

**ACTIVE CLASSIFICATION OF NEUTRAL SPECTRA FOR CRISM IMAGES** Arun M. Saranathan and Mario Parente, Department of Electrical & Computer Engineering, University of Massachusetts Amherst, Amherst, MA 01003

**Introduction:** Spectra acquired by planetary hyperspectral imagers like the Compact Reconnaissance Imaging Spectrometer for Mars (CRISM) are affected by the atmospheric conditions, dust/debris in the scene and instrument artifacts [1] [2]. In spite of best efforts to minimize the effects of these distortions in the CRISM pipeline some noise and distortions/artifacts persist in the CRISM product. Practitioners commonly attempt to simultaneously improve the spectral contrast of various Regions Of Interest (ROIs) and mitigate the effects of the atmosphere and other artifacts by manually ratioing the average spectra of a ROI with spectra from an “spectrally unremarkable (neutral) region” [3] (an example is shown in Fig 1 (A)). Neutral Spectra (NS) are defined as “spectra from spectrally unremarkable regions in the scene” [3] and are expected to contain only “background continuum and noise artifacts that are common across the entire image” [1]. For push-broom spectrometers such as CRISM the identification of neutral regions is complicated by column-specific distortions effects, making it essential to choose NS from the same column as the spectra of interest.

For large image databases it quite infeasible to perform manual identification of NS for each ROI and an automated approach for finding the NS would be very useful. While the notion of a NS is quite intuitive, automating the process of finding appropriate NS for a specific column is quite complicated. The major hurdle in this process is that “suitability” of a specific spectrum as a NS is “left to intuition and observable desirable effects” [1]. Some attempts have been made to use the column-mean or column-median as the NS, e.g. [2], such attempts are successful if only a few pixels in the column contain useful information. In many cases such techniques average out useful mineral bands present in the spectra (as shown in Fig. 1 (B)). Another alternative technique based on superpixels attempts to find the flattest spectrum in the column, by fitting a line to each  $L1$ -normalized mean super-pixel segment spectrum [1]. While the technique finds an approximately flat spectrum as the NS, such spectra often contain significant noise and other local distortions which it may introduce into the ratio (as shown in Fig. 1 (C)).

Since NS are described and analyzed qualitatively, it very hard to find an objective metric to evaluate the quality of a NS. In the absence of such a metric/quantitative analysis the most viable path to identifying a “good” NS spectra would be to learn a classification model from a teacher (expert). We propose a

method to learn a model for classifying as a candidate NS as “good” (+1) or “bad” (−1).

**Active Learning Based Classifier for Neutral Spectrum:** The quality of a NS is evaluated subjectively and on a case-by-case basis, due to the subjective nature of the labeling it is expected that the classifier will have to learn a complicated boundary. For complicated problems the number of labeled examples needed to achieve good classification performance is quite large. This is a problem as expert labeling is expensive (in terms availability and the expert’s time). To minimize the number of examples needed for training the classifier we use the *selective sampling* technique [4]. *Selective sampling* is an *active learning* [5] algorithm in which the learner interactively chooses the most informative data-points for labeling, thus reducing both labeling-cost and number of examples needed for training. In particular we use the *Query-by-committee* [6] approach, wherein a committee of models are trained on the same training-set (set of labeled examples) and the next training examples is chosen by the “*principle of maximal disagreement*” [6].

The scheme of the algorithm is described in Algorithm 1. If we are given a large set of unlabeled examples  $X$  as the input, the algorithm begins by splitting the dataset  $X$  into  $n$  smaller subsets of size  $k$  (with the first subset containing the first  $k$  samples and so on). Following this the algorithm (randomly) initializes  $p$  different models for the classifier. Now for each subset the algorithm uses the models to predict the labels for the all the samples in  $X_i$ . The algorithm then identifies the unlabeled point on which there is maximal differences (in a binary classification problem this would be the point which half the classifiers classify as positive the other half as negative). The algorithm then queries the teacher for the label of this example, this labeled example is then added to the training set  $T$ . All the different models are trained using this training set. The number of passes fixes the number times the algorithm iterates over the entire dataset. Such a framework is advantageous and the model can be trained whenever the expert is available and over time can be improved with a special focus on problem areas.

**Experiments:** In this study we considered 30 TRR3 CRISM cubes acquired by the IR (L) detector and in particular we focus on 230 spectral bands between 1.112 and 2.622  $\mu m$ . [Note: All images were acquired prior to 2009]. Simple atmospheric correction is performed using the CRISM Analysis Toolkit (CAT)

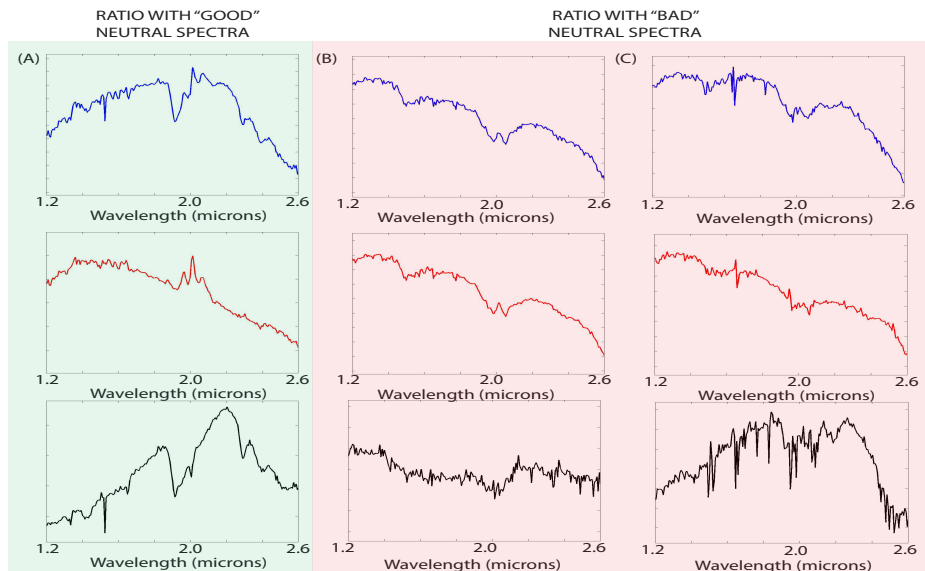


Figure 1: The effect of ratioing the Top row (blue): displays the original spectrum, Middle Row (red): the NS identified by automatic techniques and the Bottom row (black): shows the ratioed spectra (A) an example where the ratioing works and enhances the spectra features while removing a lot of the spectral distortions (B) an example where ratioing with a column-mean eliminates surface spectral information and (C) an example where the ratio introduces significant distortions into the spectra

**Data:** A dataset  $X$  with  $N$  unlabeled samples;

**Result:** An Classification model;

Split the Dataset  $X$  into  $n$  smaller subsets

$[X_1 X_2 \dots X_n]$  with  $k$  samples each;

Randomly initialize  $p$  classification models

$[\theta_1 \theta_2 \dots \theta_p]$ ;

**for**  $pass=1: \text{number of passes}$  **do**

**for**  $i=1:n$  **do**

    Use the models  $[\theta_1 \theta_2 \dots \theta_p]$  to predict the labels for the points in  $X_i$ ;

    Ask the expert for the labels  $y_j$  corresponding to the point  $x_j \in X_i$  on which there is maximum disagreement between the models;

    Add  $[x_j, y_j]$  to the training set  $T$  and train all models using the training set;

**end**

**end**

**Algorithm 1:** Query-by-committee Algorithm [6]

<sup>1</sup>. 20 of these images were chosen for the training set the others were used for the test set, and for 200 randomly chosen columns in each training image NS were calculated according to all three automatic methods, i.e. column-mean, column-median and flattest  $L1$ -normalized spectra in the column. The unlabeled dataset  $X$  consists of these  $N = (20 \times 200 \times 3)$  12000 unlabeled examples. The dataset  $X$  was broken up into

<sup>1</sup>available at <http://geo.pds.nasa.gov/missions/mro/crisp.htm#Tools>

Table 1: Classification Performance of Best Model

	# of images	# of examples	% error
Training Set	20	240	12.500
Test Set	10	160	14.3750

$n = 120$  smaller subsets of  $k = 100$  samples each. We then built a query-by-committee model with  $p = 30$  classifiers. Each classifier is an fully connected feed-forward Artificial Neural Network (ANN) with one hidden layer (similar to the one in [4]). The input-layer contains 230 nodes and the hidden layer contains 25 nodes and an output layer with two nodes (one-hot encoding). We use the sigmoid activation function for the hidden layer and output nodes. In this classifier we minimize soft-max of the logistic error. We performed 2 passes over the dataset  $X$ . The final training set contained 240 labeled examples. The test set consists of 160 examples of NS from the test images. The performance of model that does best on the test set is reported in Table 1. In the future we will look to extend this work to images with higher noise levels and leverage labels from multiple experts (by say polling) etc..

**References:** [1] L. Mandrake, et al. (2010) in *2010 2nd WHISPERS* IEEE. [2] B. Bultel, et al. (2015) *IEEE JSTARS* 8(6):3039. [3] B. L. Ehlmann, et al. (2009) *J of Geophys Res: Planets* 114(E2). [4] Z. Lu, et al. (2008) in *10th GECCO* 2047–2054 ACM. [5] B. Settles (2009) Active learning literature survey *Computer Sciences Technical Report 1648* University of Wisconsin–Madison. [6] H. S. Seung, et al. (1992) in *COLT92* 287–294 ACM.