

Differentiable Programming for Hyperspectral Unmixing using a Physics-based Dispersion Model

John Janiczek¹, Parth Thaker¹, Gautam Dasarathy¹, Christopher Edwards², Philip Christensen¹, and Suren Jayasuriya¹

¹ Arizona State University

² Northern Arizona University

Abstract. Hyperspectral unmixing is an important remote sensing task with applications including material identification and analysis. Characteristic spectral features make many pure materials identifiable from their visible-to-infrared spectra, but quantifying their presence within a mixture is a challenging task due to nonlinearities and factors of variation. In this paper, spectral variation is considered from a physics-based approach and incorporated into an end-to-end spectral unmixing algorithm via differentiable programming. The dispersion model is introduced to simulate realistic spectral variation, and an efficient method to fit the parameters is presented. Then, this dispersion model is utilized as a generative model within an analysis-by-synthesis spectral unmixing algorithm. Further, a technique for inverse rendering using a convolutional neural network to predict parameters of the generative model is introduced to enhance performance and speed when training data is available. Results achieve state-of-the-art on both infrared and visible-to-near-infrared (VNIR) datasets, and show promise for the synergy between physics-based models and deep learning in hyperspectral unmixing in the future.

Keywords: hyperspectral imaging, spectral unmixing, differentiable programming

1 Introduction

Hyperspectral imaging is a method of imaging where light radiance is densely sampled at multiple wavelengths. Increasing spectral resolution beyond a traditional camera’s red, green, and blue spectral bands typically requires more expensive detectors, optics, and/or lowered spatial resolution. However, hyperspectral imaging has demonstrated its utility in computer vision, biomedical imaging, and remote sensing [38, 8]. In particular, spectral information is critically important for understanding material reflectance and emission properties, important for recognizing materials.

Spectral unmixing is a specific task within hyperspectral imaging with application to many land classification problems related to ecology, hydrology, and

mineralogy [25, 28]. It is particularly useful for analyzing aerial images from aircraft or spacecraft to map the abundance of materials in a region of interest. While pure materials have characteristic spectral features, mixtures require algorithms to identify and quantify material presence.

A common model for this problem is linear mixing, which assumes electromagnetic waves produced from pure materials combine linearly and are scaled by the material abundance. Mathematically this is expressed as $\mathbf{b} = \mathbf{A}\mathbf{x} + \boldsymbol{\eta}$ where \mathbf{b} is the observed spectra, \mathbf{A} is a matrix whose columns are the pure material spectra, $\boldsymbol{\eta}$ is the measurement noise, and \mathbf{x} is the abundance of each pure material. The model assumes that the pure material spectra, referred to as endmember spectra, is known before-hand. Nonlinear effects are known to occur when photons interact with multiple materials within a scene for which we refer readers to the review by Heylen et al. for techniques to account for the non-linear mixing [25].

A key challenge that affects both linear and nonlinear mixing models is that pure materials have an inherent variability in their spectral signatures, and thus cannot be represented by a single characteristic spectrum. Spectral variability of endmembers is caused by subtle absorption band differences due to factors such as different grain sizes [39, 48, 43, 44] or differing ratios of molecular bonds [7, 51] as shown in Figure 1. Since variability causes significant errors in unmixing algorithms, it is an active area of research [60, 16, 65].

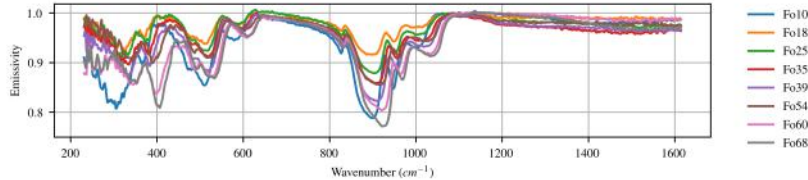


Fig. 1. Endmember Variation: Several spectra of olivine are plotted to demonstrate its spectral variability. The olivine mineral is a solid solution with continuous compositional variation of Iron (Fe_2) and Magnesium (Mg_2) bonds. This ratio of bonds (indexed by the Fo number), causes absorption bands to shift in frequency and strength

Recently, differentiable programming has become a popular research area due to its potential to bridge gaps between physics-based and machine learning-based techniques for computer vision and graphics [53, 21, 1]. Our key insight is to leverage differentiable programming by modelling the variation of spectra with a physics-based dispersion model, and incorporating this differentiable model into an end-to-end spectral unmixing algorithm. Such an approach has the capacity to unmix scenes with a large amount of variability, while constraining the predictions to be physically plausible. These physically plausible variations of endmember spectra also provide additional science data as the variation of absorption bands can reveal properties about the composition and history of the

material. To our knowledge, we are the first to use a generative physics model to account for spectral variability in an unmixing algorithm.

Contributions: Our specific contributions in this paper are the following:

- We introduce a physics-based dispersion model (first presented in [49, 30, 55]) to generate and render spectral variation for various pure materials. We provide an efficient optimization method via gradient descent to find dispersion model parameters for this spectral variation.
- We incorporate this dispersion model into an end-to-end spectral unmixing algorithm utilizing differentiable programming to perform analysis-by-synthesis optimization. Analysis-by-synthesis is solved via alternating minimization optimization and requires no training data.
- We further design an inverse rendering algorithm consisting of a convolutional neural network to jointly estimate dispersion model parameters and mineral abundances for spectral unmixing. This method requires training data, but is computationally efficient at test time and outperforms analysis-by-synthesis and other state-of-the-art methods.

We provide extensive analysis of our proposed methods with respect to noise and convergence criteria. To validate our contributions, we test on both synthetic and real datasets using hyperspectral observations in the visible and near infrared (VNIR), and mid to far infrared (IR). The datasets also span three different environments from laboratory, aircraft, and satellite based spectrometers. Our methods achieve state-of-the-art across all datasets, and we compare against several baselines from literature. Our code is openly available and accessible here: <https://github.com/johnjaniczek/InfraRender>. We hope this work inspires more fusion between physics models and machine learning for hyperspectral imaging and computer vision more generally in the future.

2 Related Work

Optimization-based Approaches. Standard optimization techniques for linear unmixing include projection, non-negative least squares, weighted least squares, and interior point methods [43, 24, 47, 12, 23]. Further, sparsity-based optimization can improve abundance prediction [63, 9]. However, most optimization have not leveraged physics priors as we do in our model.

Spectral Variability. Spectral variability has been a topic of recent interest [60, 6]. One approach is to augment \mathbf{A} with multiple variations or spectra for each endmember. To do this, multiple endmember spectral mixture analysis (MESMA) [46] and multiple-endmember linear spectral unmixing (MELSUM) [15] both require labeled data of the spectral variation for each endmember. In contrast, unsupervised techniques learn endmember sets from unlabelled hyperspectral images, including semi-automated techniques [3], k-means clustering [3], and the sparsity promoting iterated constrained endmember algorithm (SPICE) [58, 59] which simultaneously finds endmember sets while unmixing

for material abundances. These techniques are limited by the amount of sets in the endmember library, and computational complexity increases with more additions. Our method by contrast finds an efficient parameter set to physically model the spectral variation.

Another category of endmember variability techniques models the endmember spectral variation as samples from a multivariate distribution $\mathbf{P}(\mathbf{e}|\theta)$ where \mathbf{e} is the endmember spectra, and θ are the distribution parameters. Common statistical distributions proposed include the normal compositional model [50], Gaussian mixture models [65], and the beta compositional model [16]. These distribution models have large capacity to model spectral variations, however sometimes they can render endmember spectra that are not physically realistic.

Deep Learning for Hyperspectral Classification and Unmixing. Deep learning has recently improved many hyperspectral imaging tasks [62, 33]. In particular, networks process hyperspectral pixel vectors using both deep belief networks [36] and CNNs [26, 10, 56]. For spatial hyperspectral data, CNNs [11], joint spectral-spatial feature extraction [64], and 3D CNNs [35] are used. All these methods require large hyperspectral datasets that are annotated correctly. One of our methods uses analysis-by-synthesis and differentiable programming to avoid low training data issues, but our technique can be made complementary to deep learning architectures as we show in our inverse rendering CNN.

Differentiable Programming and Rendering. Differentiable programming refers to the paradigm of writing algorithms which can be fully differentiated end-to-end using automatic differentiation for any parameter [53, 54, 4]. This has been applied for audio [18] and 3D geometry processing [45]. In graphics, differentiable rendering has improved ray tracing [34, 40, 37, 61], solved analysis-by-synthesis problems in volumetric scattering [22, 21], estimated reflectance and lighting [1], and performed 3D reconstruction [52]. In our paper, we write a forward imaging model utilizing the physics of dispersion in spectral variation to allow our pipeline to be differentiable end-to-end.

3 Method

Our approach to hyperspectral unmixing features two main components: (1) use of a physically-accurate dispersion model for pure endmember spectra, and (2) a differentiable programming pipeline to perform spectral unmixing. This approach has synergistic benefits of leveraging prior domain knowledge while learning from data. Our first algorithm solves spectral unmixing in a self-supervised fashion using analysis-by-synthesis optimization with the dispersion model as the synthesis step. Further, we show how inverse rendering via a convolutional neural network (CNN) can learn parameters of this model to help speed up our end-to-end pipeline and improves performance when training data is available.

3.1 Dispersion Model

We first describe the dispersion model for generating endmember spectra. **End-member and/or endmember spectra** is what we call the spectral curve for

emissivity ϵ as a function of wavenumber ω . Each pure material has a characteristic endmember spectrum, although spectra can vary, which is the problem we are trying to solve/disambiguate. Let $\epsilon^{\text{measured}}(\omega)$ be endmember spectra we have measured, typically in a lab or in the field, whose emissivity is sampled at different wavenumbers: $[\epsilon^{\text{measured}}(\omega_1), \dots, \epsilon^{\text{measured}}(\omega_N)]^T$. Our goal is to propose a model $\epsilon^{\text{model}}(\mathbf{A}; \omega)$ with parameters \mathbf{A} such that the following loss is minimized: $L(\mathbf{A}) = \sum_{i=1}^N (\epsilon^{\text{measured}}(\omega_i) - \epsilon^{\text{model}}(\mathbf{A}; \omega_i))^2$. That is, we fit the model emissivity of an endmember spectrum to the measured spectrum. In practice, we need to add regularization and constraints to this endmember loss for better fitting which we describe after the derivation of the dispersion model.

Derivation of the Dispersion model: Our model of endmember spectra is derived from an atomistic oscillator driven by electromagnetic waves impinging on the molecular structure of the pure material [49, 30]. In Figure 2, we show a conceptual diagram of this model, and how it generates emissivity curves as a function of wavelength. For the full derivation of the model from first principles, we refer the reader to Appendix A in the supplementary material. Instead, we outline the model below based on the equations derived from that analysis.

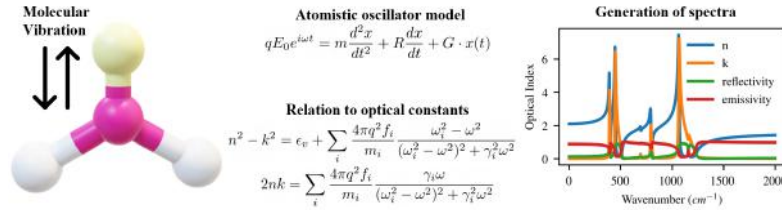


Fig. 2. Dispersion Model Concept Figure: The insight of the dispersion model is that optical properties can be related to molecular structure through first principles via an atomistic oscillator model. We use this generative model for the formation of spectral variation in our spectral unmixing algorithm

Let $\mathbf{A} = [\boldsymbol{\rho}, \boldsymbol{\omega}_o, \boldsymbol{\gamma}, \boldsymbol{\epsilon}_r]$ be a matrix of parameters, where $\boldsymbol{\rho}, \boldsymbol{\omega}_o, \boldsymbol{\gamma}, \boldsymbol{\epsilon}_r \in \mathbb{R}^K$ and K is a model hyperparameter corresponding to the number of distinct mass-spring equations used to model the emissivity. $\boldsymbol{\rho}$ is the band strength, $\boldsymbol{\omega}_o$ is the resonant frequency, $\boldsymbol{\gamma}$ is the frictional force (dampening coefficient), and $\boldsymbol{\epsilon}_r$ is relative dielectric permeability. Please see the supplemental material for the physical significance of these parameters to the atomistic oscillator model, and their control over the shape of spectral absorption bands. Note: usually $\boldsymbol{\epsilon}_r$ is a constant vector which does not vary with K . Thus $\mathbf{A} \in \mathbb{R}^{K \times 4}$. The refractive index terms n, k are given as follows [49, 30]:

$$n(\mathbf{A}; \omega) = \sqrt{\frac{\theta + b}{2}}, \quad k(\mathbf{A}; \omega) = \frac{\phi}{n(\mathbf{A}; \omega)}, \quad (1)$$

where the expressions for θ, b, ϕ are given as follows:

$$\theta = \epsilon_r + \sum_{k=1}^K 4\pi\rho_k\omega_0^2 \frac{(\omega_0^2 - \omega^2)}{(\omega_0^2 - \omega^2)^2 + \gamma_k^2\omega_0^2\omega^2}, \quad (2)$$

$$b = \sqrt{\theta^2 + 4\phi^2}, \quad \phi = \sum_{k=1}^K 2\pi\rho_k\omega_0^2 \frac{\gamma_k\omega_0\omega}{(\omega_0^2 - \omega^2)^2 + \gamma_k^2\omega_0^2\omega^2}. \quad (3)$$

We note that subscript k denotes the k -th coordinate of the corresponding vector. Also there is another useful relation (derived in Appendix A) that $n^2 - k^2 = \theta, nk = \phi$. We then define the complex refractive index as $\hat{n}(\mathbf{A}; \omega) = n(\mathbf{A}; \omega) - i \cdot k(\mathbf{A}; \omega)$, where $i = \sqrt{-1}$ is the imaginary number. Hence, we can calculate the emissivity as follows:

$$\epsilon(\mathbf{A}; \omega) = 1 - R(\mathbf{A}; \omega), \quad \text{where } R(\mathbf{A}; \omega) = \left| \frac{\hat{n}(\mathbf{A}; \omega) - 1}{\hat{n}(\mathbf{A}; \omega) + 1} \right|^2. \quad (4)$$

When considering minerals, we introduce $M \in \mathbb{N}$, the number of optical axes of symmetry in crystal structures, (eg. 2 axes of symmetry in quartz [49, 55]), to define the full model:

$$\epsilon^{\text{model}}(\mathbf{A}; \omega) = \sum_{m=1}^M \alpha_m \cdot \epsilon(\mathbf{A}_m; \omega) \quad \text{such that } \sum_{m=1}^M \alpha_m = 1, \alpha_m \geq 0, \quad (5)$$

where we use a different parameter matrix \mathbf{A}_m and weight α_m for each optical axis of symmetry.

The dispersion model has been primarily used to analyze optical properties of materials to determine n and k , which then can be subsequently applied to optical models like radiative transfer [49, 55]. After n and k are found, spectra such as reflectance, emissivity, and transmissivity can be generated. In particular, we notice that fine-grained control of the dispersion model parameters can realistically render spectral variation that occurs in hyperspectral data. Our contribution is to leverage these properties in a differentiable programming pipeline for spectral unmixing.

Endmember Fitting. Using the dispersion model presented above, we want to robustly estimate the model parameters to fit the spectra $\epsilon^{\text{measured}}$ captured in a lab or in the field. To fit the model, we wish to perform gradient descent to efficiently find these parameters. Using chain rule on the loss function, we see that $\frac{\partial L}{\partial \mathbf{A}_{i,j}} = \frac{\partial L}{\partial \epsilon^{\text{model}}} \frac{\partial \epsilon^{\text{model}}}{\partial \mathbf{A}_{i,j}}$, where (ij) corresponds to that element of the parameter matrix, and all expressions are scalars once the coordinate is specified. While the partial derivatives can be calculated explicitly via symbolic toolboxes, the resulting expressions are too long to be presented here. For simplicity and ease of use, we use the autograd function [4, 41] in PyTorch [42] to automatically compute derivatives for our model as we are performing backpropagation.

One main challenge in performing endmember fitting is that the dispersion model is not an injective function, and hence is typically not identifiable, that

is more than one \mathbf{A} can result in the same fit. This can be solved, in-part, through regularization to enforce sparsity, especially since a preference for fewer dispersion parameters has been suggested in the literature [49, 55]. In our implementation, we initialize our model with $K = 50$ rows of the parameter matrix. Since the parameter ρ controls the strength of the absorption band, small values of ρ do not contribute much energy to the spectra (unnecessary absorption bands), and can be pruned. After performing sparse regression by penalizing the L_1 norm of ρ , K is typically around 10-15 in our experiments.

Thus, our modified sparse regression problem may be written as

$$\arg \min_{\mathbf{A}_{\min} \leq \mathbf{A} \leq \mathbf{A}_{\max}} \sum_{i=1}^N (\epsilon^{real}(\omega_i) - \epsilon^{model}(\mathbf{A}; \omega_i))^2 + \lambda_\rho \|\rho\|_1, \quad (6)$$

where \mathbf{A}_{\min} and \mathbf{A}_{\max} restrict the variation of the dispersion parameters to a plausible range. In addition, endmembers (particularly minerals) can have multiple optical axes of symmetry described by separate spectra, which has been noted in the literature [49, 55]. Without prior knowledge of the number of axes for every material we encounter, we run this optimization for a single and double axes, and pick the one with the lowest error. See Section 4 for results on endmember fitting and Figure 6 for examples of modelled vs. measured spectra.

Despite the fact that this regression problem is non-convex, we solve it using gradient descent with a random initialization; this is known to converge to a local minimum with probability 1 [31]. A global minimum is not necessary at this stage, since we use endmember fitting to provide a good initialization point for the subsequent alternating minimization procedure introduced in the next subsection.

3.2 Differentiable Programming for End-to-End Spectral Unmixing

Analysis-by-Synthesis Optimization. In Figure 3, we show our full end-to-end spectral unmixing pipeline. Here, $\epsilon^{model}(\mathbf{A}; \omega)$, which is initially fit to $\epsilon^{measured}$, is then aggregated into the columns of \mathbf{A} . Then, the observed spectra \mathbf{b} is linearly unmixed by solving a regularized least-squares optimization: $\arg \min_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_p$ subject to sum-to-one and non-negativity constraints $\|\mathbf{x}\|_1 = 1, \mathbf{x} \geq 0$. Given these constraints, one cannot impose sparsity with the usual L_1 norm. Instead, we use the L_p norm to induce sparsity for the predicted abundances; this has been proposed before for spectral unmixing [9].

The key to our pipeline is that everything is fully differentiable, and thus we can actually minimize the following equation:

$$\arg \min_{\mathbf{x}, \mathbf{A} \in [\mathbf{A}_{\min}, \mathbf{A}_{\max}]} \|\mathbf{b} - \mathbf{A}(\mathbf{A})\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_p \text{ such that } \|\mathbf{x}\|_1 = 1, \mathbf{x} \geq 0. \quad (7)$$

with respect to both the parameters of the dispersion model \mathbf{A} and the abundances \mathbf{x} . This gives us our recipe for hyperspectral unmixing: first, perform endmember fitting to initialize $\mathbf{A}(\mathbf{A})$, then, solve Equation 7 in an alternating

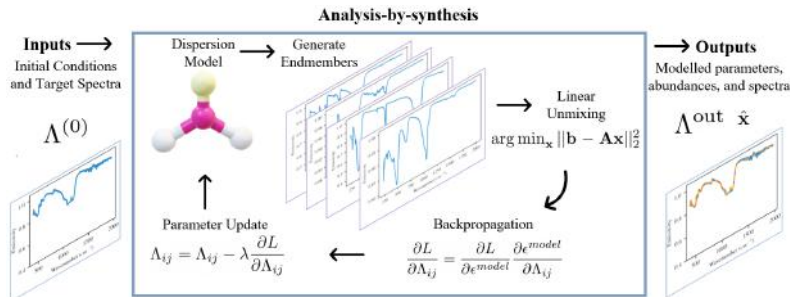


Fig. 3. Analysis-by-Synthesis: The analysis-by-synthesis algorithm uses differentiable programming to find optimal dispersion parameters and abundances. The initial dispersion parameters and the target spectra are fed as inputs, and the algorithm alternates between optimizing the abundances in the least squares sense and updating the dispersion parameters with respect to the gradient

fashion for \mathbf{x} and \mathbf{A} . One could also solve this equation jointly for both unknowns, however, we found that the alternating optimization was faster and converged to better results.

The optimization problem established in equation (7) is an alternating minimization problem and is unfortunately not convex [27]. One popular approach to tackle nonconvex problems is to find a good initialization point [17, 5], and then execute a form of gradient descent. Inspired by this, we first initialize $\mathbf{A}(\mathbf{A})$ by performing endmember fitting using Equation 6 as described in the previous subsection. Our experiments indicate that this provides a useful initialization for our subsequent step. We then perform alternating minimization on Equation 7 for \mathbf{x} and \mathbf{A} . Note that each iterate of the resulting alternating minimization involves the solution of a subproblem which has a convergence rate which depends on the condition number of the matrix $\mathbf{A}(\mathbf{A})$. For more details on this, we refer the reader to Appendix C where we discuss on the properties of $\mathbf{A}(\mathbf{A})$ across multiple runs.

In the ideal scenario, this initial matrix $\mathbf{A}(\mathbf{A})$ would consist of the endmember spectra that fully characterizes the mixed spectra \mathbf{b} . However, since spectra for the same material can significantly vary [39, 48, 43, 44, 7, 51] (see Figure 1), the initialization can be slightly off and we follow up with (7) to obtain a better fit. Note that this optimization problem is solving for the maximum likelihood estimator under a Gaussian noise model. Our optimization technique is performing **analysis-by-synthesis**, as given a single observation \mathbf{b} , the dispersion model synthesizes endmember variation until a good fit is achieved.

Inverse Rendering of Dispersion Model Parameters. The previous analysis-by-synthesis optimization does not require training data (labeled abundances in spectral mixtures) in order to perform spectral unmixing. However, there is room for even more improvement by using labeled data to help improve the parameter fitting of the model in the synthesis step. We train a CNN to pre-

dict the parameters for a generative model, known as inverse rendering in other domains [57]. In Figure 4, we show this inverse rendering conceptually, and how it can be fed into our differentiable programming pipeline for end-to-end spectral unmixing.

Our CNN architecture consists of convolutional layers followed by a series of fully-connected layers. We refer the reader to the supplemental material for the exact network structure and implementation details. Using a CNN for inverse rendering is significantly faster at test time as compared to the analysis-by-synthesis optimization. However, it does have a drawback of requiring training data which is unavailable for certain tasks/datasets.

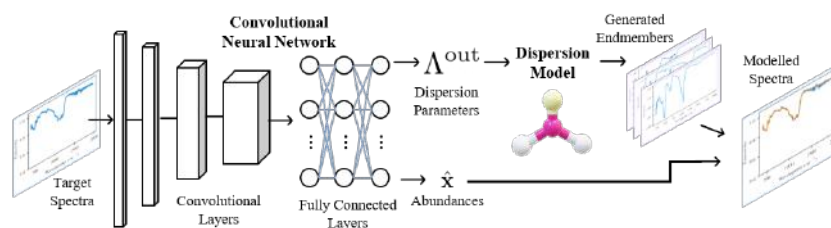


Fig. 4. Inverse Rendering: A CNN is trained to “inversely render” pixels of the hyperspectral image, by predicting both the dispersion parameters that control the spectral variability, and the abundances that control the mixing model. During training, the reconstruction error is back-propagated through the differentiable dispersion model to boost the performance of the network at making physically realistic predictions

4 Experimental Results

Datasets. We utilize three separate datasets to validate our spectral unmixing algorithms. In Figure 5, we visually represent these datasets and their exemplar data. For specific implementation details and dataset pre-processing, please see Appendix D in the supplemental material.

Feely et al. Dataset. We utilize 90 samples from the Feely et al. dataset [19] of thermal emission spectra in the infrared for various minerals measured in the lab. Ground truth was determined via optical petrography [19], and a labeled endmember library is provided. The limited amount of data is challenging for machine learning methods, so we utilize the dispersion model to generate 50,000 additional synthetic spectra for dataset augmentation.

Gulfport dataset. The Gulfport dataset from Gader et al. [20] contains hyperspectral aerial images in the VNIR along with ground truth classification labels segmenting pixels into land types (e.g. grass, road, building). Although the dataset is for spectral classification, it can also be used to benchmark unmixing algorithms by creating synthetic mixtures of pure pixels from the Gulfport dataset with random abundances as done by [16, 65]. We perform both spectral

classification (Gulfport) and unmixing (Gulfport synthetic) tasks in our testing. Both datasets are split into a train and test set (although some methods do not require training data), and the training data is augmented with 50,000 synthetically generated mixtures from the dispersion model.

One main difficulty of this dataset is the endmembers identified correspond to coarse materials such as grass and road as opposed to pure materials. Such endmembers can significantly vary across multiple pixels, but this spectral variation is not physically described by the dispersion model. To solve this problem, we utilize K-means clustering to learn exemplar endmembers for each category (e.g. grass, road, etc). Then the resulting centroid endmember can be fit to the dispersion model to allow further variation such as absorption band shifts in the spectra. We found that $K = 5$ worked the best for the Gulfport dataset.

TES Martian Dataset. The Thermal Emission Spectrometer (TES) [13] uses Fourier Transform Infrared Spectroscopy to measure the Martian surface. We utilize pre-processing from Bandfield et al. [2], and the endmember library used by Rogers et al. to analyze Mars TES data [47]. There is no ground truth for this dataset, as the true abundance of minerals on the Martian surface is unknown, so other metrics such as reconstruction error of the spectra are considered.

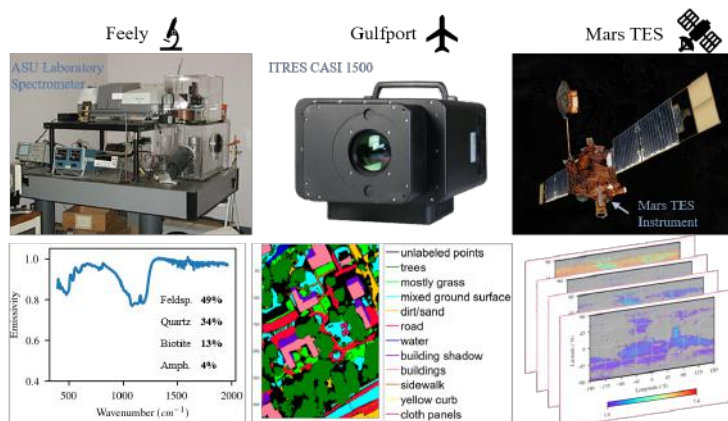


Fig. 5. Datasets: This figure shows representative data and instrumentation for the three datasets considered in this paper. Data includes laboratory, aircraft, and satellite measurements, and ground truth ranges from detailed abundance analysis under a microscope (Feely [19]) to pure pixel labels of land type for spectral classification (Gulfport [20]) to no ground truth for the Martian data (TES [13])

Baselines. We compare against several state-of-the-art baselines in the literature. The basic linear unmixing algorithm is Fully Constrained Least Squares (FCLS) [24] which solves least squares with sum-to-one and non-negativity constraints on the abundances. We also implement two state-of-the-art statistical methods for modelling endmember variability as distributions: the Normal Com-

positional Model (NCM) [50] and the Beta Compositional Model (BCM) [16]. NCM and BCM use a Gaussian and Beta distribution respectively, perform expectation-maximization for unmixing, and require a small amount of training data to determine model parameters.

We also compare against two state-of-the-art deep learning networks by Zhang et al. [63]. The first network utilizes a 1D CNN (CNN-1D) architecture, while the second network utilizes a 3D CNN (but with 1D convolutional kernels) (CNN-3D). CNN-3D is only applicable to datasets with spatial information, and not testable on the Feely and Gulfport synthetic data. We further created a modified CNN architecture (CNN-1D Modified) to maximize the performance on our datasets by changing the loss function to MSE, removing max-pooling layers, and adding an additional fully connected layer before the output. In the supplemental material, we provide information about the parameters, network architectures, and training procedures we used for these baselines as well as details for our own methods. We also have all of our code available here: <https://github.com/johnjaniczek/InfraRender>.

Endmember Fitting Results. To bootstrap both the analysis-by-synthesis and inverse rendering algorithms, good initial conditions for the dispersion parameters need to be input to the model. Determining dispersion parameters typically required detailed molecular structure analysis or exhaustive parameter searching methods [49, 55, 32]. One main advantage of our method is that we utilize gradient descent to efficiently find parameter sets for different materials. In Appendix B of the supplemental material, we share some of these parameter sets and our insights using the dispersion model for the scientific community.

In Figure 6, we show qualitative results of our endmember fitting by minimizing the loss in Equation 6 using gradient descent. The reconstructed spectra achieves a low MSE with the measured spectra with an average MSE of 0.016 for the TES library, 0.0019 for the Feely library, and an MSE of $2.6e-5$ on the Gulfport cluster centroids. Note that there is noise in the measurements, and so MSE is not an absolute metric of the fit to the true unknown spectra.

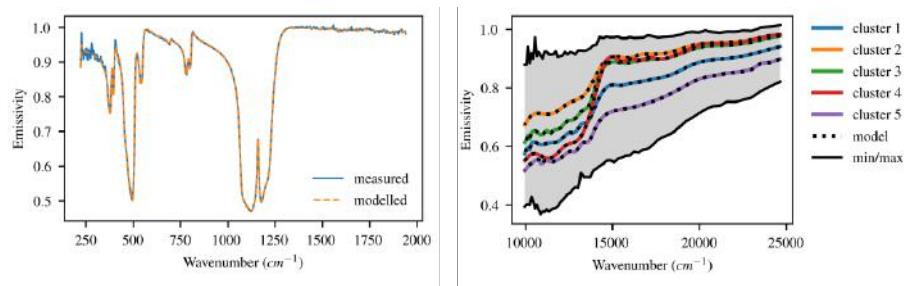


Fig. 6. Endmember Fitting: (Left) Measured and modelled spectra for a quartz sample in the IR. (Right) Cluster centroids found for pixels labelled as grass in the Gulfport dataset, and the model fit to these centroids. Note the high fidelity of fit via the dispersion model for both these cases

Table 1. Results: Table - Mean squared error of the abundance predictions vs. ground truth for Feely, Gulfport, and Gulfport synthetic datasets. The bold entries indicate top performance

Dataset	FCLS [24]	NCM [50]	BCM [16]	CNN-1D [63]	CNN-3D [63]	CNN-1D Modified	Analysis- by- synthesis	Inverse Rendering
Feely [19]	0.121	0.119	0.131	0.469	N/A	0.205	0.052	0.188
Gulfport [20]	0.75	0.799	0.800	1.000	0.497	0.297	0.45	0.272
Gulfport Synthetic	0.911	0.471	0.136	0.824	N/A	0.148	0.147	0.059

Spectral Unmixing Results. In Table 1, we show results on the Feely, Gulfport, and the Gulfport synthetic mixture datasets. For Feely, the analysis-by-synthesis method achieved a MSE of 0.052, with the next closest method (NCM) achieving 0.119. Due to the Feely dataset only containing 90 test samples, the machine learning methods were trained on synthetic data which explains their lower performance as data mismatch. Thus, the low error of analysis-by-synthesis shows the utility of the dispersion model for modelling endmember variability, particularly in cases with low training data.

For the Gulfport dataset, the task was to predict the material present since the labeled data is for single coarse materials (e.g. road, grass, etc) at 100% abundance per pixel. Here, the deep learning methods of CNNs and Inverse Rendering have the highest performance. This is expected as there exists a large amount of training data to learn from. Note that Inverse Rendering performs the best at 0.272 MSE, demonstrating that the addition of a generative dispersion model to the output of the CNN improves performance over purely learned approaches. Also note that our analysis-by-synthesis method still has relatively high performance (0.45 MSE) without using any training data at all.

For the Gulfport synthetic mixture dataset, Inverse Rendering achieves the lowest MSE of 0.059, leveraging both physics-based modeling for spectral mixing as well as learns from available training data. The BCM and the analysis-by-synthesis methods both outperform the CNN methods, even though they do not have access to the training data. In fact, BCM even slightly outperforms the analysis-by-synthesis method, which could be because the sources of variation in this data are well-described by statistical distributions.

Speed of Methods. The additional capacity of adding statistical and physical models usually has a cost of speed in implementation. Averaged over 90 mixtures, the convergence for a single operation was FCLS - 10ms, BCM - 1.23s, NCM - 18ms, CNN - 33ms, Inverse Rendering - 39ms, and analysis-by-synthesis - 10.2s. Future work could potentially increase the speed of analysis-by-synthesis with parallel processing.

Noise analysis. Prior to spectral unmixing, emissivity is separated from radiance by dividing out the black-body radiation curve at the estimated tem-

perature [43, 14]. In general, a Gaussian noise profile in the radiance space with variance $\sigma_{\text{radiance}}^2$ results in wavenumber dependent noise source in the emissivity space with the profile $\sigma^2(\omega) = \sigma_{\text{radiance}}^2 \cdot 1/B(\omega, T)$ where B is the black-body function given by Planck’s law. In our noise experiments we use a black-body radiation curve for a 330K target, which is the approximate temperature the Feely dataset samples were held at. In Figure 7 left, we see that the emissivity noise is higher where the radiance signal is lower.

We simulated varying the noise power to determine the methods’ robustness tested on 30 samples from the Feely dataset. In Figure 7, you can see that analysis-by-synthesis still has the best performance in the presence of noise, and is relatively flat as noise increases compared to other methods. We note that statistical methods, while having higher average error, seem to be robust to increased noise as they can handle random perturbations of each spectral band statistically. CNN and Inverse Rendering methods perform the worst for high noise, as these methods were trained on data without noise.

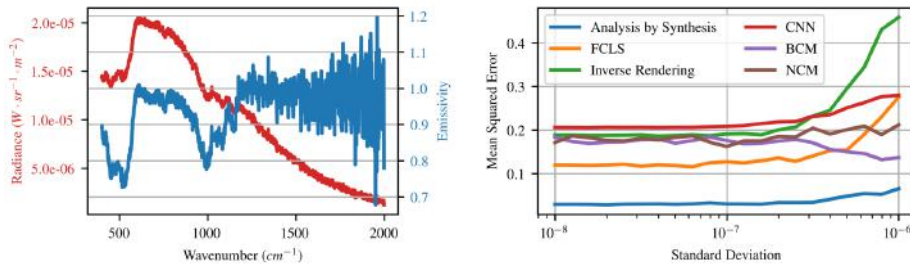


Fig. 7. The left plot shows the radiance profile of a spectra perturbed by Gaussian noise and the resulting emissivity profile after separating out the blackbody radiance. The right figure shows the robustness of the algorithms to increasing amounts of noise

TES Data. The Mars TES data was unmixed using our analysis-by-synthesis method to demonstrate it’s utility on tasks where zero training data is available. The method produces mineral maps which correctly finds abundances of the mineral hematite at Meridiani Planum in Figure 8. This is an important Martian mineral which provides evidence for liquid water having existed at some point on Mars, and has been verified by NASA’s Opportunity Rover [29]. Note how FCLS predicts many sites for hematite, while our method narrows down potential sites on the Martian surface, which is useful for planetary scientists. By allowing for spectral variation through our physics-based approach, our method has lower RMS reconstruction error than previous analysis of TES data. FCLS, which was previously used on TES because of the zero training-data problem, has an average RMS reconstruction error of 0.0043 while analysis-by-synthesis has an average of 0.0038. This is an exciting result as our methods could provide a new suite of hyperspectral analysis tools for scientists studying the Martian surface.

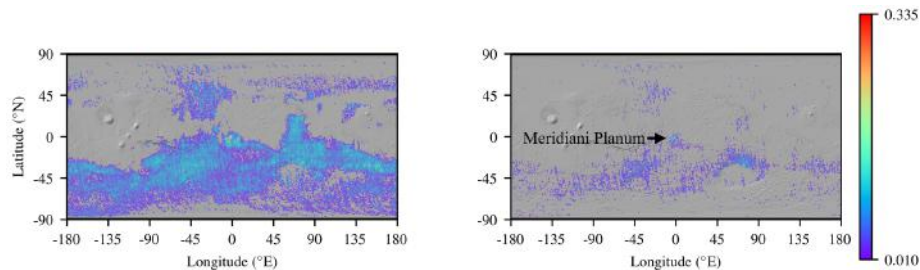


Fig. 8. Martian Surface Map: The images show the mineral map for hematite of the Martian surface produced by FCLS (left) and analysis-by-synthesis (right) using TES data. Both algorithms find the known deposit of hematite on Meridiani Planum, but analysis-by-synthesis predicts a sparser map which matches expected distributions.

5 Discussion

This paper incorporated generative physics models into spectral unmixing algorithms via differentiable programming. We adopt a physics-based dispersion model to simulate spectral variability, and show how this model can realistically fit several real measured spectra via gradient descent. We further show how to jointly optimize for the dispersion parameters and material abundances with an analysis-by-synthesis optimization. A second algorithm is introduced for tasks where additional data is available by training a CNN to “Inversely Render” a hyperspectral image with the differentiable dispersion model in the loop.

We validate these contributions extensively on three datasets ranging from mid to far IR and VNIR, and compared against state-of-the-art optimization, statistical and deep learning benchmarks. From these experiments we observe that analysis-by-synthesis has the best performance when training data is not available, and that Inverse Rendering has the best performance when training data is available. We also see that analysis-by-synthesis is noise resilient, and reconstructs Mars spectra with lower error than previous techniques.

There are still limitations for the methods proposed. First, analysis-by-synthesis has a large computational cost compared to other methods, although this could be mitigated through parallelization of the algorithm. Secondly, the spectral unmixing community is limited by the training data available. This is difficult to overcome, because it is expensive to produce quality datasets, and it is not easy for experts to label remote sensing datasets from prior knowledge alone. Future work could investigate generating realistic synthetic data suitable for training machine learning based algorithms for better performance. We hope using generative physics-based models inspires others to produce realistic synthetic data as well as differentiable programming methods which require low training data.

Acknowledgements. This work was supported by NSF grant IIS-1909192 as well as GPU resources from ASU Research Computing. We thank Dr. Alina Zare, Christopher Haberle, and Dr. Deanna Rogers for their helpful discussions, and Kim Murray (formerly Kim Feely) for providing the laboratory measurements and analysis contributing to this paper.

References

1. Azinovic, D., Li, T.M., Kaplanyan, A., Niessner, M.: Inverse path tracing for joint material and lighting estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2447–2456 (2019)
2. Bandfield, J.L.: Global mineral distributions on mars. *Journal of Geophysical Research: Planets* **107**(E6), 9–1 (2002)
3. Bateson, C.A., Asner, G.P., Wessman, C.A.: Endmember bundles: A new approach to incorporating endmember variability into spectral mixture analysis. *IEEE Transactions on Geoscience and Remote Sensing* **38**(2), 1083–1094 (2000)
4. Baydin, A.G., Pearlmutter, B.A., Radul, A.A., Siskind, J.M.: Automatic differentiation in machine learning: a survey. *The Journal of Machine Learning Research* **18**(1), 5595–5637 (2017)
5. Bhojanapalli, S., Kyriillidis, A., Sanghavi, S.: Dropping convexity for faster semi-definite optimization. In: Conference on Learning Theory. pp. 530–582 (2016)
6. Borsoi, R.A., Imbiriba, T., Bermudez, J.C.M., Richard, C., Chanussot, J., Drumetz, L., Tourneret, J.Y., Zare, A., Jutten, C.: Spectral variability in hyperspectral data unmixing: A comprehensive review. arXiv preprint arXiv:2001.07307 (2020)
7. Burns, R.G.: Crystal field spectra and evidence of cation ordering in olivine minerals. *American Mineralogist: Journal of Earth and Planetary Materials* **55**(9-10), 1608–1632 (1970)
8. Chang, C.I.: *Hyperspectral imaging: techniques for spectral detection and classification*, vol. 1. Springer Science & Business Media (2003)
9. Chen, F., Zhang, Y.: Sparse hyperspectral unmixing based on constrained lp-1 2 optimization. *IEEE Geoscience and Remote Sensing Letters* **10**(5), 1142–1146 (2013)
10. Chen, Y., Jiang, H., Li, C., Jia, X., Ghamisi, P.: Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing* **54**(10), 6232–6251 (2016)
11. Cheng, G., Li, Z., Han, J., Yao, X., Guo, L.: Exploring hierarchical convolutional features for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* **56**(11), 6712–6722 (2018)
12. Chouzenoux, E., Legendre, M., Moussaoui, S., Idier, J.: Fast constrained least squares spectral unmixing using primal-dual interior-point optimization. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **7**(1), 59–69 (2014)
13. Christensen, P., Bandfield, J., Hamilton, V.E., Ruff, S.W., Kieffer, H.H., Titus, T.N., Malin, M.C., Morris, R.V., Lane, M.D., Clark, R.L., Jakosky, B.M., Mellon, M.T., Pearl, J.C., Conrath, B.J., Smith, M.D., Clancy, R.T., Kuzmin, R.O., Roush, T., Mehall, G.L., Gorelick, N., Bender, K., Murray, K., Dason, S., Greene, E., Silverman, S., Greenfield, M.: Mars global surveyor thermal emission spectrometer experiment: investigation description and surface science results. *Journal of Geophysical Research: Planets* **106**(E10), 23823–23871 (2001)
14. Christensen, P.R., Hamilton, V.E., Mehall, G., Pelham, D., O'Donnell, W., Anwar, S., Bowles, H., Chase, S., Fahlgren, J., Farkas, Z., Fisher, T., James, I., Kubik, I., Lazbin, M., Miner, M., Rassas, L., Schulze, K., Shamordola, T., Tourville, G., West, R. and Woodward, D., Lauretta: The osiris-rex thermal emission spectrometer (otes) instrument. *Space Science Reviews* **214**(5), 87 (2018)

15. Combe, J.P., Le Mouélic, S., Sotin, C., Gendrin, A., Mustard, J., Le Deit, L., Launeau, P., Bibring, J.P., Gondet, B., Langevin, Y., Pinet, P.: Analysis of omega/mars express data hyperspectral data using a multiple-endmember linear spectral unmixing model (melsum): Methodology and first results. *Planetary and Space Science* **56**(7), 951–975 (2008)
16. Du, X., Zare, A., Gader, P., Dranishnikov, D.: Spatial and spectral unmixing using the beta compositional model. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **7**(6), 1994–2003 (2014)
17. Duchi, J.C., Ruan, F.: Solving (most) of a set of quadratic equalities: Composite optimization for robust phase retrieval. *Information and Inference: A Journal of the IMA* **8**(3), 471–529 (2019)
18. Engel, J., Hantrakul, L.H., Gu, C., Roberts, A.: Ddsp: Differentiable digital signal processing. In: *International Conference on Learning Representations* (2020)
19. Feely, K.C., Christensen, P.R.: Quantitative compositional analysis using thermal emission spectroscopy: Application to igneous and metamorphic rocks. *Journal of Geophysical Research: Planets* **104**(E10), 24195–24210 (1999)
20. Gader, P., Zare, A., Close, R., Aitken, J., Tuell, G.: Muufi gulfport hyperspectral and lidar airborne data set. Univ. Florida, Gainesville, FL, USA, Tech. Rep. REP-2013-570 (2013)
21. Gkioulekas, I., Levin, A., Zickler, T.: An evaluation of computational imaging techniques for heterogeneous inverse scattering. In: *European Conference on Computer Vision*. pp. 685–701. Springer (2016)
22. Gkioulekas, I., Zhao, S., Bala, K., Zickler, T., Levin, A.: Inverse volume rendering with material dictionaries. *ACM Transactions on Graphics (TOG)* **32**(6), 162 (2013)
23. Goudge, T.A., Mustard, J.F., Head, J.W., Salvatore, M.R., Wiseman, S.M.: Integrating crism and tes hyperspectral data to characterize a halloysite-bearing deposit in kashira crater, mars. *Icarus* **250**, 165–187 (2015)
24. Heinz, D.C., et al.: Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing* **39**(3), 529–545 (2001)
25. Heylen, R., Parente, M., Gader, P.: A review of nonlinear hyperspectral unmixing methods. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **7**(6), 1844–1868 (2014)
26. Hu, W., Huang, Y., Wei, L., Zhang, F., Li, H.: Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors* **2015** (2015)
27. Jain, P., Kar, P., et al.: Non-convex optimization for machine learning. *Foundations and Trends® in Machine Learning* **10**(3-4), 142–336 (2017)
28. Keshava, N., Mustard, J.F.: Spectral unmixing. *IEEE Signal Processing Magazine* **19**(1), 44–57 (2002)
29. Klingelhöfer, G., Morris, R.V., Bernhardt, B., Schröder, C., Rodionov, D.S., De Souza, P., Yen, A., Gellert, R., Evlanov, E., Zubkov, B., et al.: Jarosite and hematite at meridiani planum from opportunity’s mössbauer spectrometer. *Science* **306**(5702), 1740–1745 (2004)
30. Larkin, P.: *Infrared and Raman spectroscopy: principles and spectral interpretation*. Elsevier (2017)
31. Lee, J.D., Simchowitz, M., Jordan, M.I., Recht, B.: Gradient descent converges to minimizers. *arXiv preprint arXiv:1602.04915* (2016)
32. Lee, S., Tien, C.: Optical constants of soot in hydrocarbon flames. In: *Symposium (international) on combustion*. vol. 18, pp. 1159–1166. Elsevier (1981)

33. Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., Benediktsson, J.A.: Deep learning for hyperspectral image classification: An overview. *IEEE Transactions on Geoscience and Remote Sensing* **57**(9), 6690–6709 (2019)
34. Li, T.M., Aittala, M., Durand, F., Lehtinen, J.: Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* **37**(6), 222:1–222:11 (2018)
35. Li, Y., Zhang, H., Shen, Q.: Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network. *Remote Sensing* **9**(1), 67 (2017)
36. Liu, P., Zhang, H., Eom, K.B.: Active deep learning for classification of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **10**(2), 712–724 (2016)
37. Loubet, G., Holzschuch, N., Jakob, W.: Reparameterizing discontinuous integrands for differentiable rendering. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* **38**(6) (Dec 2019). <https://doi.org/10.1145/3355089.3356510>
38. Lu, G., Fei, B.: Medical hyperspectral imaging: a review. *Journal of Biomedical Optics* **19**(1), 010901 (2014)
39. Moersch, J., Christensen, P.R.: Thermal emission from particulate surfaces: A comparison of scattering models with measured spectra. *Journal of Geophysical Research: Planets* **100**(E4), 7465–7477 (1995)
40. Nimier-David, M., Vicini, D., Zeltner, T., Jakob, W.: Mitsuba 2: A retargetable forward and inverse renderer. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* **38**(6) (Dec 2019). <https://doi.org/10.1145/3355089.3356498>
41. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch (2017)
42. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: *Advances in Neural Information Processing Systems*. pp. 8024–8035 (2019)
43. Ramsey, M.S., Christensen, P.R.: Mineral abundance determination: Quantitative deconvolution of thermal emission spectra. *Journal of Geophysical Research: Solid Earth* **103**(B1), 577–596 (1998)
44. Ramsey M., C.P.: Mineral abundance determination: Quantitative deconvolution of thermal emission spectra: Application to analysis of martian atmospheric particulates. *Journal of Geophysical Research: Solid Earth* (2000)
45. Ravi, N., Reizenstein, J., Novotny, D., Gordon, T., Lo, W.Y., Johnson, J., Gkioxari, G.: Pytorch3d. <https://github.com/facebookresearch/pytorch3d> (2020)
46. Roberts, D.A., Gardner, M., Church, R., Ustin, S., Scheer, G., Green, R.: Mapping chaparral in the santa monica mountains using multiple endmember spectral mixture models. *Remote Sensing of Environment* **65**(3), 267–279 (1998)
47. Rogers, A., Aharonson, O.: Mineralogical composition of sands in meridiani planum determined from mars exploration rover data and comparison to orbital measurements. *Journal of Geophysical Research: Planets* **113**(E6) (2008)
48. Salisbury, J.W., D’Aria, D.M., Sabins Jr, F.F.: Thermal infrared remote sensing of crude oil slicks. *Remote Sensing of Environment* **45**(2), 225–231 (1993)
49. Spitzer, W., Kleinman, D.: Infrared lattice bands of quartz. *Physical Review* **121**(5), 1324 (1961)
50. Stein, D.: Application of the normal compositional model to the analysis of hyperspectral imagery. In: *IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, 2003. pp. 44–51. IEEE (2003)

51. Sunshine, J.M., Pieters, C.M.: Determining the composition of olivine from reflectance spectroscopy. *Journal of Geophysical Research: Planets* **103**(E6), 13675–13688 (1998)
52. Tsai, C.Y., Sankaranarayanan, A.C., Gkioulekas, I.: Beyond volumetric albedo — A surface optimization framework for non-line-of-sight imaging. In: *IEEE Intl. Conf. Computer Vision and Pattern Recognition (CVPR)* (2019)
53. Wang, F., Decker, J., Wu, X., Essertel, G., Rompf, T.: Backpropagation with callbacks: Foundations for efficient and expressive differentiable programming. In: *Advances in Neural Information Processing Systems*. pp. 10180–10191 (2018)
54. Wang, F., Zheng, D., Decker, J., Wu, X., Essertel, G.M., Rompf, T.: Demystifying differentiable programming: Shift/reset the penultimate backpropagator. *Proceedings of the ACM on Programming Languages* **3**(ICFP), 1–31 (2019)
55. Wenrich, M.L., Christensen, P.R.: Optical constants of minerals derived from emission spectroscopy: Application to quartz. *Journal of Geophysical Research: Solid Earth* **101**(B7), 15921–15931 (1996)
56. Yang, X., Ye, Y., Li, X., Lau, R.Y., Zhang, X., Huang, X.: Hyperspectral image classification with deep learning models. *IEEE Transactions on Geoscience and Remote Sensing* **56**(9), 5408–5423 (2018)
57. Yu, Y., Smith, W.A.: Inverserendernet: Learning single image inverse rendering. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 3155–3164 (2019)
58. Zare, A., Gader, P.: Sparsity promoting iterated constrained endmember detection in hyperspectral imagery. *IEEE Geoscience and Remote Sensing Letters* **4**(3), 446–450 (2007)
59. Zare, A., Gader, P.: Hyperspectral band selection and endmember detection using sparsity promoting priors. *IEEE Geoscience and Remote Sensing Letters* **5**(2), 256–260 (2008)
60. Zare, A., Ho, K.: Endmember variability in hyperspectral analysis: Addressing spectral variability during spectral unmixing. *IEEE Signal Processing Magazine* **31**(1), 95–104 (2013)
61. Zhang, C., Wu, L., Zheng, C., Gkioulekas, I., Ramamoorthi, R., Zhao, S.: A differential theory of radiative transfer. *ACM Transactions on Graphics (TOG)* **38**(6), 1–16 (2019)
62. Zhang, L., Zhang, L., Du, B.: Deep learning for remote sensing data: A technical tutorial on the state of the art. *IEEE Geoscience and Remote Sensing Magazine* **4**(2), 22–40 (2016)
63. Zhang, S., Li, J., Li, H.C., Deng, C., Plaza, A.: Spectral–spatial weighted sparse regression for hyperspectral image unmixing. *IEEE Transactions on Geoscience and Remote Sensing* **56**(6), 3265–3276 (2018)
64. Zhao, W., Du, S.: Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach. *IEEE Transactions on Geoscience and Remote Sensing* **54**(8), 4544–4554 (2016)
65. Zhou, Y., Rangarajan, A., Gader, P.D.: A gaussian mixture model representation of endmember variability in hyperspectral unmixing. *IEEE Transactions on Image Processing* **27**(5), 2242–2256 (2018)

Supplemental Material: Differentiable Programming for Hyperspectral Unmixing using a Physics-based Dispersion Model

John Janiczek¹, Parth Thaker¹, Gautam Dasarathy¹, Christopher Edwards²,
Philip Christensen¹, and Suren Jayasuriya¹

¹ Arizona State University

² Northern Arizona University

Appendix A: Derivation of Dispersion Model

In this section, we derive the dispersion model from first principles, modeling the generation of spectral radiance as a sum of mass-spring oscillations driven by an electromagnetic wave. This induces changes in the index of refraction, which governs the reflectance of the material with respect to light wavelength/frequency. This is based on earlier work by Garbuny and by Spitzer et al. [3, 5].

We first start with the equation for a mass-spring oscillator driven by an external force:

$$F = m \frac{d^2x}{dt^2} + R \frac{dx}{dt} + G \cdot x(t). \quad (1)$$

For a charged particle, $F = qE$, where q is charge and $E = E_0e^{i\omega t}$ for a propagating electromagnetic wave. Thus we can substitute these in to get:

$$F = qE_0e^{i\omega t} = m \frac{d^2x}{dt^2} + R \frac{dx}{dt} + G \cdot x(t) \quad (2)$$

which has the solution:

$$x = \frac{qE_0e^{i\omega t}}{m} \frac{1}{\frac{G}{m} - \omega^2 + i\frac{R}{m}\omega} = \frac{qE_0e^{i\omega t}}{m} \frac{1}{\omega_0^2 - \omega^2 + i\gamma\omega}. \quad (3)$$

where $\omega_0^2 = G/m$ and $\gamma = R/m$.

At the same time, we can also relate x to E via the band strength:

$$x = \frac{\alpha E}{q} \quad (4)$$

where α is the polarizability. Using the identity $\epsilon = 1 + 4\pi\alpha$, we can derive the following band strength equation:

$$x = \frac{(\epsilon - 1)E_0e^{i\omega t}}{4\pi q}. \quad (5)$$

Combining Eq. 3 and Eq. 5, we get

$$\frac{(\epsilon - 1)E_0 e^{i\omega t}}{4\pi q} = \frac{qE_0 e^{i\omega t}}{m} \frac{1}{\omega_0^2 - \omega^2 + i\gamma\omega}. \quad (6)$$

Solving for ϵ :

$$\epsilon = \frac{4\pi q^2}{m} \frac{1}{\omega_0^2 - \omega^2 + i\gamma\omega} + 1. \quad (7)$$

Relating ϵ to the refractive index, $\epsilon\mu = \hat{n}^2$ where $\mu = 1$ and $\hat{n} = n - ik$, we get

$$\epsilon = (n - ik)^2 = n^2 - k^2 - 2nki = \frac{4\pi q^2}{m} \frac{1}{\omega_0^2 - \omega^2 + i\gamma\omega} + 1 \quad (8)$$

This yields the refractive index equations:

$$n^2 - k^2 = \frac{4\pi q^2}{m} \frac{\omega_0^2 - \omega^2}{(\omega_0^2 - \omega^2)^2 + \gamma^2\omega^2} + 1 \quad (9)$$

$$2nk = \frac{4\pi q^2}{m} \frac{\gamma\omega}{(\omega_0^2 - \omega^2)^2 + \gamma^2\omega^2}. \quad (10)$$

Using the Lorentz-Lorenz formula, we can get

$$\hat{n}^2 = 1 + \frac{4\pi q^2}{m} \frac{1}{\omega_1^2 - \omega^2 + i\gamma\omega} \quad (11)$$

where $\omega_1^2 = \omega_0^2 - \frac{4\pi q^2}{3m}$ where $\omega_1 < \omega_0$. So plugging in ω_1 for ω_0 yields:

$$n^2 - k^2 = \frac{4\pi q^2}{m} \frac{\omega_1^2 - \omega^2}{(\omega_1^2 - \omega^2)^2 + \gamma^2\omega^2} + 1 \quad (12)$$

$$2nk = \frac{4\pi q^2}{m} \frac{\gamma\omega}{(\omega_1^2 - \omega^2)^2 + \gamma^2\omega^2}. \quad (13)$$

This entire derivation was for a single oscillator, but in practice, there are multiple oscillators that interact. We write this as a linear superposition given as follows:

$$n^2 - k^2 = \epsilon_r + \sum_i \frac{4\pi q^2 f_i}{m_i} \frac{\omega_i^2 - \omega^2}{(\omega_i^2 - \omega^2)^2 + \gamma_i^2\omega^2} \quad (14)$$

$$2nk = \sum_i \frac{4\pi q^2 f_i}{m_i} \frac{\gamma_i\omega}{(\omega_i^2 - \omega^2)^2 + \gamma_i^2\omega^2}. \quad (15)$$

where f_i is the strength of each individual oscillator. Using these equations, we have two equations for two unknowns (n and k), which we showed in Section 3 of the main paper is the basis of calculating reflectance and emission.

Appendix B: Spectral Variation

As stated in the main paper, the motivation for incorporating the dispersion model into a differentiable program for spectral unmixing is to allow for physically plausible spectral variation of pure materials. Because it is known that absorption bands can shift in frequency and strength between different end-member samples, the goal was to find a model that described these changes in a physically plausible way. That is we wanted a generative model for the formation of spectra with parameters that have "dials" to tune the frequency, strength, and shape of absorption bands. From the literature on analysis of the formation of spectra from an atomistic perspective [5, 7] we find that the Lorentz-Lorenz dispersion model is the correct approach to take. However, unlike previous works we go further than using the model to derive optical properties of materials, we also incorporate the dispersion model into an end-to-end spectral unmixing pipeline that allows the parameters to be fine-tuned via differentiable programming to account for spectral variability.

In Appendix A, the dispersion model is derived from first principles and each absorption band is described by the parameters $\rho, \omega_o, \gamma, \epsilon_r$. ρ is the band strength and as it increases the absorption band becomes deeper. ω_o is the resonant frequency and as it increases the absorption band shifts in wavenumber (and also slightly shifts the shape). γ is the frictional force (dampening coefficient) and controls the shape/width of the absorption bands. ϵ_r is relative dielectric permeability and as it increases the entire emissivity curve is shifted downwards. Also note that absorption bands which are close to each other interact in highly non-linear ways.

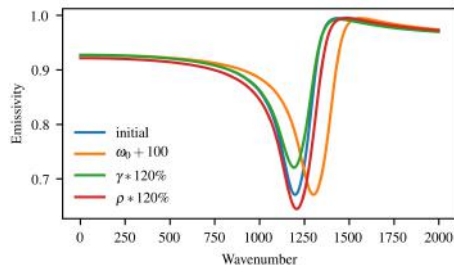


Fig. 1: A single absorption band is initialized with $\epsilon_r = 2.356, \omega_0 = 1161, \gamma = 0.1, \rho = 0.67$. Then the parameters are perturbed such that ω_0 is increased by 100, γ is increased to 120%, and ρ is increased to 120%. The plots show the effect of changing each parameter individually to show it's control over the shape and width of the absorption band.

The importance of initializing the alternating optimization with good initial dispersion parameters was emphasized in the main paper, as the problem is non-convex and good initialization is essential. It also makes intuitive sense

to initialize with parameters fit to an endmember sample to give physical significance to the generated spectra. As shown in the results of the main paper, we achieve good fits with low MSE on endmember libraries used to analyze the Martian surface as well as a semi-urban university scene. The endmember libraries used to fit the minerals to analyze the Mars TES data are of high quality from the Arizona State University Thermal Emission Spectral Library [1]. The resulting parameters from a few of the important materials from this endmember library are provided in the following tables.

Table 1: Dispersion parameters found for Olivine Fo10

Axis	Index	ω_0	γ	ρ	ϵ_r
0	0	258.45	0.018	0.022	1.07
0	1	272.71	0.038	0.070	1.07
0	2	285.33	0.027	0.035	1.07
0	3	340.81	0.021	0.015	1.07
0	4	361.06	0.067	0.187	1.07
0	5	467.03	0.060	0.091	1.07
0	6	589.36	0.032	0.043	1.07
0	7	826.60	0.011	0.015	1.07
0	8	863.05	0.030	0.083	1.07
0	9	934.94	0.018	0.038	1.07
0	10	1068.56	0.009	0.001	1.07
0	11	1349.50	0.043	0.009	1.07
0	12	1400.46	0.057	0.026	1.07
0	13	1452.82	0.064	0.020	1.07
0	14	1518.96	0.079	0.025	1.07
0	15	1597.62	0.018	0.001	1.07
0	16	1694.56	0.043	0.007	1.07
0	17	1794.69	0.032	0.002	1.07
0	18	1837.96	0.009	0.001	1.07
0	19	1934.50	0.056	0.020	1.07
1	0	293.77	0.042	0.240	1.99
1	1	303.28	0.058	0.263	1.99
1	2	317.16	0.137	0.356	1.99
1	3	473.47	0.006	0.002	1.99
1	4	496.39	0.029	0.030	1.99
1	5	504.45	0.062	0.302	1.99
1	6	562.92	0.055	0.057	1.99
1	7	577.32	0.027	0.008	1.99
1	8	891.85	0.023	0.189	1.99
1	9	990.28	0.047	0.086	1.99
1	10	1108.25	0.023	0.006	1.99

Table 2: Dispersion parameters found for Biotite

Axis	Index	ω_0	γ	ρ	ϵ_r
0	0	235.91	0.066	0.2343	1.31
0	1	432.39	0.056	0.4040	1.31
0	2	439.80	0.039	0.4131	1.31
0	3	446.34	0.014	0.0385	1.31
0	4	451.92	0.042	0.4797	1.31
0	5	594.57	0.073	0.0147	1.31
0	6	954.50	0.036	0.2510	1.31
0	7	1008.94	0.014	0.0578	1.31
0	8	1013.39	0.017	0.0184	1.31
0	9	1041.20	0.048	0.0178	1.31
0	10	1075.68	0.025	0.0198	1.31
0	11	1116.66	0.007	0.0003	1.31
0	12	1152.61	0.019	0.0012	1.31
0	13	1390.98	0.044	0.0177	1.31
0	14	1460.91	0.061	0.0280	1.31
0	15	1524.44	0.065	0.0676	1.31
0	16	1629.72	0.025	0.0271	1.31
0	17	1661.44	0.007	0.0034	1.31
0	18	1687.84	0.068	0.0723	1.31
0	19	1772.30	0.074	0.0877	1.31
0	20	1813.27	0.006	0.0009	1.31
0	21	1865.48	0.064	0.0731	1.31
0	22	1964.44	0.055	0.0131	1.31
1	0	268.77	0.073	0.4634	2.61
1	1	294.51	0.045	0.1965	2.61
1	2	313.92	0.064	0.3242	2.61
1	3	337.12	0.093	0.4930	2.61
1	4	362.24	0.062	0.1954	2.61
1	5	400.00	0.209	0.5174	2.61
1	6	462.66	0.065	0.4399	2.61
1	7	492.95	0.080	0.3498	2.61
1	8	510.47	0.061	0.0664	2.61
1	9	653.21	0.078	0.0611	2.61
1	10	718.49	0.040	0.0331	2.61
1	11	873.68	0.115	0.3343	2.61
1	12	928.32	0.048	0.0488	2.61
1	13	991.97	0.015	0.3550	2.61
1	14	1588.86	0.040	0.0607	2.61
1	15	1963.15	0.004	0.0023	2.61
1	16	1989.53	0.001	0.0002	2.61

Table 3: Dispersion parameters found for Hematite

Axis	Index	ω_0	γ	ρ	ϵ_r
0	0	258.29	0.11	0.110	1.27
0	1	279.35	0.13	0.141	1.27
0	2	294.73	0.11	0.149	1.27
0	3	335.86	0.08	0.130	1.27
0	4	471.32	0.07	0.098	1.27
0	5	526.58	0.05	0.029	1.27
0	6	543.94	0.07	0.062	1.27
0	7	563.14	0.08	0.067	1.27
0	8	609.37	0.04	0.041	1.27
0	9	619.61	0.04	0.041	1.27
0	10	632.43	0.07	0.067	1.27
0	11	654.46	0.09	0.054	1.27
0	12	686.74	0.12	0.038	1.27
0	13	798.98	0.04	0.011	1.27
0	14	890.21	0.03	0.009	1.27
0	15	916.82	0.02	0.005	1.27
0	16	958.26	0.04	0.014	1.27
0	17	1002.55	0.04	0.010	1.27
0	18	1100.72	0.03	0.022	1.27
0	19	1167.07	0.02	0.010	1.27
0	20	1238.37	0.01	0.005	1.27
0	21	1282.36	0.03	0.019	1.27
1	0	234.31	0.02	0.007	1.25
1	1	238.56	0.06	0.031	1.25
1	2	312.13	0.09	0.255	1.25
1	3	356.47	0.04	0.032	1.25
1	4	430.53	0.09	0.085	1.25
1	5	444.75	0.06	0.032	1.25
1	6	457.95	0.04	0.011	1.25
1	7	486.07	0.03	0.019	1.25
1	8	577.56	0.08	0.160	1.25
1	9	727.69	0.06	0.049	1.25
1	10	748.13	0.07	0.040	1.25
1	11	773.90	0.06	0.013	1.25
1	12	1049.92	0.10	0.058	1.25
1	13	1069.60	0.01	0.003	1.25
1	14	1140.36	0.02	0.012	1.25
1	15	1197.28	0.04	0.022	1.25
1	16	1256.54	0.02	0.010	1.25

Appendix C: Alternating Optimization Convergence

Our ultimate goal is to solve the spectral unmixing problem which can be formulated as $\min_{\mathbf{A}, \mathbf{x}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2$, where the minimization occurs over both the matrix \mathbf{A} and the unmixing vector \mathbf{x} . This is a standard case of alternate minimization which is known to be nonconvex [?]. In practice, alternating minimization are particularly hard to tackle due to the presence of suboptimal local minimas. Recent progress on tackling nonconvex problems involves either characterizing the optimization landscape [?, ?, ?] or providing initialization to descent algorithms [?, ?] to assure convergence to the global optimum. It is known that gradient descent applied to alternate minimization problem faces the issue of getting stuck at local minimas [?] and hence initialization plays an important role in solving Equation 7 in the main paper. With that in mind, in this paper we provide a mechanism to provide good initialization to gradient descent algorithm with the hope of tackling the alternating minimization problem effectively.

Initialization using dictionary $A(\epsilon^{\text{model}})$: We investigate the properties of matrix A as relates to the convergence of the alternating optimization. We denote the measured the emissivity spectrum of various materials in the lab as $\epsilon^{\text{measured}}$, and the physics-based dispersion model as ϵ^{model} . We then use these emissivity spectrum to construct a dictionary $A(\epsilon)$ which servers as the initialization for A in the alternating minimization approach in Equation 7 of the main paper. The intuition behind this revolves around the ability of matrix A as a dictionary of known emissivity spectra and we expect that the unknown spectra $\epsilon^{\text{unknown}}$ would be described as a linear combination of columns from matrix A .

Consider the following subproblem of the alternating minimization:

$$\min_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}(A)\mathbf{x}\|_2^2$$

without the regularization terms. In order to ensure the uniqueness of the solution \mathbf{x}^* , we need to ensure that the matrix A is full rank. The rate of convergence for the above minimization is inversely dependent on the condition number of the matrix $\mathbf{A}(A)$. While it is difficult to analyze this matrix analytically, we perform an experimental characterization of the rank, condition number, and eigenvalues of the matrix for several different runs of the optimization algorithm with random initializations.

From the plots, we can note that the matrix $\mathbf{A}(A)$ has full rank with condition number of around 6000. The minimum and maximum eigenvalues are not showing drastic difference which goes well with our motive to incorporate small changes using alternate minimization to fit the spectral differences due to geographic differences. The high condition number is the reason for the relatively slow performance for running the alternating minimization framework, with our method taking tens of seconds to converge.

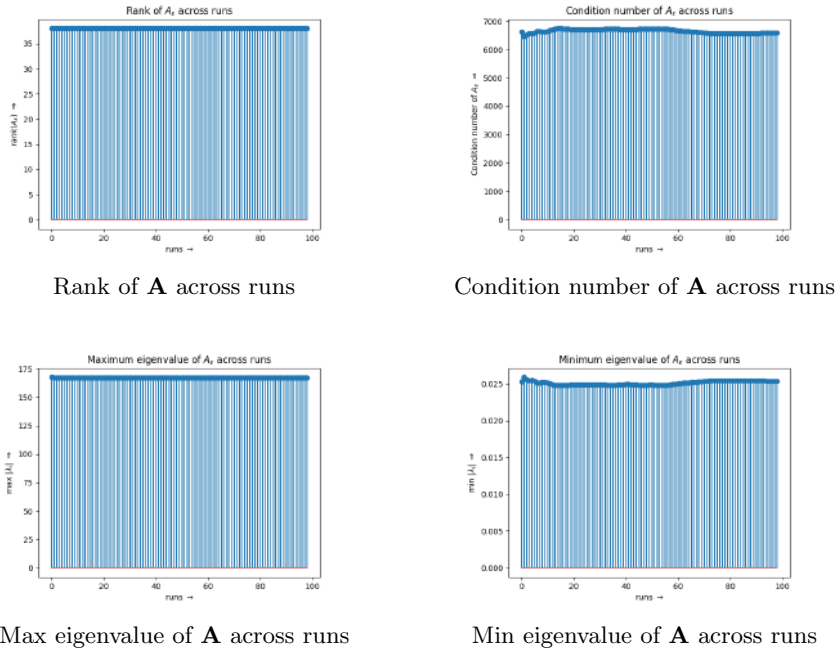


Fig. 3: Behaviour of \mathbf{A} across runs

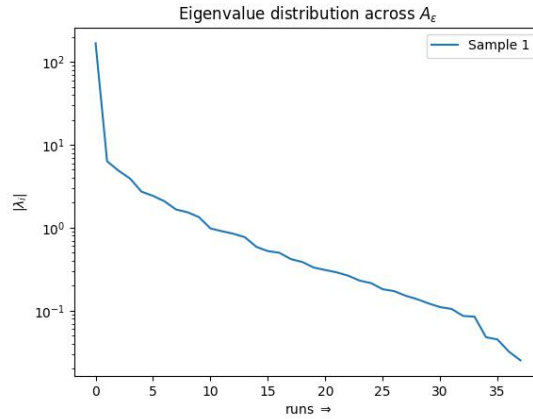


Fig. 4: Random sample of \mathbf{A} has the following eigenvalue distribution

Appendix D: Implementation Details

Fully Constrained Least Squares We compare against Fully Constrained Least Squares (FCLS) [4] which is a popular classical unmixing algorithm.

FCLS solves for the the (aerial) abundances, \mathbf{x} : $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2$ subject to $\|\mathbf{x}\|_1 = 1, \mathbf{x} \geq 0$. The constraints, referred to as the sum-to-one constraint and the non-negativity constraint, are enforced since abundances are interpreted as percentages.

Normal Compositional Model. The Normal Compositional Model (NCM) [6] is one of the most popular methods for modelling endmember variability via statistical methods. The method requires a small amount of training data (roughly 50 samples per endmember) to learn the mean and variance of reflectivity (or emissivity) of each spectral band, and modelling the variation as a Gaussian distribution. During unmixing, Expectation Maximization is used to simultaneously learn the abundances and the endmember variation, subject to the abundance sum-to-one and non-negativity constraints.

The NCM is run using the Matlab code provided by Du et al. [2]. Training data of about 50 samples of endmember variation were provided to the NCM for each dataset. There are no hyperparameters needed for this method.

Beta Compositional Model. The Beta Compositional Model (BCM) [2] is a more recent method for modelling spectral variability via a statistical method. Similar to the NCM, a small amount of training data is used to learn the beta parameters of each spectral band, and an Expectation Maximization algorithm is used during unmixing. The beta parameters allow each spectral band to be modelled as a more complex distribution than the NCM and has been shown to increase performance.

The BCM is run using the Matlab code provided by Du et al. [2]. Training data of about 50 samples of endmember variation were provided to the BCM for each dataset. For datasets without sufficient endmember samples, we generated synthetic endmember variation with the dispersion model. We search for the optimal hyperparameters through repeated experiments and report the best results. The optimal BCM across all datasets was run with $K = 3$, $\sigma_V = 100$, and $\sigma_M = 0.001$.

CNN for Spectral Classification and Unmixing. We baseline against the architecture recently proposed by Zhang et al. [8] for hyperspectral unmixing using both a 1D and 3D CNN. The main difference between the 1D and 3D CNN is that the 3D CNN is operates on 3×3 bundles of pixels while the 1D CNN is provided a single pixel. However, in both architectures the convolutional kernel is 1D and operates along the spectral dimension. Both architectures have four convolutional layers with Rectified Linear Unit (ReLU) and max-pooling non-linear operations. These layers are followed by 2 fully connected layers, where the last layer is the output abundance predictions. The ReLU of the fully connected layers ensure non-negativity, and the sum-to-one constraint is ensured by normalizing the output layer. The network is trained to minimize $-\hat{\mathbf{x}} \log(\mathbf{x})$, where $\hat{\mathbf{x}}$ and \mathbf{x} are the predicted and ground truth abundances respectively. While the 1D architecture performs well on the dataset it was designed for in [8], we found that modifications were necessary to maximize the performance on our datasets. Namely, we found that training the network with respect to the mean squared error (MSE) loss function, removing the max-pooling layers, and adding an ad-

ditional fully connected layer before the output improved performance of the 1D CNN. The 3D CNN is only applicable to datasets with spatial information, thus is ignored for the Feely and Synthetic datasets.

CNN Baselines [8]. The architecture from Zhang et al. was used to baseline against our method (CNN-1D and CNN-3D), as well as introducing our own modified baseline (CNN-1D modified). All CNNs were trained for 100 epochs using the Adam optimizer with learning = 0.0005, betas = (0.9, 0.999) and weight decay = 0.

CNN-1D: A 1x1 hyperspectral pixel is input into the network, four convolutional layers with alternating 1x5 and 1x4 kernels and a depth of 3, 6, 12, and 24 kernels in each layer respectively. All convolutional layers have ReLU activations a 1x2 maxpooling layer. The convolutional layers are followed by a fully connected layer with 150 hidden units, and an output fully connected layer with a size that depends on the number of abundances. Normalization is used to enforce the abundance sum-to-one constraint and the ReLU activation enforces the non-negativity constraint. The CNN is trained to minimize the log loss between the predicted and ground truth abundances. The network converges in about 100 epochs with a learning rate of 1e-3.

CNN-3D: CNN-3D has an almost identical architecture, although it accepts a 3x3 set of pixels at the input. Although a spatial dimension exists at the input, the convolutions only occur in the spectral dimension. four convolutional layers with alternating 1x5 and 1x4 kernels and a depth of 16, 32, 64, and 128 kernels in each layer respectively. The convolutional layers are followed by a fully connected layer with 150 hidden units, and an output fully connected layer with a size that depends on the number of abundances. Normalization is used to enforce the abundance sum-to-one constraint and the ReLU activation enforces the non-negativity constraint. The CNN is trained to minimize the log loss between the predicted and ground truth abundances. The network converges in about 100 epochs with a learning rate of 1e-3.

CNN-1D Modified: Finally, a modified version of CNN-1D is baselined against to try to find the optimal architecture for performance on our datasets. The first 2 max-pooling layers are removed, an additional hidden fully connected layer with 150 units is added before the output, and a softmax operation is applied to the output to enforce the abundance sum-to-one constraint. Also, the network is trained to minimize the mean squared error between the predicted and ground truth abundances. The network converges in about 100 epochs with a learning rate of 1e-3.

Analysis-by-Synthesis Optimization. For analysis-by-synthesis, the sparse regularization was set with $p = 0.95$ and $\lambda_p = 0.0001$. Dispersion parameters were constrained within a tolerance of their initial conditions with $\rho_{tol} = 0.05$, $\gamma_{tol} = 0.005$, $\epsilon_{tol} = 0.001$, and $\omega_{tol} = 0.0001$. On the Gulfport datasets γ_{tol} and ϵ_{tol} were increased to 0.05 to compensate for increased variation. Analysis-by-Synthesis alternates between finding optimal abundances (solving a regularized least squares problem), and updating the dispersion parameters for 100 itera-

tions using the Adam optimizer with learning rate = 0.01, betas = (0.9, 0.999), and weight decay = 0.

Inverse Rendering CNN. The inverse rendering CNN uses the same CNN architecture as CNN-1D modified. The input to the CNN is the spectrum (or batch of spectra). That is there are four convolutional layers with alternating 1x5 and 1x4 kernels and a depth of 3, 6, 12, and 24 kernels in each layer respectively. The convolutional layers also have ReLU activations and the last 2 layers have a 1x2 maxpooling layer. The convolutional layers are followed by fully connected layers with 150 hidden units. The final fully connected layer has enough units for the amount of dispersion parameters and abundances depending on the size of the endmember library and number of dispersion parameters per endmember. Then, the dispersion parameters are used to render endmember spectra and the mixture is reconstructed under the linear mixing model with the predicted abundances as inputs. The network only needs the input spectra and the abundances as inputs for training, as the reconstruction error of the spectra is used to back-propagated through the differentiable dispersion model to teach the network to predict good dispersion parameters. Real data (when available) and synthetic data (around 50,000 samples) are used to train the network, which converges after about 100 epochs. An Adam optimizer is used with learning rate set to 1e-3, betas set to (0.9, 0.999), and weight decay set to 0.

Appendix E: Additional Results

The mineral maps produced using analysis-by-synthesis on the Mars TES data are shown in Figures 5 - 10. The files containing the numerical abundances of each mineral will be uploaded to the project repository.

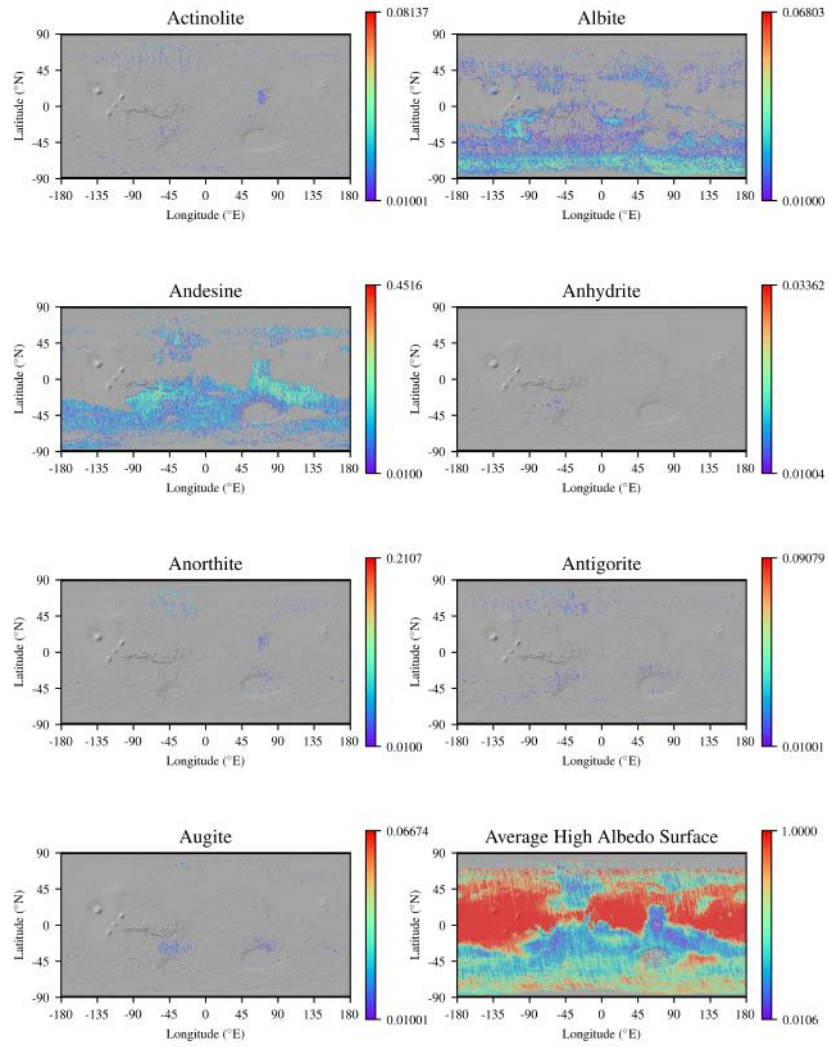


Fig. 5: Mineral Maps

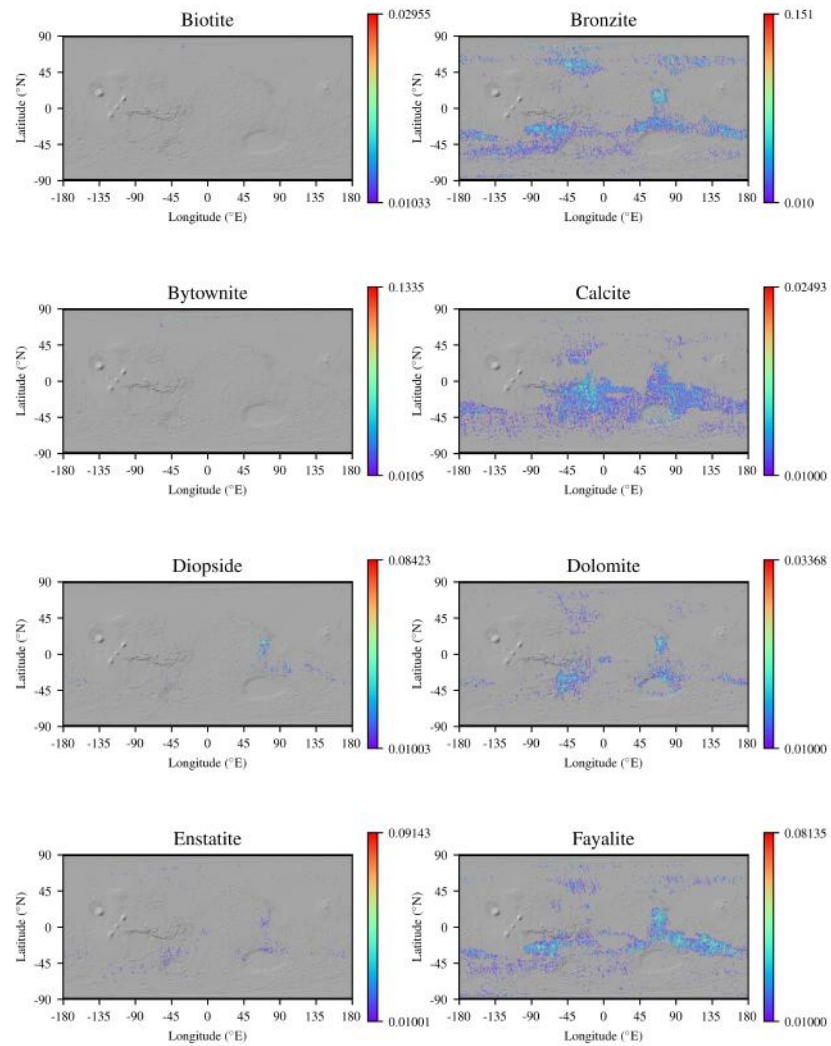


Fig. 6: Mineral Maps Continued

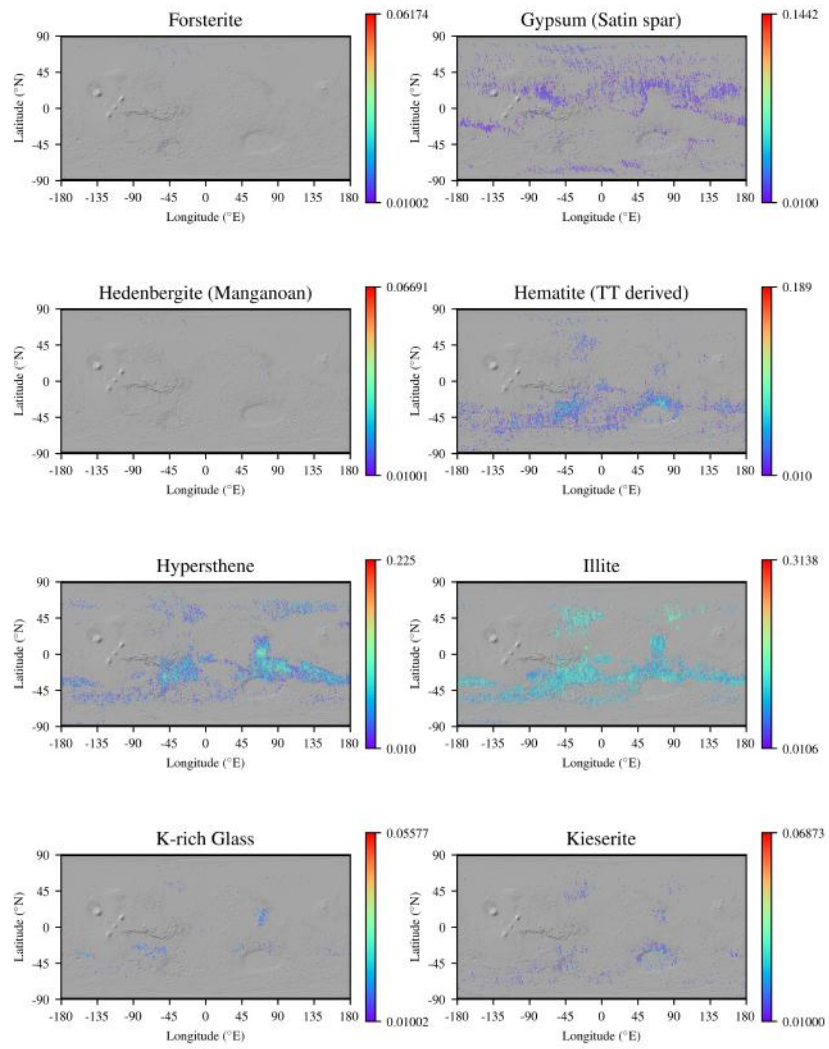


Fig. 7: Mineral Maps

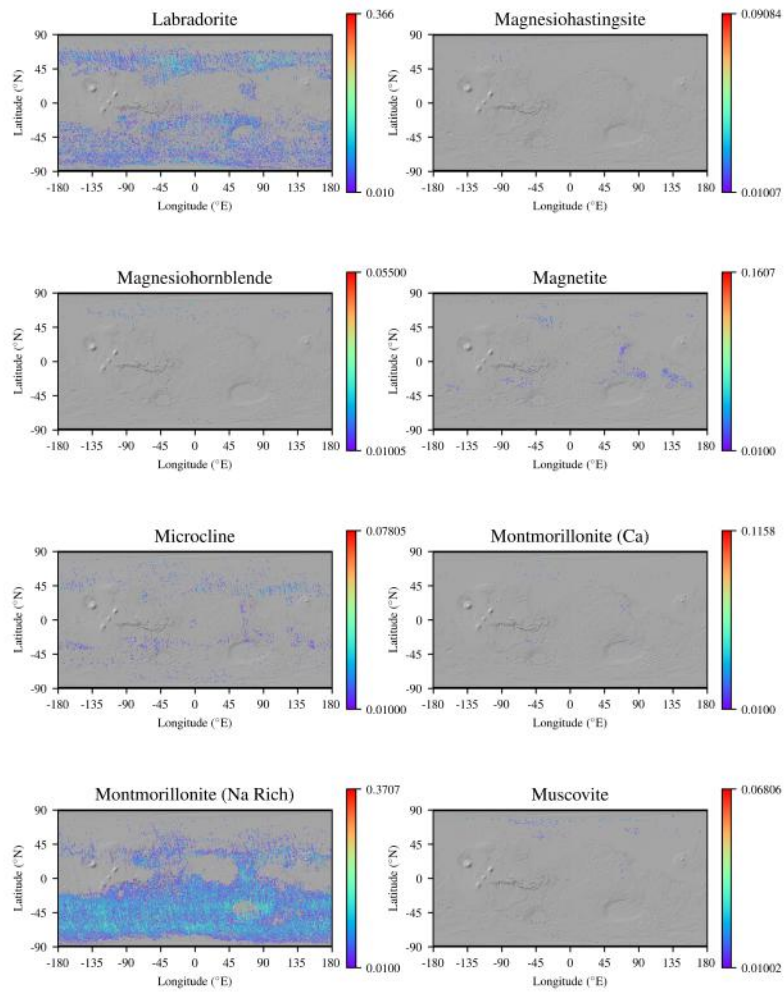


Fig. 8: Mineral Maps

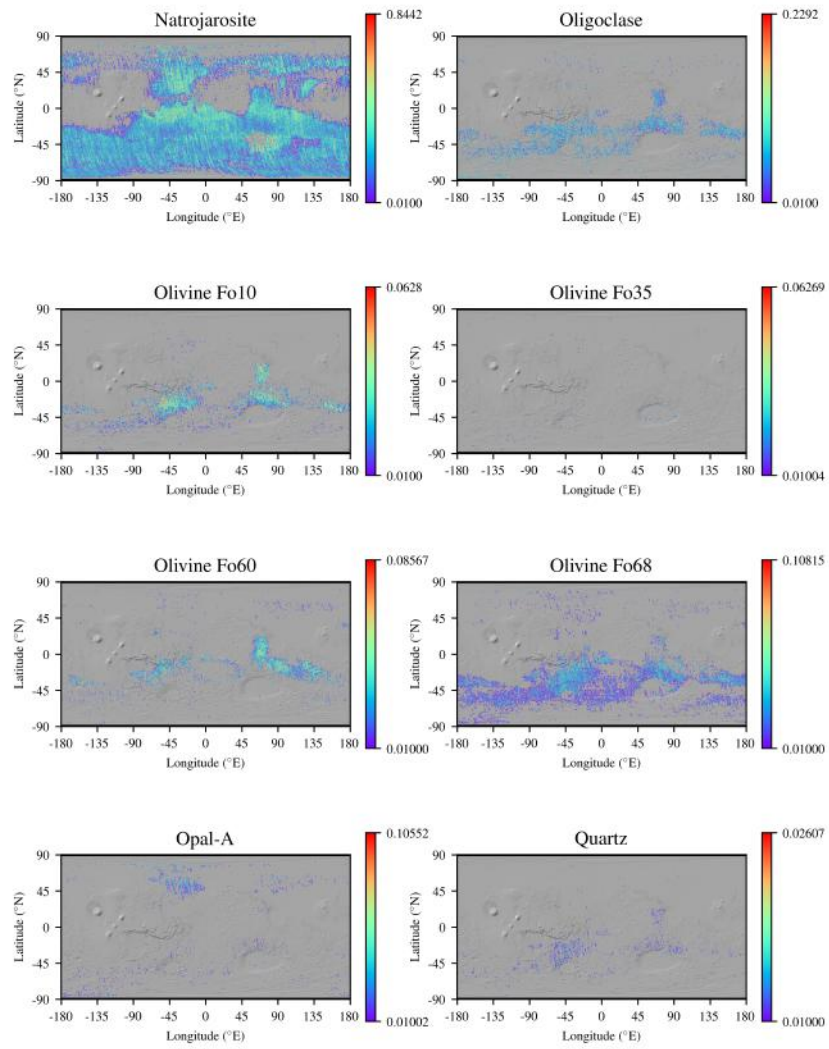


Fig. 9: Mineral Maps

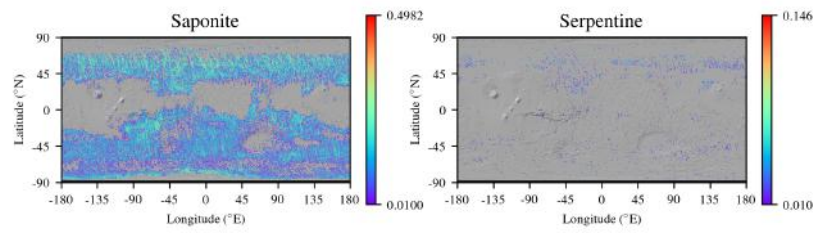


Fig. 10: Mineral Maps

References

1. Christensen, P.R., Bandfield, J.L., Hamilton, V.E., Howard, D.A., Lane, M.D., Piatek, J.L., Ruff, S.W., Stefanov, W.L.: A thermal emission spectral library of rock-forming minerals. *Journal of Geophysical Research: Planets* **105**(E4), 9735–9739 (2000)
2. Du, X., Zare, A., Gader, P., Dranishnikov, D.: Spatial and spectral unmixing using the beta compositional model. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **7**(6), 1994–2003 (2014)
3. Garbuny, M.: Optical physics. *Optical Physics by Max Garbuny* New York, NY: Academic Press, 1965 (1965)
4. Heinz, D.C., et al.: Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing* **39**(3), 529–545 (2001)
5. Spitzer, W., Kleinman, D.: Infrared lattice bands of quartz. *Physical Review* **121**(5), 1324 (1961)
6. Stein, D.: Application of the normal compositional model to the analysis of hyperspectral imagery. In: *IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, 2003. pp. 44–51. IEEE (2003)
7. Wenrich, M.L., Christensen, P.R.: Optical constants of minerals derived from emission spectroscopy: Application to quartz. *Journal of Geophysical Research: Solid Earth* **101**(B7), 15921–15931 (1996)
8. Zhang, S., Li, J., Li, H.C., Deng, C., Plaza, A.: Spectral–spatial weighted sparse regression for hyperspectral image unmixing. *IEEE Transactions on Geoscience and Remote Sensing* **56**(6), 3265–3276 (2018)