Proportion inference using deep neural networks. Applications to x-ray diffraction and hyperspectral imaging

Titouan Simonnet, Mame Diarra Fall, Bruno Galerne, Francis Claret, Sylvain Grangeon

Proportion inference using deep neural networks. Applications to X-ray diffraction and hyperspectral imaging

Titouan Simonnet [a,b], Mame Diarra Fall [a], Bruno Galerne [a,c], Francis Claret [b], Sylvain Grangeon [b]

[a] Institut Denis Poisson, Université d'Orléans, Université de Tours, CNRS, France

[b] BRGM, 3, Avenue Claude Guillemin, 45060 Orléans cedex 2, France

[c] Institut universitaire de France (IUF)

***Abstract*—Deep learning is considered as a disruptive method in the field of mineralogy and hyperspectral imaging. Many techniques exist to gain mineralogical information. Amongst them powder X-Ray diffraction (XRD) is very popular and powerful, while hyperspectral imaging is used in many applications such as Earth observation. A key issue for both XRD and hyperspectral imaging is not only to identify the endmembers constituting a mixture but also quantify the abundance of each endmember. In this study, we propose completly novel neural network (NN) training losses specifically designed for proportion inference. Extensive experiments illustrate that the proposed approach allows validated NN architectures to be trained to infer accurately on proportions.**

***Index Terms*—Proportion inference, Hyperspectral Unmixing, X-Ray Diffraction, Neural Networks, Dirichlet distribution.**

I. Introduction

Determining the nature of the mineral phases and their proportions in soil, sediments, and engineered materials is fundamental to understand and predict the chemical and mechanical properties of the studied materials. Such determination is most efficiently done with the help of powder X-ray diffraction (XRD). Methods such as the widely used Rietveld method [1] allow for an accurate and fast determination of the nature, crystal structure, and proportion of most mineral phases in samples. However, this method, which relies on a least squares algorithm, requires preliminary qualitative examination of XRD patterns to qualitatively identify all mineral phases, which may be extremely time consuming or even virtually impossible in case of large datasets such as those acquired during XRD-Computed Tomography [2]. There is therefore a genuine need to develop a method that allows for the identification and quantification of all phases contained in a mineralogical assemblage. Several recent contributions involve deep NN (DNN) [3] for mineral identification [4], [5], providing promising results for robust identification of mineralogical phases even in the presence of slight variations in crystallographic parameters (e.g. crystal size, morphology, and shape, and lattice parameters). In this work we focus on the proportion inference of mineral phases. That is given a XRD patterns, we infer the proportion of each involved phase.

While the XRD was our main motivation to introduce a method for proportion inference based on NN, other fields of application such as hyperspectral unmixing (HU) appear to be highly relevant. The HU aims at identifying the spectral signatures of the endmembers (i.e. the components) and their abundance vector (proportions are referred as abundances in the HU literature). This task is challenging and has been addressed in many contributions. Bioucas-Dias *et al.* [6] and Iordache *et al.* [7] propose an overview of many existing solutions. The most common methods are the SUnSAL algorithm [8] or methods derived from Non-negative matrix factorization (NMF) [9]. Wang and Jia [10] introduce an extended Support Vector Machine (eSVM) whereas Li, Chen and Rahardja [11] present a superpixel construction for HU.

Likewise, solutions involving NN are gaining popularity spurred by growth in computational resources. Auto Associative NN combined with Multilayer Perceptron have been proposed in [12]. Recent works focus on CNN architectures to solve the problem of abundance and endmember inference [13], [14].

In this work, we introduce a new training loss for NN-based proportion inference. This loss relies on a likelihood maximization for Dirichlet variates. The proposed modeling shares similarities with the work of Sensoy *et al* [15] that uses Dirichlet variates for assessing classification uncertainty in the context of evidential learning [16]. Our experimental results show that training with the proposed loss a NN architecture successful for XRD classification or HU yields to robust proportion inference.

The sequel of the paper is organised as follows. Section II presents the proposed Dirichlet modeling for proportion inference. Section III details our experimental setup for both XRD and HU applications. Results and comparisons with competing methods are presented in Section IV. We conclude in Section V with a brief discussion and some pointers for future research.

II. Methods

In the sequel, $K \geq 2$ represents the number of classes. We denote $\Delta_K = \{\mathbf{x} = (x_1, ..., x_K) \in \mathbb{R}^K | x_j \geq 0,\ j = 1, ..., K,\ \text{and}\ \sum_{j=1}^K x_j = 1\}$ the K-dimensional simplex, that is, the set of proportion vectors. We consider a training dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i), i = 1, \ldots, N_\mathcal{D}\}$, where each input signal

$\mathbf{x}_i$ lives in $\mathbb{R}^d$ and $\mathbf{y}_i \in \Delta_K$ is its associated ground-truth proportion vector. Given a similar input signal $\mathbf{x}$ we aim at inferring its corresponding component proportions $\mathbf{y}$ using a NN.

Classification is usually performed by maximizing the likelihood of a multinomial distribution parameterized by the NN response using a softmax layer. Here we will perform proportion regression maximizing the likelihood of the Dirichlet distribution parameterized by $\boldsymbol{\alpha} \in (0, \infty)^K$.

We denote $f(\cdot|\boldsymbol{\theta}) : \mathbb{R}^d \to \mathbb{R}^K$ a NN with parameters $\boldsymbol{\theta}$, and $\mathbf{a}_i = f(\mathbf{x}_i|\boldsymbol{\theta}) \in \mathbb{R}^K$ the NN output corresponding to the input $\mathbf{x}_i$. One should define the relationship between the output vector $\mathbf{a}$ and the Dirichlet parameters $\boldsymbol{\alpha} = \phi(\mathbf{a})$, where ϕ must comply with the strictly positive constraint of $\boldsymbol{\alpha}$. It should also be strictly increasing such that an output vector $\mathbf{a}$ with large values is mapped to a Dirichlet distribution with low variability. As in [15], we consider $\phi(\mathbf{a}) = \mathrm{ReLU}(\mathbf{a}) + 1$ that satisfies both conditions. We thus associate with any input data $\mathbf{x}_i$ the random variable

$$\mathbf{P}_i = (P_{i1}, ..., P_{iK}) \sim \mathrm{Dir}(\boldsymbol{\alpha}_i), \tag{1}$$

where $\boldsymbol{\alpha}_i = \phi(\mathbf{a}_i) = \mathrm{ReLU}(f(\mathbf{x}_i|\boldsymbol{\theta})) + 1$. The proportion prediction $\hat{\mathbf{y}}_i$ is given by the mean of the Dirichlet distribution $\mathrm{Dir}(\boldsymbol{\alpha}_i)$, that is,

$$\hat{\mathbf{y}}_i = \frac{\boldsymbol{\alpha}_i}{S_{\boldsymbol{\alpha}_i}} \in \Delta_K, \quad \text{with} \quad S_{\boldsymbol{\alpha}} = \sum_{j=1}^{K} \alpha_j.$$

A. Likelihood functions for proportion regression

a) Mean square error & Dirichlet: From the Dirichlet model previously introduced, we must provide a loss function to train the NN. One can first think at minimizing the Mean Square Error (MSE). That is for any input data $\mathbf{x_i}$, minimizing the expectation of the squares of the errors between $\mathbf{y_i}$ and $\mathbf{P}_i$, the random variable associated with the input data $\mathbf{x}_i$ in Eq. (1). This loss function $\mathcal{L}_i^{SE}(\boldsymbol{\theta})$ is expressed as follows,

$$\mathcal{L}_i^{SE}(\boldsymbol{\theta}) = \mathbb{E}(\| \mathbf{y}_i - \mathbf{P}_i \|^2) = \| \mathbf{y}_i - \hat{\mathbf{y}}_i \|^2 + \mathrm{Var}(\mathbf{P}_i) \tag{2}$$

where $\mathrm{Var}(\mathbf{P}_i) = \sum_{j=1}^{K} \mathrm{Var}(P_{ij})$. We will refer to this loss as "MSE & Dirichlet".

b) Cross-entropy & Dirichlet: An alternative is the cross-entropy (CE) loss between $\mathbf{y_i}$ and $\mathbf{P}_i$ given by

$$\begin{aligned}\mathcal{L}_i^{CE}(\boldsymbol{\theta}) &= \mathbb{E}\left(\sum_{j=1}^{K} -y_{ij} \log(P_{ij})\right) \\ &= \sum_{j=1}^{K} -y_{ij} \left[\psi(\alpha_i) - \psi(S_{\alpha_i})\right],\end{aligned}$$

where ψ is the digamma function. This loss will be named "CE & Dirichlet".

c) Alternative losses without Dirichlet modeling: We also consider three alternative losses. The first, named "MSE", computes the naive MSE using the positive part of the network output: $\mathcal{L}_i = \| \mathbf{y}_i - \mathrm{ReLU}(\mathbf{a}_i) \|^2$. The second, "CE & SoftMax", evaluates the cross-entropy using the $\mathrm{Softmax}$ of the network output $\mathcal{L}_i = -\sum_{j=1}^{K} y_{ij} \log(\mathrm{Softmax}(\mathbf{a}_{ij}))$. And finally the third, "MSE & proportion", defines the loss $\mathcal{L}_i = \| \mathbf{y}_i - \hat{\mathbf{y}}_i \|^2$, where $\hat{\mathbf{y}}_{ij} = \frac{\alpha_{ij}}{S_{\boldsymbol{\alpha}_i}}$, and $\boldsymbol{\alpha}_i = \mathrm{ReLU}(\mathbf{a}_i) + 1$. This loss corresponds to the first term of Eq. (2), that is the "MSE & Dirichlet" loss without the variance term.

B. Evaluation metrics

Let $\mathcal{T} = \{(\mathbf{x}_i, \mathbf{y}_i), i = 1, \ldots, N_{\mathcal{T}}\}$ denote the testing set, where $N_{\mathcal{T}}$ is the number of labeled data in each data set. To quantity the performance of the networks, we consider three metrics: the standard Root Mean Square Error (RMSE), together with the Mean Maximum Absolute Error (MMAE) and the Rate of Recovered Support (RRS) defined respectively as,

$$\mathrm{MMAE} = \frac{1}{N_{\mathcal{T}}} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{T}} \max_{j \in \{1, \ldots, K\}} |\widehat{y}_{ij} - y_{ij}|,$$

$$\mathrm{RRS} = \frac{1}{N_{\mathcal{T}}} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{T}} \mathbb{1}(\mathrm{supp}(\mathbf{y_i}) = \mathrm{supp}_{\varepsilon}(\hat{\mathbf{y}}_\mathbf{i})),$$

where $\mathrm{supp}(\mathbf{y}) = \{j \in \{1, \ldots, K\},\ \mathbf{y} > 0\}$ and $\mathrm{supp}_{\varepsilon}(\mathbf{y}) = \{j \in \{1, \ldots, K\},\ \mathbf{y}_j > \varepsilon\}$, for $\epsilon \in (0, 1)$. In the following experiments we will set $\epsilon = 0.01$, that is, a class is considered present if its estimated proportion is greater than 1%.

III. Experimental setup

A. Mineral phase identification from X-Ray diffraction

The elastic scattering of photoelectrons from an X-ray beam results in a scattered beam that is most commonly measured as a function of the scattering angle θ. The intensity of the scattered beam depends on three main components: the polarization factor, the structure factor, and the interference function [17]. Other parameters influence the diffracted intensity, for example atomic agitation, which is here accounted for by the Debye-Waller factor (isotropic agitation factor).

We recall that our goal is to find the proportions of the different phases given an XRD patterns. To this aim, a synthetic database of XRD patterns was generated using the simplest system of equations, i.e. by assuming that the crystals are perfectly 3D-ordered, have isotropic size, and have no lattice defect, resulting in 6000 XRD patterns of pure material for the phases Calcite [18], Dolomite [19], Gibbsite [20] and Hematite [21]. Then, 15000 XRD synthetic patterns of mixtures were created by combining one to four of the different mineral phases with a given proportion vector. Examples of XRD patterns are displayed in Fig. 1.

In addition, several real XRD patterns were acquired. Phase quantification from XRD data in real material samples usually requires a preliminary step of qualitative identification. We recorded 24 XRD patterns on pure minerals and on assemblages that are mixtures of the same mineral phases as in the synthetic dataset.

XRD patterns were acquired on micronized powders, with a Bruker D8 diffractometer, equipped with a LynxEye XE detector and a Cu anode ($\lambda = 1.5418\,\text{Å}$). The proportions of each phase was quantified by successive weightings. Data

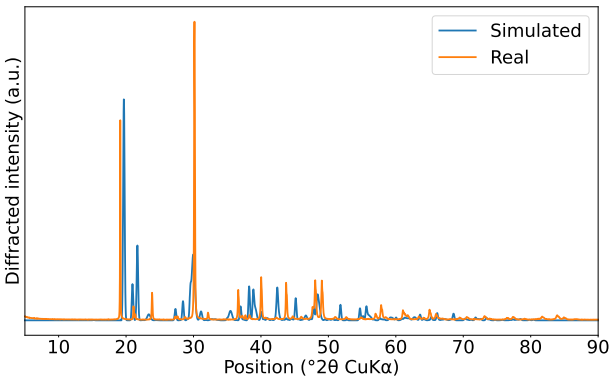


Fig. 1: Examples of XRD patterns: One from the simulated database (22% Calcite, 74% Gibbsite, 0.03% Dolomite, 0% Hematite), and one from real experiments (80% Calcite, 20% Gibbsite, 0% Dolomite, 0% Hematite).

were collected in a continuous scan mode, averaged every 0.04 $^\circ 2\theta$, and modelled with the Profex interface to the BGMN software [1].

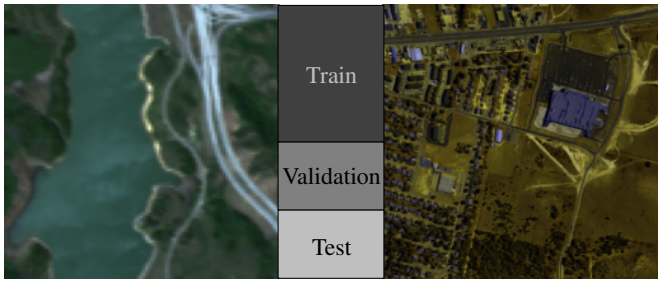
We trained a convolutional NN (CNN) following the architecture proposed in [22] for dimensionality and space group classification from XRD patterns. For each of the five loss functions introduced in Section II, we trained the CNN five times using 100 epochs (with the same initialization) and retained the most efficient network in terms of MMAE on the validation among all epochs. This is done to ensure a fair comparison of the best performances of the losses we investigate since we observed that those including a MSE term sometimes remains blocked in local minima. This can surely be solved by an adapted training policy or the use of normalization layers, but we chose to limit the complexity of the training procedure to avoid bias towards one of the losses.

B. Hyperspectral images

HU deals with inference proportion from spectral signatures. We used the Jasper Ridge and Urban images to conduct experiments on real data for which a ground-truth is provided[1]. The Jasper Ridge image has four endmembers: Road, Soil, Water and Tree. We analyzed a sub-image of 100x100 pixels for which the ground-truth has been established. On this image, the original spectra have 224 channels, but atmospheric effects and water vapour affect some spectral bands. Hence we reduced the data to 198 channels. Urban is a 307x307 pixels image with a spectral length of 162 (some channels were removed as for Jasper Ridge). We used the six ground-truth classes (Asphalt, Grass, Tree, Roof, Metal and Dirt). We splitted each image into three parts to constitute the training, validation and tests sets as shown in Fig. 2, in order to have a test set that is as independent as possible from the training set.

We used the 1D CNN architecture proposed in [13] for abundance estimation. As for the XRD experiment, the CNN

[1]Retrieved from https://rslab.ut.ac.ir/data



(a) Jasper Ridge (b) Urban

Fig. 2: Splits of the hyperspectral images for training and testing (RGB images obtained by averaging spectral bands).

was trained five times for each loss, and the reported results are the ones for the model with minimal MMAE over all epochs on the validation set.

IV. Numerical Results

A. Results on synthetic and real XRD data

The performance of the different trained models on the synthetic and the 24 real XRD patterns are reported in Table I.

For XRD simulated data, the loss functions MSE & Dirichlet and MSE & prop. provide the best results. It is worth mentioning that comparison with Rietveld refinement (detailed below) was not performed since the numerical approaches for building simulated data are, for 3D-ordered structures, similar to the calculation and hence minimization approach of the Rietveld software.

Analysing real XRD patterns allows to compare our results with the Rietveld refinement [1]. Two types of refinements were performed: in the first one, all geometrical and crystallographic parameters were allowed to vary. In the second, the refinement parameters were limited to the same as those considered when creating the database (norm of lattice vectors, Debye-Waller factors, and isotropic crystallite size). The first refinement is hence assumed to be representative of the state of the art while the second one allows for the most objective comparison with our NN method. However, some instrument parameters had to be left floated to allow for a proper functioning of the Rietveld refinement, namely a background determination function and a slight z sample displacement. It is hence not suprising that both standard and constrained Rietveld refinements are more efficient than our methods since the network was only trained on simulated data and Rietveld refinement had slightly more refinement parameters and aims at minimizing the root mean square error. Despite this, the losses using MSE provide satisfying performance. This confirms the interest of a NN-based method for XRD analysis. Indeed real data are affected by uncertainties from instrumental parameters (e.g. detector efficiency, source brightness, etc.). This makes the analysis of these real data harder than that of simulated data and also explains the performance gap, which is however of limited amplitude.

TABLE I: Results for XRD (measures in percentage)

	Simulated data			Real data		
Loss function	RMSE↓	MMAE↓	RRS↑	RMSE↓	MMAE↓	RRS↑
MSE & Dirichlet	**3.67**	**3.49**	87.1	6.07	8.23	91.7
CE & Dirichlet	4.57	4.21	90.8	9.10	11.0	83.3
CE & Softmax	4.75	4.36	**92.6**	7.22	8.12	**95.8**
MSE	4.18	4.81	86.2	**5.20**	7.77	83.3
MSE & prop.	3.74	3.54	88.5	5.53	7.26	87.5
Stand. Rietveld	Depends on the sample			**1.31**	**2.07**	**100**
Const. Rietveld	(see text IV-A)			1.74	3.12	100

TABLE II: Results for both hyperspectral images

	Urban			Jasper Ridge		
Loss function	RMSE↓	MMAE↓	RRS↑	RMSE↓	MMAE↓	RRS↑
MSE & Dirichlet	4.50	5.68	55.7	2.48	**2.34**	**80.3**
CE & Dirichlet	5.44	7.77	51.1	2.90	3.09	77.8
CE & Softmax	4.95	6.48	43.7	4.03	3.87	59.5
MSE	5.07	8.13	**59.7**	3.29	4.21	74.0
MSE & prop.	**4.17**	**5.45**	56.1	**2.34**	2.37	**80.3**
SUnSAL	19.7	28.89	17.5	7.81	7.83	51.4
HyperAE	36.3	54.79	6.73	25.9	32.18	37.5
UnDIP	28.1	52.0	0.68	18.8	22.7	15.1

B. Results on hyperspectral unmixing data

The five proposed loss functions introduced in Section II were compared with three competitive methods. The first one is the SUnSAL algorithm [8], a constrained sparse regression method[2] based on the alternating direction method of multipliers. It allows to obtain the fractional abundance of components in each pixel of an hyperspectral image, without using ground-truth data. We also compare our method with two deep-learning based methods: HyperAE[3] [23] and UnDIP[4] [24]. HyperAE proposes a NN autoencoder that extracts both the endmembers spectral signatures and the fractional abundances in each pixel. UnDIP is a two-step algorithm that first extracts the endmembers using a geometric method, and second estimates the abundances using deep image prior.

Table II summarizes the results on both Jasper Ridge and Urban images. Results obtained on the Jasper Ridge basis highlight the effectiveness of the proposed two loss functions, namely MSE & Dirichlet and MSE & prop. The three other losses also perform well on the error measurements and the RRS. The results on the Urban dataset are quite similar to those obtained on Jasper Ridge, MSE & Dirichlet and MSE & prop. provide the best results in terms of minimum errors. For both images, the SUnSAL, UnDIP and HyperAE methods perform relatively poorly compared to the proposed NN models.

These results can be illustrated by comparing the ground-truth abundances and the predicted ones. Fig. 3 depicts the ground-truth (first row) and the absolute difference between this truth and the prediction obtained with the two best performing loss functions MSE & Dirichlet (second row) and MSE & prop. (third row) as well as the UnDIP and SUnSal methods (resp. in the fourth and fifth rows). One can observe that the results are quite good and close for both proposed losses. As expected, the areas where the predictions reach 30% of error are the borders between different endmembers. In comparison, the SunSal method produces larger errors in the same areas while the UnDIP method outputs wrong predictions for the Dirt and Road classes.

[2]https://github.com/Laadr/SUNSAL
[3]https://github.com/dv-fenix/HyperspecAE
[4]https://github.com/BehnoodRasti/UnDIP

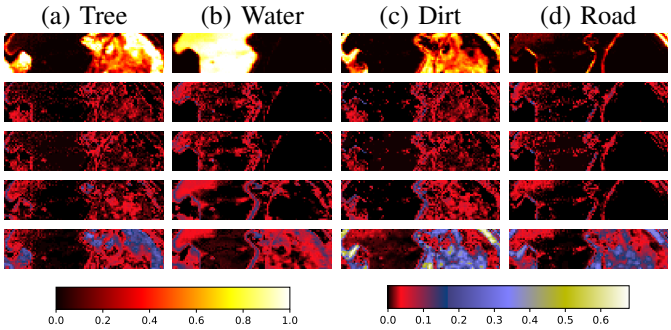


Fig. 3: For each class of Jasper Ridge: ground-truth (GT) (1^{st} row), absolute difference between GT and prediction resp. with Dirichlet & MSE (2^{nd} row), MSE & prop (3^{rd} row), SUnSAL (4^{th} row) and UnDIP (5^{th} row).

V. Conclusion and Perspectives

We have proposed a NN training loss based on Dirichlet modeling that allows to quantify phases in a mixture. The proposed method has been successfully applied on synthetic and real XRD and HU data. One of the strengths of our method is its ease of use. Indeed there is no need to create a specific architecture for the network. One can use a validated NN that is appropriate for the studied data, and train the model with the Dirichlet loss function for proportion inference.

In addition, to our knowledge, machine learning based approaches for XRD analysis are limited and do not allow an exact estimate of proportion. Those approaches are reduced to classifications problems, by identifying space groups and crystal dimensionality [5], [22], or by sorting the percentages of abundance of the different classes [4].

Therefore, these methods lead to results that cannot be used operationally in a mineralogy laboratory. On the contrary, the method we propose combines the advantages of a deep method while providing results close to those of Rietveld. We believe that the remaining performance gap between Rietveld and our method stems from a combination of more refinement parameters for the Rietveld method and the difference between the real and simulated data.

Last but not least, our NN-based alternative opens the way for the automatic analysis of a large amounts of signals involving in settings where the Rietveld algorithm generally fails, e.g. when a large amount of phases is involved. This

would undoubtedly be a key advantage in the case of XRD-computed tomography [2], which typically contains millions of XRD patterns. Current work is undergoing in this direction.

REFERENCES

[1] N. Doebelin and R. Kleeberg, "Profex: a graphical user interface for the rietveld refinement program bgmn," *Journal of applied crystallography*, vol. 48, no. 5, pp. 1573–1580, 2015.

[2] F. Claret, S. Grangeon, A. Loschetter, C. Tournassat, W. De Nolf, N. Harker, F. Boulahya, S. Gaboreau, Y. Linard, X. Bourbon, et al., "Deciphering mineralogical changes and carbonation development during hydration and ageing of a consolidated ternary blended cement paste," *IUCrJ*, vol. 5, no. 2, pp. 150–157, 2018.

[3] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016, http://www.deeplearningbook.org.

[4] J-W. Lee, W. B. Park, J. H. Lee, S. P. Singh, and K-S. Sohn, "A deep-learning technique for phase identification in multiphase inorganic compounds using synthetic xrd powder patterns," *Nature communications*, vol. 11, no. 1, pp. 1–11, 2020.

[5] W. B. Park, J. Chung, J. Jung, K. Sohn, S. P. Singh, M. Pyo, N. Shin, and K-S Sohn, "Classification of crystal structure using a convolutional neural network," *IUCrJ*, vol. 4, no. 4, pp. 486–494, 2017.

[6] J. M Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 354–379, 2012.

[7] M-D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 6, pp. 2014–2039, 2011.

[8] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2010, pp. 1–4.

[9] Paris V Giampouras, Athanasias A Rontogiannis, and Konstantinos D Koutroumbas, "Low-rank and sparse nmf for joint endmembers' number estimation and blind unmixing of hyperspectral images," in *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, 2017, pp. 1430–1434.

[10] L. Wang and X. Jia, "Integration of soft and hard classifications using extended support vector machines," *Geoscience and Remote Sensing Letters, IEEE*, vol. 6, pp. 543 – 547, 08 2009.

[11] Zeng Li, Jie Chen, and Susanto Rahardja, "Superpixel construction for hyperspectral unmixing," in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 647–651.

[12] G. A Licciardi and F. Del Frate, "Pixel unmixing in hyperspectral data by means of neural networks," *IEEE transactions on Geoscience and remote sensing*, vol. 49, no. 11, pp. 4163–4172, 2011.

[13] X. Zhang, Y. Sun, J. Zhang, P. Wu, and L. Jiao, "Hyperspectral unmixing via deep convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 11, pp. 1755–1759, 2018.

[14] L. Wan, T. Chen, A. Plaza, and H. Cai, "Hyperspectral unmixing based on spectral and sparse deep convolutional neural networks," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 11669–11682, 2021.

[15] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential deep learning to quantify classification uncertainty," *Advances in neural information processing systems*, vol. 31, 2018.

[16] A. Amini, W. Schwarting, A. Soleimany, and D. Rus, "Deep evidential regression," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, Eds. 2020, vol. 33, pp. 14927–14937, Curran Associates, Inc.

[17] D. L. Bish and J. E. Post, *Modern powder diffraction*, Walter de Gruyter GmbH & Co KG, 1990.

[18] S. A. Markgraf and R. J. Reeder, "High-temperature structure refinements of calcite and magnesite," *American Mineralogist*, vol. 70, no. 5-6, pp. 590–600, 1985.

[19] H. Steinfink and F. J. Sans, "Refinement of the crystal structure of dolomite," *American Mineralogist*, vol. 44, pp. 679–682, 1959.

[20] E. Balan, M. Lazzeri, G. Morin, and F. Mauri, "First-principles study of the oh-stretching modes of gibbsite," *American Mineralogist*, vol. 91, no. 1, pp. 115–119, 2006.

[21] R. L. Blake, R. E. Hessevick, T. Zoltai, and L. W. Finger, "Refinement of the hematite structure," *American Mineralogist: Journal of Earth and Planetary Materials*, vol. 51, no. 1-2, pp. 123–129, 1966.

[22] F. Oviedo, Z. Ren, S. Sun, C. Settens, Z. Liu, N. T. P. Hartono, S. Ramasamy, B. L. DeCost, S. I. P. Tian, G. Romano, et al., "Fast and interpretable classification of small x-ray diffraction datasets using data augmentation and deep neural networks," *npj Computational Materials*, vol. 5, no. 1, pp. 1–9, 2019.

[23] B. Palsson, J. Sigurdsson, J. R. Sveinsson, and M. O. Ulfarsson, "Hyperspectral unmixing using a neural network autoencoder," *IEEE Access*, vol. 6, pp. 25646–25656, 2018.

[24] B. Rasti, B. Koirala, P. Scheunders, and P. Ghamisi, "Undip: Hyperspectral unmixing using deep image prior," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2021.