LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-744504

Overhead longwave infrared hyperspectral material identification using radiometric models

M. E. Zelinski

January 12, 2018

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Overhead longwave infrared hyperspectral material identification using radiometric models

Michael E. Zelinski[a]
[a]Lawrence Livermore National Laboratory, Livermore, CA 94550

Abstract. Material detection algorithms used in hyperspectral data processing are computationally efficient but can produce relatively high numbers of false positives. Material identification performed as a secondary processing step on detected pixels can help separate true and false positives. This paper presents a material identification processing chain for longwave infrared hyperspectral data of solid materials collected from airborne platforms. The algorithms utilize unwhitened radiance data and an iterative algorithm that determines the temperature, humidity, and ozone of the atmospheric profile. Pixel unmixing is done using constrained linear regression and Bayesian Information Criteria for model selection. The resulting product includes an optimal atmospheric profile and full radiance material model that includes material temperature, abundance values, and several fit statistics. A logistic regression method utilizing all model parameters to improve identification is also presented. This paper details the processing chain and provides justification for the algorithms used. Several examples are provided using modeled data at different noise levels.

Keywords: Hyperspectral, LWIR, radiometric modeling, detection, identification

1 Introduction

Longwave infrared (LWIR) hyperspectral imagers (HSI) can capture high spectral resolution measurements of the electromagnetic spectrum between 7.5-13.5um.[1,2] In this spectral region many gas[3] and solid materials[4,5] have spectral emission/absorption features that are observable by LWIR HSI. Spectral analysts match observed spectral features found in the data with those found in the spectral database of materials. This is often performed by using automated material or target detection algorithms.[6,7] Detection algorithms are designed to be computationally efficient and process spectra quickly; however, they have typical false alarms rates of approximately 10^{-5} for common threshold settings[6]. The detections above the threshold will be examined by a spectral analyst. As the number of sensors grow and as the sensors themselves improve, collecting greater numbers of pixels, the amount of false alarms will begin to overload the current working number of spectral analysts. There is an increasing need to further reduce the number of pixels inspected by analysts. To mitigate the effect of costly false alarms a "material identification" algorithm can be used[8,9]. Material identification performs a more thorough analysis on a single pixel (or region) of interest that passed the detection threshold. It is often more time consuming than detection algorithms and is not practical to run on a full scene. The resulting information can provide the spectral analyst with more information about the contents of the pixel. This information, which is often quantitative, can also be used to set additional thresholds on the data to further suppress false alarms.

This work will require estimation of atmospheric parameters (transmission, downwelling radiance, and upwelling radiance). Temperature Emissivity Separation (TES) algorithms provide a method of acquiring these terms. The In-Scene Atmospheric Compensation (ISAC) algorithm[10] utilizes in-scene blackbodies in an algorithm that provides an estimate of atmospheric transmission and upwelling. This work was developed for Aerospace Corporation's Spatially Enhanced Broadband Array Spectrograph System (SEBASS). SEBASS utilizes a liquid helium cooled focal plane array that has very well-behaved noise structure and few dead pixels. Not all

LWIR HSI sensors have these characteristics. Another important issue is that not all scenes have blackbodies present and not all scenes have spatially uniform atmospheric profiles. Because of these issues the ISAC algorithm is not appropriate for all LWIR HSI imaging scenarios. Other approaches[11] require measuring the atmosphere by sounding, and then using this data in a radiative transport code such as MODTRAN (MODerate resolution atmospheric TRANsmission)[12] to simulate the atmospheric terms found in the radiance equation. This approach has problems relating to the availability of time/location appropriate sounding data. There are also several methods to "search" for correct atmospheric and model terms. This is done by using precomputed look up tables for the atmosphere[13-15] and spectral emissivity smoothness as a metric for appropriate model parameters (solid materials have broader spectral features than atmospheric gasses). The work presented here will utilize some of these concepts in a new approach that does not depend on in-scene blackbodies, sounding, or a spatially uniform atmosphere. It should be noted that these traditional algorithms are likely more computationally efficient than the processing chain presented in this document.

This paper demonstrates a physics based processing chain for performing material identification by unmixing non-whitened LWIR HSI radiance data. Spectra are unmixed by producing radiance models that match measured scene spectra. The models are comprised of background endmembers and emissivity spectra that are forward modeled to radiance. Models and scene measurements are compared by using RMS error.

As mentioned above, researchers have used atmospheric sounding to obtain an atmospheric temperature/humidity/ozone profile. This paper demonstrates how to acquire the atmospheric profile by searching for it using an optimization algorithm. Temperatures and abundances for the material of interest are also determined. If all parameters in the model are correct then it should match the measurement. If it does not match, then it is unlikely that the pixel under inspection contains the material of interest.

The processing chain consists of two primary steps, *atmospheric inference* and *radiometric modeling/pixel unmixing*. Figure 1 provides the reader with a summary view of the algorithms.

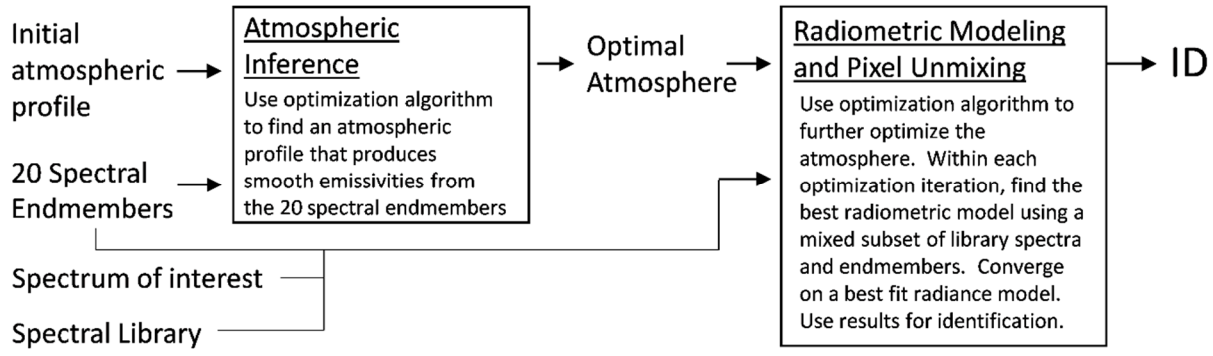


Figure 1. This flowchart provides a basic view of the data processing pipeline.

Whether this processing chain is applied to real data or simulated data there are several important assumptions that need to be stated.

- Sensors accurately and precisely measure radiance at the sensor aperture

- MODTRAN simulates atmospheric transmission, upwelling, and downwelling both accurately and precisely
- Lambertian radiance models are appropriate

This paper is organized as follows. Section 2 contains a description of MODTRAN and provides several important points on using it in this study. Section 3 describes an approach for obtaining an optimal atmospheric column parameterization. Section 4 details the Lambertian radiance model used. Section 5 describes the calculations that occur within each iteration of the *atmospheric inference* and *spectral unmixing* algorithms. Section 6 describes how logistic regression can be used with multiple model output parameters for identification and reducing false alarms. Section 7 uses the data processing chain on multiple experiments and results. Section 8 is the conclusion. Section 9 is an appendix with detailed flow charts of the algorithms. Section 10 provides a list of references. Throughout the document are several implementation notes that readers should follow if they choose to implement these algorithms.

The radiance unit used here is a micro-Flick (μF) which is a $\mu W/(cm^2\ sr\ \mu m)$.

2 MODTRAN

MODTRAN is a highly capable tool for radiative transport calculations in the Earth's atmosphere at altitudes below sea level (e.g. Death Valley) to 100km for wavelengths between (0.2 to 10,000μm) at a spectral resolution of $0.1cm^{-1}$.

This research makes use of the 'Card2C1' to define the temperature, humidity, and ozone. The algorithms define the profile at four altitudes. The lowest altitude is ground level. The second altitude is at 300 meters where the atmospheric boundary layer could exist. The third altitude is at the aircraft altitude. If the aircraft has an onboard temperature/humidity/ozone sensor that data can be used by the algorithm – this will be discussed in the results. The fourth altitude is at 10km. Between each of the defined altitudes, seven additional atmospheric layers are computed using linear interpolation for a total of 25 layers. The algorithms described below will control the atmospheric profile values at the four layers and find an optimal profile for establishing realistic radiance models.

For the radiance models defined below the atmospheric transmission and the upwelling radiance values are determined by positioning the MODTRAN observer at the altitude of the HSI sensor. At this setting the target temperature should be set to 0K. To find the downwelling term the observer should be placed 1m above the target and the target reflectance should be set to 1. The Lambertian reflectance model should be used here.

Atmospheric band radiance and transmission values vary significantly in narrow wavelength regions - much narrower than the spectral resolution studied here.[10] Therefore, spectral calibration (band center and width) is critical for this work. Because this work makes use of simulated data the spectral calibration is known; however, in a real HSI system, wavelength calibration bandcenters should be known to within $1/10^{th}$ of a spectral bin across the entire focal plane array (accounting for spectral smile and keystone). The spectral band line shape should also be known, for this work a Gaussian was used. Bands between 8.86μm and 13.1μm are used,

outside of this region the water vapor features are large and high frequency enough that they are not well sampled by MODTRAN at 0.1cm^{-1} resolution.

The code used to interact with MODTRAN was written in MATLAB and makes use of the MODTRAN class wrapper[16] to set the values in the tape5 file. It also includes commands to run MODTRAN as well as read the output in the tape7 file.

MODTRAN calculations are performed at 0.1cm^{-1} spectral resolution between 1400cm^{-1} and 700cm^{-1}. The resulting atmospheric arrays have 7001 elements. When appropriate, radiometric calculations are done in high spectral resolution. The flowcharts in the Appendix detail when this is done. Downsampling is done by integrating under the Gaussian (sigma value of 0.024um) spectral line shape for each spectral band. This work uses a hypothetical sensor with 178 bands, bandwidth 0.024um/pixel spanning 8.86 – 13.1um. Downsampling is done efficiently by creating a bandpass array (dimensions 7001x178) that is applied to the high-resolution vectors with a dot product.

3 Atmospheric Column Parameter Optimization

The *atmospheric inference* and *spectral unmixing* methods utilize MODTRAN estimates of the atmospheric spectral transmission, downwelling, and upwelling. MODTRAN atmospheres are parameterized by defining the temperature, dewpoint, and ozone profiles of the atmospheric column. As will be described, if these atmospheric terms are known one can expect smooth emissivities and low error radiance models. Typically, atmospheric sounding is used to measure these parameters. An alternate method is to search for them by using metrics relating to emissivity smoothness and model error. In the approach used here an optimization algorithm is used to guide the search.

Nelder-Meade (a.k.a. “simplex” or “amoeba” algorithm)[17] is a common numerical optimization algorithm that does not require an analytical derivative. There are implementations in many coding packages such as Python or MATLAB. An open source constrained version of the algorithm in MATLAB allows users to set boundaries on each optimized variable[18]. This is particularly useful when the objective function has local minima and the user knows the initialization is close to an optimal solution.

Numerical optimization algorithms require initialization points. There are several acceptable ways this could be done. The author uses the median scene brightness temperature to bias the temperature and dewpoints of a set of atmospheric profiles that loosely resemble standard atmospheres. Using this set, the *atmospheric inference* step is repeated using the same atmospheres but with varying amounts of tropospheric ozone. Readers might find it useful to use the standard atmospheres. The best atmosphere in the set is selected as the initialization point for the *atmospheric inference*, which is then run for 60 iterations. The output of this is then used as an initialization point for the *spectral unmixing* method, is run several times with different initialization constraints (see Table 1). Each successive run improves upon the previous estimate of the atmospheric parameters. The first run improves the estimate of the atmospheric column temperature and humidity. The second run improves the estimate of the tropospheric ozone

concentration. The third run fine tunes the most sensitive portions of the profile. As the optimization algorithm defines new atmospheres they are saved in a database.

Table 1 presents the constraints used on atmospheric optimizations. 'tp_aX', 'dp_aX', and 'oz_aX' indicate the temperature, dewpoint, and ozone constraints at altitude 'X'. The values in the table indicate the boundary around the initialization value "i" at each altitude, for example, if tp_a1 is 20, then the boundary for the *atmospheric inference* method would be [5C, 35C]. The ozone layer uses the initial value +/- the initial value divided by 2. This prevents negative ozone values from being used. Bold numbers indicate hard boundaries that are not relative to the initialization value.

Algorithm	tp_a1	tp_a2	tp_a3	tp_a4	dp_a1	dp_a2	dp_a3	dp_a4	oz_a1	oz_a2	oz_a3	oz_a4	Iter.
Atm. Infer.	-15, 15	-16, 16	-16, 16	-16, 16	-16, 16	-16, 16	-16, 16	-16, 16	-i/2, +i/2	-i/2, +i/2	-i/2, +i/2	-i/2, +i/2	60
Unmix	-5, 5	-5, 5	-5, 5	0, 0	-5, 5	-5, 5	0, 0	0, 0	0, 0	0, 0	0, 0	0, 0	15
Unmix	0, 0	0, 0	0, 0	0, 0	0, 0	0, 0	0, 0	0, 0	**0, .4**	**0, .4**	0, 0	0, 0	25
Unmix	-5, 5	-5, 5	-5, 5	0, 0	-5, 5	-5, 5	-5, 5	0, 0	0, 0	-i/2, .1	0, 0	0, 0	60

4 LWIR Lambertian Radiance Model

The LWIR Lambertian radiance model[19] is defined as:

$$\boldsymbol{L = B(T)\varepsilon\tau + (1-\varepsilon)\tau L_d + L_u} \quad (1)$$

Where L is at sensor radiance [μF], $B(T)$ is blackbody radiance defined at temperature T, ε is the material emissivity, τ is the transmission of the atmosphere from the ground to sensor, L_d is the downwelling radiance, and L_u is the upwelling radiance. The spectral nature of each component is implied and the λ has therefore been omitted from the equation.

Section 5.2 makes use of a target leaving radiance model that incorporates mixtures of target spectra as well as scene endmembers. This can be defined as:

$$\boldsymbol{L_{leaving} = \sum_i^M \frac{f_i(L_{em,i}-L_u)}{\tau} + \sum_j^N f_j\left(B(T_j)\varepsilon_j - (1-\varepsilon_j)L_d\right)} \quad (2)$$

Where M is the number of endmembers, N is the number of target spectra, $\boldsymbol{L_{em,i}}$ is the i[th] scene endmember, and $\boldsymbol{f_i}$ and $\boldsymbol{f_j}$ are the fractional abundances of each component. The abundances are constrained such that they sum to 1 and are non-negative.[20]

5 Calculations Occurring Within Each Iteration

The in Section 3, the a*tmospheric inference* and *pixel unmixing* algorithms utilize iterative optimization to find a solution that is close to optimal. At each iteration, tests are done to assess how well model parameters approximate the optimal solution. This section will provide details on the calculations done within each iteration.

A key part of this work is finding the correct material temperature in the radiometric model. Both algorithms used in the paper have separate approaches to finding this temperature for the pixels(s) under inspection. The *atmospheric inference* method uses an approach inspired by ARTEMIS (Automatic Retrieval of Temperature and Emissivity using Spectral Smoothness)[13] that determines optimal temperature by examining the smoothness of the calculated emissivity. While the *pixel unmixing* algorithm uses a library based method, where the material temperature is adjusted to find the lowest root mean squared error between the model and the measurement. The details of both methods are discussed in the following two sections.

5.1 Temperature Determination for Atmospheric Inference

A key assumption used here is that most solid materials tend to have smoothly varying emissivity relative to both the sensor's spectral resolution and the spectral features of atmospheric gasses.[18] If Equation 1 is solved for emissivity and the atmospheric parameters and material temperatures are known, then the calculated emissivity should be smooth for Lambertian materials.

The method of temperature determination used by the *atmospheric inference* algorithm is shown in Figure 2. Emissivity vectors are created at temperature interval of .1K spanning -30/+80K of the median brightness temperature. For each emissivity vector a smoothness calculation was performed. The smoothness metric used here is the product of two numbers. The first number is the $abs(median(\varepsilon_T - .95))$, which biases the metric such that high emissivity vectors are favored. The second number is calculated by down-sampling the emissivity vector by a factor of 2, taking the difference along the adjacent elements of the array (analogous to a derivative), raising that vector to the 4th power (this accentuates rough spectral features caused by an incorrect atmosphere but not emissivity variation), and then taking its mean. A minimum value indicates the least rough (or smoothest) emissivity that is close to .95. The product of these two numbers is used in the cost function for the optimization. This process is repeated for all 20 endmembers. Readers should examine Appendix A.1, which shows all steps in the *atmospheric inference* algorithm.

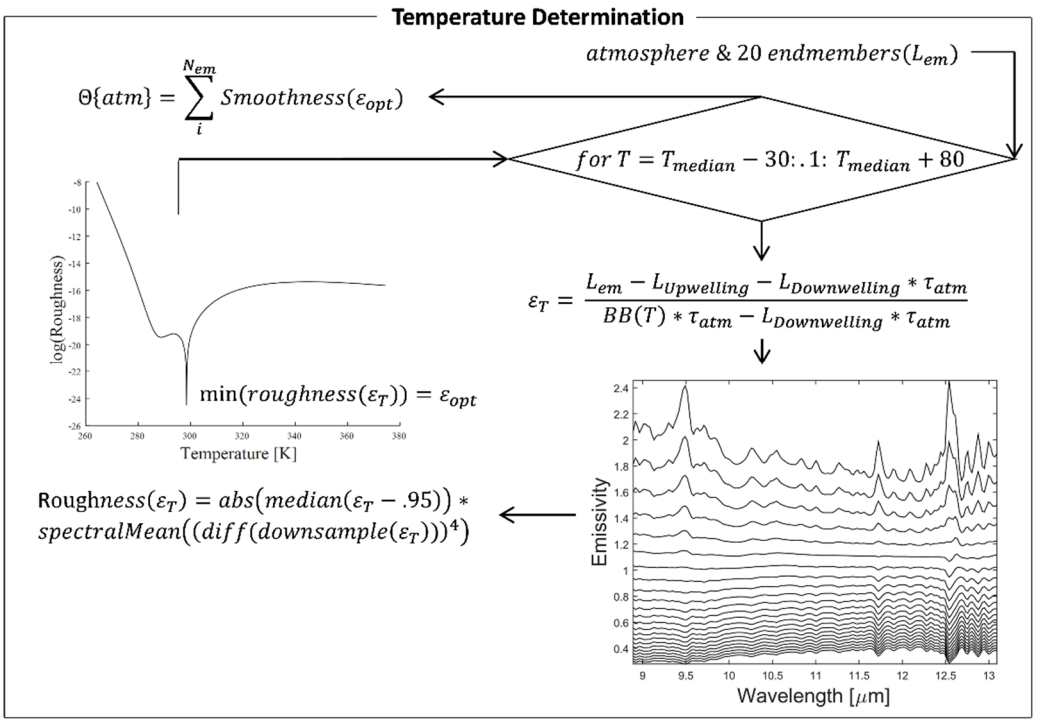


Figure 2. This flowchart shows the temperature determination method used in the *atmospheric inference* algorithm. Eq. 1 solved for emissivity is used to calculate emissivity spectra at a range of temperatures (shown in right hand plot). A smoothness metric is used to select the ideal temperature (left hand plot) at the minimum roughness value. This is repeated for each of the 20 endmembers. The summation of smoothness values is used as the cost function in the atmospheric optimization.

This procedure is computationally fast and can be applied to many pixels, temperature ranges, and atmospheres. In a real-world implementation, it is preferable to have pixels that are at different temperatures and have different emissivities. The 20 endmembers should be collected from an area around the pixel of interest. Using a rectangle or circle with size determined by padding the perimeter around the region of interest by 10 pixels is an appropriate approach. Endmembers can be selected using the maxD[19] algorithm, where the 20 most orthogonal pixel vectors are chosen.

5.2 Model-Based Temperature Determination and Spectral Unmixing

A key component of material identification is finding the signal model that matches the measurement. In this section, radiance models are created from a subset of library spectra and the 20 local background endmembers. As in Section 5.1, this method is also part of an iterative atmospheric optimization used to determine an optimal set of atmospheric parameters, however here a spectrum of interest is unmixed and several statistics useful for material identification are found.

Unmixing using radiometric models is a multistep process. To aid in understanding this process readers should refer to Appendix A.2 and then to the flow charts presented in this section. The first step is to obtain an initial temperature estimate. Using MATLAB's *lsqlin*[20] models with all material spectra and background endmembers are fit at a course range of temperatures (dT =

.5K). Figure 3 details this method. The temperature found at this step is used in the following modeling selection step.

The best-fit model from the initial temperature determination step will likely be overfit. Reducing the number of variables within the model may result in more reliable model statistics. This can be done by using Bayesian Information Criteria (BIC). The definition of BIC in model fitting scenarios where the log-likelihood is being maximized $BIC = -2 \cdot loglik + (logN) \cdot d$.[21,22] Where $loglik$ is the log-likelihood, N is number of samples, and d is the number of variables. Including the number of variables in a summation term has a regularizing effect. A modified version of this equation ($mbic$) is used here for model selection (see Figure 3B). The primary modification is that the number of variables is now included in a squared term, this results in an increased preference for simple models.

The reduced model is then used in a final temperature determination. This step is identical to the initial temperature determination except that a finer temperature increment is used (dT=.1K). An important implementation note is that the final temperature found here is then used as an initialization point for the next iteration of atmospheric optimization.

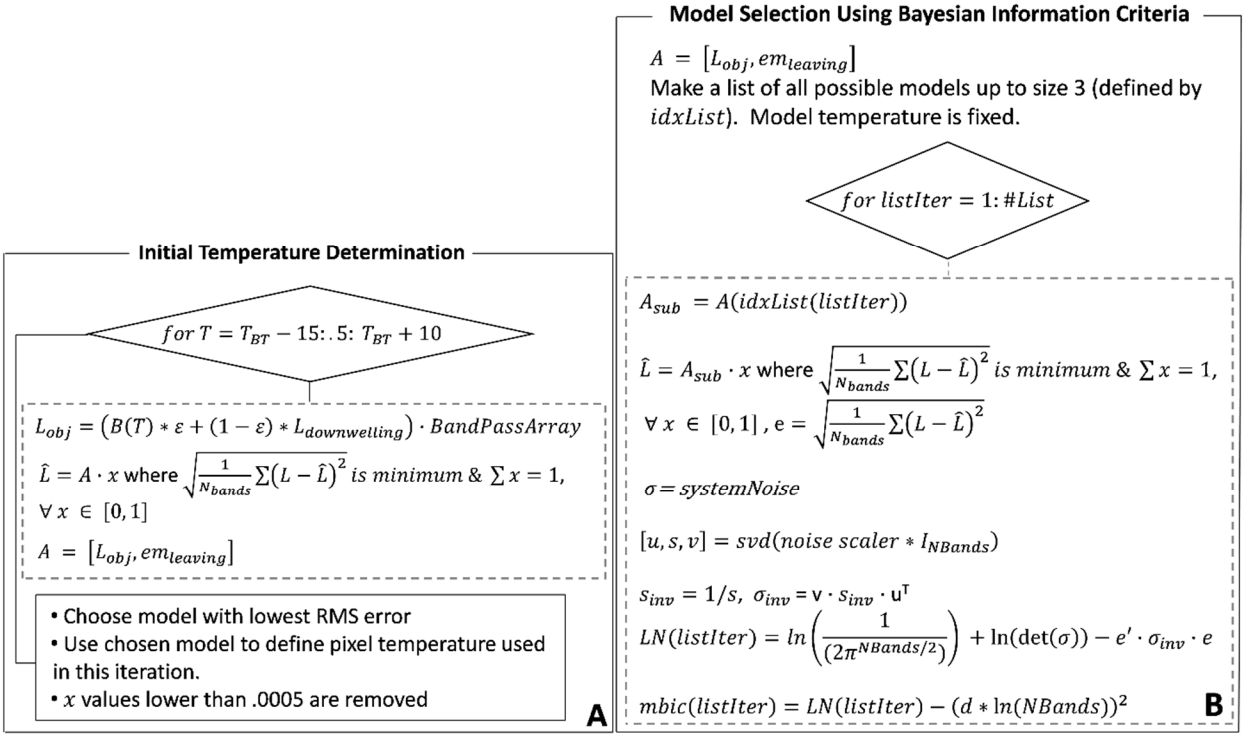


Figure 3. A) A method for determining model constituents and material temperature is provided here. A non-negativity and sum to 1 constraint is placed on the abundances of the model. Referring to Eq. 3, $em_{leaving}$ has dimension [M x numbands] and L_{obj} has dimension [N x numbands]. B) With material temperature held static a search of models is performed. A modified version of BIC that more strongly preferences simpler models is used. The maximum mbic value is chosen as the best model.

6 Logistic Regression for Spectral Identification

Traditional detection algorithms utilize a user defined threshold to establish which pixels will be presented to the analyst. Identification offers a different approach where information can come from multiple sources. A spectral analyst might find it useful to inspect the detection score (such as the Adaptive Cosine Estimator[19]), RMS Error, overall F-statistic, partial F-statistic[23], target material temperature, number of target materials, and target abundances. If the analyst views identification results from many detections, some patterns might appear in the identification results that would allow for additional thresholding. Another option is to acquire an equal number of true and false positives and use the fit statistics with a logistic regression algorithm[21,22] to establish a decision surface that optimally separates the true and false positives. Once the surface is established future identification results can be tested against this decision surface to determine whether the detection is a true or false positive. A 2-dimensional illustration of this is provided in Figure 4. The parameters shown here could be *RMS Error* vs. *determined target abundance.* Reflecting that pixels fit with models that have low RMS Error and high abundance are more likely to be true positives. Adding additional information such as partial F-statistics, model size, temperature, etc. can increase the ability of this method to separate true vs. false positive detections.

The author has found this to be a powerful approach to eliminate large numbers of false alarms in real datasets. If readers choose to employ this method caution should be taken when using data from multiple sensors. A decision surface established using data from a one sensor may not be useful for analyzing data from other sensors with different noise character. Results using the simulated data will be provided in Section 7.6.

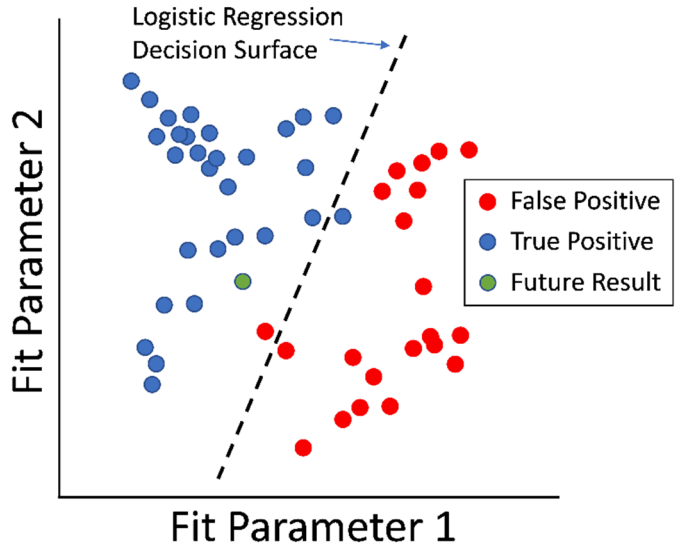


Figure 4. Illustration of logistic regression used to optimally adjust identification thresholds. The true and false positives are used to train the logistic regression decision surface. The future identification result (green) will use the decision surface to classify whether it is a true or false positive detection.

7 Experiments and Results

Using synthetic data makes it possible to create and test algorithms under many different conditions. Presenting the processing chain with challenging conditions, such as increased noise or decreasing target abundance, allows us to understand the limitations of the processing chain. This section will present a variety of tests allowing us to understand how well the processing chain determines abundance, temperature, and the overall error when compared to the measurement. Simulated false positive detections will also be examined with the algorithm and their results be compared to those of the simulated true positives.

NASA's ASTER spectral library has reflectance measurements of many common materials. The experiments in this paper utilize *jhu.becknic.rock.sedimentary.limestone.coarse.limest1.spectrum* as a surrogate target and *jhu.becknic.manmade.roofing.rubber.solid.0833uuu.spectrum* as the background material. All "scene measurements" are modeled using mixtures of these two materials at different abundances and temperatures. The input target spectral library used in unmixing includes other ASTER limestones with 11.4μm features:

'jhu.becknic.rock.sedimentary.limestone.coarse.limest1.spectrum'
'jhu.becknic.rock.sedimentary.limestone.coarse.limest2.spectrum'
'jhu.becknic.rock.sedimentary.limestone.coarse.limest3.spectrum'
'jhu.becknic.rock.sedimentary.limestone.coarse.limest4.spectrum'
'jhu.becknic.rock.sedimentary.limestone.coarse.limest5.spectrum'
'jpl.nicolet.rock.sedimentary.limestone.solid.fge3.spectrum'

The background endmember pixels were created by selecting other common roofing materials also in the ASTER library:

'jhu.becknic.manmade.roofing.rubber.solid.0833uuu.spectrum'
'jhu.becknic.manmade.roofing.rubber.solid.0834uuu.spectrum'
'jhu.becknic.manmade.roofing.shingle.solid.0490uuu.spectrum'
'jhu.becknic.manmade.roofing.shingle.solid.0597uuu.spectrum'
'jhu.becknic.manmade.roofing.shingle.solid.0672uuu.spectrum'
'jhu.becknic.manmade.roofing.shingle.solid.0680uuu.spectrum'
'jhu.becknic.manmade.roofing.shingle.solid.0683uuu.spectrum'

The 20 background spectra were modeled using a random material temperature selected from a Gaussian distribution with mean 30C and standard deviation 2.5C.

In the following sections scene radiance spectra are modeled with the middle-latitude summer standard atmosphere, sensor altitude at 5.5km above sea level, and 4.5km above ground level.

Ideally an analyst would like to see a model with low RMS Error, realistic temperature, target material abundances above ~.3 if it is the only target predictor variable in the model, and high partial F-statistic.

7.1 Impact of System Noise on Model Parameters

This section examines the effect of additive Gaussian noise and target material abundances on several fit parameters. The radiance in the scene measurement was modeled using *jhu.becknic.rock.sedimentary.limestone.coarse.limest1.spectrum* and *jhu.becknic.manmade.roofing.rubber.solid.0833uuu.spectrum* at 25 and 30C respectively. Five sets of spectra (10 scene pixels and 20 endmembers) were created where the pixel of interest was modeled with abundance values .1 to 1 in increments of .1. At five additive Gaussian noise at levels (0, .5, 1, 1.5, and 2μF) this amounts to 150 spectra total. The limestone spectrum has a ~7% spectral emissivity feature at 11.25μm. This feature at 25C and fractional abundance of .1 should be about 4μF in depth prior to atmospheric attenuation.

The full pipeline was applied to all datasets. Figure 5 shows a summary of the algorithm's performance for these tests. The first column shows the RMS Error in μF between the modeled and measured radiances. The second column shows the determined target fractional abundance from the radiance model. The third column shows the determined temperature. The dashed lines in the second and third columns are the true value values, deviations from these values are errors. Each of the 5 rows pertains to a different noise level (0, .5, … 2μF). The horizontal axis of each plot is the fractional abundance of the limestone spectra in each modeled target vector. The plots show disagreement in the RMS Error between modelled and measured spectra in the 0μF noise data. This is expected as the optimized atmospheric profile will not match the original profile and that will be reflected in the fit as error. Other contributing effects are found in the values for the "Determined Fractional Abundance" and the "Determined Temperature". At higher target abundances the error consistently increases. The reason for this is that at lower target abundances (higher background abundances) the background endmembers can explain more of the variance, as they have a variety of temperatures and spectral shapes. The algorithm shows good matching between the modeled and predicted abundance (less than 10% error). This behavior might change if there is a large discrepancy between predicted and real temperatures, for example if a predicted temperature is low a higher abundance might produce a best-fit model as it would compensate for the low temperature. Temperature estimations have precision of (+/- 1C) for abundances greater than 50% at all noise levels.

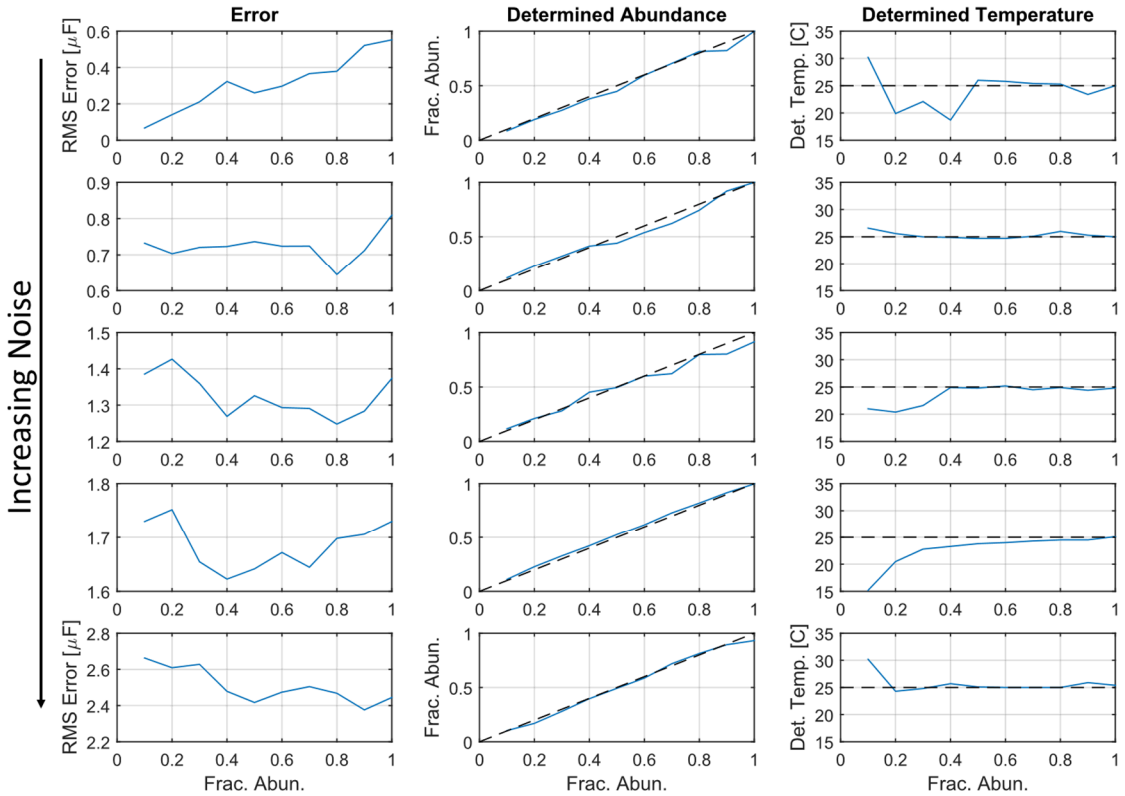


Figure 5. Demonstration of algorithm performance at several noise levels (increasing from 0, .5, 2μF). Performance is consistent even at higher levels of noise.

7.2 Importance of Atmospheric Interrogation

The *spectral unmixing* can be run without estimating the atmosphere with the *atmospheric interrogation* algorithm. The results displayed in the Figure 6 utilized the same zero-noise dataset from the first row of Figure 5. Here we see a dramatic increase in RMS Error and a reduction in the accuracy of the determined temperature and abundance. The *spectral interrogation* algorithm is therefore helpful in the retrieval of abundance and temperature.

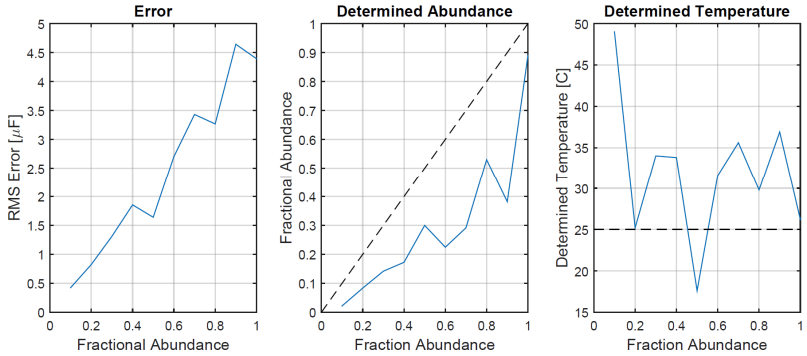


Figure 6. Retrieval of model abundance and temperature is significantly improved by using the *atmospheric interrogation* algorithm.

7.3 Demonstration of Initial Atmospheric Search

The *pixel unmixing* calculations specified in the second and third row of Table 1 can be omitted from the processing chain. Using the zero-noise dataset the reduced pipeline is run the results are shown in Figure 7. The RMS error values for the model are not substantially high. The determined abundance values are also quite reasonable. The determined model temperature shows much more error than when the full processing chain is used. The calculations defined in rows 2 and 3 of Table 1 act as priming steps for the last run of the algorithm (row 4) to help with determining correct material temperatures.

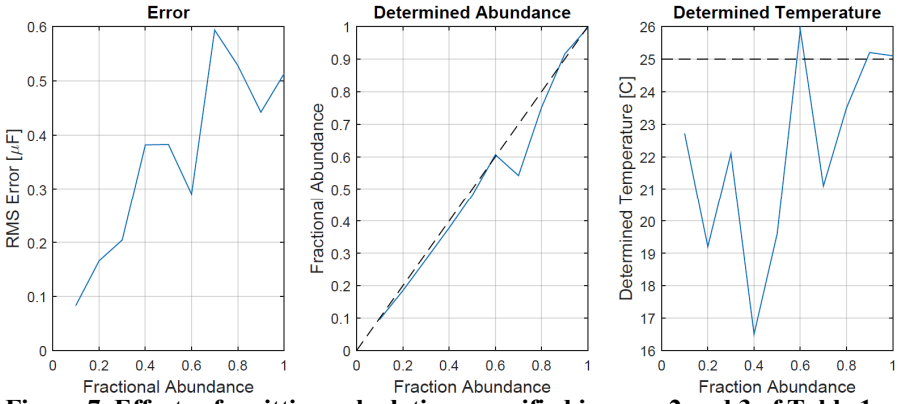


Figure 7. Effects of omitting calculations specified in rows 2 and 3 of Table 1.

7.4 Examining Effects of Object Temperature Relative to Atmospheric Temperature

One could imagine scenarios where the object temperature is either above or below the ground atmosphere temperature. An experiment was conducted where the limestone target temperature and rubber roof background temperature were increased from 8 and 13C respectively to 35 and 40C, in increments of 3C. The purpose of the experiment was to monitor the behavior of the algorithm as object temperature changed relative to a static atmospheric profile. This experiment used zero-noise data with target fractional abundance of .6. The plots in Figure 8 show that an object having low temperature (8C) relative to the atmospheric temperature could have high RMS error. In the low temperature case the determined abundance is significantly less than it should be, and the determined temperature has high error. The behavior at higher temperatures resembles results observed in previous experiments.

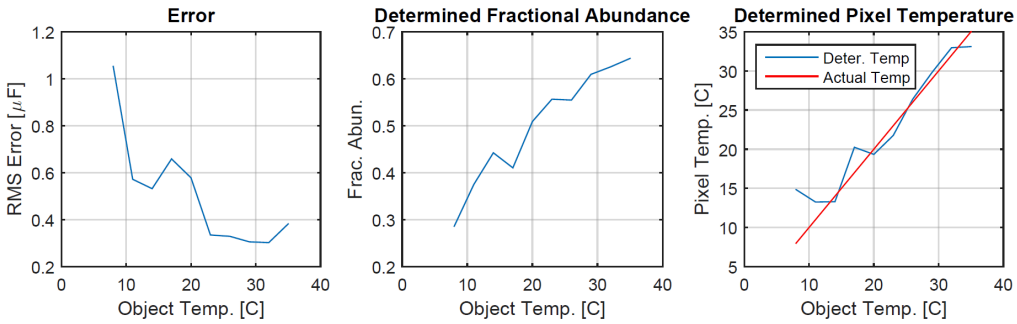


Figure 8. The lowest temperatures here show higher error rates and larger deviations in determined fractional abundance compared to results shown in Figure 5.

7.5 Atmospheric Profile Retrieval

As described in Section 3, the algorithms used here attempt to find optimal temperature and humidity profiles. Using the zero-noise data with target fractional abundance of .6 the temperature and humidity are estimated. The results in Figure 9A show little agreement between the determined temperature/dewpoint profile with that of the middle-summer latitude. The test is repeated using a narrow boundary in the Nelder-Meade optimization at the aircraft altitude – simulating a temperature/humidity sensor onboard the aircraft. Results from this test in Figure 9B show more agreement to the middle-summer latitude atmosphere, however there is still some error. A future test using just 3 altitudes in the atmospheric optimization might show improved results. The author uses 4 altitudes to guard against atmospheres with strong boundary layer conditions.

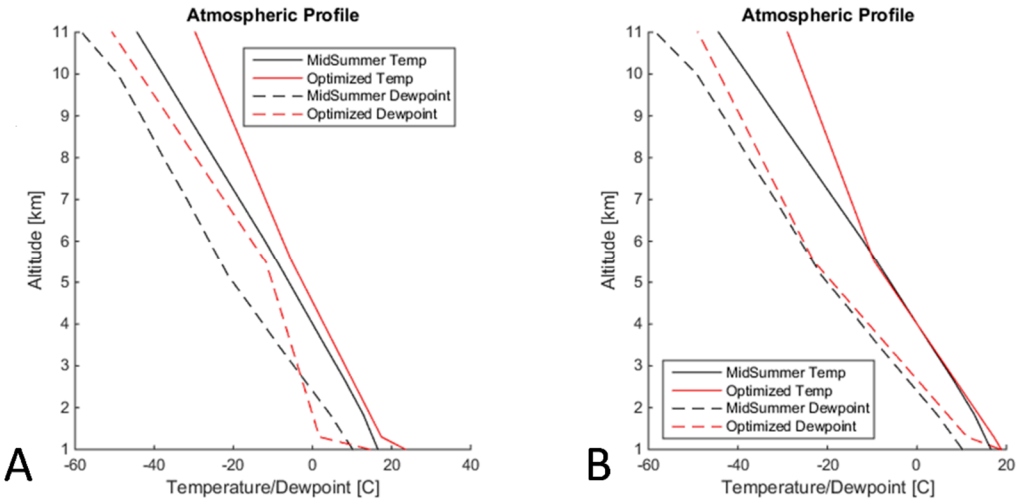


Figure 9. A) As can be seen there is a significant amount of error in this retrieval. The dewpoint estimate has errors of over 10ºC. Examining the dewpoint closely there is an undershoot at low altitudes and overshoot at higher altitudes, therefore the overall effect here is likely similar to approximations of the true atmosphere. This is an example of the algorithm getting trapped in a local minima. B) There is a slight improvement made by placing a temperature/humidity sensor onboard the aircraft.

7.6 Examination of Outputs for True and False Positive Detections

Material detection steps often produce false alarm rates greater than 10^{-5}.[19, 24] The reason for false alarming can be related to the statistical whitening process used in many detection algorithms. Reprocessing the data in radiance space using algorithms such as the one described in this paper will allow users to produce statistics and parameters that can be used to further suppress false alarms. This section illustrates how multiple parameters can be used together to improve detection statistics.

4 sets of spectra were created for this experiment:

1) 20 spectra, w/abundances of 70% limestone materials (mean temp 25, std 1), 30% rubber roofing materials (temp 30C).

2) 20 spectra, w/abundances of 40% limestones materials (mean temp 25, std 1), 60% rubber roofing materials (temp 30C).
3) 20 spectra, w/abundances of 70% non-limestone materials (mean temp 25, std 1), 30% rubber roofing materials (temp 30C).
4) 20 spectra, w/abundances of 40% non-limestone materials (mean temp 25, std 1), 60% rubber roofing materials (temp 30C).

The non-limestone materials were comprised of 20 random materials that were selected from the ASTER library. The only non-limestone material that shared common spectral features to the limestone was *jhu.becknic.manmade.concrete.paving.solid.0425uuu.spectrum*. Using these 80 spectra, separate low and high noise datasets were created using .5 and 1.5μF Gaussian noise producing a total of 160 spectra.

The full processing chain was used to gather fit results. For many of the non-limestone materials the RMS Error was very high. The most interesting results occur when the algorithm processes a false alarm that produces a low RMS Error. In this case a threshold on the RMS Error alone will not be enough to suppress all the false positives.

Figure 10A contains the results for the low noise dataset. Here we see a near perfect separation between target and non-limestone pixels when using the RMS Error. Including the determined target abundance makes possible a linear separation. The logistic regression line is drawn as a dashed line here. The two circled non-limestone results belong to pixels containing the concrete. The concrete has similar spectral features to the limestone but is not a perfect match and is therefore still separable. At the higher 1.5μF noise level, as shown in Figure 10B, the concrete pixels are not separable when using just the RMS Error and determined target abundance alone.

Plotting fit parameters in two dimensions shows that true and false positives can be separated using linear logistic regression. Using additional fit data to extend the logistic regression to higher dimensions will likely help with separability. An in-depth discussion of logistic regression has been left out of this paper as there are many resources available[21]. General machine learning principles (class balancing, training/test/validation datasets, etc.) should be followed.

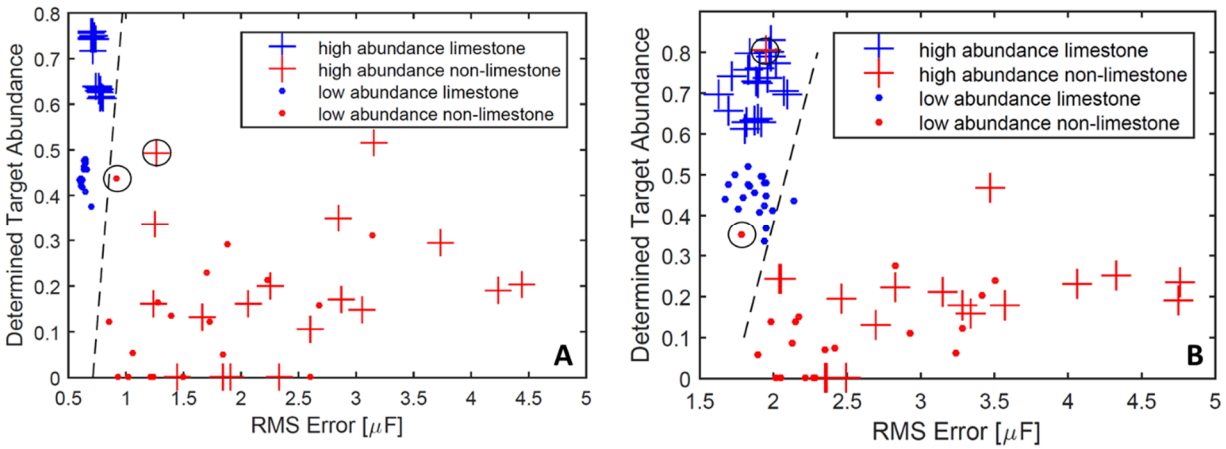


Figure 10. A) The results from the low noise (.5µF) data has a clear separation between all target and non-target materials. B) The high noise (1.5µF) data does not have a clear separation between the target and non-target materials. The circled red data points belong to spectra modeled with concrete. The dashed line is a logistic regression line that creates a decision surface between the two classes.

8 Conclusion

This paper describes a longwave infrared material identification method using radiometric models. The foundation of this work is that accurate radiometric models can be created for pixels of interest using mixtures of local endmembers and library spectra. If a scene pixel contains the materials found in the model, then the fit between the measurement and model will have a RMS Error approaching the instrument noise and the fit parameters will be close to their true values. If the pixel does not contain the materials found in the model, then the fit between the observation and model will have a high RMS Error and the model parameters might be unrealistic. The parameters and statistics derived from the model fitting can be used with a decision surface created by a logistic regression algorithm to reduce false alarms.

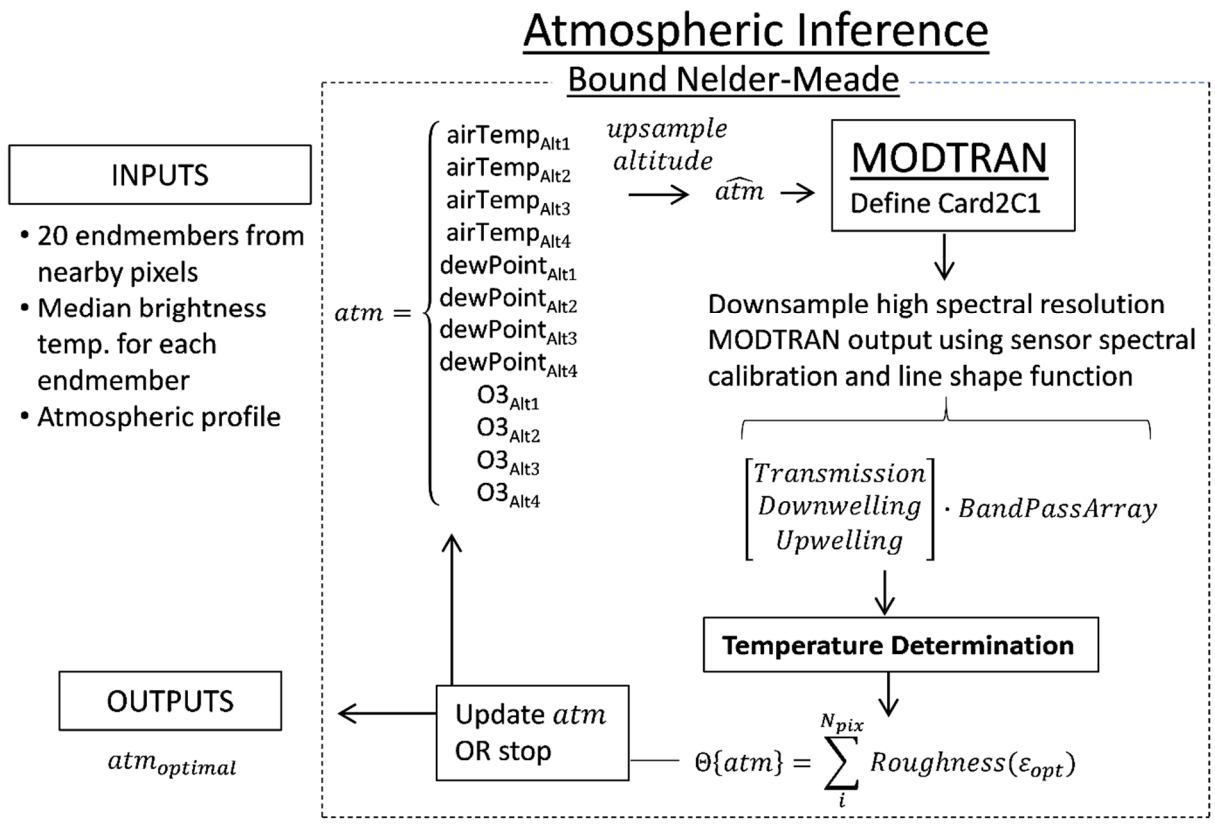
The experiments provided in Section 7 demonstrate the behavior of the processing chain under a variety of conditions. Section 7.1 shows a consistent rise in model error with system noise. Section 7.2 and 7.3 demonstrate the importance of the *atmospheric inference* and the initialization runs of the *unmixing* algorithm. In Section 7.4 scenarios where the ground is either hotter or cooler than the surrounding air temperature were examined. The algorithm appears to work best when ground temperatures are greater than or equal to the air temperature. When ground temperatures are less than 10C that of the air temperature the performance drops off significantly. Section 7.5 shows that this pipeline cannot accurately retrieve the atmospheric temperature/humidity profile. And Section 7.6 shows how multiple fit parameters from the pipeline can be used to separate true and false positives from the detection step. In one case the false positive material was a concrete which had strong calcite spectral features. At .5µF noise level the algorithm was capable of discerning between the calcite and concrete.

This work was performed on a Dell Precision with an Intel i7 processor. No GPUs were utilized. The algorithm requires about 20 minutes to process a single spectral vector. During the atmospheric optimization process atmospheres are saved in a database for subsequent use. Because of the time requirements this algorithm is not practical processing data onboard an

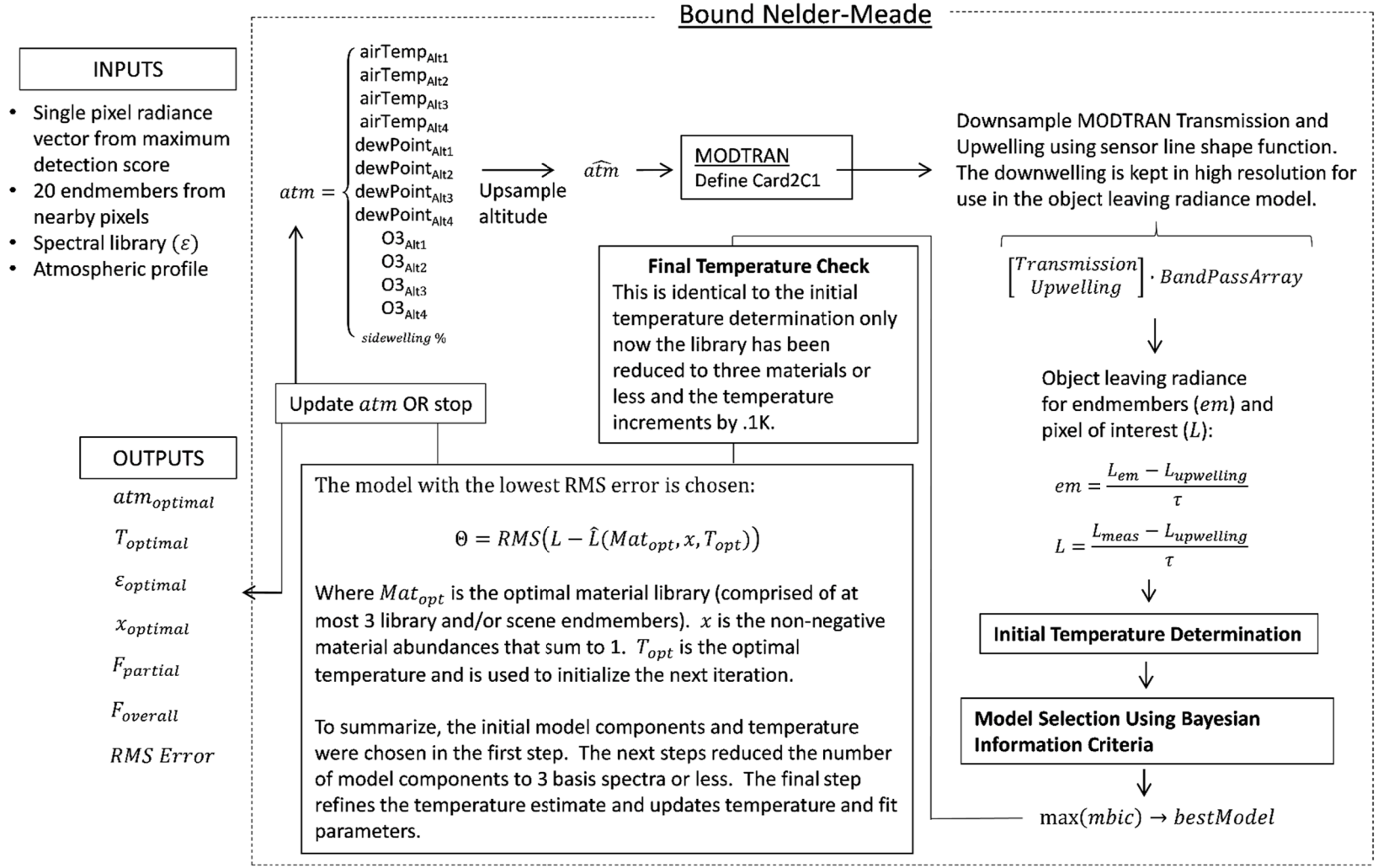
aircraft. However, because the required amount of data is only 21 pixel vectors the data could be transmitted back to a computer cluster for parallel processing. There is also likely additional work that could be done to speed up the search for an optional atmosphere.

Appendix

A.1



Radiometric Modeling and Pixel Unmixing



References

1. J.A. Hackwell, D.W. Warren, R.P. Bongiovi, S.J. Hansel, T.L. Hayhurst, D.J. Mabry, M.G. Sivjee, J.W. Skinner, LWIR/MWIR imaging hyperspectral sensor for airborne and ground-based remote sensing, *Proc. SPIE* **2819**, Imaging Spectrometry II, (13 Nov. 1996); doi: 10.1117/12.258057

2. W. R. Johnson, S. J. Hook, P. Mouroulis, D. W. Wilson, S. D. Gunapala, V. Realmuto, A. Lamborn, C. Paine, J. M. Mumolo, B.T. Eng, HyTES: Thermal imaging spectrometer development, Aerospace Conference, 2011 IEEE, (5-12 March 2011). DOI: 10.1109/AERO.2011.5747394

3. N. B. Gallagher; B. M. Wise; D. M. Sheen, Estimation of trace vapor concentration-pathlength in plumes for remote sensing application from hyperspectral images, *Analytica Chimica Acta* 490 (2003). 139-152. Doi: 10.1016/S0003-2670(03)00177-6

4. C. C. Borel, Iterative retrieval of surface emissivity and temperature for a hyperspectral sensor, Proceedings for the First JPL Workshop on Remote Sensing of Land Surface Emissivity, (May 6-8, 1997).

5. C. C. Borel, Surface emissivity and temperature retrieval for a hyperspectral sensor. Geoscience and Remote Sensing Symposium Proceedings, 1998. IGARSS '98 IEEE International, (6-10 July 1998). Seattle, WA, USA, DOI: 10.1109/IGARSS.1998.702966

6. D. Manolakis, D. Marden, G. Shaw, Target detection algorithms for hyperspectral imaging applications, *Lincoln Lab. J.* **14**(1), 79-116(2003).

7. D. Manolakis, Taxonomy of detection algorithms for hyperspectral imaging applications, *Optical Engineering* **44**(6), 066403 (June 2005).

8. B. Basener, A. Schlamm, D. Messinger, E. Ientilucci, A detection-identification process with geometric target detection and subpixel spectral visualization, 2011 3rd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), (6-9 June 2011). DOI: 10.1109/WHISPERS.2011.6080948

9. P. V. Villeneuve, A. R. Boisvert, and A. D. Stocker, Hyperspectral sub-pixel target identification using least angle regression, *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery*. XVI 7695, 76951V (2010).

10. S. J. Young, R. B. Johnson, J. A. Hackwell (2002). An in-scene method for atmospheric compensation of thermal hyperspectral data. *Geophysical Research, Journal of*, Vol. 107, No. D24, 4774, doi:10.1029/2001JD001 266.

11. M. Cubero-Castan, J. Chanussot, X. Briottet, M. Shimoni, A physics-based unmixing method to estimate subpixel temperatures on mixed pixels, *IEEE Transactions on Geoscience and Remote Sensing* Vol. 53, No. 4 (April 2015).

12. Berk, A., & et.al. (2005). *MODTRAN(TM)-5 Version 3 Revision 4 USER'S MANUAL.* Spectral Sciences, Inc.

13. Borel, C. (2003). ARTEMISS – an Algorithm to Retrieve Temperature and Emissivity from Hyper-Spectral Thermal Imaging Data. *GOMACTech, 28th Annual Conference; Hyperspectral Imaging Session.* Tampa, FL: LA-UR-027907.

14. M. Boonmee, *Land Surface Temperature and Emissivity Retrieval from Thermal Infrared Hyperspectral Imagery.* Ph.D. Thesis, Chester F. Carlson Center for Imaging Science, Rochester Institute of Technology (2007).

15. C. C. Borel, R. F. Tuttle, Recent advances in temperature-emissivity separation algorithms, *Aerospace Conference, 2011 IEEE*, Big Sky, MT, 5-12 March 2011.

16. D. J. Griffith, *matlab-modtran-5*. https://www.mathworks.com/matlabcentral/fileexchange/31961-derekjgriffith-matlab-modtran-5 30 December 2011.

17. W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed., Cambridge University Press (2007).

18. J. D'Errico, *fminsearchbnd, fminsearchcon.* https://www.mathworks.com/matlabcentral/fileexchange/8277-fminsearchbnd--fminsearchcon Release: 4, 23 July 2006.

19. J. R. Schott, *Remote Sensing: The Image Chain Approach*, 2nd Edition, Oxford University Press, Oxford, New York (2007).

20. https://www.mathworks.com/help/optim/ug/lsqlin.html

21. T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd Edition, Springer (2009).

22. C. Bishop, *Pattern Recognition and Machine Learning*, Springer (2006).

23. M. H. Kutner, C. J. Nachtsheim, J. Neter, W. Li, *Applied Linear Statistical Models*, 5th Edition, McGraw-Hill Irwin (2005).

24. D. Manolakis, D. Marden, G. A. Shaw, Hyperspectral image processing for automatic target detection applications, *Lincoln Laboratory Journal*, Vol. 14, 1 (2003).