-point Detection: begins by identifying key-points in the image using the Features from Accelerated Segment Test (FAST) algorithm, which detects corners based on a circle of pixels around the candidate pixel.
- Orientation Assignment: calculates an orientation for each key-point based on intensity-weighted centroid of the path around the key-point to achieve rotation invariance.
- Image Pyramids: achieves scale invariance by detecting key-points at multiple scales using a pyramid of scaled images.
- Descriptor Generation: calculates Binary Robust Independent Elementary Features (BRIEF) descriptor using the orientation calculated earlier. BRIEF extracts a small patch around the key-point and performs binary intensity test creating a binary string descriptor which is rotation and scale invariant. ORB descriptor produces a feature vector of size 32.

While ORB is recognized for its efficiency, using fewer computational resources and operating on binary strings instead of float vectors like SIFT, SIFT still maintains an advantage in terms of rotation and scale invariance in certain scenarios.

Following this, the discussion will shift to exploring various statistical texture methods.

5.2.3 Gray Level Co-occurence Matrix

Gray Level Co-occurence Matrix (GLCM) is a second order statistical method popular in the remote sensing domain. GLCM method focusses on the frequency of pixel-pair-intensities instead of the spatial relationship of the pixels.

Examples of two image patches with their GLCM matrices are shown in Figure 5.5 . The first patch, being benign, exhibits lower dissimilarity and higher correlation values compared to the second patch, which possesses more texture.

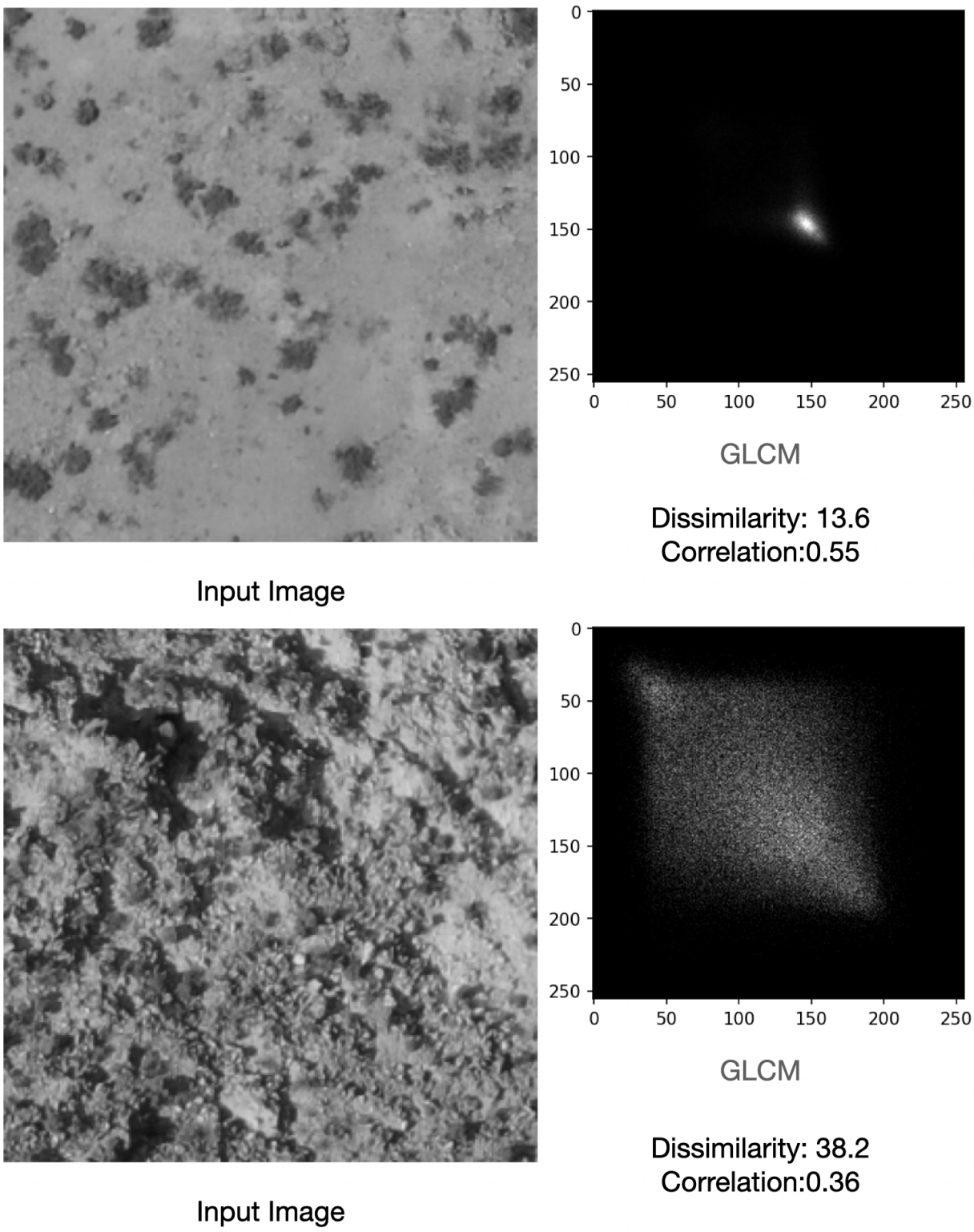


Figure 5.5: Examples of two image patches with their corresponding GLCM matrices. The top patch is mostly benign and hence leads to a smaller dissimilarity value compared to the bottom patch which has a lot of texture in it.

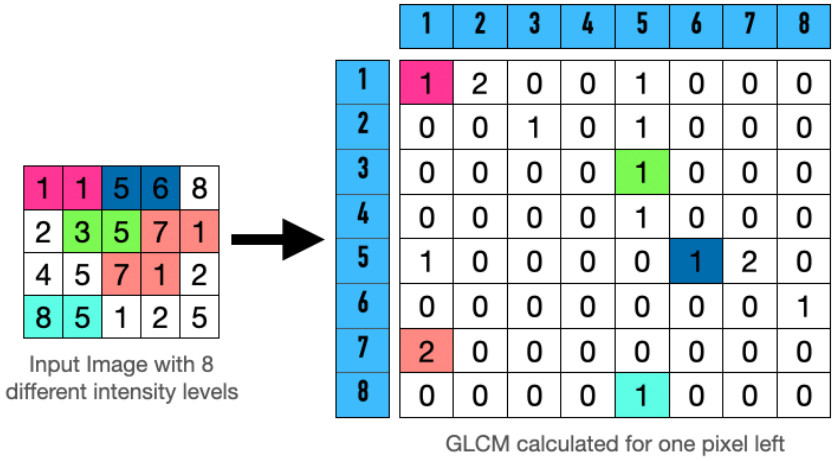


Figure 5.6: Calculation of GLCM is demonstrated with a sample 4x5 image with 8 gray levels

GLCM can be understood using a simple example demonstrating how the GLCM matrix is calculated (refer Figure 5.6). The GLCM is a square matrix whose size equals the number of gray levels in the image patch for which GLCM is calculated. The sample shows a 4x5 image with 8 gray levels, resulting in an 8x8 GLCM matrix.

The process is initiated with the extraction of the pixel value from the first pixel in the sample image, along with its left neighbor (in this case, (1,1)). The image is then searched to count occurrences of the (1,1) pair, and the corresponding cell in the GLCM is updated with the value 1, reflecting that the (1,1) pair occurs only once in the sample image. Next, consider the pair (7,1), finding two such pairs in the image, and updating the (7,1) cell in the GLCM with the value 2. This process is repeated for all such possible pairs. The process can be extended to look for pairs in other directions as well.

From the GLCM matrix, statistical measures such as correlation, dissimilarity, energy, homogeneity, contrast, and angular second moment (ASM) can be calculated and used in the feature vector.

GLCM is highly regarded for providing detailed and interpretable data on the textural properties of images. However, its effectiveness is sensitive to several factors, including the parameters used for the distance and angle at which pixel pairs are examined, the image size and resolution, and the quantization of gray levels. To ensure reliable and meaningful texture analysis results, meticulous pre-processing of images and careful consideration of GLCM parameters are required.

5.2.4 Local Binary Pattern

Compared to GLCM, the local binary pattern (LBP) operates on the local pixel scale. The intuition behind LBP can be understood by examining Figure 5.7. In flat regions, all

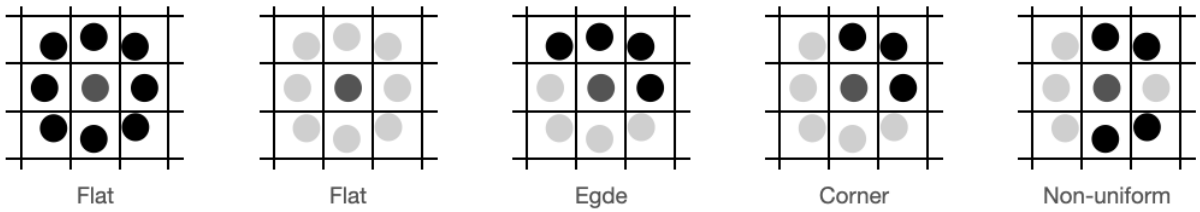


Figure 5.7: The intuition behind LBP calculation lies in how the points are distributed around a pixel which reveals whether the region is flat, an edge or a corner

neighboring pixels tend to have either greater or smaller values than the current pixel. Edges result in pixels with greater values grouped on one side and pixels with smaller values grouped on the opposite side. Corners exhibit a similar pattern to edges, but with an uneven distribution of pixels with values greater or smaller than the current pixel. In all other cases, the pixel distribution tends to be non-uniform.

The LBP calculation for a single pixel is illustrated in Figure 5.8. To begin, a 3x3 patch is extracted around the pixel of interest. The pixel value is then compared to each of its neighbors, and a binary value of 1 or 0 is assigned depending on whether the neighbor's value is greater or lesser, respectively. Starting from a fixed location, such as the left, top, or right, and going in the clockwise direction, the binary code is built. The LBP for that pixel is represented as the decimal equivalent of the binary code. This process is repeated for every pixel in the image. The feature vector is obtained by dividing the LBP image into segments and concatenating the histograms of all the segments together.

The LBP images of two sample patches along with the corresponding histograms is shown in Figure 5.9. LBP yields a feature vector of size 26.

5.2.5 Multi-Block-LBP

The Multi-Block LBP (MB-LBP) (Zhang et al., 2007) extends the original LBP technique by analyzing blocks of pixels rather than individual pixels, as shown in Figure 5.10. This modification allows MB-LBP to be resistant to noise, as minor image variations have less impact. The scalability of feature size is another advantage, with larger blocks capturing broader features and smaller blocks focusing on finer details. The capability to extract information from extensive areas and across various scales significantly boosts the discriminative power of MB-LBP, making it more effective for detailed image analysis.

When given an input image, MB-LBP divides the image into 3x3 grids. The sum of each grid is computed and compared to the sum of the central grid cell. Based on whether

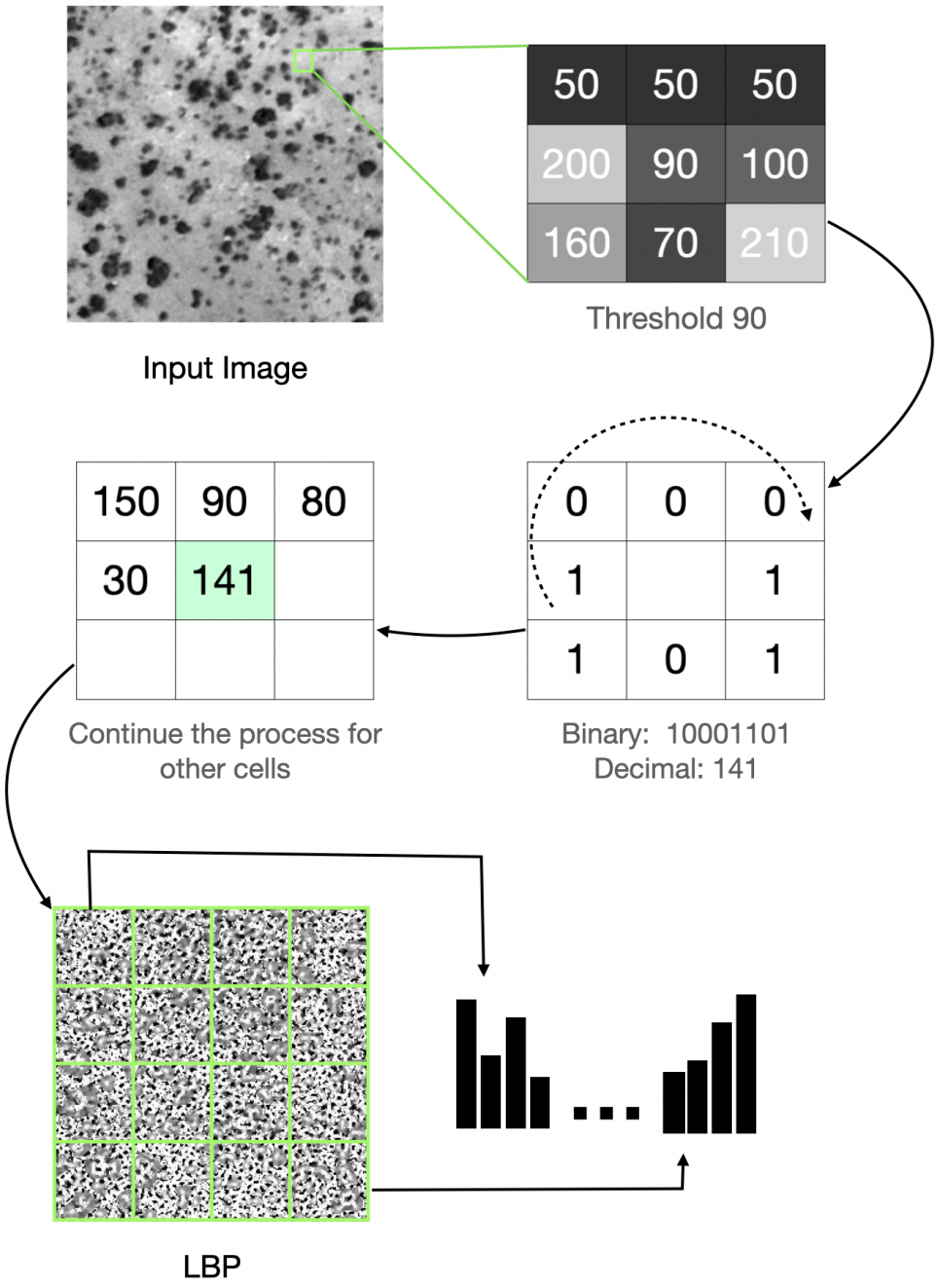


Figure 5.8: To calculate the Local Binary Pattern (LBP) for a single pixel, the value of that pixel is compared with those of its neighbors. These comparisons result in a binary pattern, which is then converted into a decimal number representing the LBP value for that pixel. This process is applied to every pixel in the image to create the LBP image. The LBP image is divided into grids, and the histograms of these grids are concatenated to form the feature vector, capturing the textural information of the entire image

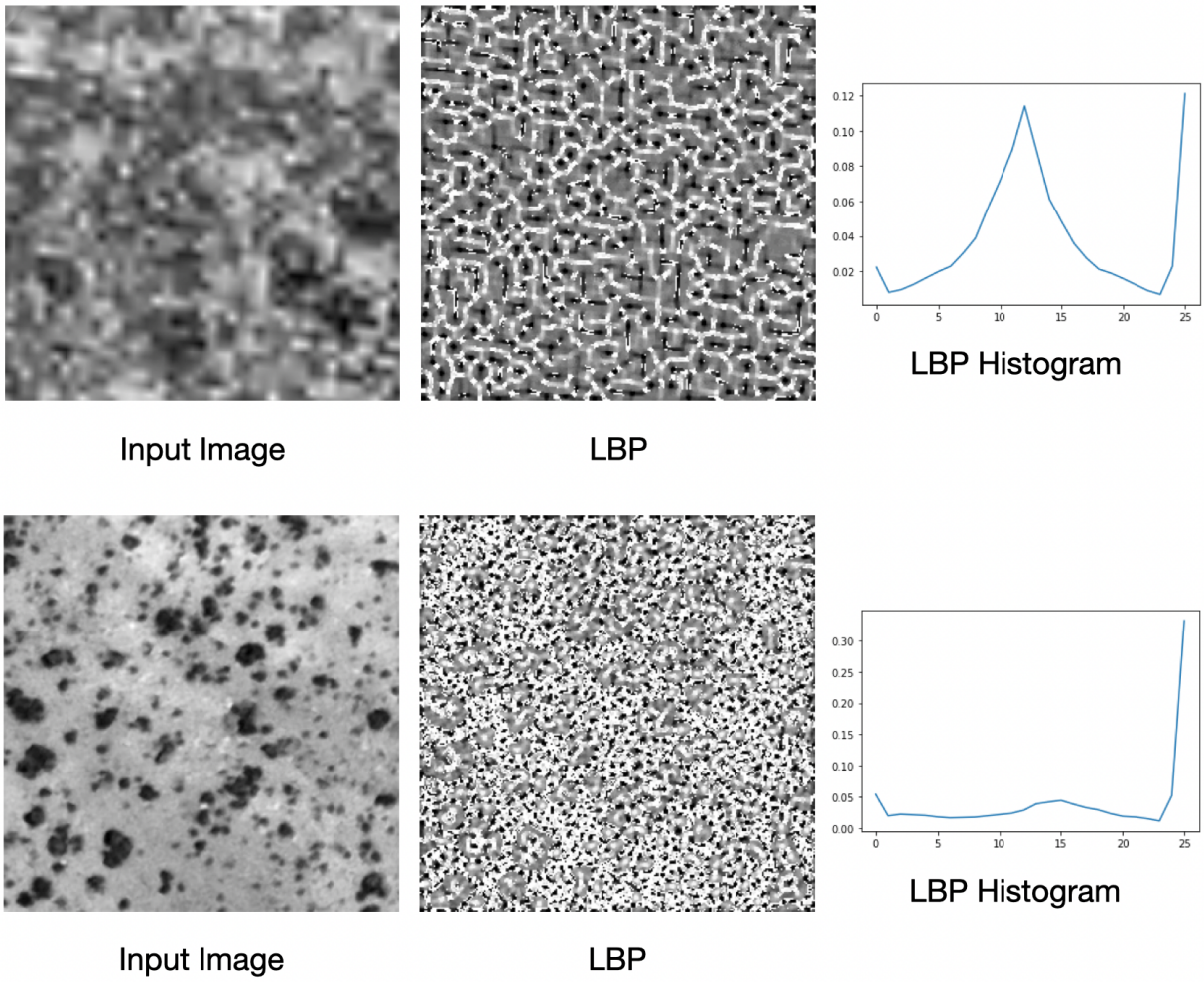


Figure 5.9: Following the process described in Sec. 5.2.4, two sample image patches along with their corresponding LBP values are presented. The LBP image is segmented into grids and a histogram is calculated for each grid cell. These histograms are concatenated to create the LBP histogram as illustrated

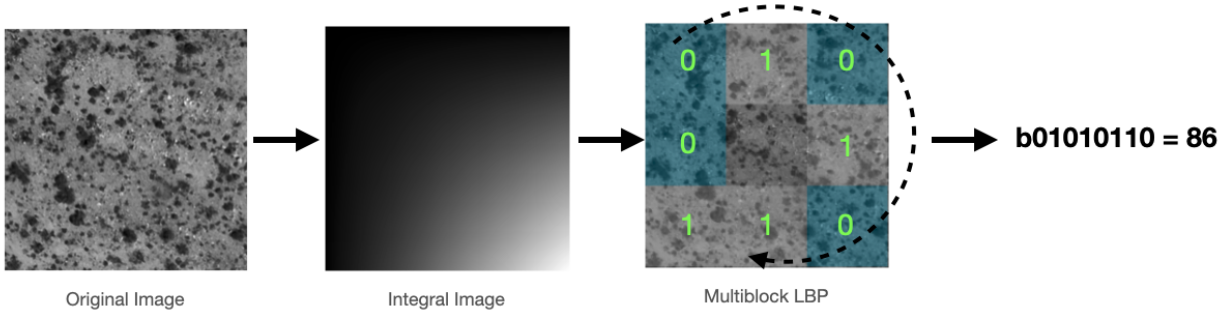


Figure 5.10: Example of MB-LBP calculation on a sample image patch: The image patch is divided into a 3x3 grid. Using an integral image, the sum of pixel values within each grid cell is computed. Similar to LBP, the central grid cell's value is compared to those of its neighbors. If a neighbor's value is greater, a '1' is assigned; otherwise, a '0' is assigned. These binary values are then accumulated to form a binary pattern. The corresponding decimal value of this pattern is used as the MB-LBP descriptor.

the value is greater or smaller, a binary value of 1 or 0 is assigned to the grid cell respectively. These values are then accumulated to create a binary pattern, which serves as the MB-LBP value for that image patch and also acts as a feature vector.

MB-LBP utilizes an integral image to compute the sum of blocks in constant time. An illustration demonstrates how an integral image is constructed and the sums of blocks are calculated in constant time (refer to Figure 5.11).

5.2.5.1 Integral Image

Each pixel in the integral image represents the sum of the pixels to its left and top, along with its own value. For instance, in the 2x2 sample input image depicted in Figure 5.11, the top-left corner element (1) lacks any elements to its left or top, resulting in its value being directly copied to the integral image. Conversely, the element (5) has a neighboring pixel (1) to its left, thus its value becomes 6 (=5+1) in the integral image. Similarly, the element (4) possesses pixels (1 and 2) to its left and a pixel (5) above it, resulting in a value of 12 (=4+1+2+5).

After computing the integral image, the sum of any block can be determined by examining just four values within the integral image. For instance, consider the block highlighted in the input image in Figure 5.11. To calculate the sum of this block, begin by taking the value of the bottom-right corner pixel (46). Then, subtract the values of the top-right corner pixel (22, which already includes the sum of all pixels above), and the bottom-left corner pixel value (20, which includes all pixels to the left). Finally, add the value of the top-left corner pixel (10) to account for the double subtraction. This process yields the sum of the block.

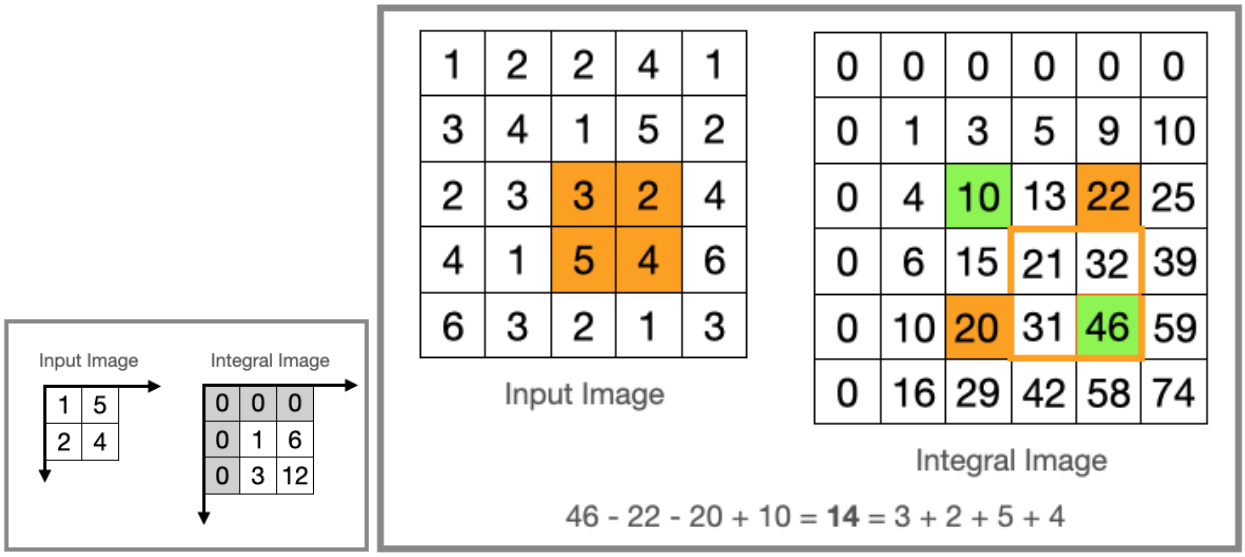


Figure 5.11: An integral image is a data structure designed for the efficient computation of the sum of values within rectangular regions of a grid. On the left, there is a 2x2 sample image alongside its integral image, where each pixel represents the sum of itself and all pixels above and to its left. On the right, a demonstration illustrates how the sum of a block of pixels can be calculated quickly using just four values from the integral image

5.2.6 Histogram of Gradients Filters

Histogram of Gradients (HOG) calculates and records the direction and strength of gradients in localized portions of an image, as gradients often contain rich information about the shape and appearance of objects within an image. The steps to calculate HOG feature descriptor are detailed below:

- Gradient Computation: apply gradient operators like Sobel to calculate gradient (magnitude and direction) values for each pixel
- Histogram Calculation: image is divided into a grid and a histogram of gradient directions are calculated within each grid cell.
- Descriptor Calculation: to reduce the effects of lighting variations, the histograms are normalized over larger, spatially connected blocks of cells and concatenated to form the feature vector

A sample image with gradients calculated for each segment is shown in Figure 5.12.

The HOG method is adept at capturing edge and texture information, demonstrating robustness against geometric and photometric transformations, although it is sensitive to rotation changes.

The following section explores transform-based texture methods, which analyze textures by converting the image data into a different domain for processing.

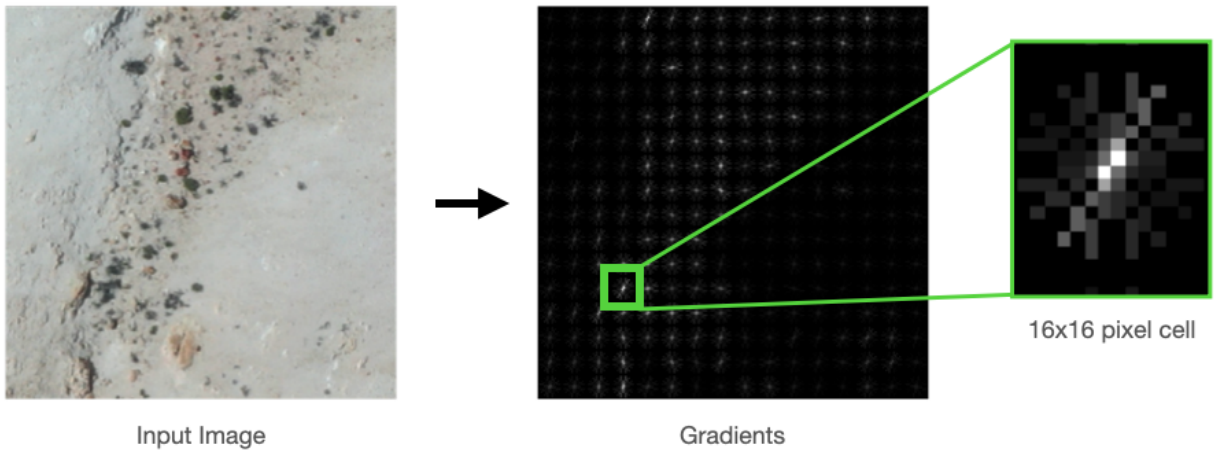


Figure 5.12: Example of HOG calculation for a sample image patch

5.2.7 MOSAIC Features

Deep learning models such as Convolutional Neural Networks (CNNs) have demonstrated efficacy in distilling RGB information into intermediate feature representations. However, many CNNs operate in a supervised setup, necessitating abundant training data. Rolf et al. (2021) proposed an alternative unsupervised approach, particularly suitable for orbital or ortho images. This method involves randomly selecting small patches from input images and employing them as convolution kernels to generate feature responses. These responses are more generalizable and applicable across a wide range of tasks.

Following this approach, 1024 random patches (see Figure 5.13) are extracted from the dronemap dataset and used as convolution kernels. The MOSAIC calculation yields a feature vector of size 1024, which the latent encoder then uses to learn the latent association.

5.2.8 Gabor Filters

Gabor filters are characterized by a sinusoidal wave modulated by a Gaussian function. They possess advantageous localization properties in both the spatial and frequency domains. Gabor functions exhibit responses akin to cells found in the visual cortex of mammalian brains (Marĉelja, 1980). The mathematical expression for the Gabor filter function is given by:

$$g(x, y; \lambda, \theta, \psi, \alpha, \gamma) = exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) exp\left(i\left(2\pi\frac{x'}{\lambda} + \psi\right)\right)$$

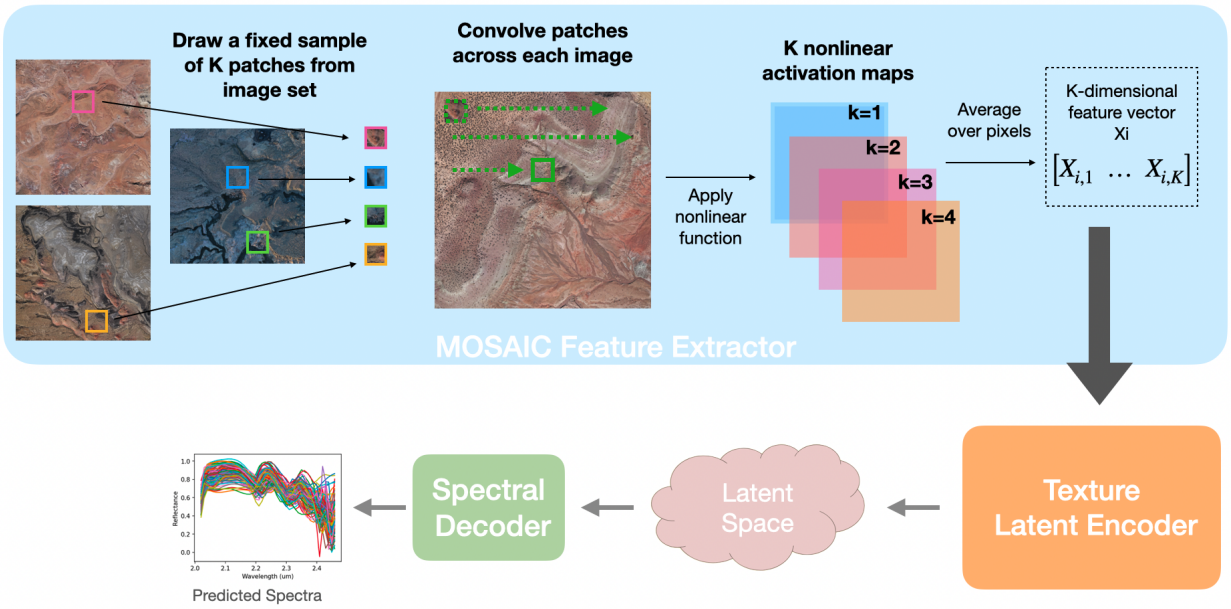


Figure 5.13: An approach to integrate MOSAIC features into the texture-latent-encoder. The top row details the extraction of MOSAIC features from images and the bottom row shows how it gets incorporated into our pipeline

where $x^{'} = xcos\theta + ysin\theta$ and $y^{'} = -xsin\theta + ycos\theta$. This equation yields both real and imagery parts and the real part is used for our purposes. The parameters λ, θ, ψ control the sinusoidal part: λ represents the wavelength, where higher values widen the ellipses; θ governs the rotation of the ellipses, with zero aligning them vertically; ψ determines the phase shift of the sinusoidal. σ is the standard deviation of the gaussian function and ψ dictates the aspect ratio of the ellipses. By adjusting these parameters, any number of Gabor filter kernels can be generated. In our study, 32 such filters are generated (refer to Figure 5.14).

When applied to an image, the filter response is obtained by convolving the filter over the image. The feature vector is generated by computing the mean and variance of the filter responses, resulting in a vector with a size of 64.

Gabor filters excel in spatial localization, allowing for the precise identification of edges and texture boundaries within images. They also demonstrate robustness against variations in illumination. However, they face challenges as well; Gabor filters are prone to noise sensitivity because the convolution process can enhance high-frequency noise components. Additionally, the success of these filters heavily relies on the careful selection of parameters, such as the size or wavelength of the kernels.

The following section shifts focus from transform-based methods to a model-based approach using fractal dimension.

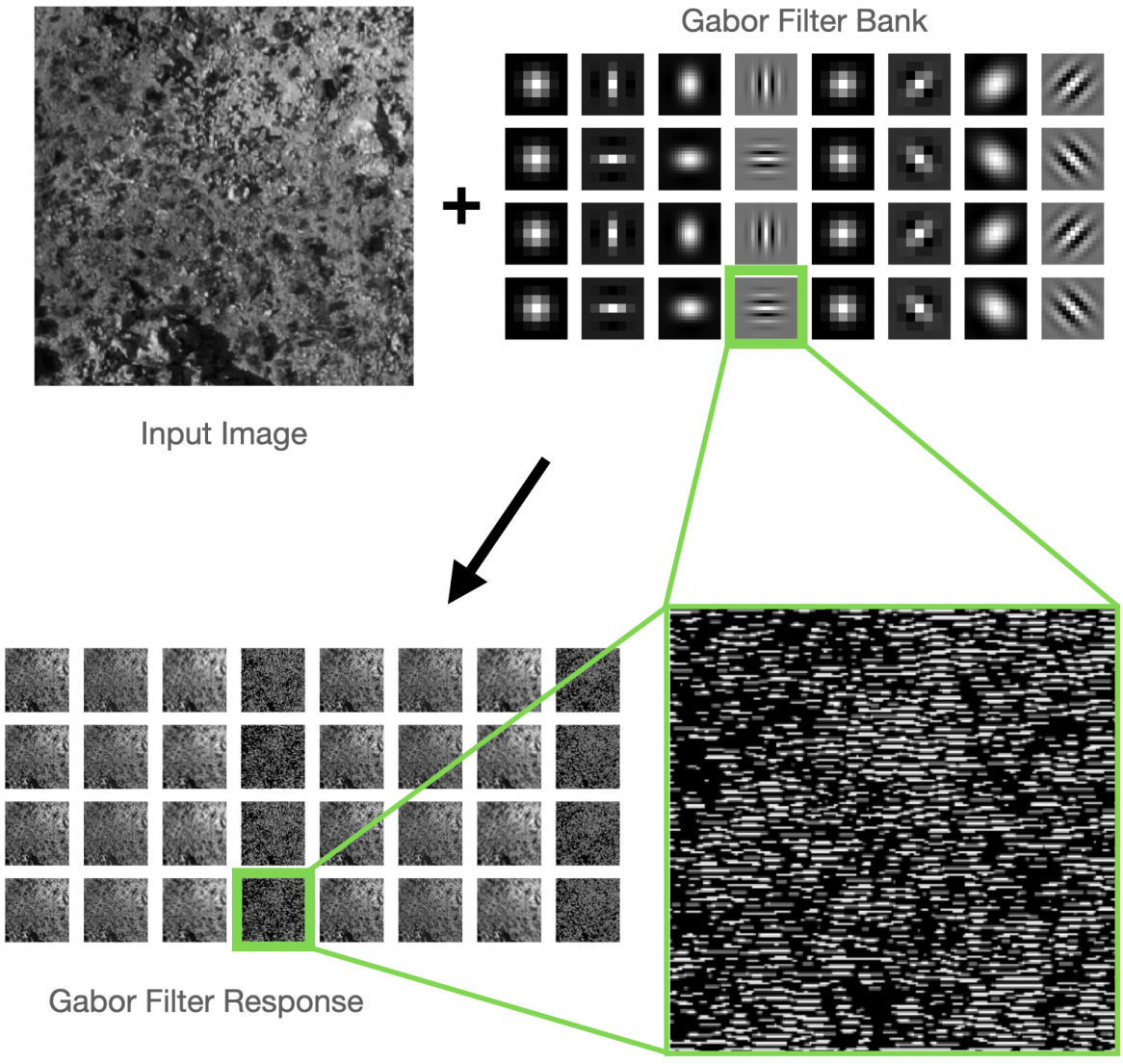


Figure 5.14: Example of filter response for sample input image using Gabor filters

Figure 5.15: Simple example demonstrating the Box counting method on a coastline (Image credit: Wikipedia)

5.2.9 Fractal Dimensions

Fractals are intricate patterns formed through a repetitive process, exhibiting self-similarity across various scales. The fractal dimension quantifies the space-filling capacity or roughness of such fractal objects. In this work, Box counting method (Conci and Proença, 1998) is employed to compute the fractal dimension (refer to Figure 5.15). Initially, the image is thresholded to form a binary image. The binary image is overlaid with a grid, and the number of grid elements necessary to cover the entire area is counted. Then, the grid size is reduced, forming a finer grid, and the process of counting the number of grid elements is repeated. The fractal dimension is calculated using the equation:

$$Fractal\ Dimension = \lim_{r \to 0} \frac{log(N)}{log(1/r)}$$

where N is the number of boxes and r is the length of the grid element.

The following section presents a series of experiments that utilize all the texture algorithms previously discussed, applied to the dronemaps dataset.

5.3 Experiments

5.3.1 Comparing RGB and Texture Latent Encoders

An experiment is setup to evaluate these methods using dronemaps from all three field sites and the cumulative results are shown in Figure 5.16. The spectral predictions from

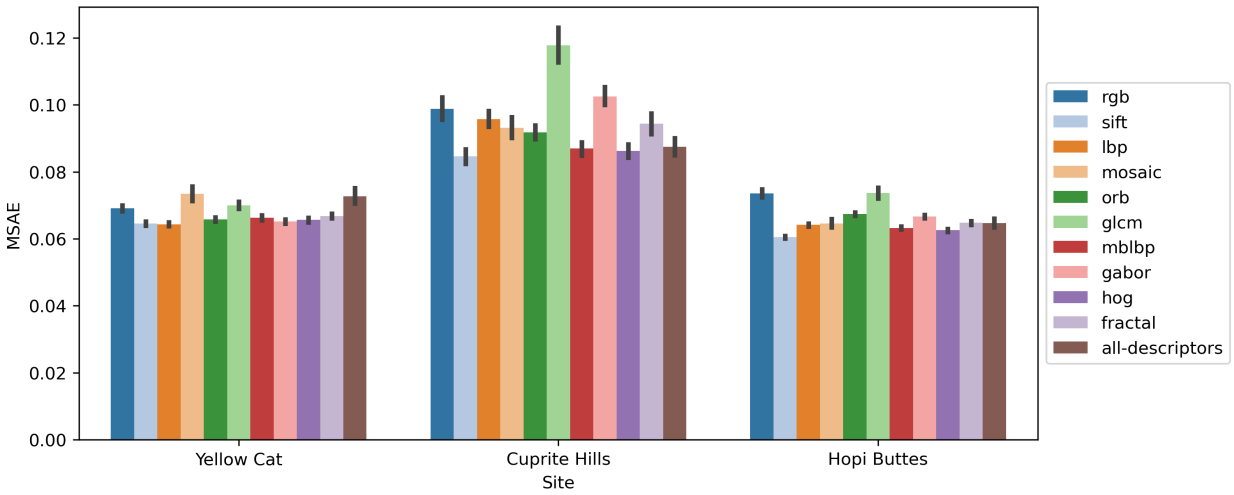


Figure 5.16: Cumulative results of all the texture descriptors on the three field datasets. Shorter bar signify better performance.

the different texture based latent encoders are compared to the original AVIRIS spectra to calculate the MSAE. Overall, most of the texture based latent encoders perform better than RGB latent encoder showing that using texture improves spectral reconstruction accuracy. GLCM appears to be the next worst performer, possibly because GLCM is more sensitive to intensity changes as it encodes the intensity values directly, which could vary greatly across the image. Overall, SIFT, LBP, and HOG performed best across all these datasets. These three descriptors encode only the texture information, not relying on the underlying color. Though HOG feature does not have scale or rotation invariance, it still performs better due to the lack of scale or rotation changes during the training or mapping phase.

The *all-descriptors* latent encoder is formed by concatenating the feature vectors from all the textures together. The *all-descriptors* latent encoder, although including all the features, is not the best performer, due to some parts of the feature vector playing against other parts.

5.3.2 Adding RGB to Texture Latent Encoders

The previous experiment assessed the performance of latent encoders trained on RGB against those trained on various texture descriptors. In this experiment, both RGB and texture feature vectors are combined to train latent encoders. The output from these latent encoders are then compared to those trained solely on texture feature vectors (without any RGB data) (see Figure 5.17).

The latent encoders trained on both texture and RGB consistently underperform across all textures, mirroring the performance of the *all-descriptor* latent encoder where less-informative additions to the feature vector diminish overall performance. GLCM shows

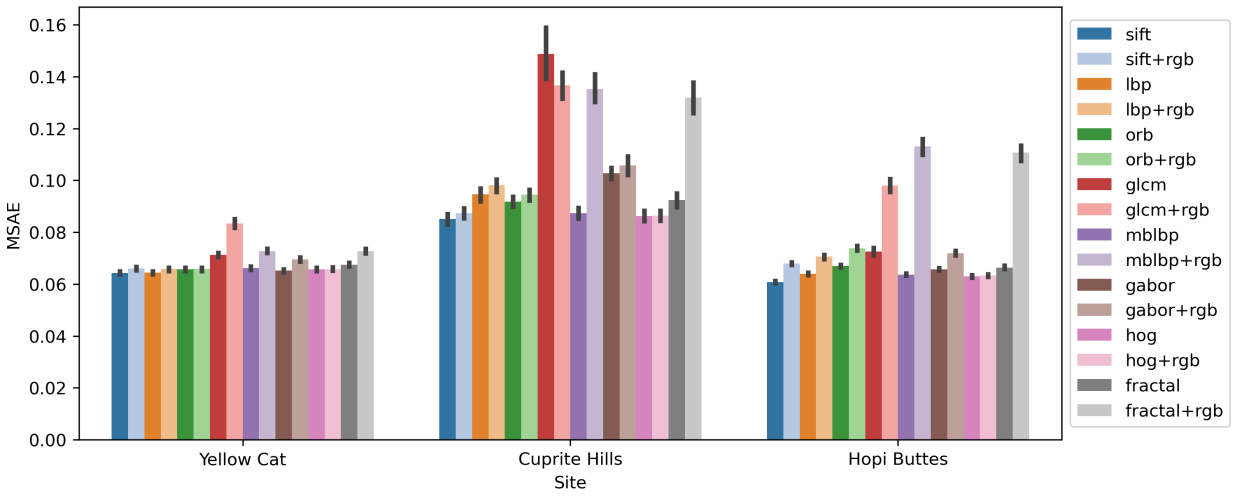


Figure 5.17: Comparing latent encoders that utilize only texture in the feature vector with those that incorporate both texture and RGB. Adding RGB to the texture feature vector results in decreased overall performance

a slight improvement with RGB in the Cuprite Hills dataset. However, this scenario is also akin to the *all-descriptor* case, where GLCM is the less informative component (as evident from previous experiment in Figure 5.16) and is enhanced by the inclusion of RGB.

5.3.3 Better Texture Latent Encoders

Given that SIFT and LBP are performing better individually, a new feature called *texture* is created that combined SIFT and LBP feature vectors together (refer to Figure 5.18). It is evident that the *texture* latent encoder outperforms the individual features.

For the same experiment, the results are displayed for individual dronemaps as illustrated in Figure 5.19. It is clear from these results that the texture latent encoder consistently outperforms the individual features, SIFT or LBP, in nearly all cases.

5.3.4 Comparing RGB, Texture and Multispectral Latent Encoders

Although texture performs better than RGB, multispectral latent encoders consistently outperform texture, as demonstrated in Figure 5.20. This improvement is attributed to the fact that the spectra being reconstructed are primarily in the short-wave-infrared (SWIR) region and the multispectral data includes some of the SWIR bands.

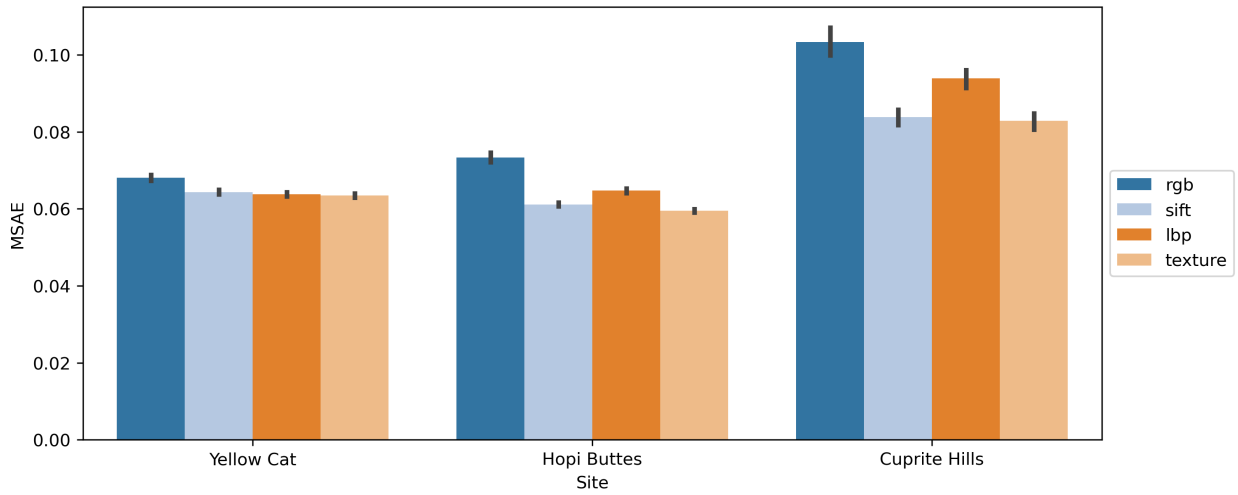


Figure 5.18: Showing results of the combined SIFT-LBP latent encoder which outperforms each of the individual latent encoders

5.4 Summary

In this chapter, various texture extraction approaches, including structural, statistical, transform-based, and model-based methods, were explored. These texture methods were integrated into the simple sensor spectral model, and their performances were assessed and compared. Specifically, superior performance was demonstrated by SIFT from the structural category and LBP from the statistical category. A new texture, created by combining feature vectors from these two methods, consistently outperformed others across all datasets. It was also shown that adding less-informative RGB to texture led to diminished performance. Finally, adding spectral bands has been demonstrated to outperform texture when the spectral bands include the wavelengths of the reconstructed spectra.

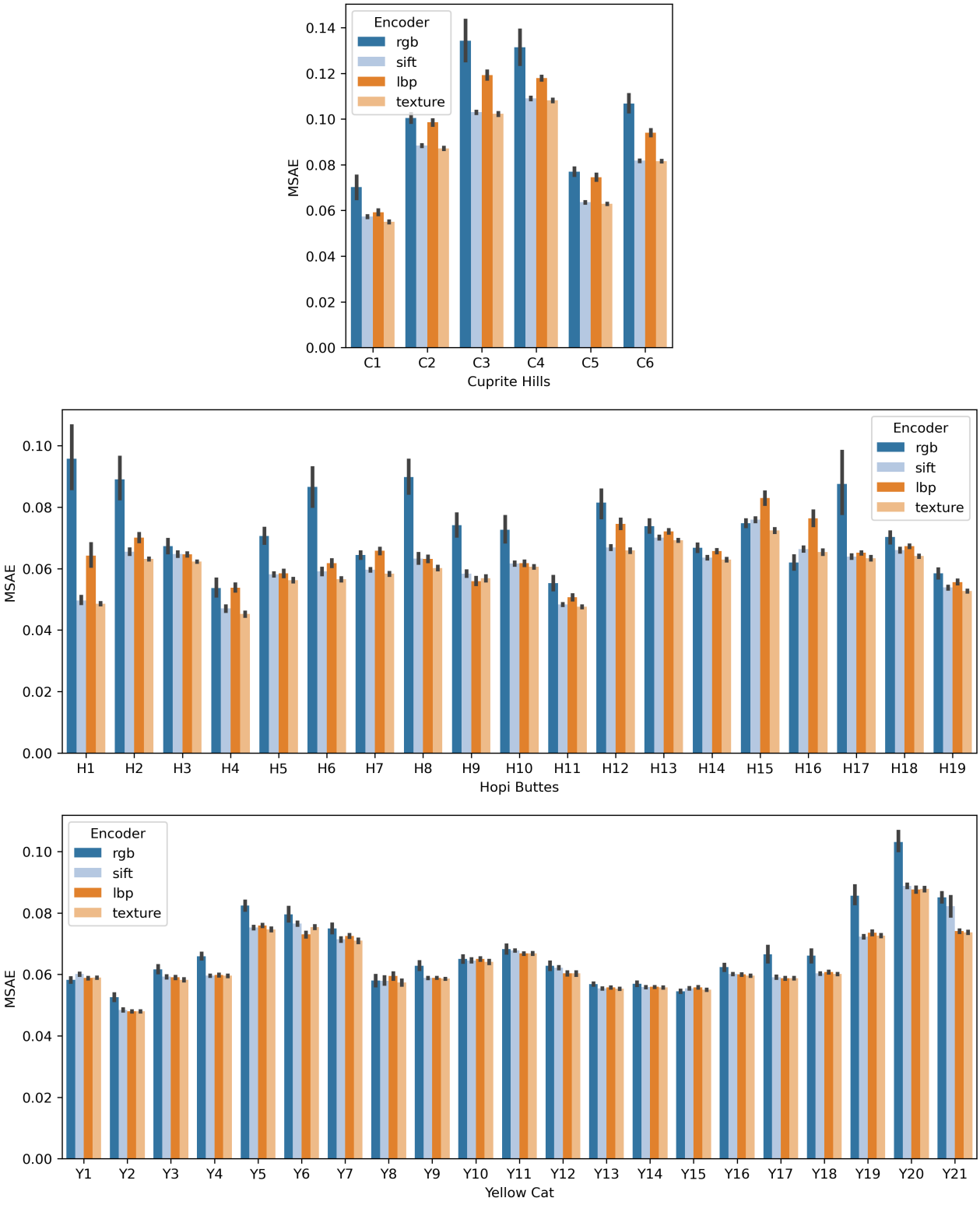


Figure 5.19: Results from the combined SIFT-LBP latent encoder for the drone map at each site

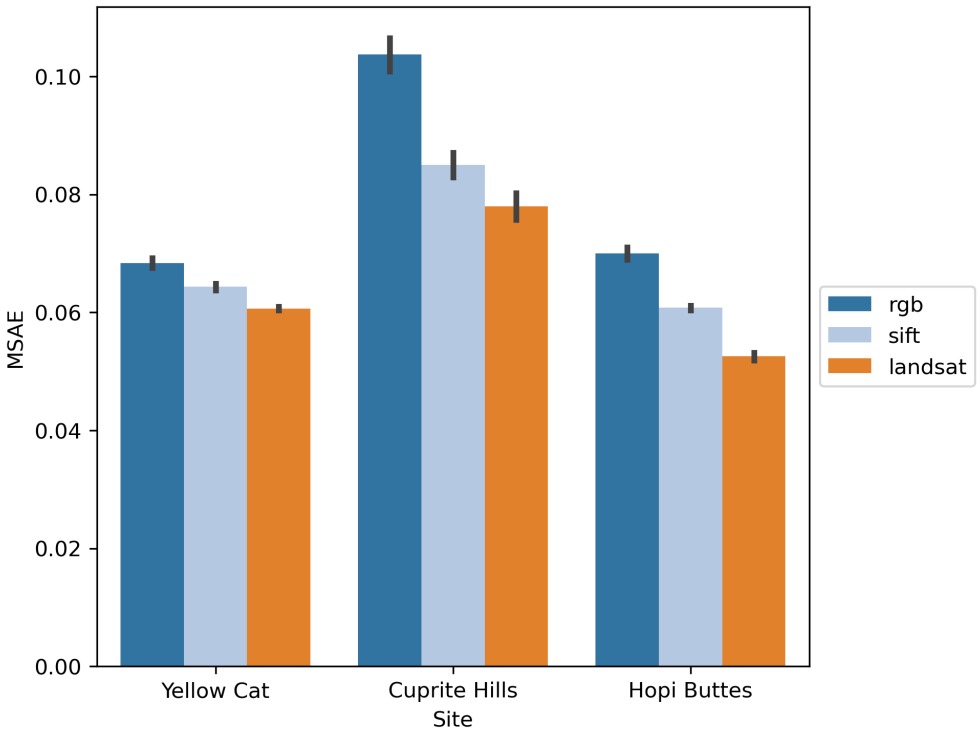


Figure 5.20: Comparing RGB, texture and multispectral latent encoders

Chapter 6

Mineral Prediction

A significant contribution of this research lies in developing a framework capable of utilizing RGB camera data to generate hyperspectral signals. While our analyses have been thorough, the comparative nature of our results limits their interpretive depth. To address this issue, a practical application is needed that can effectively gauge the utility of the achieved accuracies. Identifying underlying minerals stands as a key application of spectral mapping and a primary objective of exploration endeavor which is explored in this chapter. First, the minerals predicted by the spectral compressor is evaluated to validate if spectral compression and reconstruction yields usable spectra. Next, the mineral prediction accuracies of the different latent encoders (multispectral and texture) are evaluated. Finally, an example scenario to assess the feasibility of employing solely RGB data for spectral mapping purposes is explored.

Next, let us look at the tool used for mineral identification, Tetracorder .

6.1 Tetracorder

Tetracorder[1](Clark et al., 2003, 2024) is an open-source, expert system designed to process and analyze spectra. Renowned as the gold standard for mineral identification applications, Tetracorder relies on decades of meticulous work by mineralogists and spectroscopists in cataloging spectra and spectral features. Its extensive database, catalogs reflectance spectra from various minerals, identifying unique absorption features that distinguish each mineral type. Tetracorder executes a series of evaluations to pinpoint characteristic features of spectra while carefully isolating noise to identify the underlying mineral.

Tetracorder predictions are categorized into groups based on the spectral wavelengths under examination. The 2.0μm to 2.5μm region stands out as the most diagnostic for mineral identification for carbonates and 0.35μm to 2.5μm region for hydroxyl-bearing

[1]https://github.com/PSI-edu/spectroscopy-tetracorder

minerals. The 2.0μm to 2.5μm region has been the primary focus of this investigation. Tetracorder identifies minerals using a total of 227 reference spectra within this group, which are categorize into 9 mineral categories as noted in Chapter 3. The mineral categories are: kaolinite group (*kaolgrp*), kaolinite-muscovite (*kaol+musc*), mica group (*micagrp*), *smectite, carbonate, calcite, sepiolite, NHfeldspar*, and *others*.

Following this, let us assess the efficacy of spectral compression, demonstrating how reconstructed spectra remain viable for mineral predictions despite the loss of information in the compression step.

6.2 Efficacy of Spectral Compressor

As spectral signals undergo spectral compression, the extent of information loss depends on the effectiveness of the compressor and the bottleneck of the latent space. This loss results in variations between the predicted spectra and the input spectra.

Assessing the ability of these information-depleted, reconstructed spectra to identify minerals, is crucial. This evaluation is significant because the output of the latent encoders can, at most, mirror the results obtained from spectral compressor. Thus, the spectral compressor establishes the upper limit for the prediction accuracy of the latent encoders.

Although RMSE and MSAE quantify the deviation of spectra from the original observed spectra, they do not consistently reflect the mineral identification accuracy. Predictions from both a spectral compressor and a RGB-latent encoder alongside the original AVIRIS signal is shown in Figure 6.1. The spectral compressor is trained on wavelength from 2.0μm to 2.5μm. Despite the visual similarity between all the predicted spectra and the original AVIRIS spectra, and the low MSAE, there are instances where different minerals are identified.

6.2.1 Mineral prediction accuracy

The mineral prediction accuracy for spectra processed through the spectral compressor across the three field sites are presented in Table 6.1. The prediction accuracy is relatively similar for Cuprite Hills and Yellow Cat. Hopi Buttes exhibits a higher prediction accuracy of 80% due to its low mineral diversity. This is because its most abundant mineral (smectite group) is identified correctly (see Figure 6.3) and hence a boost in the performance. To compare the minerals identified using AVIRIS spectra refer to Figures 3.12,3.13 and 3.14 in Chapter 3.

The mineral maps generated by Tetracorder from the spectra compressed by the spectral compressor are shown in Figures 6.2, 6.3 and 6.4. A significant number of mineral categories are correctly identified by the compressed spectra. This bolsters the credibility of this approach.

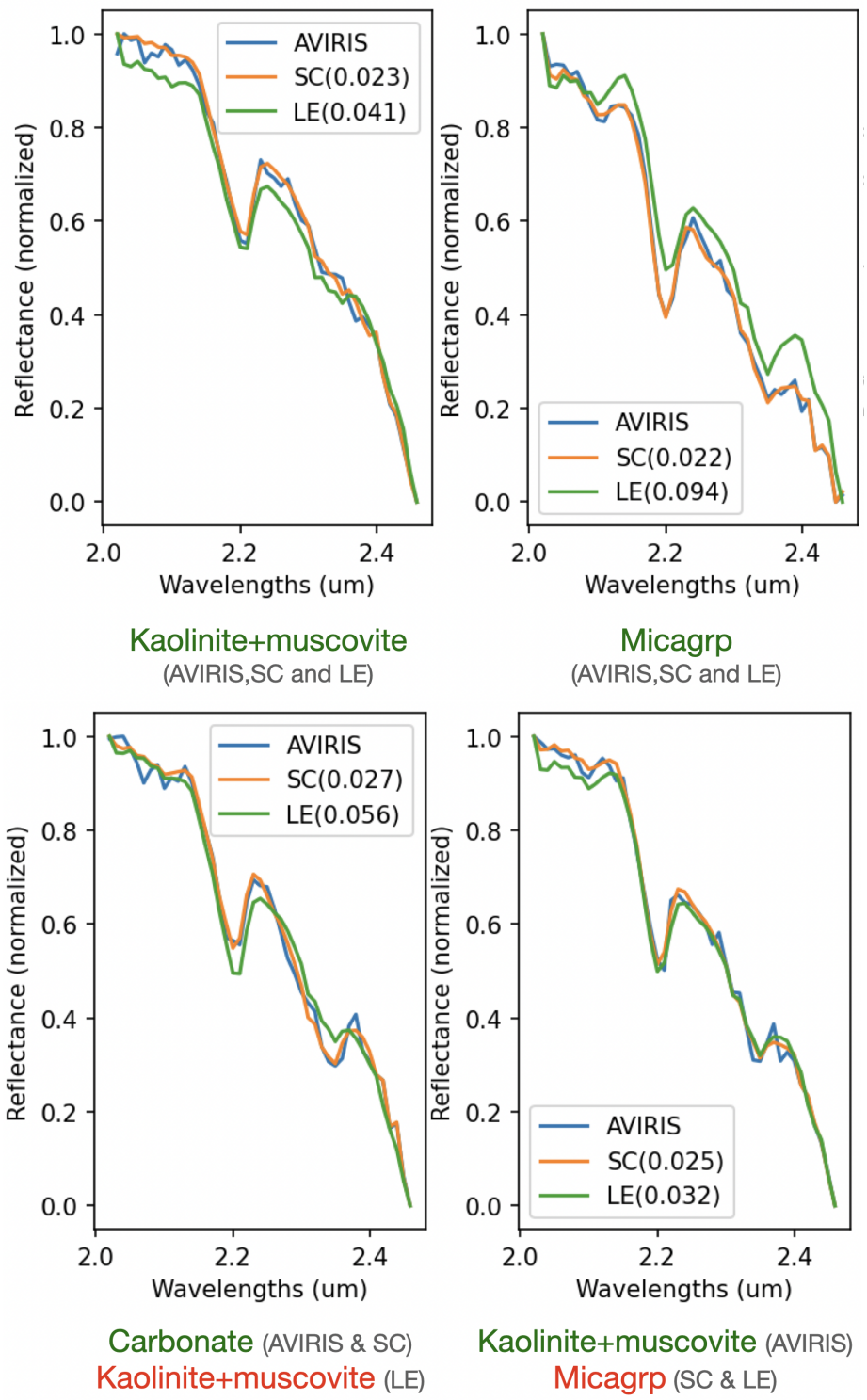
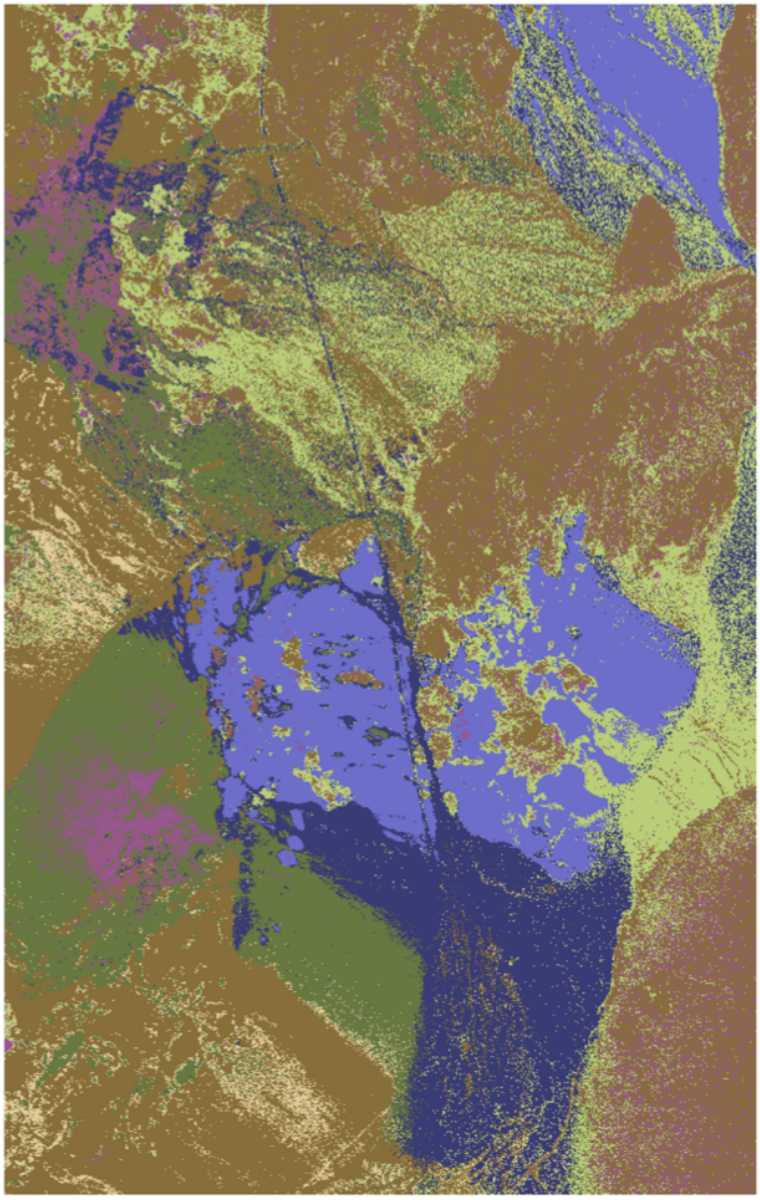
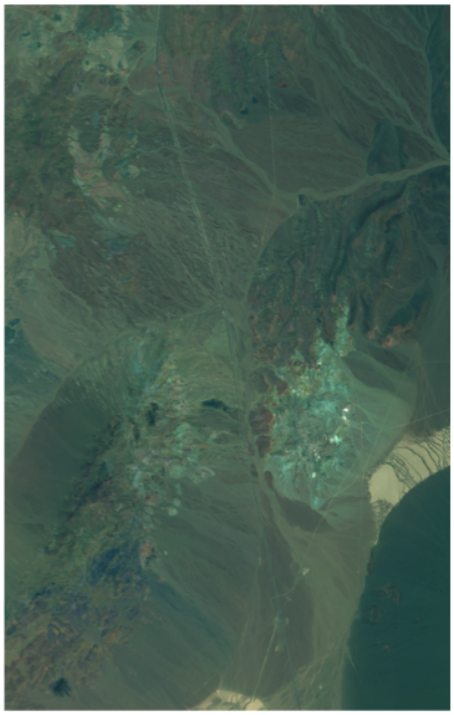


Figure 6.1: Qualitative comparison between the original spectra and the spectral predictions generated by spectral compressor and a latent encoder for four distinct pixels accompanied by mineral predictions from Tetracorder. The MSAE is shown in parentheses in the legend. In the first two graphs, minerals are correctly identified from both the original and predicted spectra by both spectral compressor and the latent encoder. However, in the third graph, carbonate is identified as a Kaolinite+muscovite mixture by the latent encoder, while the Kaolinite+muscovite mixture is identified as micagrp by both spectral compressor and the latent encoder

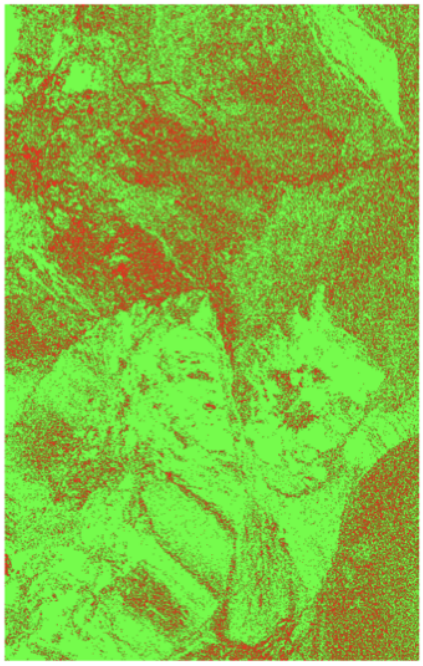


Mineral map (false color) from spectra processed through spectral compressor

28.9% carbonate
18.2% kaol+musc
18.0% smectite
12.9% micagrp
11.0% kaolgrp
7.5% others
3.4% calcite
0.2% sepiolite
0.0% NHfeldspar

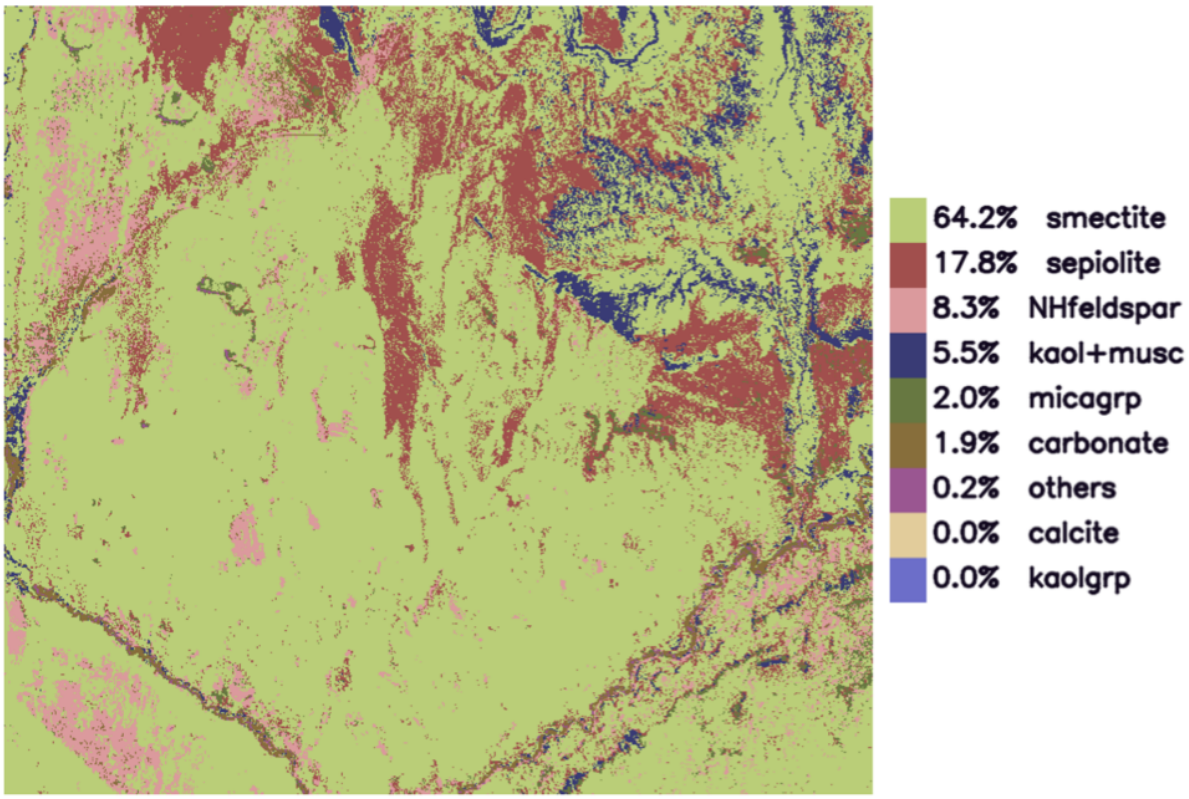


Cuprite Hills, Nevada (RGB image)

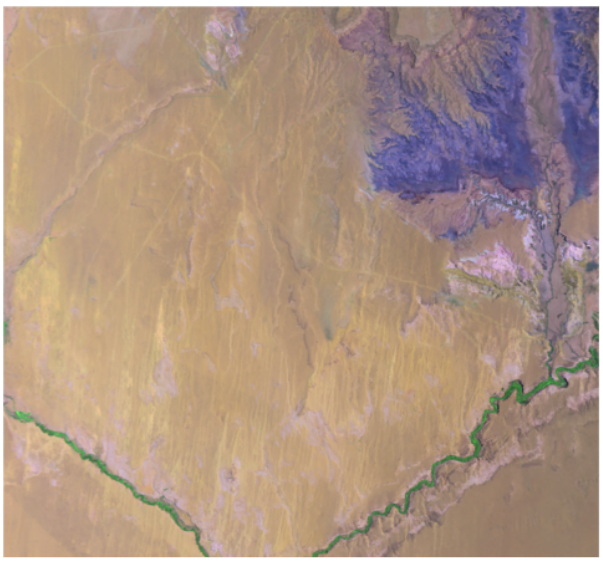


Correct predictions are marked in green (72.4%) and incorrect predictions are in red

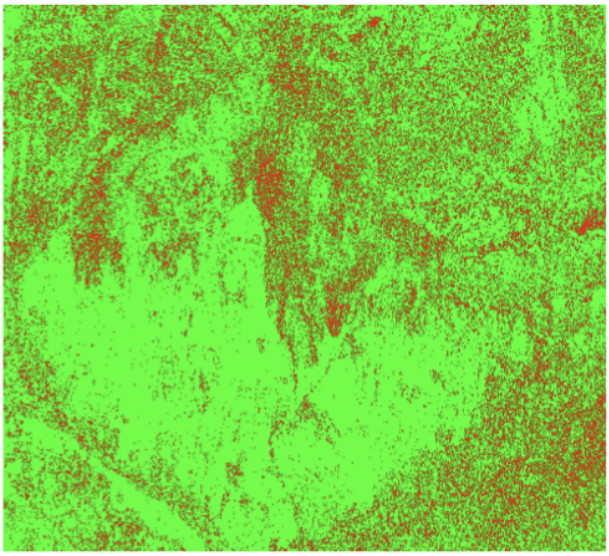
Figure 6.2: Minerals identified by Tetracorder for Cuprite Hills after being processed through the spectral compressor. The correct predictions *graphs compare the minerals predicted from the compressed spectra to those predicted from the original AVIRIS spectra, marking correctly identified pixels in green and incorrectly identified pixels in red. The mineral categories corresponding to the false colors in the mineral map are listed along with the abundance percentage. The RGB image of the site is included for reference*



Mineral map (false color) from spectra processed through spectral compressor

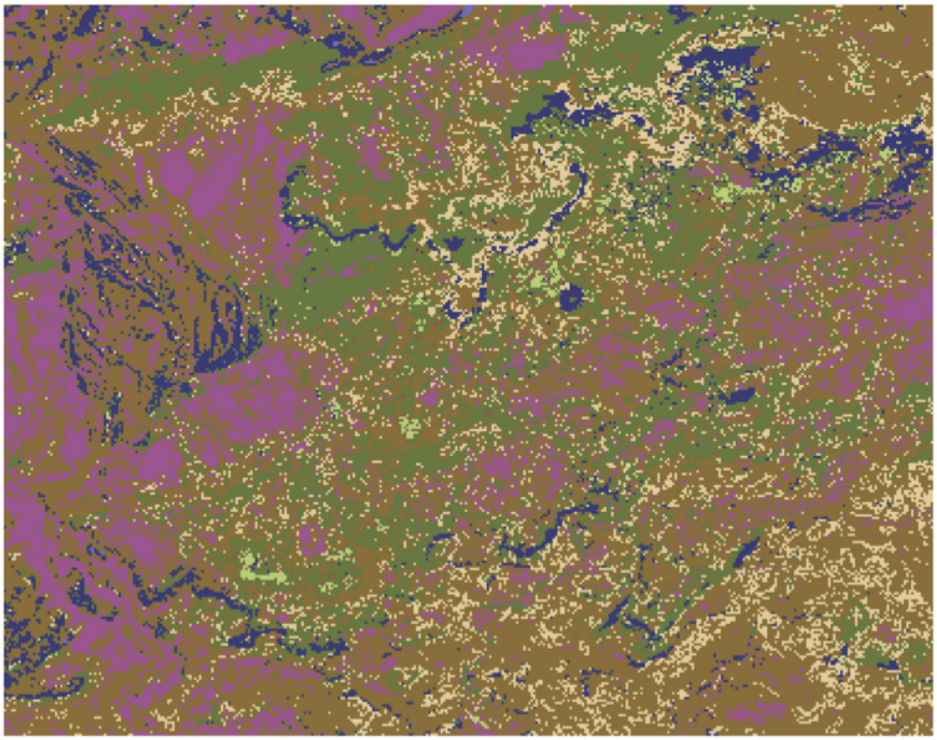


Hopi Buttes, Arizona (RGB Image)



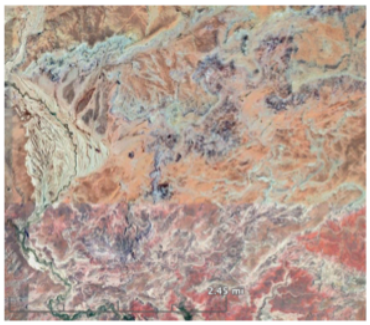
Correct predictions are marked in green (80.3%) and incorrect predictions are in red

Figure 6.3: Minerals identified by Tetracorder for Hopi Buttes after being processed through the spectral compressor. The correct predictions *graphs compare the minerals predicted from the compressed spectra to those predicted from the original AVIRIS spectra, marking correctly identified pixels in green and incorrectly identified pixels in red. The mineral categories corresponding to the false colors in the mineral map are listed along with the abundance percentage. The RGB image of the site is included for reference*

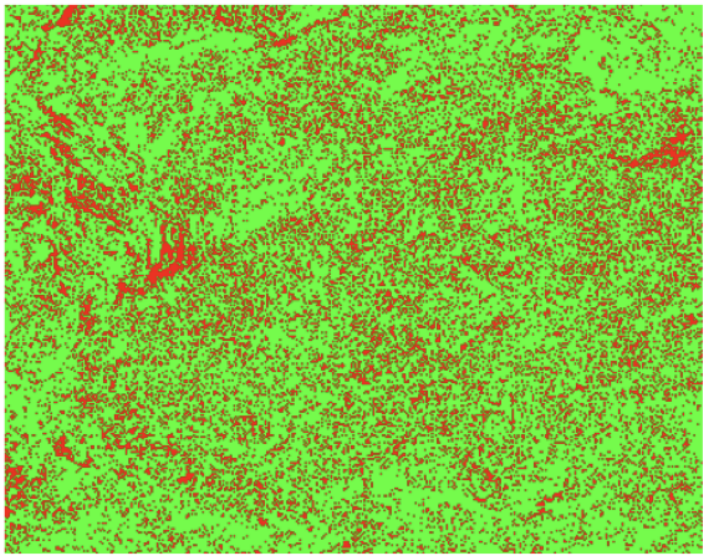


Mineral map (false color) from spectra processed through spectral compressor

47.9% carbonate
23.8% micagrp
12.0% calcite
11.4% others
2.4% kaol+musc
1.4% smectite
0.6% sepiolite
0.3% kaolgrp
0.0% NHfeldspar



Yellow Cat, Utah
(RGB Image)



Correct predictions are marked in green (74.8%) and incorrect predictions are in red

Figure 6.4: Minerals identified by Tetracorder for Yellow Cat after being processed through the spectral compressor. The correct predictions *graphs compare the minerals predicted from the compressed spectra to those predicted from the original AVIRIS spectra, marking correctly identified pixels in green and incorrectly identified pixels in red. The mineral categories corresponding to the false colors in the mineral map are listed along with the abundance percentage. The RGB image of the site is included for reference*

Site	Mineral Prediction Accuracy
Cuprite Hills	72.4%
Hopi Buttes	80.3%
Yellow Cat	74.8%

Table 6.1: Mineral prediction accuracy for the spectra processed through spectral compressor for the three field sites

Spectral Evaluation Metric	Correlation to mineral prediction accuracy
Cross-correlation (CC)	52.9%
Root Mean Squared Error (RMSE)	−93.0%
Mean Spectral Angle Error (MSAE)	−92.9%
Mean Latent Error (MLE)	−94.7%

Table 6.2: Correlation between the evaluation metrics to the mineral prediction accuracy

6.2.2 Validity of Spectral Evaluation Metrics

In Chapter 2 (Sec. 2.3), several evaluation metrics were introduced to assess spectral predictions, which are consistently employed to compare outputs from the spectral compressor and different latent encoders. To understand the relationship between these metrics and the mineral prediction accuracies, their correlations are calculated, as shown in Table 6.2. These metrics, widely acknowledged in remote sensing literature (Loncan et al., 2015), demonstrated strong correlations with mineral prediction accuracy, with three metrics surpassing 93%. This reaffirms the validity of the evaluation results showcased in preceding chapters utilizing these metrics. However, the cross-correlation metric, with a correlation of 53%, proved inadequate for robust comparison of spectral predictions and therefore warrants exclusion from further consideration. The RMSE, MSAE, and MLE exhibit an inverse relationship with mineral prediction accuracy. This is because as these error values decrease, mineral prediction accuracy increases, resulting in the observed negative correlation between them.

6.3 Spectral Compressor Variations: Based on Training Input

In Chapter 3 (Sec. 3.4), three versions of spectral compressors trained on varying data sources were introduced. Namely,

- *SiteSpecific*: SCs trained exclusively on data from specific sites and used solely for those sites.

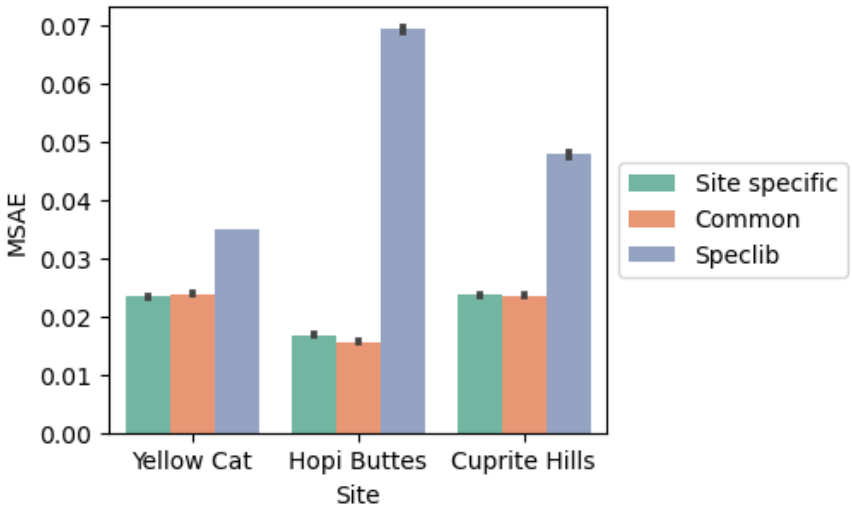


Figure 6.5: Mean spectral angle error comparisons of SiteSpecific, Common and Speclib SCs

Mineral prediction accuracy of sites	Cuprite Hills	Hopi Buttes	Yellow Cat
SiteSpecific SC	72.4%	80.3%	74.8%
Common SC	71.8%	82.6%	73.5%
Speclib SC	67.5%	57.6%	62.4%

Table 6.3: Mineral prediction accuracy for the three sites using the SCs trained on different data sources

- *Common*: a SC trained on data from all the sites and tested to each site.
- *Speclib*: a SC trained solely on data from the USGS spectral library, augmented with random mixtures.

The spectral angle error when employing the different SCs across the three field sites is shown in Figure 6.5. The mineral prediction accuracy of these SCs across the three sites are presented in Table 6.3.

The mineral prediction accuracy for the Cuprite Hills and Yellow Cat regions shows a slight decline with the Common SC compared to the Site-Specific SC, which is also reflected in the MSAE between the two SCs. This reflects the penalty incurred when training a single SC using data from all sites. Despite this, the benefits of training and maintaining a single SC outweigh the slight drop in accuracy. Additionally, while there is a further decrease in accuracy with the Speclib SC, it still maintains a mineral prediction accuracy above 60% for both Cuprite Hills and Yellow Cat.

In the case of Hopi Buttes, the mineral prediction accuracy for the Common SC is slightly better than that of the SiteSpecific SC, consistent with the pattern observed in the MSAE between the two SCs. One possible explanation for this difference is that the SiteSpecific SC for Hopi Buttes underwent fewer training epochs compared to the Common SC,

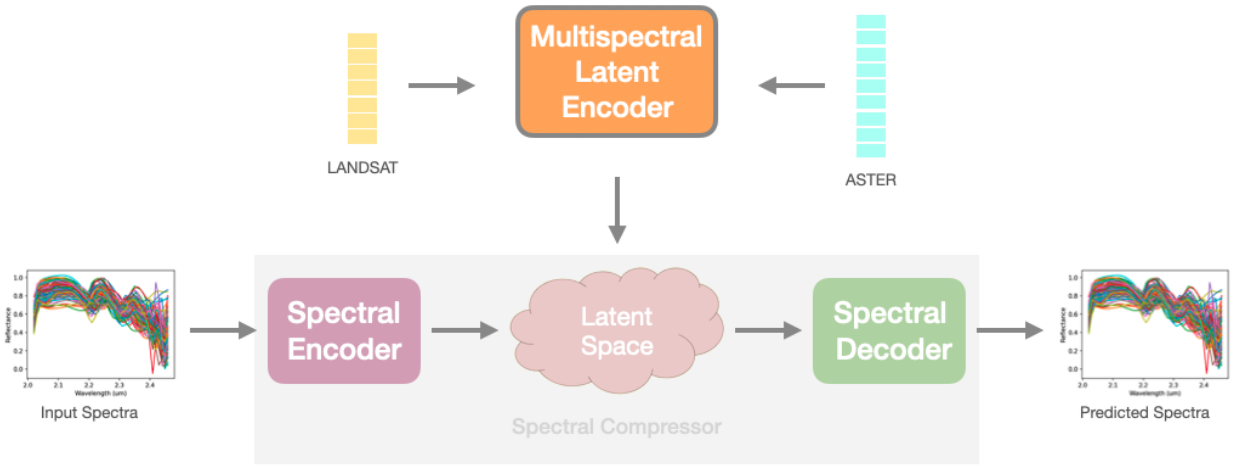


Figure 6.6: Simple-sensor-spectral model updated to accommodate multispectral sensor input

leading to its reduced performance. The mineral prediction accuracy for the Speclib SC is modest at around 58%.

It is noteworthy that the Speclib SC achieves this level of accuracy without relying on any prior information about these sites. This indicates that the Speclib SC can be applied in any unexplored site, even without prior knowledge, and still demonstrate a modest level of accuracy in mineral prediction which is an exciting result.

6.4 Multi-Spectral Mapping

Multispectral latent encoders using ASTER and Landsat multispectral data were introduced in Chapter 4 (refer to Figure 6.6). In this section, let us evaluate their efficacy for the mineral identification task. The cumulative results of mineral prediction accuracy across the three sites along with breakdown for individual drone maps is shown in Figure 6.7.

Overall, landsat-LE outperforms aster-LE, which aligns with the observed pattern in the MSAE analysis. The higher precision of the Landsat instrument is attributed for the better performance.

The mineral prediction accuracy of both landsat-LE and aster-LE is lower for the Yellow Cat region, despite the high accuracy observed with spectral compression. One potential explanation for this discrepancy is that the bands used by ASTER and Landsat instruments may not be sufficiently diagnostic for the minerals present in this particular region.

While the overall mineral prediction accuracy is higher for Hopi Buttes, there is considerable variance in the prediction accuracies across different areas within the region.

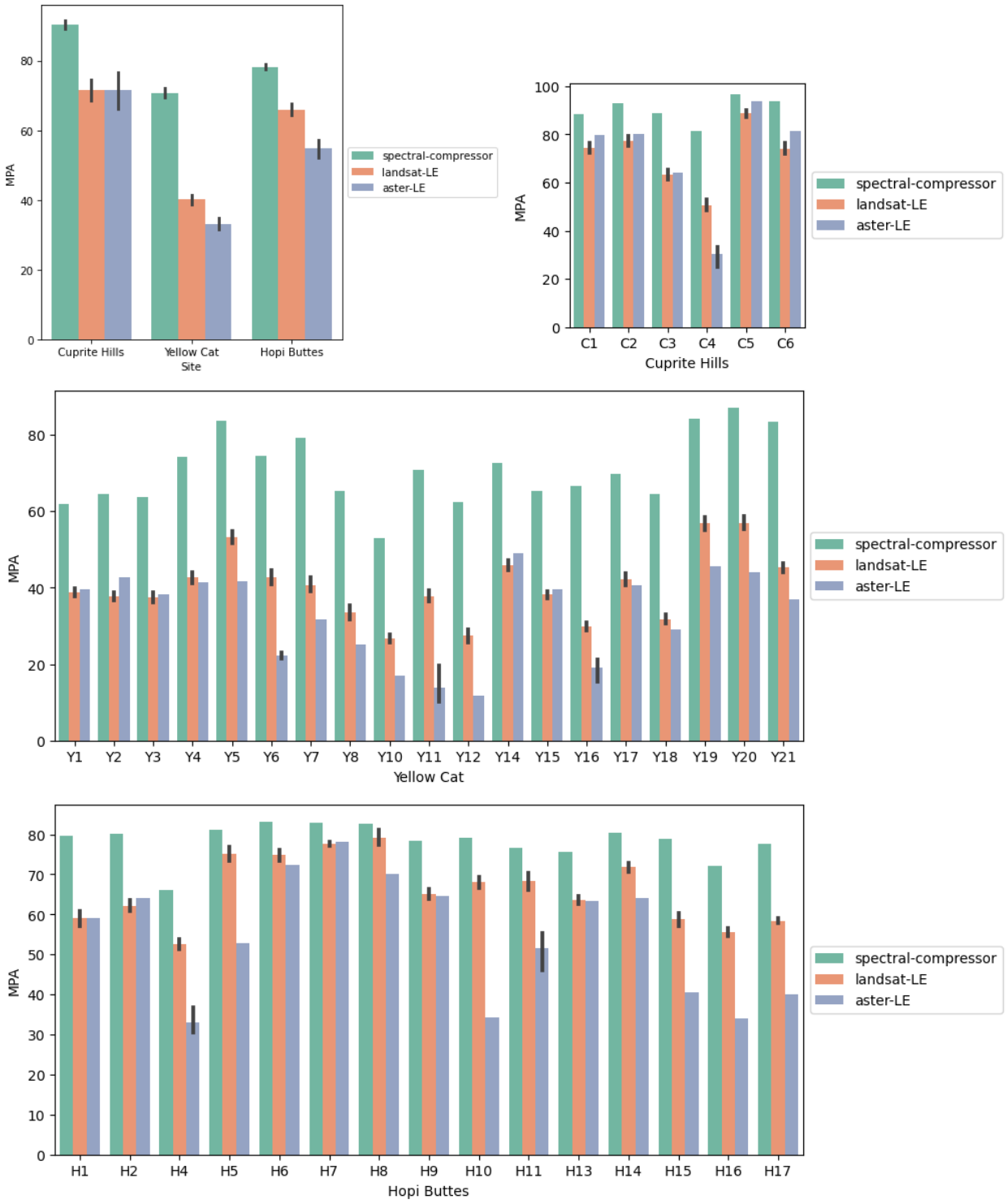


Figure 6.7: Mineral prediction accuracy for multispectral latent encoders. Taller bar represents better performance

For example, the aster-LE accuracy for H4 dronemap is only 30%, whereas for H7, it is close to 80%. This variability suggests that the accuracy of mineral predictions is heavily influenced by the specific characteristics of each region, including the minerals present and the spectral bands utilized by the latent encoder.

In summary, these results show that latent encoders trained on few extra spectral bands of the multispectral data, can achieve up to 70% accuracy in predicting minerals, emphasizing the fact that adding diagnostic spectral bands helps the latent encoder in making better predictions.

6.5 RGB Spectral Mapping

6.5.1 Texture Latent Encoder

In this section, two latent encoders are introduced and compared against the mineral prediction accuracy of the spectral compressor as shown in Figure 6.8.

- *Texture* latent encoder combines SIFT and LBP textural features that were identified as the best textural features in Chapter 5.
- *Enhanced-texture* latent encoder combines *texture* with Landsat data to form an enhanced version of the texture latent encoder

Enhanced-texture relies on the idea that adding diagnostic bands could improve the mineral prediction accuracy. Indeed, incorporating multispectral bands of Landsat into the *texture* latent encoder improves mineral prediction accuracy, particularly in Yellow Cat and Hopi Buttes, with a marginal improvement in Cuprite Hills as shown in Figure 6.8. It is important to note that not all mineral spectra are easily reconstructed, and thus the accuracy of reconstruction varies depending on the composition of minerals in a particular site. For sites with diverse mineral compositions such as Cuprite Hills, variance in prediction accuracy is expected.

6.5.2 RGB Latent Encoder

All three field sites were specifically chosen for exploration due to their mineral diversity. However in typical exploration scenarios, large areas are surveyed which are mostly benign, with occasional small sections exhibiting mineralogical diversity. It is therefore beneficial to cover the mineralogical benign regions with simple sensors and reserve the use of imaging spectrometer to the mineralogically rich regions. To illustrate that the

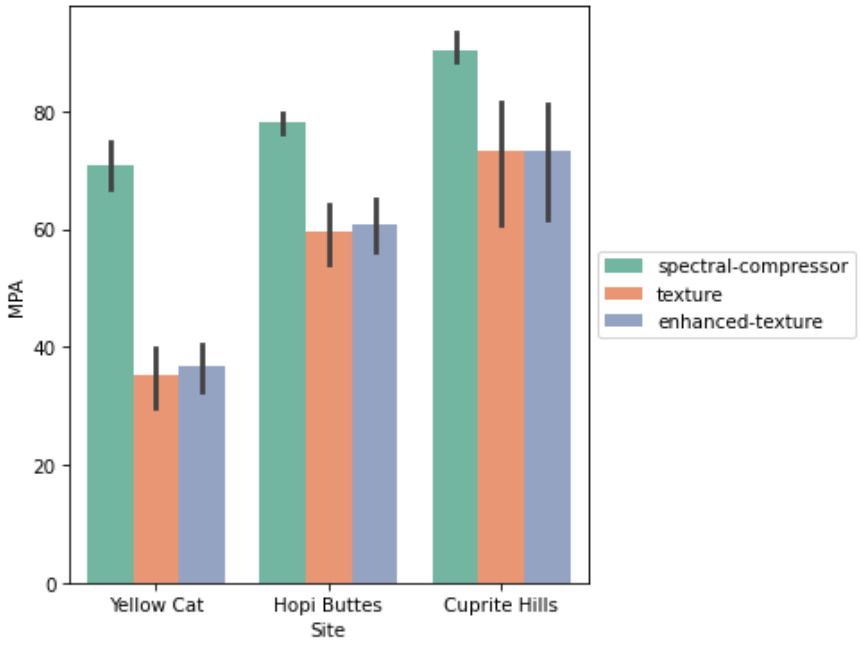


Figure 6.8: Mineral prediction accuracies for texture and enhanced-texture latent encoders

Location	Mineral Prediction Accuracy (%)
Locale-A	81.1
Locale-B	70.3
Locale-C	59.5
Locale-D	65.2

Table 6.4: Results of employing the RGB latent encoder across various locales within the Cuprite Hills region

spectral predictions from simple sensors (using the latent encoder) can produce sufficient mineral prediction accuracies an experiment is setup in the Cuprite Hills region. Four locales are selected for analysis (refer to Figure 6.9), each showcasing varying levels of accuracy for the RGB latent encoder. These locations, each covering an area of at least 2 square kilometers, are suitable as field sites for ground robot missions spanning multiple days (refer to Table 6.4).

Although visually similar, the mineral map reveals significant spectral diversity for all locations. The latent encoder effectively distinguishes visual features and generates accurate mineral predictions. Locale-A yielded the highest prediction accuracy at 81.1%, followed by Locale-B at 70.3%, with slightly lower accuracies observed for Locale-C and D. The ability of the latent encoder to achieve such accuracy using only RGB data and a limited number of hyperspectral training points is noteworthy, particularly within the spectrally diverse Cuprite Hills region.

Locale-C presents an intriguing case, as one might expect greater diversity in the RGB space to result in improved mineral prediction accuracies. However, Locale-C exhibits

the lowest accuracy value at 59.5%. This discrepancy may be attributed to several factors. Firstly, the minerals in this region may be less prevalent and thus inadequately represented in the latent encoder training data. Alternatively, conflicting RGB values within the training samples, particularly regarding white materials, such as the playa (the light area on the right edge) containing Kaolinite and other white materials, could contribute to this disparity.

Notably, the minerals vary across these locales, underscoring the robustness of the latent encoder in predicting different types of minerals.

6.6 Summary

In this chapter, the efficacy of the spectral compressors was validated through mineral predictions, achieving accuracies up to 80%. Next, the mineral prediction capability of latent encoders utilizing multispectral and texture information was validated. Feature vectors from texture and multispectral data were combined to form an enhanced texture, which performed better than either individually. Finally, an experiment was showcased demonstrating the efficacy of RGB data in predicting mineralogy, illustrating that spectral mapping can be conducted using solely RGB cameras in an area large enough for a field experiment with a ground or aerial vehicle, achieving up to 80% in mineral prediction accuracies.

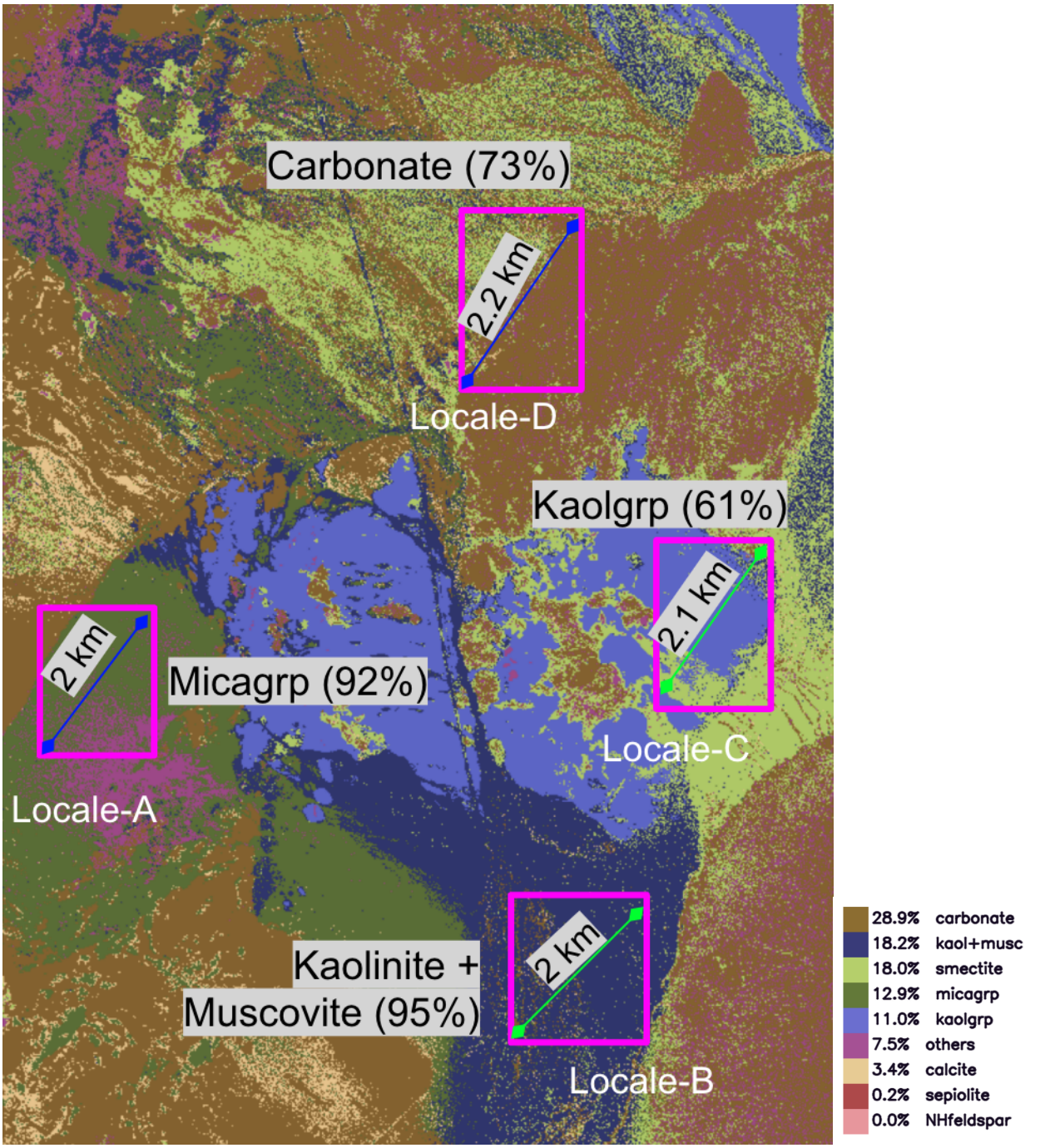


Figure 6.9: False color mineral map of Cuprite Hills with four specific locales utilized for assessment delineated by pink boxes. Each locale spans approximately 2 sq. km in size.

Chapter 7

Conclusions

7.1 Summary

In this research, the simple sensor spectral model, a framework capable of generating hyperspectral maps using only RGB images and a limited number of training measurements, is introduced (Figure 7.1).

This approach comprises a two fundamental steps:

- Compression of the high-dimensional spectra to a low-dimensional latent space.
- Learning the association between the latent space and the data acquired from simple sensors.

Chapter 2 introduced spectral compressors (SC) that utilized variational auto-encoders (VAEs) for the compression of hyperspectral data. A comprehensive evaluation of the spectral compression process was conducted, reviewing various qualitative and quantitative metrics to assess its performance. Different network architectures for spectral compression were explored, with a design incorporating Long Short-Term Memory (LSTM)

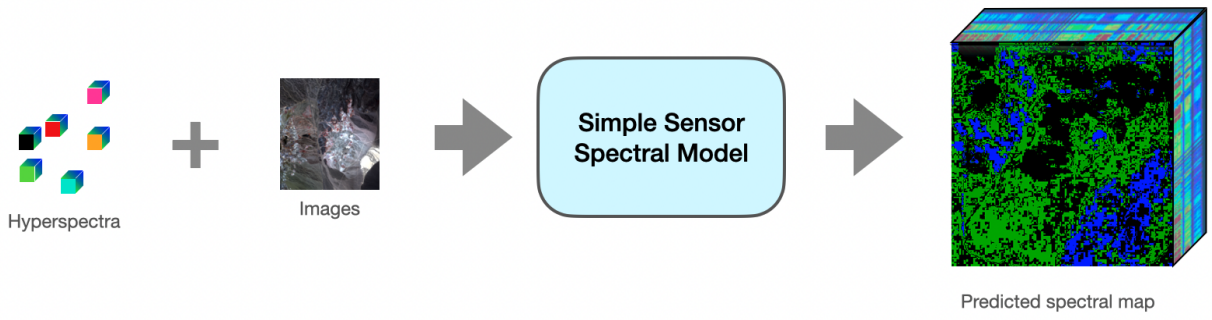


Figure 7.1: A framework to produce hyperspectral maps with just a few hyperspectral measurements and RGB images

modules ultimately being selected due to their efficacy in compression and reconstruction tasks. Additionally, the impact of latent dimensions was briefly investigated, revealing that larger dimensions generally enhance the network's ability to compress and reconstruct hyperspectral signals. However, diminishing returns were observed beyond a certain point, leading to the identification of an optimal latent size of 12 as a balanced compromise.

In Chapter 3, the spectral mapping process was explored, providing an overview of what a spectral map entails and elucidating its construction methodology. Subsequently, the datasets utilized in the experiments were outlined. These datasets were sourced from three mineralogically diverse sites and encompassed both hyperspectral and multispectral data, alongside high-resolution RGB ortho projections obtained via drone imaging. Additionally, mineral maps for each site were introduced, accompanied by details regarding mineral abundances within these regions. Furthermore, three distinct sources for training the spectral compressor were explored, each with unique advantages and limitations. The possibility of training the spectral compressor exclusively on spectra from a spectral library was discussed, and an augmented training set was created with random mixtures of spectra from said library.

In Chapter 4, latent encoders were introduced, a key contribution of our work, which facilitates the use of simple sensors in the spectral mapping process. The generalization of the feature space by the latent encoder was elucidated along with a detailed description of the latent encoder model utilizing Gaussian processes. The response from the latent encoders was analyzed to ascertain its utility, demonstrating that it offers valuable information. Spectral predictions were then assessed using the latent encoder, allowing for qualitative evaluation of prediction accuracy followed by quantitative comparisons. Moreover, it was demonstrated that latent encoders are not confined to RGB data alone and can incorporate additional diagnostic bands to enhance results. Specifically, the multispectral latent encoder was introduced and experimental results showcased its effectiveness in improving spectral predictions across all dronemaps in the field dataset.

In Chapter 5, the concept of the RGB latent encoder was expanded, exploring how the higher spatial resolution of RGB data can be utilized to enhance spectral predictions through texture information. Different texture algorithms sourced from literature, spanning methods in computer vision such as SIFT and ORB to those in remote sensing domains like GLCP and MBP, were experimented with. Through extensive testing of these texture algorithms on the available dronemaps, SIFT and LBP were identified as the best performers. Furthermore, combining the SIFT and LBP feature vectors to form a new texture feature resulted in even better performance.

One of the fundamental applications of spectral mapping is mineral identification. In Chapter 6, the capability of the spectral compressor to accurately predict minerals was first shown, with the reconstructed spectra achieving an impressive accuracy rate of up to 80%. Subsequently, spectral reconstructions generated by multispectral and texture-based latent encoders were explored, achieving similarly high accuracies for mineral

prediction. It was observed that prediction accuracies were significantly influenced by the specific area being mapped and the underlying minerals present in those regions. Additionally, scenarios using solely RGB data were explored, delivering impressive performance, and the generalization of this approach to most real-world situations was discussed.

The outcomes of this research is particularly noteworthy as previous research predominantly concentrated on identifying mineral mixtures (Thompson et al., 2015) or employing basic classifiers to predict minerals based on spectral signals (Candela Garza, 2021). In contrast, our work achieves mineral predictions utilizing an industry gold-standard tool, attaining a commendable level of accuracy. Moreover, the novelty and significance of achieving this accuracy is demonstrated using only simple RGB cameras, marking a substantial contribution to the field.

7.2 Compute Duration/Resources

Training a spectral compressor is computationally intensive task. This process requires training a VAE with several thousand spectra, with each field site contributing approximately half a million hyperspectral samples. The training is conducted on a desktop computer equipped with an Nvidia 1080Ti GPU and an Intel i9 processor with 8 cores. Depending on the complexity of the network architecture, training durations range from 2 to 6 hours. However, inference with the VAE is nearly instantaneous, taking less than a second. The latent encoders utilizes Gaussian Processes (GPs), which are also trained on the GPU using the GPyTorch library [1]. With only 120 points used for training the GPs, this phase completes in under 10 seconds, and inference occurs in less than a second.

In a typical deployment scenario, an aerial vehicle collects overhead imagery of the region mapped from a higher altitude. The aerial imagery, combined with a few hyperspectral measurements from either orbital/aerial instruments or ground-based sources, is used to train a latent encoder. This latent encoder can then generate a spectral map for the region in just a few minutes. The resulting spectral map can be used by ground rovers or aerial vehicles for tasks such as information-based path planning (Candela et al., 2017). Alternatively, mineral maps can be constructed using the spectral maps which could take from a few minutes to hours depending on the spatial resolution of the spectral map.

7.3 Conclusions

Four major conclusions can be drawn from this work. Namely,

[1] GPytorch library - gpytorch.ai

- Hyperspectral data can be effectively compressed and reconstructed to include all the spectral features for effective mineral identification.
- Using simple sensors like RGB cameras, high dimensional spectral data can be synthesized.
- Complimentary information like textures from RGB images can compensate for the limited spectral information.
- Spectral reconstruction accuracy varies with the choice of spectral bands, as some bands are more diagnostic than others.

7.4 Contributions

The following are the major contributions of this research.

- Developed a framework for building spectral maps using camera images.
- Achieved effective compression and reconstruction of hyperspectral data, to sufficient fidelity, to maintain up to 80% accuracy in mineral identification tasks.
- Introduced a generalized spectral compressor that can achieve at least 60% accuracy in mineral identification tasks on any site without needing specific training spectra from that site.
- Enhanced framework accuracy by integrating additional spectral bands as available.
- Leveraged the high resolution of RGB data to improve spectral map accuracy using textures.
- Curated a diverse dataset from field sites including registered hyperspectral, multispectral, and high-resolution RGB drone map data, and demonstrated the efficacy of the framework on this dataset.

7.5 Future Directions

The following directions are proposed in each of these areas to advance this research further.

7.5.1 SC Improvements: Transformers and BERTS

Optimal performance in spectral compression is achieved when recurrent neural networks, particularly those with Long Short-Term Memory (LSTM) modules, are utilized, surpassing simpler architectures such as fully connected networks or deep convolutional networks. However, recent advancements in recurrent neural network technology have introduced new possibilities. Notably, attention-based networks, pioneered by Vaswani et al. (2017), represent a significant breakthrough, leading to the development of Transformer architectures and, more recently, Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) . These advancements present a natural progression from LSTMs.

In the past, a primary challenge was the scarcity of training data for spectral models, compounded by the limited availability of calibrated hyperspectral datasets. However, with our unique approach to training in a self-supervised manner and the availability of high-quality datasets like EMIT (Green et al., 2021), new opportunities arise to leverage these advanced network architectures for even more accurate spectral reconstructions.

7.5.2 Efficacy Improvements: Deep Textures

In this research, a wide variety of texture algorithms from the literature are utilized to extract textures from high-resolution RGB images. While some algorithms that consistently outperformed others are identified, none exhibited consistent performance across all datasets. To address this variability and enhance texture extraction, it would be advantageous to develop a deep learning model or an ensemble of models capable of identifying the optimal texture descriptor for any given scene and mineralogy.

An even more effective approach would involve developing a deep learning model capable of generating custom texture descriptors that are universally effective across diverse scenarios. This approach could significantly improve the accuracy and robustness of texture-based mineral identification in spectral mapping applications.

7.5.3 Experiments: Realistic Exploration Scenarios

Our results are validated on three mineralogically diverse sites. However, in most exploration scenarios, it is common to observe mineralogical homogeneity in large sections, with significant changes in mineralogy occurring infrequently. It would be beneficial to conduct experiments in such realistic exploration scenarios where the true utility of this method becomes apparent. Multiple camera-based systems can be deployed for spectral mapping in larger, benign regions while resource-constrained, ground robots equipped with imaging spectrometers can be reserved for mineralogically diverse regions which require detailed mineralogical analysis. This approach allows for cost-effective exploration and enhances both the coverage and scientific yield of exploration missions.

7.6 Concluding Remarks

This research facilitates the utilization of simple sensors in spectral mapping, marking a significant advancement in the field. It introduces a novel strategy that bridges the gap between low-dimensional and high-dimensional data, expanding their functionality across various applications.

In the realm of remote sensing and planetary exploration, our method stands as a beacon of progress, streamlining spectral mapping processes. Through meticulous experimentation and rigorous analysis, the effectiveness of our approach is demonstrated across diverse terrains. By compressing high-dimensional spectral data and effectively leveraging the latent space, new possibilities for simple sensor-based spectral mapping have been unlocked, overcoming challenges in diverse landscapes from Cuprite Hills to Yellow Cat.

Looking ahead, our method promises unparalleled insights into planetary geology. With the integration of camera-based systems and resource-constrained ground robots equipped with imaging spectrometers, a future is imagined where exploration boundaries are pushed further. In this vision, our method empowers researchers and explorers to unravel the mysteries of our solar system's mineral wealth with unprecedented clarity and efficiency. As the journey of exploration continues, guided by our method, the potential for discovery remains as vast as the cosmos itself.

References

Achieng, J.; Mutua, J.; Mibei, G.; Olaka, L.; and Waswa, A. K., 2017. Mapping of Hydrothermal Minerals Related to Geothermal Activities Using Remote Sensing and GIS: Case Study of Paka Volcano in Kenyan Rift Valley. *International Journal of Geosciences*, 8, 5 (May 2017), 711–725. doi:10.4236/ijg.2017.85039.

Akhtar, N. and Mian, A., 2020. Hyperspectral Recovery from RGB Images using Gaussian Processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42, 1 (Jan. 2020), 100–113. doi:10.1109/TPAMI.2018.2873729.

Aly, H. A. and Sharma, G., 2014. A Regularized Model-Based Optimization Framework for Pan-Sharpening. *IEEE Transactions on Image Processing*, 23, 6 (Jun. 2014), 2596–2608. doi:10.1109/TIP.2014.2316641.

Andrearczyk, V., 2017. Deep learning for texture and dynamic texture analysis.

Arad, B. and Ben-Shahar, O., 2016. Sparse Recovery of Hyperspectral Signal from Natural RGB Images. In *Computer Vision – ECCV 2016*, 19–34. Springer International Publishing, Cham. doi:10.1007/978-3-319-46478-7_2.

Azarang, A.; Manoochehri, H. E.; and Kehtarnavaz, N., 2019. Convolutional Autoencoder-Based Multispectral Image Fusion. *IEEE Access*, 7 (2019), 35673–35683. doi:10.1109/ACCESS.2019.2905511.

Bajpai, S.; Kidwai, N. R.; Singh, H. V.; and Singh, A. K., 2019. Low memory block tree coding for hyperspectral images. *Multimedia Tools and Applications*, 78, 19 (Oct. 2019), 27193–27209. doi:10.1007/s11042-019-07797-6.

Balaram, B.; Canham, T.; Duncan, C.; Grip, H. F.; Johnson, W.; Maki, J.; Quon, A.; Stern, R.; and Zhu, D., 2018. Mars Helicopter Technology Demonstrator. In *2018 AIAA Atmospheric Flight Mechanics Conference*. American Institute of Aeronautics and Astronautics, Kissimmee, Florida. doi:10.2514/6.2018-0023.

Bandara, W. G. C.; Valanarasu, J. M. J.; and Patel, V. M., 2022. Hyperspectral Pansharpening Based on Improved Deep Image Prior and Residual Reconstruction. *IEEE Transactions on Geoscience and Remote Sensing*, 60 (2022), 1–16. doi:10.1109/TGRS.2021.3139292.

Bandos, T. V.; Bruzzone, L.; and Camps-Valls, G., 2009. Classification of Hyperspectral Images With Regularized Linear Discriminant Analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 47, 3 (Mar. 2009), 862–873. doi:10.1109/TGRS.2008.2005729.

Bansal, S., 2018. Language Modelling and Text Generation using LSTMs — Deep Learning for NLP.

Barburiceanu, S.; Terebes, R.; and Meza, S., 2021. 3D Texture Feature Extraction and Classification Using GLCM and LBP-Based Descriptors. *Applied Sciences*, 11, 5 (Jan. 2021), 2332. doi:10.3390/app11052332.

Becker, E.; Pandit, P.; Rangan, S.; and Fletcher, A. K., 2022. Instability and Local Minima in GAN Training with Kernel Discriminators. *Advances in Neural Information Processing Systems*, 35 (Dec. 2022), 20300–20312.

Bell, J. F.; Maki, J. N.; Mehall, G. L.; Ravine, M. A.; Caplinger, M. A.; Bailey, Z. J.; Brylow, S.; Schaffner, J. A.; Kinch, K. M.; Madsen, M. B.; Winhold, A.; Hayes, A. G.; Corlies, P.; Tate, C.; Barrington, M.; Cisneros, E.; Jensen, E.; Paris, K.; Crawford, K.; Rojas, C.; Mehall, L.; Joseph, J.; Proton, J. B.; Cluff, N.; Deen, R. G.; Betts, B.; Cloutis, E.; Coates, A. J.; Colaprete, A.; Edgett, K. S.; Ehlmann, B. L.; Fagents, S.; Grotzinger, J. P.; Hardgrove, C.; Herkenhoff, K. E.; Horgan, B.; Jaumann, R.; Johnson, J. R.; Lemmon, M.; Paar, G.; Caballo-Perucha, M.; Gupta, S.; Traxler, C.; Preusker, F.; Rice, M. S.; Robinson, M. S.; Schmitz, N.; Sullivan, R.; and Wolff, M. J., 2021. The Mars 2020 Perseverance Rover Mast Camera Zoom (Mastcam-Z) Multispectral, Stereoscopic Imaging Investigation. *Space Science Reviews*, 217, 1 (Feb. 2021), 24. doi:10.1007/s11214-020-00755-x.

Binney, J.; Krause, A.; and Sukhatme, G. S., 2010. Informative path planning for an autonomous underwater vehicle. In *2010 IEEE International Conference on Robotics and Automation*, 4791–4796. doi:10.1109/ROBOT.2010.5509714.

Binney, J. and Sukhatme, G. S., 2012. Branch and bound for informative path planning. In *2012 IEEE International Conference on Robotics and Automation*, 2147–2154. doi:10.1109/ICRA.2012.6224902.

Bodrito, T.; Zouaoui, A.; Chanussot, J.; and Mairal, J., 2021. A Trainable Spectral-Spatial Sparse Coding Model for Hyperspectral Image Restoration. In *Advances in Neural Information Processing Systems*, vol. 34, 5430–5442. Curran Associates, Inc.

Candela, A.; Kodgule, S.; Edelson, K.; Vijayarangan, S.; Thompson, D. R.; Noe Dobrea, E.; and Wettergreen, D., 2020. Planetary Rover Exploration Combining Remote and In Situ Measurements for Active Spectroscopic Mapping. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 5986–5993. doi:10.1109/ICRA40945.2020.9196973.

Candela, A.; Thompson, D.; Dobrea, E. N.; and Wettergreen, D., 2017. Planetary robotic exploration driven by science hypotheses for geologic mapping. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3811–3818. doi:10.1109/IROS.2017.8206231.

Candela, A.; Thompson, D. R.; and Wettergreen, D., 2018. AUTOMATIC EXPERIMENTAL DESIGN USING DEEP GENERATIVE MODELS OF ORBITAL DATA. In *Proceedings of International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS '18)*, 8.

Candela Garza, A., 2021. *Bayesian Models for Science-Driven Robotic Exploration*. Ph.D. thesis, Carnegie Mellon University.

Chapman, J. W.; Thompson, D. R.; Helmlinger, M. C.; Bue, B. D.; Green, R. O.; Eastwood, M. L.; Geier, S.; Olson-Duvall, W.; and Lundeen, S. R., 2019. Spectral and Radiometric Calibration of the Next Generation Airborne Visible Infrared Spectrometer (AVIRIS-NG). *Remote Sensing*, 11, 18 (Jan. 2019), 2129. doi:10.3390/rs11182129.

Charou, E.; Stefouli, M.; Dimitrakopoulos, D.; Vasiliou, E.; and Mavrantza, O. D., 2010. Using Remote Sensing to Assess Impact of Mining Activities on Land and Water Resources. *Mine Water and the Environment*, 29, 1 (Apr. 2010), 45–52. doi:10.1007/s10230-010-0098-0.

Chen, Y.; Nasrabadi, N. M.; and Tran, T. D., 2011. Hyperspectral Image Classification Using Dictionary-Based Sparse Representation. *IEEE Transactions on Geoscience and Remote Sensing*, 49, 10 (Oct. 2011), 3973–3985. doi:10.1109/TGRS.2011.2129595.

Choi, M., 2006. A new intensity-hue-saturation fusion approach to image fusion with a tradeoff parameter. *IEEE Transactions on Geoscience and Remote Sensing*, 44, 6 (Jun. 2006), 1672–1682. doi:10.1109/TGRS.2006.869923.

Clark, R. N.; Swayze, G. A.; Livo, K. E.; Brodrick, P.; Dobrea, E. N.; Vijayarangan, S.; Green, R. O.; Wettergreen, D.; Garza, A. C.; Hendrix, A.; García-Pando, C.; Pearson, N.; Lane, M. D.; González-Romero, A.; and Querol, X., 2024. Imaging spectroscopy: Earth and planetary remote sensing with the PSI Tetracorder and expert systems: From Rovers to EMIT and Beyond. *Planetary Science Journal*, (2024).

Clark, R. N.; Swayze, G. A.; Livo, K. E.; Kokaly, R. F.; King, T. V. V.; Dalton, J. B.; Vance, J. S.; Rockwell, B. W.; Hoefen, T.; and McDougal, R. R., 2002. Surface Reflectance Calibration of Terrestrial Imaging Spectroscopy Data: A Tutorial Using AVIRIS. *In Proceedings of the 10th Airborne Earth Science Workshop*, 2 (2002).

Clark, R. N.; Swayze, G. A.; Livo, K. E.; Kokaly, R. F.; Sutley, S. J.; Dalton, J. B.; McDougal, R. R.; and Gent, C. A., 2003. Imaging spectroscopy: Earth and planetary remote sensing with the USGS Tetracorder and expert systems. *Journal of Geophysical Research: Planets*, 108, E12 (2003). doi:10.1029/2002JE001847.

Coggins, James Michael, 1982. *A Framework for Texture Analysis Based on Spatial Filtering*. Ph.D. thesis, [object Object]. doi:10.25335/WAW5-P810.

Conci, A. and Proença, C. B., 1998. A fractal image analysis system for fabric inspection based on a box-counting method. *Computer Networks and ISDN Systems*, 30, 20 (Nov. 1998), 1887–1895. doi:10.1016/S0169-7552(98)00211-6.

Costa, A. F.; Humpire-Mamani, G.; and Traina, A. J. M., 2012. An Efficient Algorithm for Fractal Analysis of Textures. In *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, 39–46. doi:10.1109/SIBGRAPI.2012.15.

Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; and Bharath, A. A., 2018. Generative Adversarial Networks: An Overview. *IEEE Signal Processing Magazine*, 35, 1 (Jan. 2018), 53–65. doi:10.1109/MSP.2017.2765202.

Cross, G. R. and Jain, A. K., 1983. Markov Random Field Texture Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5, 1 (Jan. 1983), 25–39. doi:10.1109/TPAMI.1983.4767341.

Dadrass Javan, F.; Samadzadegan, F.; Mehravar, S.; Toosi, A.; Khatami, R.; and Stein, A., 2021. A review of image fusion techniques for pan-sharpening of high-resolution satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 171 (Jan. 2021), 101–117. doi:10.1016/j.isprsjprs.2020.11.001.

De Sanctis, M. C.; Coradini, A.; Ammannito, E.; Filacchione, G.; Capria, M. T.; Fonte, S.; Magni, G.; Barbis, A.; Bini, A.; Dami, M.; Ficai-Veltroni, I.; and Preti, G., 2012. The VIR Spectrometer. In *The Dawn Mission to Minor Planets 4 Vesta and 1 Ceres* (Eds. C. Russell and C. Raymond), 329–369. Springer, New York, NY. ISBN 978-1-4614-4903-4. doi:10.1007/978-1-4614-4903-4_13.

Delashmit, W.; Missiles, L.; and Manry, M., 2005. Recent Developments in Multilayer Perceptron Neural Networks.

Dennison, P. E.; Halligan, K. Q.; and Roberts, D. A., 2004. A comparison of error metrics and constraints for multiple endmember spectral mixture analysis and spectral angle mapper. *Remote Sensing of Environment*, 93, 3 (Nov. 2004), 359–367. doi:10.1016/j.rse.2004.07.013.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K., 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. doi:10.48550/arXiv.1810.04805.

Dua, Y.; Kumar, V.; and Singh, R. S., 2020. Comprehensive review of hyperspectral image compression algorithms. *Optical Engineering*, 59 (Sep. 2020), 090902. doi:10.1117/1.OE.59.9.090902.

Dua, Y.; Singh, R. S.; Parwani, K.; Lunagariya, S.; and Kumar, V., 2021. Convolution Neural Network based lossy compression of hyperspectral images. *Signal Processing: Image Communication*, 95 (Jul. 2021), 116255. doi:10.1016/j.image.2021.116255.

Duda, K. A. and Abrams, M., 2012. ASTER Satellite Observations for International Disaster Management. *Proceedings of the IEEE*, 100, 10 (Oct. 2012), 2798–2811. doi:10.1109/JPROC.2012.2191929.

Fasbender, D.; Radoux, J.; and Bogaert, P., 2008. Bayesian Data Fusion for Adaptable Image Pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 46, 6 (Jun. 2008), 1847–1857. doi:10.1109/TGRS.2008.917131.

Fekri-Ershad, S., 2019. TEXTURE IMAGE ANALYSIS AND TEXTURE CLASSIFICATION METHODS - A REVIEW. 2 (2019).

Feng, X.-R.; Li, H.-C.; Wang, R.; Du, Q.; Jia, X.; and Plaza, A., 2022. Hyperspectral Unmixing Based on Nonnegative Matrix Factorization: A Comprehensive Review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15 (2022), 4414–4436. doi:10.1109/JSTARS.2022.3175257.

Fouchet, T.; Reess, J.-M.; Montmessin, F.; Hassen-Khodja, R.; Nguyen-Tuong, N.; Humeau, O.; Jacquinod, S.; Lapauw, L.; Parisot, J.; Bonafous, M.; Bernardi, P.; Chapron, F.; Jeanneau, A.; Collin, C.; Zeganadin, D.; Nibert, P.; Abbaki, S.; Montaron, C.; Blanchard, C.; Arslanyan, V.; Achelhi, O.; Colon, C.; Royer, C.; Hamm, V.; Beuzit, M.; Poulet, F.; Pilorget, C.; Mandon, L.; Forni, O.; Cousin, A.; Gasnault, O.; Pilleri, P.; Dubois, B.; Quantin, C.; Beck, P.; Beyssac, O.; Le Mouélic, S.; Johnsson, J. R.; McConnochie, T. H.; Maurice, S.; and Wiens, R. C., 2022. The SuperCam infrared spectrometer for the perseverance rover of the Mars2020 mission. *Icarus*, 373 (Feb. 2022), 114773. doi:10.1016/j.icarus.2021.114773.

Fu, W.; Li, S.; Fang, L.; and Benediktsson, J. A., 2017. Adaptive Spectral–Spatial Compression of Hyperspectral Image With Sparse Representation. *IEEE Transactions on Geoscience and Remote Sensing*, 55, 2 (Feb. 2017), 671–682. doi:10.1109/TGRS.2016.2613848.

Fubara, B. J.; Sedky, M.; and Dyke, D., 2020. RGB to Spectral Reconstruction via Learned Basis Functions and Weights. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 480–481.

Fujisada, H., 1995. Design and performance of ASTER instrument. In *Advanced and Next-Generation Satellites*, vol. 2583, 16–25. SPIE. doi:10.1117/12.228565.

Gandhi, V., 2020. PCA - Beginner's guide to Dimensionality Reduction. https://kaggle.com/code/vipulgandhi/pca-beginner-s-guide-to-dimensionality-reduction.

Gao, B.-C.; Montes, M. J.; Ahmad, Z.; and Davis, C. O., 2000. Atmospheric correction algorithm for hyperspectral remote sensing of ocean color from space. *Applied Optics*, 39, 6 (Feb. 2000), 887–896. doi:10.1364/AO.39.000887.

Ghassemian, H., 2016. A review of remote sensing image fusion methods. *Information Fusion*, 32 (Nov. 2016), 75–89. doi:10.1016/j.inffus.2016.03.003.

Goetz, A. F. H.; Rock, B. N.; and Rowan, L. C., 1983. Remote sensing for exploration; an overview. *Economic Geology*, 78, 4 (Jul. 1983), 573–590. doi:10.2113/gsecongeo.78.4.573.

Green, R. O.; Eastwood, M. L.; Sarture, C. M.; Chrien, T. G.; Aronsson, M.; Chippendale, B. J.; Faust, J. A.; Pavri, B. E.; Chovit, C. J.; Solis, M.; Olah, M. R.; and Williams, O., 1998. Imaging Spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS). *Remote Sensing of Environment*, 65, 3 (Sep. 1998), 227–248. doi:10.1016/S0034-4257(98)00064-9.

Green, R. O.; Thompson, D. R.; and Team, EMIT., 2021. NASA's Earth Surface Mineral Dust Source Investigation: An Earth Venture Imaging Spectrometer Science Mission. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, 119–122. doi:10.1109/IGARSS47720.2021.9554217.

Greenacre, M.; Groenen, P. J. F.; Hastie, T.; D'Enza, A. I.; Markos, A.; and Tuzhilina, E., 2022. Principal component analysis. *Nature Reviews Methods Primers*, 2, 1 (Dec. 2022), 1–21. doi:10.1038/s43586-022-00184-w.

Guan, B.; Zhang, J.; Sethares, W. A.; Kijowski, R.; and Liu, F., 2021. SpecNet: Spectral Domain Convolutional Neural Network. doi:10.48550/arXiv.1905.10915.

Hamlin, L.; Green, R. O.; Mouroulis, P.; Eastwood, M.; Wilson, D.; Dudik, M.; and Paine, C., 2011. Imaging spectrometer science measurements for Terrestrial Ecology: AVIRIS and new developments. In *2011 Aerospace Conference*, 1–7. doi:10.1109/AERO.2011.5747395.

Han, X.-H.; Shi, B.; and Zheng, Y., 2018. Residual HSRCNN: Residual Hyper-Spectral Reconstruction CNN from an RGB Image. In *2018 24th International Conference on Pattern Recognition (ICPR)*, 2664–2669. doi:10.1109/ICPR.2018.8545634.

He, L.; Rao, Y.; Li, J.; Chanussot, J.; Plaza, A.; Zhu, J.; and Li, B., 2019a. Pansharpening via Detail Injection Based Convolutional Neural Networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12, 4 (Apr. 2019), 1188–1204. doi:10.1109/JSTARS.2019.2898574.

He, L.; Zhu, J.; Li, J.; Plaza, A.; Chanussot, J.; and Li, B., 2019b. HyperPNN: Hyperspectral Pansharpening via Spectrally Predictive Convolutional Neural Networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12, 8 (Aug. 2019), 3092–3100. doi:10.1109/JSTARS.2019.2917584.

Heincke, B.; Jackisch, R.; Saartenoja, A.; Salmirinne, H.; Rapp, S.; Zimmermann, R.; Pirttijärvi, M.; Sörensen, E. V.; Gloaguen, R.; Ek, L.; Bergström, J.; Karinen, A.; Salehi, S.; Madriz, Y.; and Middleton, M., 2019. Developing multi-sensor drones for geological mapping and mineral exploration: Setup and first results from the MULSE-DRO project. *GEUS Bulletin*, 43 (Jul. 2019). doi:10.34194/GEUSB-201943-03-02.

Heylen, R.; Parente, M.; and Gader, P., 2014. A Review of Nonlinear Hyperspectral Unmixing Methods. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7, 6 (Jun. 2014), 1844–1868. doi:10.1109/JSTARS.2014.2320576.

Hochreiter, S. and Schmidhuber, J., 1997. Long Short-term Memory. *Neural computation*, 9 (Dec. 1997), 1735–80. doi:10.1162/neco.1997.9.8.1735.

Huang, H.; Shi, G.; He, H.; Duan, Y.; and Luo, F., 2020. Dimensionality Reduction of Hyperspectral Imagery Based on Spatial–Spectral Manifold Learning. *IEEE Transactions on Cybernetics*, 50, 6 (Jun. 2020), 2604–2616. doi:10.1109/TCYB.2019.2905793.

Hubbard, B. E. and Crowley, J. K., 2005. Mineral mapping on the Chilean–Bolivian Altiplano using co-orbital ALI, ASTER and Hyperion imagery: Data dimensionality issues and solutions. *Remote Sensing of Environment*, 99, 1 (Nov. 2005), 173–186. doi:10.1016/j.rse.2005.04.027.

Humeau-Heurtier, A., 2019. Texture Feature Extraction Methods: A Survey. *IEEE Access*, 7 (2019), 8975–9000. doi:10.1109/ACCESS.2018.2890743.

Ihlen, V., 2019. Landsat 8 Data Users Handbook | U.S. Geological Survey. https://www.usgs.gov/media/files/landsat-8-data-users-handbook.

Irons, J. R.; Dwyer, J. L.; and Barsi, J. A., 2012. The next Landsat satellite: The Landsat Data Continuity Mission. *Remote Sensing of Environment*, 122 (Jul. 2012), 11–21. doi:10.1016/j.rse.2011.08.026.

Javan, F. D.; Samadzadegan, F.; and Fathollahi, F., 2018. Spectral and Spatial Quality assessment of IHS and Wavelet Based Pan-sharpening Techniques for High Resolution Satellite Imagery. *Advances in Image and Video Processing*, 6, 2 (Apr. 2018). doi:10.14738/aivp.62.4226.

Jia, Y.; Zheng, Y.; Gu, L.; Subpa-Asa, A.; Lam, A.; Sato, Y.; and Sato, I., 2017. From RGB to Spectrum for Natural Scenes via Manifold-Based Mapping. In *Proceedings of the IEEE International Conference on Computer Vision*, 4705–4713.

Jiang, B.; Wei, D.; Liu, J.; Wang, S.; Cheng, L.; Wang, Z.; and Qu, M., 2020. Automated Classification of Massive Spectra Based on Enhanced Multi-Scale Coded Convolutional Neural Network. *Universe*, 6, 4 (Apr. 2020), 60. doi:10.3390/universe6040060.

Kalra, T., 2023. Texture Analysis with Deep Learning for improved Computer Vision.

Kamano, A.; Morimoto, M.; and Nagura, R., 2001. Multispectral image compression using hierarchical vector quantization. In *IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No.01CH37217)*, vol. 4, 1856–1858 vol.4. doi:10.1109/IGARSS.2001.977095.

Kamarainen, J.-K.; Kyrki, V.; and Kalviainen, H., 2006. Invariance properties of Gabor filter-based features-overview and applications. *IEEE Transactions on Image Processing*, 15, 5 (May 2006), 1088–1099. doi:10.1109/TIP.2005.864174.

Kedia, A. C.; Kapos, B.; Liao, S.; Draper, J.; Eddinger, J.; Updike, C.; and Frazier, A. E., 2021. An Integrated Spectral–Structural Workflow for Invasive Vegetation Mapping in an Arid Region Using Drones. *Drones*, 5, 1 (Mar. 2021), 19. doi:10.3390/drones5010019.

Kingma, D. P. and Welling, M., 2013. Auto-Encoding Variational Bayes. (Dec. 2013).

Kokaly, R. F.; Clark, R. N.; Swayze, G. A.; Livo, K. E.; Hoefen, T. M.; Pearson, N. C.; Wise, R. A.; Benzel, W. M.; Lowers, H. A.; Driscoll, R. L.; and Klein, A. J., 2017. USGS Spectral Library Version 7. USGS Numbered Series 1035, U.S. Geological Survey, Reston, VA. doi:10.3133/ds1035.

Kruse, F., 2002. Comparison of AVIRIS and Hyperion for hyperspectral mineral mapping. (Jan. 2002).

Kruse, F. A.; Lefkoff, A. B.; Boardman, J. W.; Heidebrecht, K. B.; Shapiro, A. T.; Barloon, P. J.; and Goetz, A. F. H., 1993. The spectral image processing system (SIPS)—interactive visualization and analysis of imaging spectrometer data. *Remote Sensing of Environment*, 44, 2 (May 1993), 145–163. doi:10.1016/0034-4257(93)90013-N.

Kumar, S.; Luo, W.; Kantor, G.; and Sycara, K., 2019. Active Learning with Gaussian Processes for High Throughput Phenotyping. doi:10.48550/arXiv.1901.06803.

Li, C.; Ma, Y.; Mei, X.; Liu, C.; and Ma, J., 2016. Hyperspectral Image Classification With Robust Sparse Representation. *IEEE Geoscience and Remote Sensing Letters*, 13, 5 (May 2016), 641–645. doi:10.1109/LGRS.2016.2532380.

Li, J. and Liu, Z., 2019. Multispectral Transforms Using Convolution Neural Networks for Remote Sensing Multispectral Image Compression. *Remote Sensing*, 11, 7 (Jan. 2019), 759. doi:10.3390/rs11070759.

Li, J.; Wu, C.; Song, R.; Li, Y.; and Liu, F., 2020. Adaptive Weighted Attention Network With Camera Spectral Sensitivity Prior for Spectral Reconstruction From RGB Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 462–463.

Li, S. and Milliken, R. E., 2017. Water on the surface of the Moon as seen by the Moon Mineralogy Mapper: Distribution, abundance, and origins. *Science Advances*, 3, 9 (Sep. 2017), e1701471. doi:10.1126/sciadv.1701471.

Liu, X.; Wang, Y.; and Liu, Q., 2018. Psgan: A Generative Adversarial Network for Remote Sensing Image Pan-Sharpening. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, 873–877. IEEE, Athens. doi:10.1109/ICIP.2018.8451049.

Loncan, L.; de Almeida, L. B.; Bioucas-Dias, J. M.; Briottet, X.; Chanussot, J.; Dobigeon, N.; Fabre, S.; Liao, W.; Licciardi, G. A.; Simoes, M.; Tourneret, J.-Y.; Veganzones, M. A.; Vivone, G.; Wei, Q.; and Yokoya, N., 2015. Hyperspectral Pansharpening: A Review. *IEEE Geoscience and Remote Sensing Magazine*, 3, 3 (Sep. 2015), 27–46. doi:10.1109/MGRS.2015.2440094.

Lowe, D. G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60, 2 (Nov. 2004), 91–110. doi:10.1023/B:VISI.0000029664.99615.94.

Lundeen, S. R., 2022. AVIRIS - Airborne Visible / Infrared Imaging Spectrometer - Data Processing. https://aviris.jpl.nasa.gov/aviris/data_facility.html.

Lunga, D.; Prasad, S.; Crawford, M. M.; and Ersoy, O., 2014. Manifold-Learning-Based Feature Extraction for Classification of Hyperspectral Data: A Review of Advances in Manifold Learning. *IEEE Signal Processing Magazine*, 31, 1 (Jan. 2014), 55–66. doi:10.1109/MSP.2013.2279894.

Mallick, J.; Rahman, A.; and Singh, C. K., 2013. Modeling urban heat islands in heterogeneous land surface and its correlation with impervious surface area by using night-time ASTER satellite data in highly urbanizing city, Delhi-India. *Advances in Space Research*, 52, 4 (Aug. 2013), 639–655. doi:10.1016/j.asr.2013.04.025.

Manjunath, B. and Ma, W., 1996. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18, 8 (Aug. 1996), 837–842. doi:10.1109/34.531803.

Mantripragada, K.; Dao, P. D.; He, Y.; and Qureshi, F. Z., 2022. The effects of spectral dimensionality reduction on hyperspectral pixel classification: A case study. *PLOS ONE*, 17, 7 (Jul. 2022), e0269174. doi:10.1371/journal.pone.0269174.

Marĉelja, S., 1980. Mathematical description of the responses of simple cortical cells*. *JOSA*, 70, 11 (Nov. 1980), 1297–1300. doi:10.1364/JOSA.70.001297.

Markham, B. L.; Jenstrom, D.; Masek, J. G.; Dabney, P.; Pedelty, J. A.; Barsi, J. A.; and Montanaro, M., 2016. Landsat 9: Status and plans. In *SPIE Optical Engineering + Applications*, 99720G. San Diego, California, United States. doi:10.1117/12.2238658.

Meier, D., 2012. Analyzing Spectral Cubes. Technical report.

Młynarczyk, A., 2020. Using FieldSpec ASD to create hyperspectral imaging. 236–242. ISBN 978-83-7986-302-0.

Nanni, L.; Lumini, A.; and Brahnam, S., 2010. Local binary patterns variants as texture descriptors for medical image analysis. *Artificial Intelligence in Medicine*, 49, 2 (Jun. 2010), 117–125. doi:10.1016/j.artmed.2010.02.006.

Nath, B.; Ni-Meister, W.; and Özdoğan, M., 2021. Fine-Scale Urban Heat Patterns in New York City Measured by ASTER Satellite—The Role of Complex Spatial Structures. *Remote Sensing*, 13, 19 (Jan. 2021), 3797. doi:10.3390/rs13193797.

Nouri, M.; Nasehi, B.; Goudarzi, M.; and Abdanan Mehdizadeh, S., 2018. Non-destructive Evaluation of Bread Staling Using Gray Level Co-occurrence Matrices. *Food Analytical Methods*, 11, 12 (Dec. 2018), 3391–3395. doi:10.1007/s12161-018-1319-6.

Padwick, C.; Deskevich, M.; Pacifici, F.; and Smallwood, S., 2010. WORLDVIEW-2 PAN-SHARPENING. *San Diego*, (2010).

Pandey, V. and Jha, S. B., 2020. Incorporating Image Gradients as Secondary Input Associated with Input Image to Improve the Performance of the CNN Model. doi: 10.48550/arXiv.2006.04570.

Pauca, V. P.; Piper, J.; and Plemmons, R. J., 2006. Nonnegative matrix factorization for spectral data analysis. *Linear Algebra and its Applications*, 416, 1 (Jul. 2006), 29–47. doi:10.1016/j.laa.2005.06.025.

Peng, H.; Chen, X.; and Zhao, J., 2020. Residual Pixel Attention Network for Spectral Reconstruction From RGB Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 486–487.

Pizzolante, R. and Carpentieri, B., 2016. Multiband and Lossless Compression of Hyperspectral Images. *Algorithms*, 9, 1 (Mar. 2016), 16. doi:10.3390/a9010016.

Popović, M.; Vidal-Calleja, T.; Hitz, G.; Chung, J. J.; Sa, I.; Siegwart, R.; and Nieto, J., 2020. An informative path planning framework for UAV-based terrain monitoring. *Autonomous Robots*, 44, 6 (Jul. 2020), 889–911. doi:10.1007/s10514-020-09903-2.

Portillo, S. K. N.; Parejko, J. K.; Vergara, J. R.; and Connolly, A. J., 2020. Dimensionality Reduction of SDSS Spectra with Variational Autoencoders. *The Astronomical Journal*, 160, 1 (Jun. 2020), 45. doi:10.3847/1538-3881/ab9644.

Pryzant, R. Gaussian Processes. https://cs.stanford.edu/~rpryzant/blog/gp/gp.html.

Quintano, C.; Fernández-Manso, A.; Shimabukuro, Y. E.; and Pereira, G., 2012. Spectral unmixing. *International Journal of Remote Sensing*, 33, 17 (Sep. 2012), 5307–5340. doi:10.1080/01431161.2012.661095.

Ramakrishnan, D. and Bharti, R., 2015. Hyperspectral remote sensing and geological applications. *Current Science*, 108, 5 (2015), 879–891.

Rasmussen, C. E. and Williams, C. K. I., 2006. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, Mass. ISBN 978-0-262-18253-9.

Rebillard, P. and Nguyen, T. P., 1983. An exploitation of coregistered SIR-A, Seasat and Landsat images.

Rolf, E.; Proctor, J.; Carleton, T.; Bolliger, I.; Shankar, V.; Ishihara, M.; Recht, B.; and Hsiang, S., 2021. A generalizable and accessible approach to machine learning with global satellite imagery. *Nature Communications*, 12, 1 (Jul. 2021), 4392. doi:10.1038/s41467-021-24638-z.

Roth, V. and Steinhage, V., 1999. Nonlinear Discriminant Analysis Using Kernel Functions. In *Advances in Neural Information Processing Systems*, vol. 12. MIT Press.

Routray, S.; Ray, A. K.; and Mishra, C., 2017. Analysis of various image feature extraction methods against noisy image: SIFT, SURF and HOG. In *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 1–5. doi:10.1109/ICECCT.2017.8117846.

Royer, C.; Fouchet, T.; Montmessin, F.; and Poulet, F., 2022. The detection of spectral signatures with IRS/SuperCam, Perseverance rover: Instrument performance. In *In Lunar and Planetary Science Conference*, vol. 53, 1840.

Rublee, E.; Rabaud, V.; Konolige, K.; and Bradski, G., 2011. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the IEEE International Conference on Computer Vision*, 2564–2571. doi:10.1109/ICCV.2011.6126544.

Salehinejad, H.; Sankar, S.; Barfett, J.; Colak, E.; and Valaee, S., 2018. Recent Advances in Recurrent Neural Networks. doi:10.48550/arXiv.1801.01078.

Scarpa, G.; Vitale, S.; and Cozzolino, D., 2018. Target-Adaptive CNN-Based Pansharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 56, 9 (Sep. 2018), 5443–5457. doi:10.1109/TGRS.2018.2817393.

Scheeline, A., 2017. How to Design a Spectrometer. *Applied Spectroscopy*, 71, 10 (Oct. 2017), 2237–2252. doi:10.1177/0003702817720468.

Seelos, F. P.; Seelos, K. D.; Murchie, S. L.; Novak, M. A. M.; Hash, C. D.; Morgan, M. F.; Arvidson, R. E.; Aiello, J.; Bibring, J.-P.; Bishop, J. L.; Boldt, J. D.; Boyd, A. R.; Buczkowski, D. L.; Chen, P. Y.; Clancy, R. T.; Ehlmann, B. L.; Frizzell, K.; Hancock, K. M.; Hayes, J. R.; Heffernan, K. J.; Humm, D. C.; Itoh, Y.; Ju, M.; Kochte, M. C.; Malaret, E.; McGovern, J. A.; McGuire, P.; Mehta, N. L.; Moreland, E. L.;

Mustard, J. F.; Nair, A. H.; Núñez, J. I.; O'Sullivan, J. A.; Packer, L. L.; Poffenbarger, R. T.; Poulet, F.; Romeo, G.; Santo, A. G.; Smith, M. D.; Stephens, D. C.; Toigo, A. D.; Viviano, C. E.; and Wolff, M. J., 2023. The CRISM investigation in Mars orbit: Overview, history, and delivered data products. *Icarus*, (May 2023), 115612. doi:10.1016/j.icarus.2023.115612.

Sharma, A., 2021. Variational Autoencoder in TensorFlow. https://learnopencv.com/variational-autoencoder-in-tensorflow/.

Shekhar, R. and Jawahar, C., 2012. Word Image Retrieval Using Bag of Visual Words. In *2012 10th IAPR International Workshop on Document Analysis Systems*, 297–301. doi:10.1109/DAS.2012.96.

Shen, H.; Jiang, Z.; and Pan, W. D., 2018. Efficient Lossless Compression of Multitemporal Hyperspectral Image Data. *Journal of Imaging*, 4, 12 (Dec. 2018), 142. doi:10.3390/jimaging4120142.

SILIOS, 2024. LARGE SPATIAL RESOLUTION SNAPSHOT MULTISPECTRAL CAMERA FOR YOUR SWIR APPLICATIONS. (2024).

SpectralDevices, 2024. 16-Band SWIR Camera. https://spectraldevices.com/products/16-band-swir-camera.

Stiebel, T.; Koppers, S.; Seltsam, P.; and Merhof, D., 2018. Reconstructing Spectral Images From RGB-Images Using a Convolutional Neural Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 948–953.

Swayze, G.; Clark, R. N.; Kruse, F.; Sutley, S.; and Gallagher, A., 1992. Ground-truthing AVIRIS mineral mapping at Cuprite, Nevada.

Swayze, G. A.; Clark, R. N.; Goetz, A. F.; Livo, K. E.; Breit, G. N.; Kruse, F. A.; Sutley, S. J.; Snee, L. W.; Lowers, H. A.; Post, J. L.; Stoffregen, R. E.; and Ashley, R. P., 2014. Mapping advanced argillic alteration at Cuprite, Nevada, using imaging spectroscopy. *Economic Geology*, 109, 5 (2014), 11791221. doi:10.2113/econgeo.109.5.1179.

TELOPS, 2024. Hyper-Cam Mini xLW. Technical report.

Tharwat, A.; Gaber, T.; Ibrahim, A.; and Hassanien, A. E., 2017. Linear discriminant analysis: A detailed tutorial. *AI Communications*, 30, 2 (Jan. 2017), 169–190. doi:10.3233/AIC-170729.

Thompson, D., 2008. *Intelligent Mapping for Autonomous Robotic Survey*. Ph.D. thesis.

Thompson, D.; Wettergreen, D.; Foil, G.; Furlong, M.; and Kiran, A., 2015. Spatio-Spectral Exploration Combining In Situ and Remote Measurements. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29, 1 (Mar. 2015). doi:10.1609/aaai.v29i1.9673.

Thompson, D. R.; Candela, A.; Wettergreen, D. S.; Dobrea, E. N.; Swayze, G. A.; Clark, R. N.; and Greenberger, R., 2018a. Spatial Spectroscopic Models for Remote Exploration. *Astrobiology*, 18, 7 (Jul. 2018), 934–954. doi:10.1089/ast.2017.1782.

Thompson, D. R.; Candela, A.; Wettergreen, D. S.; Dobrea, E. N.; Swayze, G. A.; Clark, R. N.; and Greenberger, R., 2018b. Spatial Spectroscopic Models for Remote Exploration. *Astrobiology*, 18, 7 (Jul. 2018), 934–954. doi:10.1089/ast.2017.1782.

Tu, T.-M.; Huang, P.; Hung, C.-L.; and Chang, C.-P., 2004. A fast intensity-hue-saturation fusion technique with spectral adjustment for IKONOS imagery. *IEEE Geoscience and Remote Sensing Letters*, 1, 4 (Oct. 2004), 309–312. doi:10.1109/LGRS.2004.834804.

Tyagi, D., 2020. Introduction to SURF (Speeded-Up Robust Features).

Vácha, P. and Haindl, M., 2024. Texture recognition under scale and illumination variations. *Journal of Information and Telecommunication*, 8, 1 (Jan. 2024), 130–148. doi:10.1080/24751839.2023.2265190.

Vane, G.; Green, R. O.; Chrien, T. G.; Enmark, H. T.; Hansen, E. G.; and Porter, W. M., 1993. The airborne visible/infrared imaging spectrometer (AVIRIS). *Remote Sensing of Environment*, 44, 2 (May 1993), 127–143. doi:10.1016/0034-4257(93)90012-M.

Vargas, E.; Arguello, H.; and Tourneret, J.-Y., 2019. Spectral Image Fusion From Compressive Measurements Using Spectral Unmixing and a Sparse Representation of Abundance Maps. *IEEE Transactions on Geoscience and Remote Sensing*, 57, 7 (Jul. 2019), 5043–5053. doi:10.1109/TGRS.2019.2895822.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; ukasz Kaiser, Ł.; and Polosukhin, I., 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc.

Vijayarangan, S.; Kohanbash, D.; Foil, G.; Zacny, K.; Cabrol, N.; and Wettergreen, D., 2018. Field Experiments in Robotic Subsurface Science with Long Duration Autonomy. In *Field and Service Robotics*, Springer Proceedings in Advanced Robotics, 515–529. Springer International Publishing, Cham. doi:10.1007/978-3-319-67361-5_33.

Vivone, G.; Dalla Mura, M.; Garzelli, A.; Restaino, R.; Scarpa, G.; Ulfarsson, M. O.; Alparone, L.; and Chanussot, J., 2021. A New Benchmark Based on Recent Advances in Multispectral Pansharpening: Revisiting Pansharpening With Classical and Emerging Pansharpening Methods. *IEEE Geoscience and Remote Sensing Magazine*, 9, 1 (Mar. 2021), 53–81. doi:10.1109/MGRS.2020.3019315.

Wang, X.; Tao, J.; Shen, Y.; Qin, M.; and Song, C., 2018. Distributed Source Coding of Hyperspectral Images Based on Three-Dimensional Wavelet. *Journal of the Indian Society of Remote Sensing*, 46, 4 (Apr. 2018), 667–673. doi:10.1007/s12524-017-0735-1.

Webster, K. and Sheppard, J., 2022. Robust Spectral Based Compression of Hyperspectral Images using LSTM Autoencoders. In *2022 International Joint Conference on Neural Networks (IJCNN)*, 01–08. IEEE, Padua, Italy. doi:10.1109/IJCNN55064.2022.9892532.

Wei, Y. and Yuan, Q., 2017. Deep residual learning for remote sensed imagery pansharpening. In *2017 International Workshop on Remote Sensing with Intelligent Processing (RSIP)*, 1–4. doi:10.1109/RSIP.2017.7958794.

Wenbin, W.; Wu, Y.; and Li, J., 2018. The Hyper-spectral Image Compression Based on K-Means Clustering and Parallel Prediction Algorithm*. *MATEC Web of Conferences*, 173 (2018), 03071. doi:10.1051/matecconf/201817303071.

Whiting, M. L.; Ustin, S. L.; Zarco-Tejada, P.; Palacios-Orueta, A.; and Vanderbilt, V. C., 2006. Hyperspectral mapping of crop and soils for precision agriculture. In *Remote Sensing and Modeling of Ecosystems for Sustainability III*, vol. 6298, 84–98. SPIE. doi:10.1117/12.681289.

Williams, D. L.; Goward, S.; and Arvidson, T., 2006. Landsat. *Photogrammetric Engineering & Remote Sensing*, 72, 10 (Oct. 2006), 1171–1178. doi:10.14358/PERS.72.10.1171.

Wulder, M. A.; Roy, D. P.; Radeloff, V. C.; Loveland, T. R.; Anderson, M. C.; Johnson, D. M.; Healey, S.; Zhu, Z.; Scambos, T. A.; Pahlevan, N.; Hansen, M.; Gorelick, N.; Crawford, C. J.; Masek, J. G.; Hermosilla, T.; White, J. C.; Belward, A. S.; Schaaf, C.; Woodcock, C. E.; Huntington, J. L.; Lymburner, L.; Hostert, P.; Gao, F.; Lyapustin, A.; Pekel, J.-F.; Strobl, P.; and Cook, B. D., 2022. Fifty years of Landsat science and impacts. *Remote Sensing of Environment*, 280 (Oct. 2022), 113195. doi:10.1016/j.rse.2022.113195.

Xie, W.; Lei, J.; Cui, Y.; Li, Y.; and Du, Q., 2020. Hyperspectral Pansharpening With Deep Priors. *IEEE Transactions on Neural Networks and Learning Systems*, 31, 5 (May 2020), 1529–1543. doi:10.1109/TNNLS.2019.2920857.

XIMEA, 2024. XIMEA - Hyperspectral Linescan USB3 camera 150 bands 470-900nm. https://www.ximea.com/en/products/hyperspectral-cameras-based-on-usb3-xispec/mq022hg-im-ls150-visnir.

Xiong, Z.; Shi, Z.; Li, H.; Wang, L.; Liu, D.; and Wu, F., 2017. HSCNN: CNN-Based Hyperspectral Image Recovery From Spectrally Undersampled Projections. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 518–525.

Xu, P.; Chen, B.; Xue, L.; Zhang, J.; and Zhu, L., 2018. A Prediction-Based Spatial-Spectral Adaptive Hyperspectral Compressive Sensing Algorithm. *Sensors*, 18, 10 (Oct. 2018), 3289. doi:10.3390/s18103289.

Xu, Q.; Zhang, Y.; and Li, B., 2014. Recent advances in pansharpening and key problems in applications. *International Journal of Image and Data Fusion*, 5, 3 (Jul. 2014), 175–195. doi:10.1080/19479832.2014.889227.

Xu, T.; Moore, I. D.; and Gallant, J. C., 1993. Fractals, fractal dimensions and landscapes — a review. *Geomorphology*, 8, 4 (Dec. 1993), 245–262. doi:10.1016/0169-555X(93)90022-T.

Yan, Y.; Zhang, L.; Li, J.; Wei, W.; and Zhang, Y., 2018. Accurate Spectral Super-Resolution from Single RGB Image Using Multi-scale CNN. In *Pattern Recognition and Computer Vision*, 206–217. Springer International Publishing, Cham. doi:10.1007/978-3-030-03335-4_18.

Yang, X.; Jian, L.; Yan, B.; Liu, K.; Zhang, L.; and Liu, Y., 2018. A sparse representation based pansharpening method. *Future Generation Computer Systems*, 88 (Nov. 2018), 385–399. doi:10.1016/j.future.2018.04.096.

Yuan, X.; Jiang, H.; Huang, G.; and Wilford, P. A., 2015. Compressive Sensing via Low-Rank Gaussian Mixture Models. doi:10.48550/arXiv.1508.06901.

Zhang, J.; Su, R.; Fu, Q.; Ren, W.; Heide, F.; and Nie, Y., 2022. A survey on computational spectral reconstruction methods from RGB to hyperspectral imaging. *Scientific Reports*, 12, 1 (Jul. 2022), 11905. doi:10.1038/s41598-022-16223-1.

Zhang, L.; Chu, R.; Xiang, S.; Liao, S.; and Li, S. Z., 2007. Face Detection Based on Multi-Block LBP Representation. In *Advances in Biometrics*, 11–18. Springer, Berlin, Heidelberg. doi:10.1007/978-3-540-74549-5_2.

Zhao, Y.; Po, L.-M.; Yan, Q.; Liu, W.; and Lin, T., 2020. Hierarchical Regression Network for Spectral Reconstruction from RGB Images. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1695–1704. IEEE, Seattle, WA, USA. doi:10.1109/CVPRW50498.2020.00219.

Zhou, J.; Civco, D. L.; and Silander, J. A., 1998. A wavelet transform method to merge Landsat TM and SPOT panchromatic data. *International Journal of Remote Sensing*, 19, 4 (Jan. 1998), 743–757. doi:10.1080/014311698215973.

Zhu, X. X. and Bamler, R., 2013. A Sparse Image Fusion Algorithm With Application to Pan-Sharpening. *IEEE Transactions on Geoscience and Remote Sensing*, 51, 5 (May 2013), 2827–2836. doi:10.1109/TGRS.2012.2213604.