

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

John Janiczek<sup>1</sup>, Parth Thaker<sup>1</sup>, Gautam Dasarathy<sup>1</sup>, Christopher S. Edwards<sup>2</sup>, Philip Christensen<sup>1</sup>, and Suren Jayasuriya<sup>1</sup>

<sup>1</sup> Arizona State University

<sup>2</sup> 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. [5, 9].

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  $E = E_0 e^{i\omega t}$  for a propagating electromagnetic wave. Thus we can substitute these in to get:

$$F = qE_0 e^{i\omega t} = m \frac{d^2x}{dt^2} + R \frac{dx}{dt} + G \cdot x(t) \quad (2)$$

which has the solution:

$$x = \frac{qE_0 e^{i\omega t}}{m} \frac{1}{\frac{G}{m} - \omega^2 + i \frac{R}{m} \omega} = \frac{qE_0 e^{i\omega t}}{m} \frac{1}{\omega_0^2 - \omega^2 + i\gamma\omega}. \quad (3)$$

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

$$x = \frac{(\epsilon - 1)E_0 e^{i\omega t}}{4\pi q}. \quad (5)$$

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  $\epsilon$ :

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

Relating  $\epsilon$  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  $\omega_1$  for  $\omega_0$  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 endmember 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 [9, 13] 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  $\rho, \omega_o, \gamma, \epsilon_r$ .  $\rho$  is the band strength and as it increases the absorption band becomes deeper.  $\omega_o$  is the resonant frequency and as it increases the absorption band shifts in wavenumber (and also slightly shifts the shape).  $\gamma$  is the frictional force (dampening coefficient) and controls the shape/width of the absorption bands.  $\epsilon_r$  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.

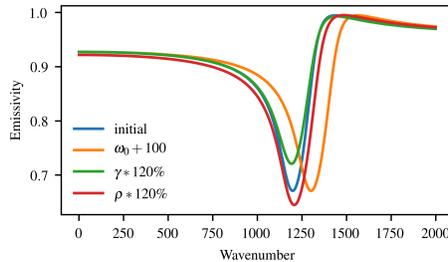


Fig. 1: A single absorption band is initialized with  $\epsilon_r = 2.356, \omega_0 = 1161, \gamma = 0.1, \rho = 0.67$ . Then the parameters are perturbed such that  $\omega_0$  is increased by 100,  $\gamma$  is increased to 120%, and  $\rho$  is increased to 120%. The plots show the effect of changing each parameter individually to show its 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 [3]. 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	$\omega_0$	$\gamma$	$\rho$	$\epsilon_r$
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	$\omega_0$	$\gamma$	$\rho$	$\epsilon_r$
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	$\omega_0$	$\gamma$	$\rho$	$\epsilon_r$
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 [8]. 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 [6, 12, 11] or providing initialization to descent algorithms [1, 2] 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 [8] 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.

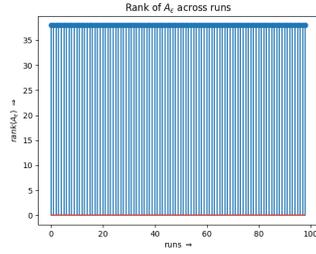
**Initialization using dictionary  $A(\epsilon^{\text{model}})$ :** We investigate the properties of matrix  $A$  as relates to the convergence of the alternating optimization. We denote the measured the emissivity spectrum of various materials in the lab as  $\epsilon^{\text{measured}}$ , and the physics-based dispersion model as  $\epsilon^{\text{model}}$ . 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:

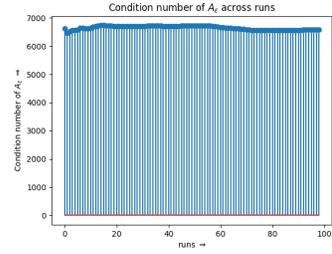
$$\min_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}(A)\mathbf{x}\|_2^2$$

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.

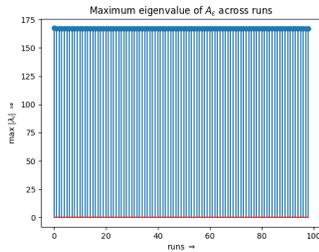
From the plots, we can note that the matrix  $\mathbf{A}(A)$  has full rank with condition number of around 6000. The minimum and maximum eigenvalues are not showing drastic difference which goes well with our motive to incorporate small changes using alternate minimization to fit the spectral differences 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.



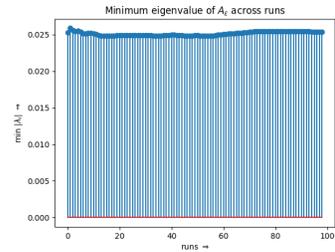
(a) Rank of  $\mathbf{A}$  across runs



(b) Condition number of  $\mathbf{A}$  across runs



(c) Max eigenvalue of  $\mathbf{A}$  across runs



(d) Min eigenvalue of  $\mathbf{A}$  across runs

Fig. 2: Behaviour of  $\mathbf{A}$  across runs

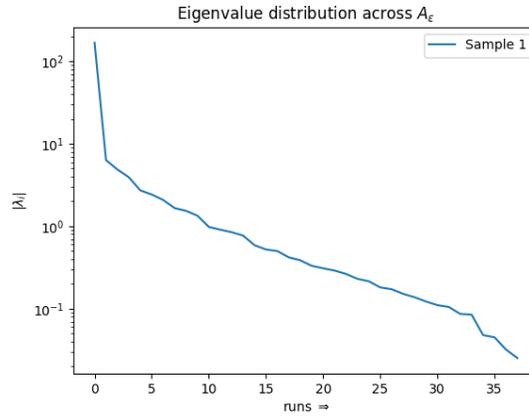


Fig. 3: Random sample of  $\mathbf{A}$  has the following eigenvalue distribution

## Appendix D: Implementation Details

**Fully Constrained Least Squares** We compare against Fully Constrained Least Squares (FCLS) [7] which is a popular classical unmixing algorithm.

FCLS solves for the the (aerial) abundances,  $\mathbf{x}$ :  $\hat{\mathbf{x}} = \operatorname{argmin}_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2$  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) [10] 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. [4]. 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) [4] 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. [4]. 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. [14] 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  $3 \times 3$  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 [14], 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 [14].** 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  $\gamma_{tol}$  and  $\epsilon_{tol}$  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 4 - 9. The files containing the numerical abundances of each mineral will be uploaded to the project repository.

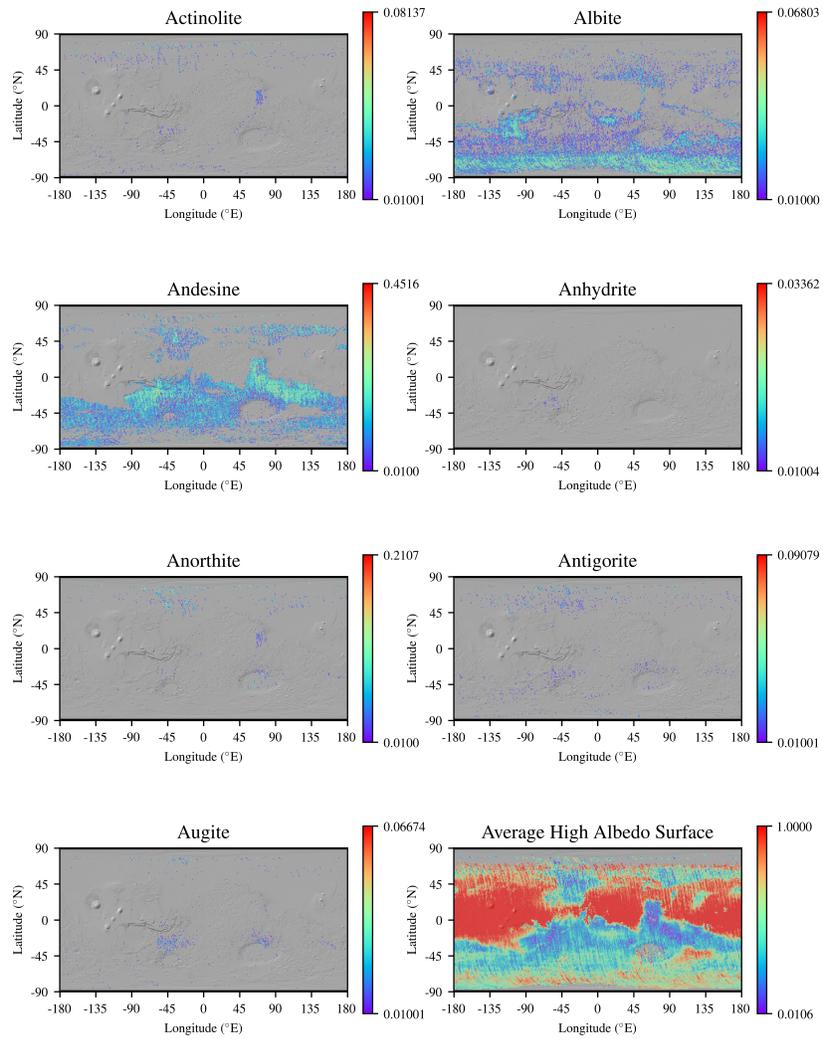


Fig. 4: Mineral Maps

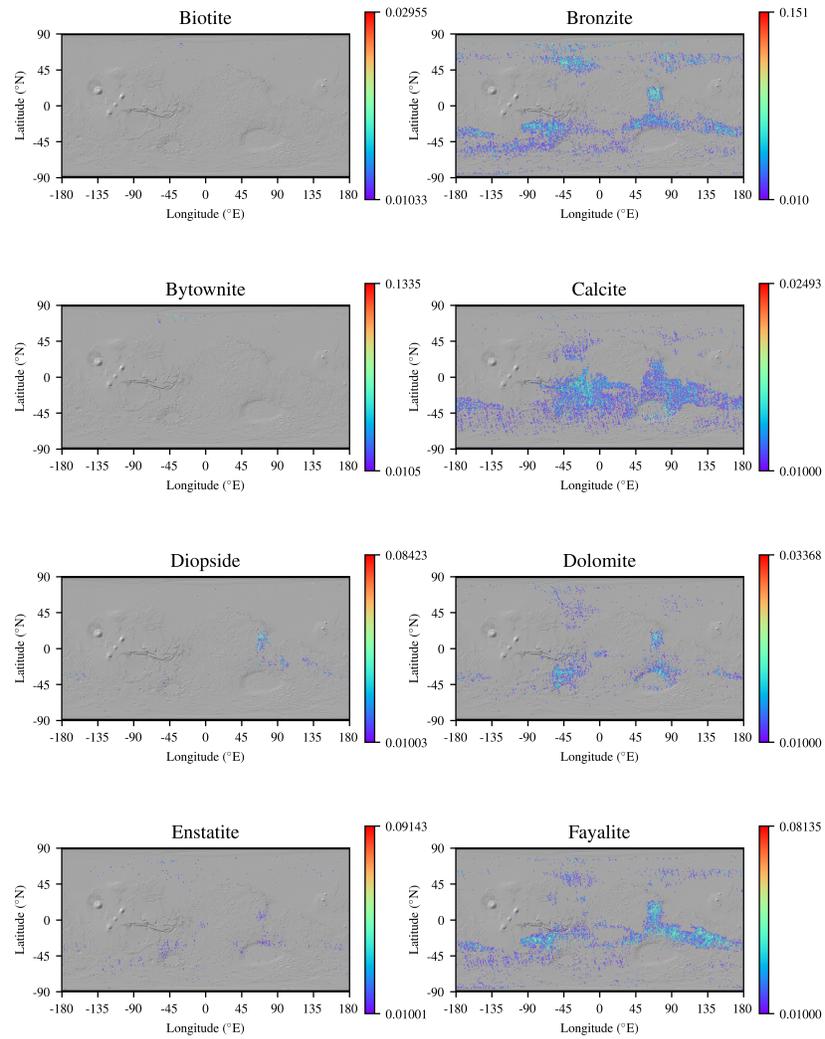


Fig. 5: Mineral Maps Continued

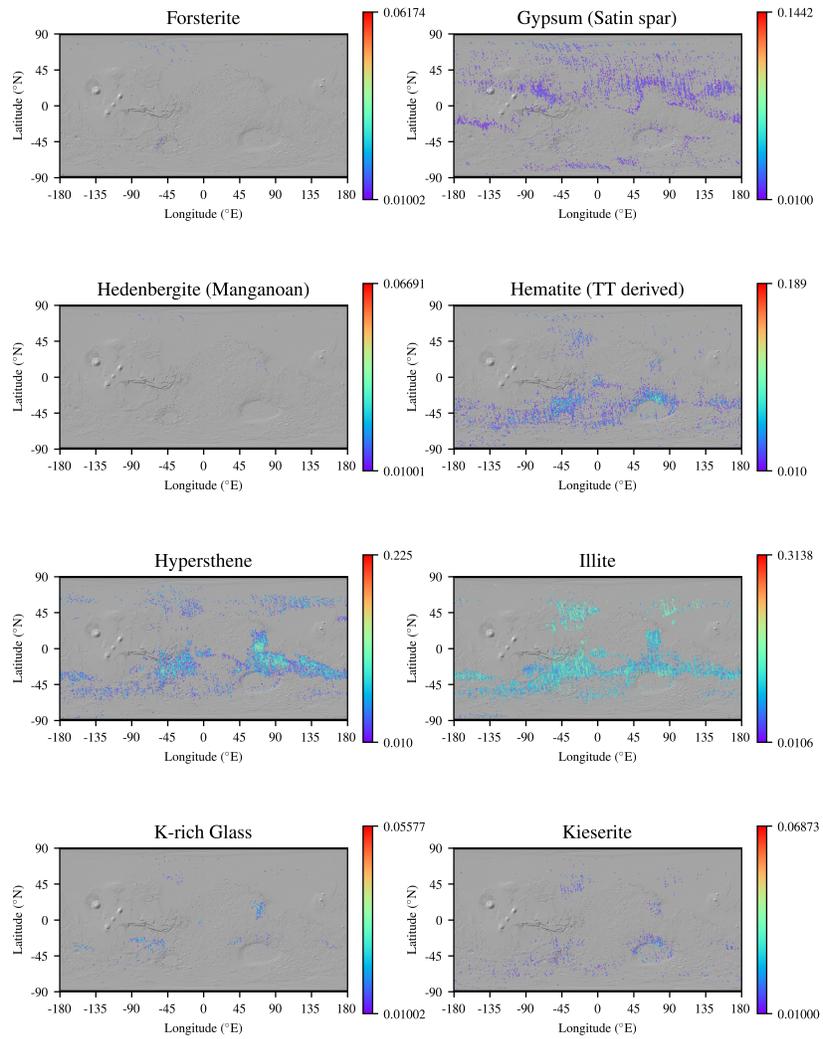


Fig. 6: Mineral Maps

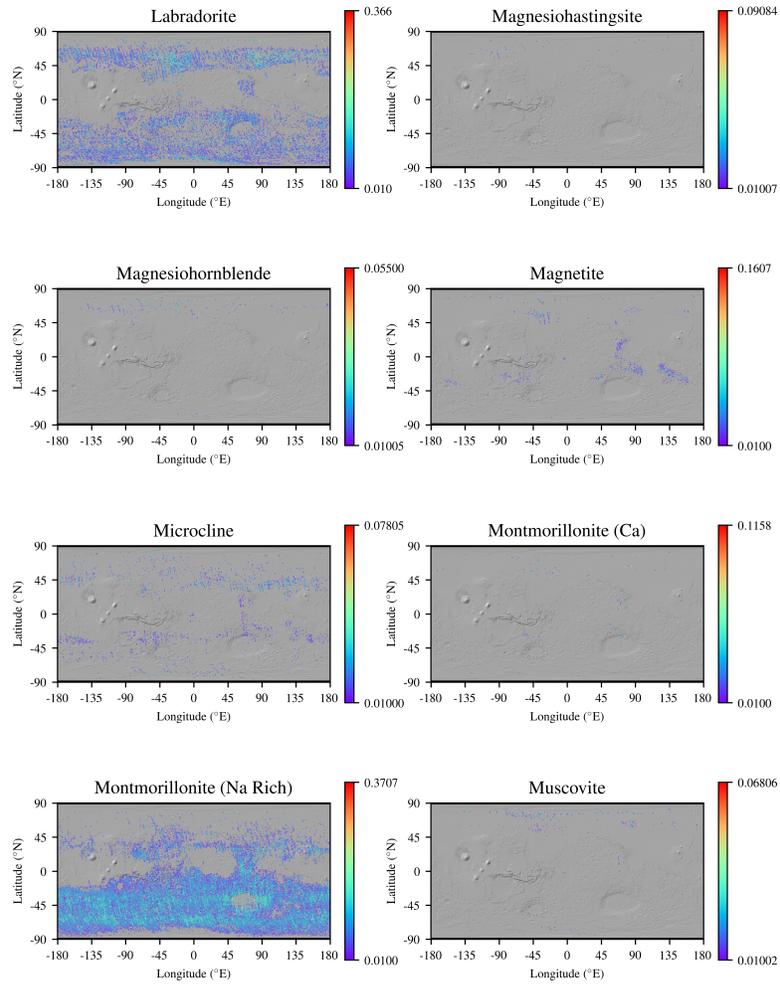


Fig. 7: Mineral Maps

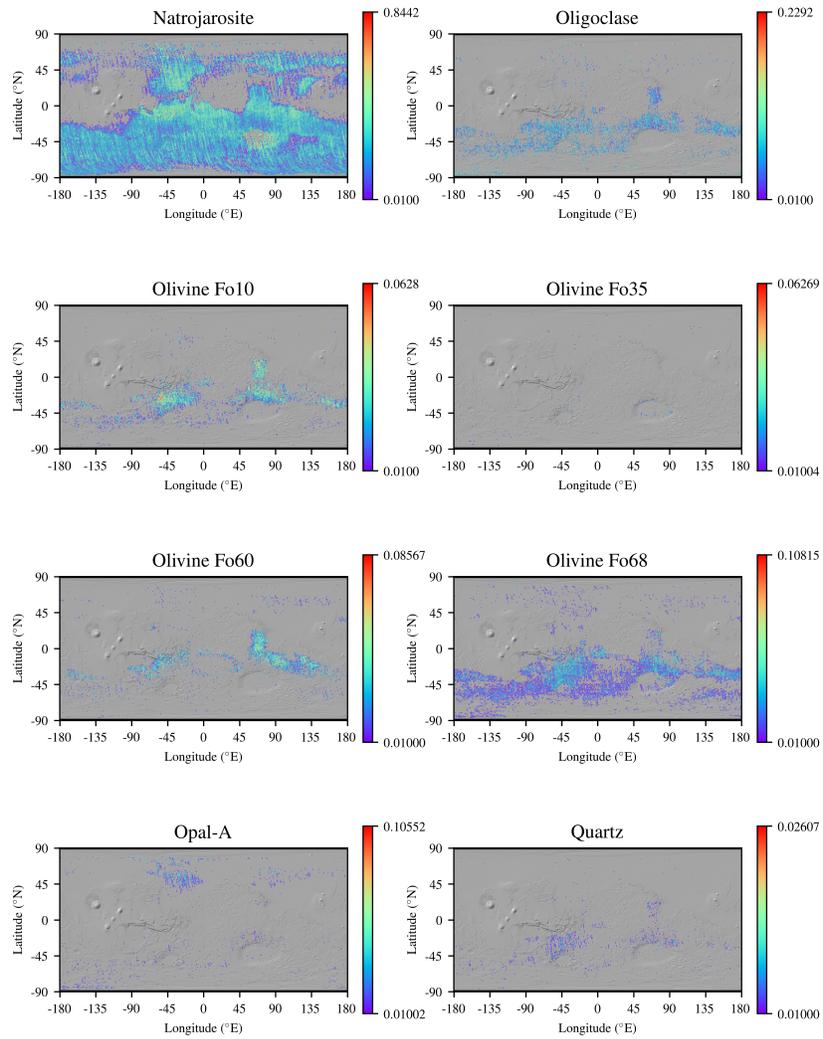


Fig. 8: Mineral Maps

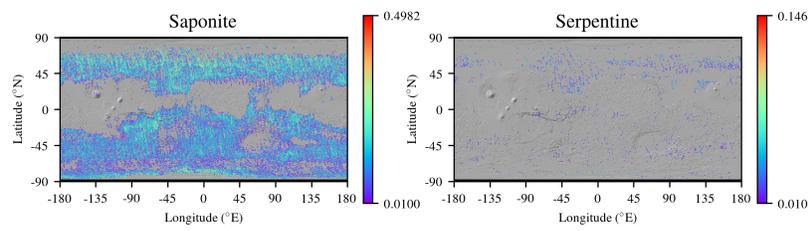


Fig. 9: Mineral Maps

## References

1. Bhojanapalli, S., Kyriillidis, A., Sanghavi, S.: Dropping convexity for faster semi-definite optimization. In: Conference on Learning Theory. pp. 530–582 (2016)
2. Chen, Y., Chi, Y., Fan, J., Ma, C.: Gradient descent with random initialization: Fast global convergence for nonconvex phase retrieval. *Mathematical Programming* **176**(1-2), 5–37 (2019)
3. 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)
4. 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)
5. Garbuny, M.: *Optical physics*. Optical Physics by Max Garbuny New York, NY: Academic Press, 1965 (1965)
6. Ge, R., Jin, C., Zheng, Y.: No spurious local minima in nonconvex low rank problems: A unified geometric analysis. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 1233–1242. JMLR. org (2017)
7. 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)
8. Jain, P., Kar, P., et al.: Non-convex optimization for machine learning. *Foundations and Trends® in Machine Learning* **10**(3-4), 142–336 (2017)
9. Spitzer, W., Kleinman, D.: Infrared lattice bands of quartz. *Physical Review* **121**(5), 1324 (1961)
10. Stein, D.: Application of the normal compositional model to the analysis of hyper-spectral imagery. In: IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data, 2003. pp. 44–51. IEEE (2003)
11. Sun, J., Qu, Q., Wright, J.: A geometric analysis of phase retrieval. *Foundations of Computational Mathematics* **18**(5), 1131–1198 (Aug 2017). <https://doi.org/10.1007/s10208-017-9365-9>, <http://dx.doi.org/10.1007/s10208-017-9365-9>
12. Thaker, P., Dasarathy, G., Nedi, A.: On the sample complexity and optimization landscape for quadratic feasibility problems (2020)
13. 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)
14. 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)