APPLICATION OF MACHINE LEARNING TO IDENTIFY SURFACE MINERALS IN CRISM IMAGERY.

J. Caggiano[1], A. M. Sessa[2], J. J. Wray[2], and C. S. Paty[1], [1]Department of Earth Sciences, University of Oregon, Eugene, OR 97403 (jcaggian@uoregon.edu), [2]School of Earth and Atmospheric Sciences, Georgia Institute of Technology, Atlanta, GA 30332.

Introduction: Visible and Near-Infrared (VNIR) spectroscopy (0.3-3 μm) has been utilized by both Observatoire pour la Minéralogie, l'Eau, les Glaces et l'Activité [1] and Compact Reconnaissance Imaging Spectrometer for Mars (CRISM) [2] to elucidate the composition of the martian surface through the detection of diagnostic absorptions attributed to iron-related electronic processes, as well as overtones and combinations of fundamental stretching and bending vibrations (e.g., metal-OH, H_2O, and CO_3^{-2}) [3,4].

Aided by the revised spectral parameters formulated by [5], these instruments have observed a wide array of minerals indicative of various formation environments once present on ancient Mars [6,7,8,9]. A RGB composite, or "browse product", is produced by combining three parameters, which allows for an observation-by-observation inspection of the spatial distribution of key mineralogic absorptions. While this type of manual inspection is suitable for a small subset of observations, it is untenable for a global or even regional large-scale spectroscopic survey of the surface.

Automations of this process for large CRISM datasets using methodologies differing from the one employed here have been demonstrated previously by [10-13]. However, the proposed program will use a Deep Neural Network (DNN) to simplify the automation process and enhance the versatility of surface mineral research. A neural network approach has been used to determine the temperature and single scattering albedo for each pixel in a group of CRISM scenes from 1-3.8 μm [14]. Unlike [14], the purpose of our program is to perform a whole-image survey to identify specific minerals in CRISM observations using defined parameters. Figure 1 shows a visualization of some surface mineral assemblages that the program is attempting to detect.

Methods: For the purpose of this abstract, two images from the Martian north polar region were utilized for the accuracy evaluation, but larger datasets will be presented at the conference. Specifically, we utilized unprojected full-resolution targeted (FRT) products with identical pixel dimensionality with a spatial sampling of 15-19m/pixel. TER (Targeted Empirical Records), which are post-processed using the pipeline highlighted in [15], were used in this study.

The classifier program was developed in the Python language and is written with two primary modules: an image parsing and preprocessing module, and a DNN classifier module. A special challenge for the image parser/preprocessor module is the formatting of the CRISM datasets. The image data are contained in two component files: a .img file, which contains the pixel data, and a .hdr file, which contains information about the file, such as size, channel index names, etc. Our program utilizes the GDAL package to open and read the image files [16]. The files are then processed into a 3-dimensional array. Each channel is then scanned by the preprocessor module using a very slow, yet highly accurate scanning algorithm to determine if an outcrop area is present. The scan parameters such as area and sensitivity are defined by the user when the program is run. After scanning, the image data are compressed into a 2-dimensional array and saved to the local hard drive along with the scan results. The scanning algorithm is only used for the purpose of creating training data for the DNN and evaluating the accuracy of the program overall. Only the parser portion of the module will be needed for mainstream use.

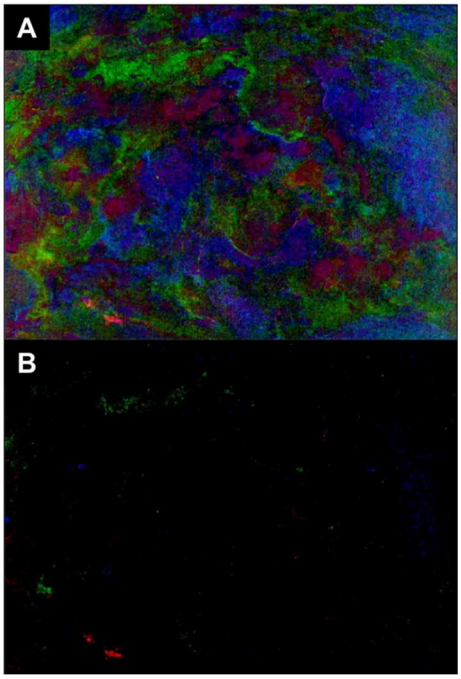

Figure 1: *Sample mineral indices from FTR0000406B (R:BD1300, G:OLINDEX3, B:SINDEX2). (A): The top 80% of pixel index values. (B): The top 0.1% of pixel index values.*

The DNN classifier module is developed using TensorFlow. The saved data from the preprocessor are loaded and reshaped to the original 3-dimensional array. The DNN algorithm compares its evaluations for each channel to the results of the scan from the preprocessor algorithm and assesses the accuracy for each epoch. The

DNN then uses the accuracy to correct and optimize the neural network weights for each new epoch. The neural network produces a binary output for two classes: the image channel shows an outcrop for a corresponding index (1), or it does not (0). The neural network will be optimized for accuracy by iterating through different batch sizes and iterations. To test the effect of neural network hidden-layer structure on the accuracy of the DNN module, the neural network layers are each tested at different values, also using the optimized batch size and iteration data from the previous evaluation.

Accuracy Results: The program was run with batch sizes ranging from 5 channels to 30 channels per batch in increments of 5 channels, and the number of epochs ranging from 1 to 20. At best, the accuracy of the program's training data was 100%, and the test data was 91%. The accuracy results from the final iteration of each parameter were printed by the program and were then used for calibration and demonstrative purposes. Figure 2 shows the accuracy of the DNN module as a function of the batch size and number of iterations. From these plots, it becomes clear how consistent the training data and test data are with regards to accuracy. It is also obvious that the accuracy of the DNN module occasionally, and apparently randomly, drops to zero for both the training and test data.

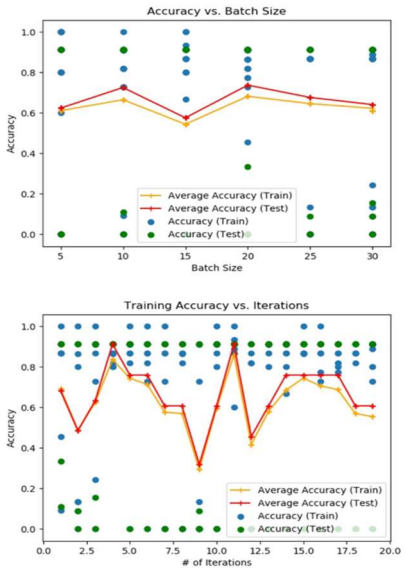

Figure 2: *Variations of accuracy with respect to batch size (top) and the number of epochs used (bottom).*

To further optimize the DNN module, the module was evaluated again, iterating through different number of neurons present in the three hidden layers. This second evaluation was performed using 100 iterations and a batch size of 60. This evaluation showed that a simpler neural network appears to be more favorable over more complex neural networks in regards to accuracy. These results are consistent with the average accuracy of the batch and epoch variation tests.

Discussion: Referencing Figure 1, since the average accuracy of each batch and each iteration are strongly varying, but not consistent, it appears the accuracy of the DNN module itself is not dependent on these two variables. A potential means of improvement for the DNN is to address the generalized nature of the data. The mineral outcrops do not appear in the same areas of each image, nor do they follow the same visual pattern, which is difficult for the DNN to properly predict. This may be resolved by using more images to train the DNN in the future, which may increase the accuracy by further generalizing the weights of the neuron inputs.

Furthermore, the DNN module of the network is unable to work with images of different sizes, which is problematic for analyzing all the CRISM data, because the CRISM images come in a variety of sizes. Using a resizing algorithm on the images would be problematic for accuracy. However, this can be resolved by converting to a Convolutional Neural Network (CNN), a type of Neural Network developed mainly for image analysis. This should be more versatile with regards to image size, and should produce more accurate results.

Conclusions: If the improvements discussed above are implemented in the program, the neural network classifier will be ready for widespread use among researchers needing such a tool. Many of these changes should be complete by the time of presentation. Future work on this program, in addition to implementing the improvements above, could include expanding the usability of the program to other types of CRISM data, including the ability to classify mineral assemblages based on spectral parameters, which will allow for more versatility for identifying rare minerals.

References: [1] Bibring J. P. et al. (2005) *Science*, *307*(5715), 1576-1581. [2] Murchie S. et al. (2007) *JGR, 112,* E05S03. [3] Clark R. N. et al. (1990) *JGR*, *95*(B8), 12653-12680. [4] Clark, R. N. (1999) *Manual of Remote Sensing, 3,* John Wiley and Sons, New York, p 3- 58. [5] Viviano-Beck C. E. et al. (2014) *JGR: Planets*, *119*(6), 1403-1431. [6] Bishop J. L. et al. (2008) *Science*, *321*(5890), 830-833. [7] Mustard J. F. et al. (2008) *Nature*, *454*(7202), 305-309. [8] Ehlmann B. L. et al. (2009) *JGR*, *114*, E00D08. [9] Wray J. J. et al. (2009) *Geology*, *37*(11), 1043-1046. [10] Carter J. et al. (2013) *Planet. Space Sci.*, *76*, 53-67. [11] Allender E. and Stepinski T. F. (2017) *Icarus*, *281*, 151-161. [12] Thomas N. H. and Bandfield J. L. (2017) *Icarus*, *291*, 124-135. [13] Amador E. S. et al. (2018) *Icarus*, *311*, 113-134. [14] Powell K. E. et al. (2018) *LPS XLIX*, Abstract #2113. [15] Seelos F. P. et al. (2011) *LPS XLII*, Abstract #1438. [16] OSGeo (2018), GDAL/OGR, Website, https://www.osgeo.org/projects/gdal/.