Deep Learning Midterm Report

Jon Calvin Wetzel
Department of EECS
University of Tennessee
Knoxville, Tennessee
jwetzel2@vols.utk.edu

Wilson Amemewon Garuba
Department of Chemistry
University of Tennessee
Knoxville, Tennessee
wgaruba@vols.utk.edu

Dong Jun Woun
Department of EECS
University of Tennessee
Knoxville, Tennessee
djwoun@gmail.com

Jonathan Edward Skeen
Department of EECS
University of Tennessee
Knoxville, Tennessee
jskeen6@vols.utk.edu

***Abstract*—Raman Spectroscopy decomposition can identify and quantify individual chemical components in mixtures. Traditional methods like Multivariate Curve Resolution (MCR) are disadvantageous in various aspects, including the overlapping signal resolution problem. In this paper, we describe a novel approach for spectral decomposition using Convolutional Neural Networks (CNNs), which aim to overcome the issues of the current state-of-the-art method and efficiently process thousands of mixture spectra. Utilizing a dataset that mocks Martian soil analogs, our research utilizes various deep-learning techniques. We preprocess the data by applying baseline correction, cosmic ray removal, and spectra smoothing. The model is fine-tuned based on the mean squared error loss function and cosine similarity metrics. Our CNN model can potentially offer a solution for accurately analyzing chemicals, with potential applications in Martian soil analysis.**

***Index Terms*—raman spectra, decomposition, deep learning, convolutional neural network**

I. Introduction

Raman spectroscopy (RS) is a vibrational technique that is used to provide the molecular 'fingerprint' of chemical structures. RS is a non-destructive technique that determines the vibrational modes of molecules through inelastic scattering of light. Because of its high specificity, it can be used to identify the presence of chemicals in samples and then provide the chemical and physical properties of the compounds. However, complex spectra of mixture need to be unmixed for us to qualitatively and quantitatively measure the individual components present in mixture. Therefore, the need to develop algorithms or mathematical models that can unmix complex spectra is indispensable.

Several techniques have been developed to decompose complex spectra to identify individual components. A method that decomposes spectra based on peak decomposition was initially developed [1]. However, this method fails to explain the problem of overlapping peaks. In the case where two or more components have similar peak positions and sizes, the technique would not be able to distinguish the components. Multivariate Curve Resolution (MCR) has been used for spectral unmixing for a very long time [2]. It has been reported to be very effective in Raman spectral decomposition. However, incorporating physical and chemical constraints is essential for the process to be feasible. Like many other techniques, MCR can also be faced with problems of noise and overfitting. These challenges have limited the use of MCR in the application of component identification, especially in medical application where specificity and sensitivity are paramount.

Machine learning and deep learning algorithms have also been used in complex spectral decomposition [3]. These algorithms can learn from the mixture data and reconstruct the spectra of each individual component present in the mixture. There are many successful cases where these have been shown to perform better than the conventional methods. However, there is need to optimize similar deep learning networks for better performance and accuracy.

In this study, we apply a Deep neural network architecture in the spectral decomposition. We aim to develop a Convolutional Neural Network (CNN) that takes spectra of mixture compound and outputs the spectra of individual components present in the mixture. Our network architecture aims to achieve similar performance metrics found in reported techniques for spectra decomposition.

II. Literature Review

Current literature within the domain of Raman Spectroscopy analysis features two neural network approaches that aid with decomposition. The classification approach attempts to label the individual components that exist in the mixture. As a result, Raman spectra data can be inputted to the classification model which can then output the labels of the individual components of the mixture. This is particularly useful for foreign mixtures, such as those found on neighboring planets e.g., Mars. While this approach aids researches during the early stages of analysis, it fails to discover finer details, such as component concentration or the spectra of the individual components. Consequently, the decomposition approaches have incorporated Convolutional layers to decompose the mixture spectra and output the individual component spectra. From these component spectra, researchers can discover information about individual component spectra/concentrations and decrease time expended through typical manual analysis techniques.

"Application of deep learning and spectral deconvolution for estimating mineral abundances of zeolite, Mg-sulfate, and montmorillonite mixtures and its implications for Mars" written by G. Kodikara, et al. discusses their approach to using deep neural networks for component mineral abundance estimations within reflectance spectrum of mineral mixtures. As shown in Fig. 1, their model was a 3-layer fully connected

network with three outputs; one for each component of the mixture.

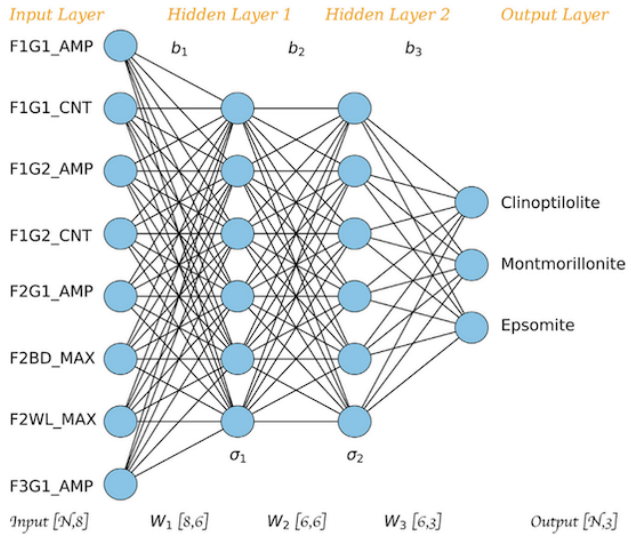

Fig. 1. 3-Layer network with an output for each mixture component [2].

After training their model, and performing some hyperparameter tuning, the researchers found the "two hidden layer architectures (with 6 hidden units) with ADAM optimization, Sigmoid activation function, and a 0.001 learning rate model gave the best results with lower RMSE" [4]. Of particular note, the authors mention adding noise to their dataset as a regularization technique. The authors concluded their research with the ability to approximate component concentrations using the 3-layer, fully-connected neural network.

Whereas the previous research focused on using neural networks for component concentration approximations, the "Deep Learning-Based Extraction for Improving the Performance of Surface-Enhanced Raman Spectroscopy Analysis on Multiplexed Indentification and Quantitation" by J. Zhang, et al. focused on constructing the individual component spectra of the mixture through Convolutional layers. More specifically, the researchers preprocessed their dataset, which consisted of mixtures with three components, and fed the dataset to their neural network which decomposed the data into individual components (Fig. 2).

To achieve this, the model consisted of a processing unit and three extractor units; one for each component. Each unit contained two Conv1D layers and two Batch Normalization layers to prevent gradient diffusion. The output of each extractor unit was passed to a fully connected layer with a sigmoid activation, which restored the component data back to the original spectra dimensions (Fig. 3).

J. Zhang, et al. utilized several evaluation and performance metrics that should be incorporated into our future model. Similar to G. Kodikara's, et al. approach, J. Zhang, et al. used the Adam optimizer for their optimization function and mean squared error (MSE) for their loss function. Since J. Zhang, et al. were concentrated on decomposing the mixture

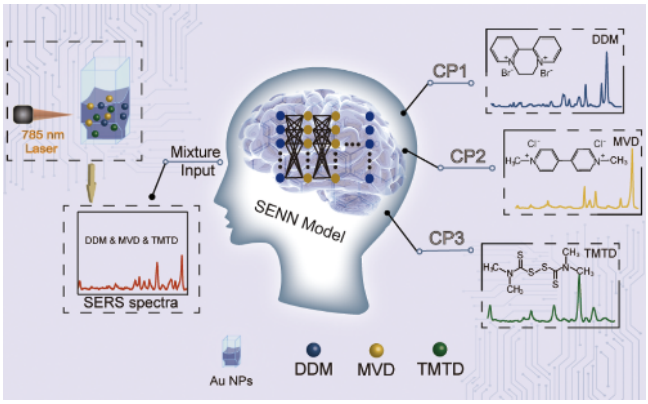

Fig. 2. Conceptual Overview of the Model's Input and Output [4].

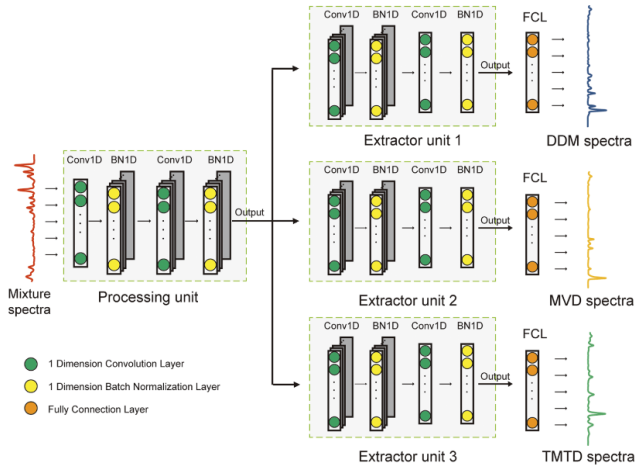

Fig. 3. Mixture Spectra Decomposition to Individual Component Spectra [4].

spectra into its individual component spectra, they measured the accuracy of the decomposition data (component spectra) using cosine similarity (1), which measures the similarity between two vectors i.e., how closely two vectors point in the same direction.

$$cos_{similarity} = \frac{s \cdot x}{max(\|s\|_2 \cdot \|x\|_2, \epsilon)} \tag{1}$$

Therefore, a comparison between the model-generated, component spectra, s, and the pure, ground truth, component spectra data, x, served as measure for the neural network's decomposition accuracy. In other words, the model, if accurate, should generate component spectra that has a cosine similarity score close to one; their model was able to produced cosine similarity scores > 0.97. Combined, the MSE and cosine similarity scores served as the primary metrics for model accuracy and success. These studies have illustrated neural networks' promising potential for Raman spectra decomposition and concentration analysis.

III. Technical Approach

Based on current research publications, our model will attempt to mimic J. Zhang's, et al. approach except with self-

gathered experimental dataset. Our model will create a central processing unit and a number of extractor units that matches the number of individual components in our experimental mixtures. Like their model, these units will contain Conv1D layers and Batch Normalization layers to minimize the gradient issues discussed in their paper. Our model will also employ Adam optimization and mean squared error (MSE) as our loss function. Most importantly, we will assess the model's decomposition effectiveness using the cosine similarity score, which compares the similarity between the model's output spectra and the ground-truth, component spectra. Therefore, our ultimate goal is to maximize the cosine similarity score of the model's output so that our model may accurately output the component spectra of input, mixture spectra.

Since the activation functions and learning rates varied amongst the literature, we intend to tune our model's learning rate, vary the types of activation functions e.g., Relu, tanh, and Leaky ReLU, and other various tweaks to optimize the cosine similarity score. Given the results of both papers, a $MSE < 0.3$ and a $cos_{similarity} > 0.90$ would indicate a successful model build and should be the anchor points for final model evaluation.

IV. Baseline Selection

A. Baseline Models

A new state-of-the-art Raman Decomposition model can improve the analysis of Raman spectra data, particularly for unknown mixtures. In order to create a state-of-the-art Raman Decomposition deep learning model, it is important to select robust, recent, and competitive baselines. The criteria used for the selection of baselines were the ability to accurately identify components of minerals, the cosine similarity between the extracted pure spectra and their respective spectra, and models that trained with datasets with numerous mixtures of minerals and spectra of the minerals. The ability to accurately identify components of minerals is important because it can enhance the selectivity and sensitivity of the processing unit. G. R.L. Kodikara's paper identifies that a DNN architecture with 8 x 6 x 6 x 3 and an optimizer as ADAM with Sigmoid can achieve a Root Mean Square Error as low as 0.176. Where the processing unit can refine the spectra and extract features used for the extractors, and the extractor unit can extract the spectra of individual components of the chemical. Figure 3 displays J. Zhang's processing unit refining data, which the extractor can utilize to compute the individual spectra.

The cosine similarity between the extracted pure spectra and their respective spectra can measure the similarity between two vectors. So, a high cosine similarity indicates the likeness of two graphs. So, a high cosine similarity between the extracted spectra and human-made spectra, indicates that the deep learning model can extract spectra at a human level. Figure 4 shows that J. Zhang's cosine similarity of their models, which achieved 0.999, 0.997, and 0.994 similarities for each subsequent type of mixture.

The utilization of various mixtures and spectra of minerals is important because it increases the variance of the model.

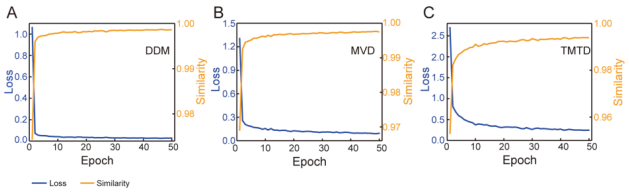

Fig. 4. Training outcomes for DDM (A), MVD (B), and TMTD (C). The blue line represents the Mean Squared Error (MSE) loss during the model's gradient descent steps, while the yellow line depicts the cosine similarity during the same steps. [4]

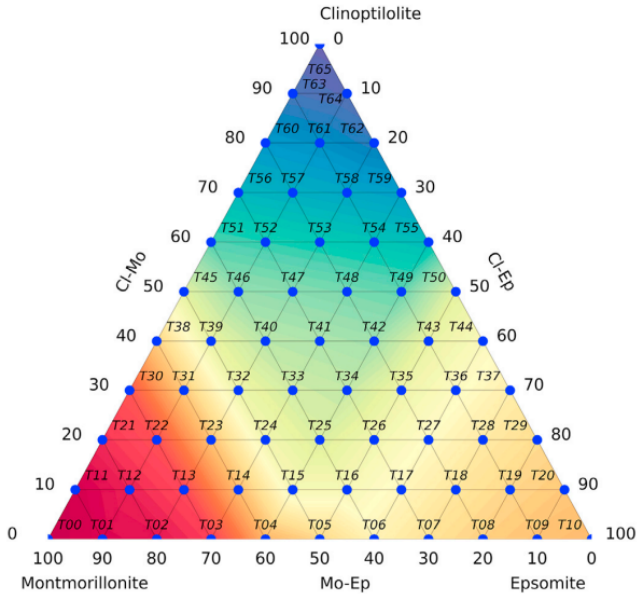

Fig. 5. A diagram depicting mixtures of clinoptilolite, montmorillonite, and epsomite is illustrated schematically with ternary axes. [5]

Models that only train on small datasets often easily overfit and fail to generalize to real-world data. So, models with low variance would not be useful for scientists with the purpose of decomposing Raman spectra. We identified that J. Zhang, et al. utilized mixtures of 3 minerals with 3600 spectra of various ratios of the minerals [4], and G. R.L. Kodikara, et al, utilized 66 mixtures with 66 spectra [2]. Figure 5 identifies the various complex ratios of each chemical component in G. R.L. Kodikara's paper. These datasets offer a plethora of variance in data.

B. Dataset

1) Experimental Setup and Data Collection: The confocal Raman microscope is designed to be used with multiple continuous wave (CW) lasers in the Near-IR and the visible region: a 785 nm diode laser, a 632.8 nm helium-neon (He-Ne) laser, and two optically pumped semiconductor lasers (OPSLs) at 532 nm and 488 nm. The optical path to the microscope (Figure 6) has been optimized with all lasers aligned on a common beam path to a single mirror that directs the beam through a set of irises (I1 and I2). Two focusing lenses, FL1 and FL2 direct the laser to a periscope, with mirrors that

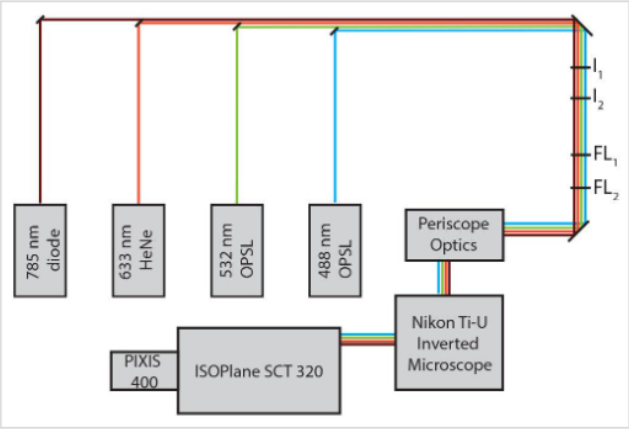

Fig. 6. Diagram of the micro-Raman system that includes four lasers, a Nikon inverted microscope, spectrometer, and CCD camera.

change the direction of the light from parallel to perpendicular to the laser table, before entering the back of a microscope. Via the Nikon Ti-U inverted microscope, the sample on the stage is illuminated through an objective from below, with the aid of a dichroic mirror to direct the light to the objective. The microscope is equipped with a 20x, 40x and 100x oil immersion objectives (Nikon, Melville, NY). The numerical aperture for these objectives is 0.45, 0.60, and 0.5 – 1.3 respectively. Before entering the ISOPlane SCT 320 spectrometer, the backscattered light from the sample is directed through a focusing lens and a long-pass filter corresponding to the excitation wavelength of interest to assist in blocking the unwanted Rayleigh scattered signal and to transmit the Raman scattered light from the sample. In the spectrometer, the light is dispersed and detected by a PIXIS 400 CCD camera (Princeton Instruments). The sample mixture to be measured contains three Martian soil analogs , Gypsum, Olivine and Soapstone. Mixture dataset of about 3000 raman spectra was collected. Raman spectra of individual components were aslo collected. This serves as our ground truths.

2) Data Preprocessing: The data preprocessing step involves baseline correction, cosmic ray removal and spectra smoothing. The baseline correction function was implemented by calling the pybaseline function from the scikit learn library. Savizky golay filter was applied for smoothing and noise reduction. The spectra were clipped or truncated between 600 – 1600 cm^{-2}, leaving out the useless part of the spectra that have no meaningful contribution to our analysis. A min-max scaling function was developed for data normalization.

C. Model

Our problem involves decomposing a mixture spectra into individual component spectra. Thus, when choosing a neural network that would best work in this scenario, we would this network to have the ability to learning features of the component spectra, so that it can detect when these samples are present in the mixture spectra. CNN's are almost the perfect candidate for this problem, since they typically excel in feature extraction. A roadblock arises, however, since we would also like to output the component spectra themselves. This is where the addition of the fully-connect layer (FCL) comes into play. The FCL can take channels from the Convolution layers as input and create a full component spectra as output, thus fulfilling our needs.

When considering Convolution layers, there are different types which work best in different scenarios. For us, we have chosen 1-dimensional Convolution layers since our dataset consists of 2-dimensional graphs / spectra.

The last missing piece to our model architecture is batch normalization. Adding Batch Norm layers to the model will help the model converge more quickly and also serves as a regularization technique, which in turn, reduces the chance of overfitting.

Thus, our model will consist of three network layer types: Convolutional layers, Batch Norm, layers, and Fully-Connected layers. The convolution section should extract features which will identify component spectra. The Batch Norm section add a regularization effect and help with the model's ability to learn. Lastly, the Fully-Connected section will take the features and create the component spectra as its output.

V. Experiments and Results Analysis

A. Experiments

Once the data was pre-processed, it was split into training and test sets using Scikit-Learn's *train test split* method. 90% of the data was given to the training set and 10% to the testing set, which resulted in 900 training examples and 100 validation examples. The training data and associated labels appear in Fig. 7, 8, 9, 10.

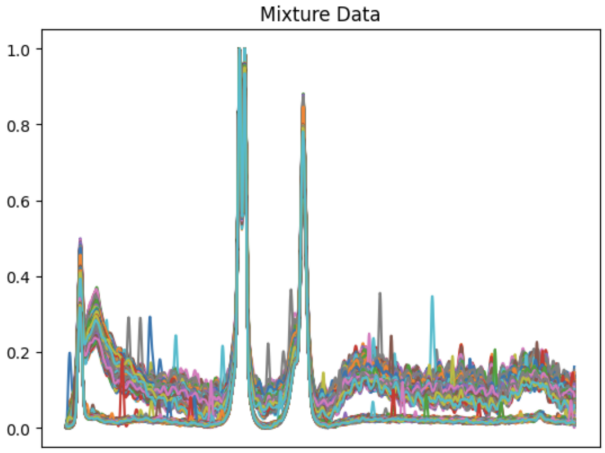

Fig. 7. Raman Spectra of Mixture Training Data

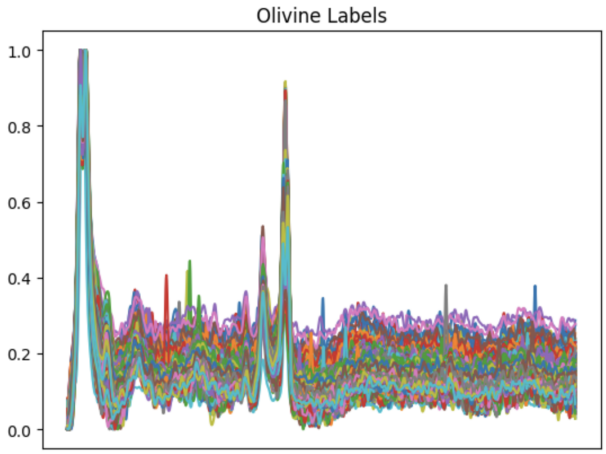

Fig. 8. Olivine Raman Spectra Labels

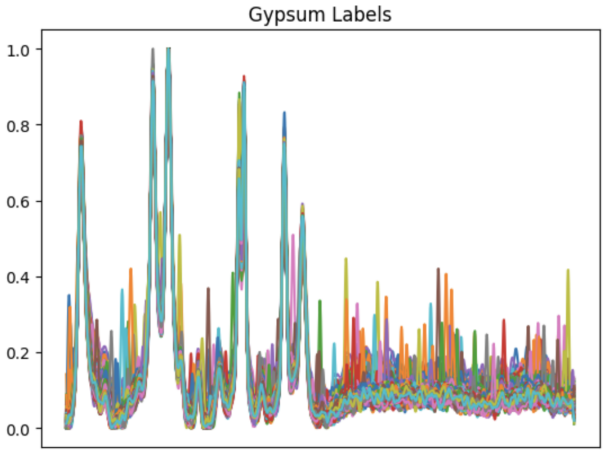

Fig. 9. Gypsum Raman Spectra Labels

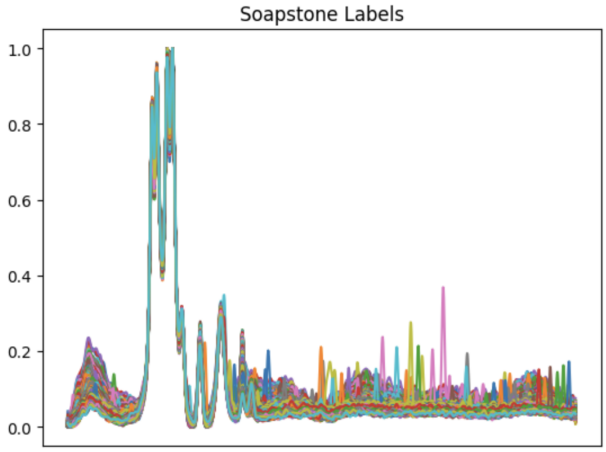

Fig. 10. Soapstone Raman Spectra Labels

Of the several trials run, the best hyperparameters found were the following: lr = 0.0001, Adam optimizer, MSE loss function, and a 90/10 train-test split. The following plots illustrate the similarity and loss trends as the model was trained.

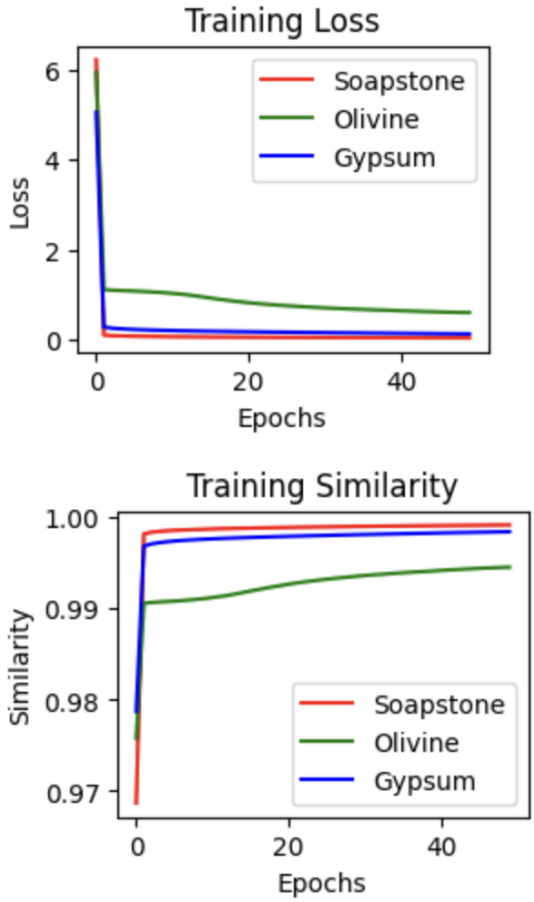

Fig. 11. Training Loss and Similarity over 50 epochs

The model's training loss and similarity values after 50 epochs can be found in Table I.

TABLE I: Training Loss and Similarity after 50 Epochs

Substance	Loss	Similarity
Soapstone	0.0497	99.92%
Olivine	0.6098	99.46%
Gypsum	0.1381	99.85%

The model achieved accurate similarity scores but began to slightly overfit the training data towards the latter epochs. This was evident through the validation plots shown in Fig. 12. Although the model featured some slight over fitting, it was able to decompose a mixture sample into the soapstone, olivine, and gypsum components with all individual component similarities above 92% as shown in Table II.

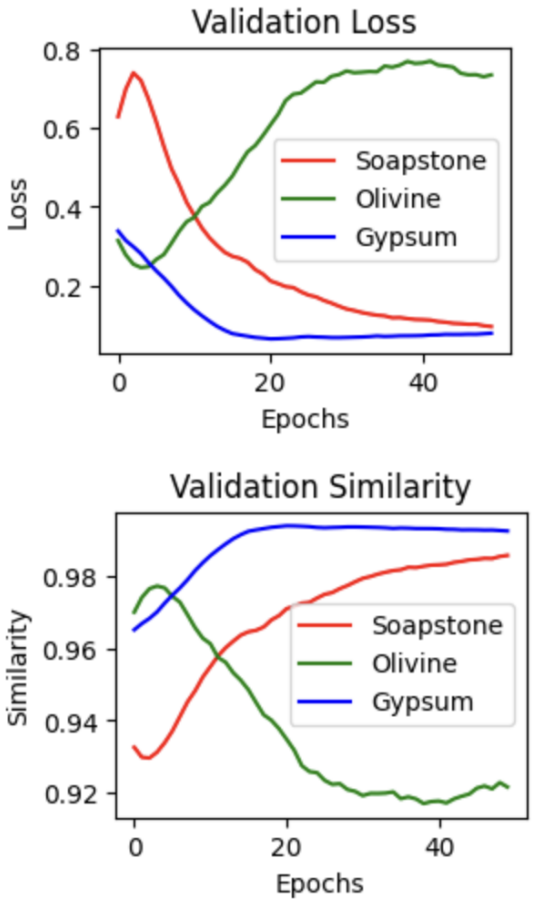

Fig. 12. Validation Loss and Similarity over 50 epochs

TABLE II: Validation Loss and Similarity after 50 Epochs

Substance	Loss	Similarity
Soapstone	0.0948	98.56%
Olivine	0.7365	92.16%
Gypsum	0.0777	99.23%

As evident in both the training and validation results, the model's mixture decomposition into Olivine featured the worst performance of the components present in the mixture. Returning to Fig. 8, the drop in performance is likely attributed to the vertical spread of the Olivine intensity labels. The model struggled to accurately decomposed the mixture into an Olivine predictions, since the loss attributed to each prediction-label pair varied significantly across the Olivine intensity labels. To improve this in future studies, the mixture training data and labels should be experimentally recollected (improve experimental data collection error) or combined with a larger dataset i.e. approx. 2,500 samples.

Considering the small, self-collected dataset (1,000 samples), the model performed above expectations and effectively decomposed a mixture sample into its components as shown in Fig. 13. Of particular note and illustrated in Fig. 14, the model accurately predicted the peaks of each component; the largest source of error appeared in the noise between the labels and predictions.

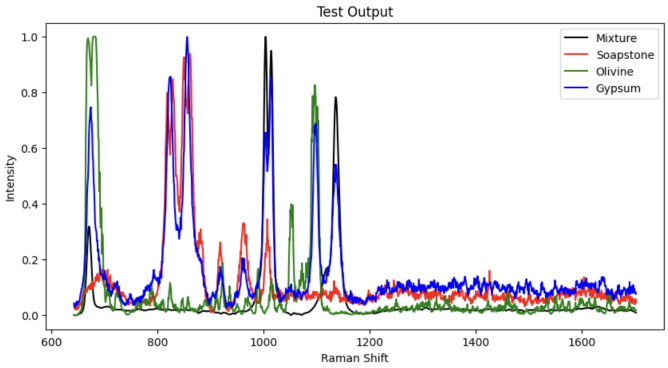

Fig. 13. Model Component Predictions based on Individual Mixture Sample.

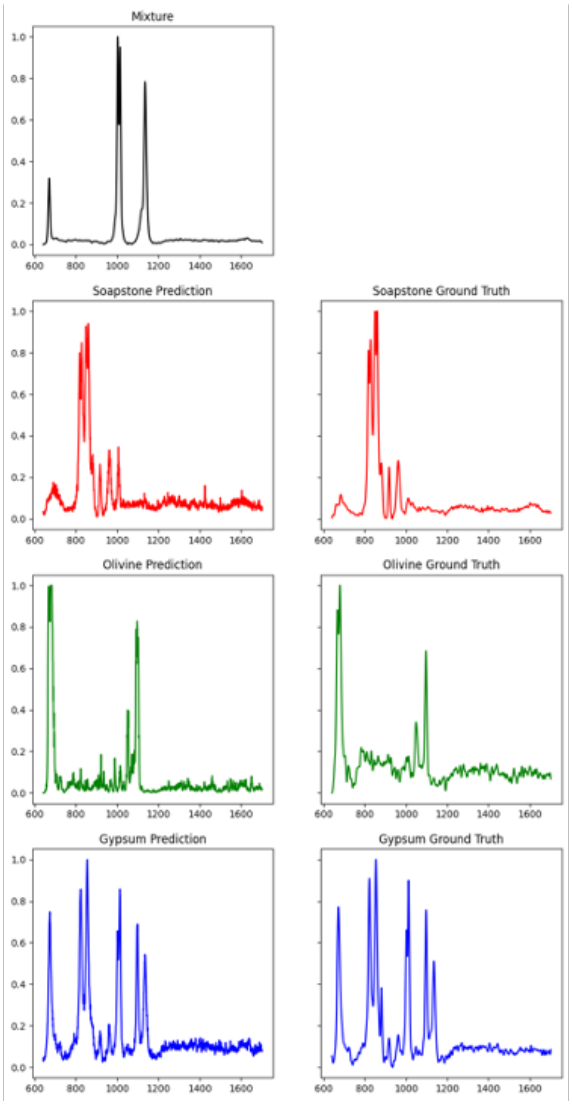

Fig. 14. Mixture Sample, Individual Component Predictions, and associated component labels from Fig. 13.

VI. Conclusion

Current literature reviews indicate neural networks can effectively decompose Raman mixture spectra. Convolutional networks are used to analyze the spectra data. The model output is then evaluated through MSE and cosine similarity scores, from which the model's optimization function can determine the model's next best learning adaptations. By iterating over a preprocessed dataset, the model learned from the training data and decomposed Raman spectroscopy samples into individual components to within 92-98% similarity. While the model performed well, there were several key takeaways from this project. First, the data collection and preprocessing stages are the most vital and time consuming phases of production. Inaccuracies in the data collection process, which results in data bias, directly affect the model. If not realized in the preprocessing stages, these data issues may seemingly stem from the model but actually originate from the data itself. Futhermore, to gain an accurate population representation, the dataset should contain more than 1,000 samples; likely closer to 2,500 samples as echoed within the Statistics field. Once the data is accurately collected, the model architecture is not difficult to understand, code, or debug thanks to libraries such as NumPy, PyTorch, etc. The final stages can then involve hyperparameter tuning across a range of trials to determine combination that produces the best results. Overall, the project served as great launch pad into the field of deep learning; more specifically, how to conduct a literature review, build a model, and use it to improve daily research tasks.

VII. Appendices

A. Code Design

1) Data preprocessing: A series of data preprocessing steps were taken to clean and transform the raw data. The standard normal variate normalization, min-max scaling, t-distributed Stochastic Neighbor Embedding, and Principal Component Analysis were considered for data preprocessing. However, after an analysis of the accuracy, only the min-max scaling and smoothing preprocessing steps were incorporated for the final model. The min-max scaling technique transforms features by scaling each feature between 0 and 1. This is done by subtracting the minimum value of a feature and then dividing it by the range of the feature.

2) Model Blueprint: Our model consists of three components, PreprocessUnit, ExtractorUnit, and SENN. The PreprocessUnit preprocesses the input, consisting of two convolutional layers followed by batch normalization and ReLU activation functions. The purpose of the unit is to extract features from the input. The ExtractorUnit extracts features from the preprocessed data, consisting of two convolutional layers followed by batch normalization and ReLU activation functions and a final linear layer followed by a sigmoid activation function. Finally, these two components are combined into the SENN. The SENN model takes one preprocessing unit and a variable number of extracting units. The model essentially processes the input spectrum and then passes the processed data through multiple extractor units to return the extracted components.

3) Model Training: The model was trained with the Adam optimizer, 0.0001 learning rate, 50 epochs, and Mean Squared Error loss function. The model was evaluated using the Cosine Similarity metric. For each epoch, the training phase processes the three components soapstone, olivine, and gypsum, and calculates the losses and similarities. The similarity and loss of the training and validation set are recorded in each epoch. The trained model is then trained again until the set number of epochs is reached.

4) Plotting: Various plots were created to evaluate the results. The training and validation's loss and similarity were plotted to view the loss and similarity of each epoch. Additionally, a plot was created to visualize the predicted spectroscopy output against the mixture data. The plot gave a visual assessment of how well the model could separate and predict the individual components in a mixed sample. Finally, the test output was plotted against the ground truth data, showcasing the model's accuracy for each material.

B. Workload Distribution

Calvin Wetzel: Wrote the literature review, technical approach, experiments, results, and conclusion. Created the model, training, and plotting functions. Experimented with standard normal variate normalization, min-max scaling, t-distributed Stochastic Neighbor Embedding, and Principal Component Analysis preprocessing steps. Created a section of the video report.

Dong Jun Woun: Wrote the abstract, baseline selection, and appendix. Started coding the model, and reconfigured the code to allow GPU utilization. Trained the model with 4 datasets, where two data datasets had a pre-processed and raw version. Tuned various hyperparameters, including the optimizer, learning rate, and the number of epochs trained. Learning rates were trained with log scaling, and the number of epochs was trained with linear scaling. The stochastic gradient descent optimizer, RMSprop optimizer, and the Adam optimizer were tried. Achieved 99.99% training accuracy and 96.65%, where previously the model achieved lower 90% and lower 60% accuracy on the training and testing set. Created a section of the video report.

Wilson Garuba: Wrote the introduction, literature review, and baseline selection. Created the dataset of the mixtures with soapstone, olivine, and gypsum. Provided two datasets, where one consisted of the raw dataset and the other was preprocessed. Created a section of the video report.

Jonathan Skeen: Wrote the Baseline implementation. Created a section of the video report. Recorded and clipped the project video.

References

[1] Chen, Y. and L. Dai, A Method for the Quantitative Analysis of a Key Component in Complex Mixtures Using Raman Spectroscopy Based on Peak Decomposition. Analytical Sciences, 2019. 35(5): p. 511-515

[2] De Juan, A., J. Jaumot, and R. Tauler, Multivariate Curve Resolution (MCR). Solving the mixture analysis problem. Analytical Methods, 2014. 6(14): p. 4964-4976.
[3] Pan, L., et al. Noise reduction technique for Raman spectrum using deep learning network. in 2020 13th International Symposium on Computational Intelligence and Design (ISCID). 2020. IEEE.
[4] J. Zhang, P.-L. Xin, X.-Y. Wang, H.-Y. Chen, and D.-W. Li, "Deep Learning-Based Spectral Extraction for Improving the Performance of Surface-Enhanced Raman Spectroscopy Analysis on Multiplexed Identification and Quantitation," J. Phys. Chem. A, vol. 126, no. 14, pp. 2278-2285, 2022. DOI: 10.1021/acs.jpca.1c10681.
[5] G. R. L. Kodikara, L. J. McHenry, and F. D. van der Meer, "Application of deep learning and spectral deconvolution for estimating mineral abundances of zeolite, Mg-sulfate and montmorillonite mixtures and its implications for Mars," Planet. Space Sci., vol. 223, 2022, Art. no. 105579. DOI: 10.1016/j.pss.2022.105579.