Hyperspectral Imaging System Modeling

John P. Kerekes and Jerrold E. Baum

To support hyperspectral sensor system design and parameter trade-off investigations, Lincoln Laboratory has developed an analytical end-to-end model that forecasts remote sensing system performance. The model uses statistical descriptions of scene class reflectances and transforms them to account for the effects of the atmosphere, the sensor, and any processing operations. System-performance metrics can then be calculated on the basis of these transformed statistics. The model divides a remote sensing system into three main components: the scene, the sensor, and the processing algorithms. Scene effects modeled include the solar illumination, atmospheric transmittance, shade effects, adjacency effects, and overcast clouds. Modeled sensor effects include radiometric noise sources, such as shot noise, thermal noise, detector readout noise, quantization noise, and relative calibration error. The processing component includes atmospheric compensation, various linear transformations, and a number of operators used to obtain detection probabilities. Models have been developed for several imaging spectrometers, including the airborne Hyperspectral Digital Imagery Collection Experiment (HYDICE) instrument, which covers the reflective solar spectral region from 0.4 to 2.5 μm. This article presents the theory and operation of the model, and provides example parameter trade studies to show the utility of the model for system design and sensor operation applications.

Hyperspectral imaging (HSI) systems are being applied to a number of areas, including the environment, land use, agricultural monitoring, and defense. Because it uniquely captures spatial and spectral information, hyperspectral imagery is often processed by traditional automated image processing tools as well as analyst-interactive approaches derived from spectroscopy. HSI products often contain both quantitative and qualitative information, arranged in an image to show the spatial relationships present in a scene.

The interaction of spatial and spectral information, the dependence on ancillary or library information in the processing, and the wide range of possible HSI products prevent using a single or multiple instrument metric(s) to characterize system performance in a general manner. Thus design and sensitivity analyses of hyperspectral systems require a more comprehensive approach than traditional imaging systems.

This goal of capturing the effects of the entire sensing and processing chain motivates our HSI system modeling. We are interested in developing tools to understand the sensitivities and relative importance of various HSI system parameters in achieving a level of performance in a given application. Examples include understanding how sensitive the detection of a subpixel object is to atmospheric haze or instrument noise. Another aspect is understanding which system parameters are most important and in what situations. Having a comprehensive modeling capability is key to exploring these issues.

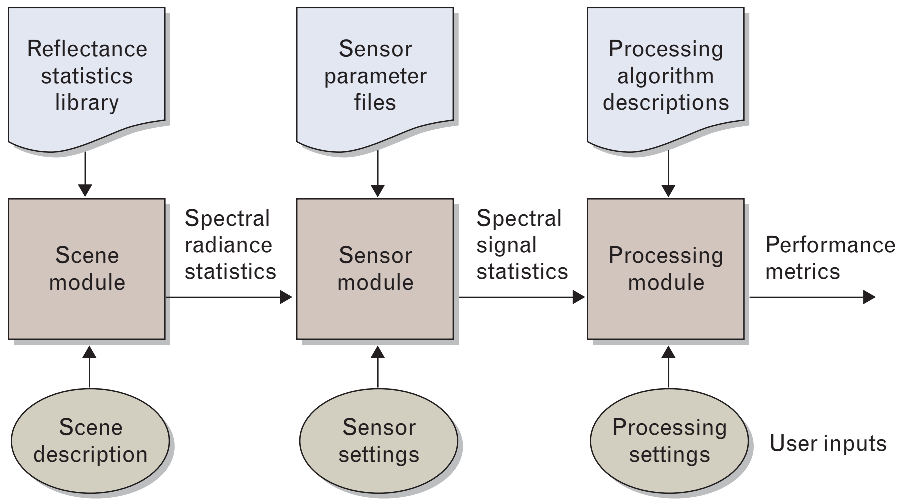


FIGURE 1. Block diagram of the end-to-end remote sensing system model. Surface classes, such as trees or roads, can be represented by first- and second-order spectral statistics. The effects of various processes in the end-to-end spectral imaging system can be modeled as transformations and functions of those statistics.

To support quick assessments of these kinds of sensitivity analyses, we have pursued a statistical parametric modeling approach, based on earlier work [1], as opposed to a physics-based system simulation method [2–4]. The method described in this article can be run quickly and efficiently through a large number of parameter configurations to understand these sensitivities.

End-to-End Remote Sensing System Model

The end-to-end remote sensing system model includes all the elements in the scene (illumination, surface, and atmospheric effects), the sensor (spatial, spectral, and radiometric effects), and the processing algorithms (calibration, feature selection, and application algorithm) that produce a data product. Figure 1 presents an overview of the model.

The underlying premises of the model are that the various surface classes of interest, such as trees or roads, can be represented by first- and second-order spectral statistics, and that the effects of various processes in the end-to-end spectral imaging system can be modeled as transformations and functions of those statistics.

The model is driven by an input set of system parameter descriptions that define the scenario, including the scene classes, atmospheric state, sensor characteristics, and processing algorithms. Table 1 contains a list of model parameters and options available, as well as their symbols used in this article.

These parameters are used in analytical functions to transform the spectral reflectance first- and second-order statistics of each surface class through the spectral imaging process. The spectral mean and spectral covariance matrix of each class are propagated from reflectance to spectral radiance to sensor signals, and finally to features, which are operated on to yield a metric of system performance. The following sections describe the scene, sensor, and processing modules of the model.

Scene Module

The end-to-end remote sensing system model considers a scene to consist of one or more background classes and an object class. The user supplies the proportion of the scene filled by each background class and the fraction of a pixel occupied by the object class. Each class is described by its first- and second-order spectral reflectance statistics (mean vector and covariance matrix). With user-supplied descriptions of the atmosphere and the observation geometry, an atmospheric code transforms weighted combinations of these reflectance vectors and matrices into surface-reflected and path-scattered radiances. These radi-

Table 1. Input System Parameters in Model

Scene	
Total number of background classes	M (≥ 1)
Area fraction of scene occupied by class m	$0 \le f_m \le 1$, $\Sigma f_m = 1$
Pixel fraction occupied by subpixel object	$0 \le f_T \le 1$
Object fraction in shadow	$0 \le f_S \le 1$
Fraction of sky visible from object in shadow	$0 \le f_{sky} \le 1$
Spectral covariance scaling factor for class m	g_m
Solar zenith angle	$0 \le \theta_s < 90°$
Atmospheric model	Tropical, midlatitude summer, midlatitude winter, subarctic summer, subarctic winter, 1976 U.S. standard
Meteorological range (visibility)	V
Aerosol model	Rural or urban
Cloud at 2.4 km altitude	Yes or no
Sensor	
Sensor type	HYDICE, Hyperion, and others
Number of spectral channels	K
Channel wavelength, bandwidth	λ, $\Delta\lambda$
Spectral quantum efficiency	η
Spectral optics transmittance	τ
Pixel integration time	t
Saturation spectral radiance	L_{max}
Number of radiometric bits	Q
Sensor platform altitude	z
Sensor view angle	$0° \le \theta_v < 90°$ (nadir = 0°)
Sensor noise factor	g_n
Relative calibration error	c_R
Data bit-error rate	B_e
Processing	
Number of features	F
Spectral regions for use as features	Wavelength regions
Feature selection algorithm	Contiguous regions, principal components, band averaging
Atmospheric compensation	Empirical line method (ELM) or none
Performance algorithm metric	Constrained energy minimization (CEM) spectral matched filter, total error, spectral characterization accuracy
Desired false-alarm rate	$10^{-6} < P_{FA} < 10^{-2}$

ances are then combined to produce the mean and covariance statistics of the spectral radiance at the input aperture of a spectral imaging sensor. The scene geometry, reflectance inputs, transformations, and at-sensor radiances are detailed below.

Scene Geometry and Subpixel Object Model

The model assumes a simple area-weighted linear mixing model for a subpixel object within a scene that contains M background classes, as shown in Figure 2. Each background class m occupies a fraction f_m of the scene, with the constraint that the fractions sum to one. The background class containing the subpixel object is denoted m^*. It is important to note that this model does not actually simulate a specific spatial layout; rather, it accounts for the effects of the multiple background classes through the area-weighting scheme.

A simple linear model is assumed for the object-class pixel. The subpixel fraction f_T, with $0 \leq f_T \leq 1$, defines the fractional area of the pixel occupied by the object with direct line of sight to the sensor. Parts of the object occluded by the background are accounted for in the background fraction.

Input Reflectance Statistics

The object and background spectral reflectance statistics are computed externally and provided as inputs to the model. They may be derived from field spectrometers, laboratory measurements, airborne spectrometer imagery converted to reflectance, or physics-based simulations. For each class, the input statistics consist of a spectral mean reflectance vector ρ and a spectral reflectance covariance matrix $\mathbf{\Gamma}_\rho$.

The model assumes that the reflectance distribution of each class, background or object, is unimodal. Thus the data used to compute the statistics must be carefully screened, through spectral clustering or histogram techniques, to ensure they form a cluster around a single mean point in the multidimensional space. In some cases, a single terrain category needs to be separated into multiple reflectance classes to ensure that each class is unimodal. For example, the single category "grass" may need to be split into classes of "dry grass" and "healthy grass," each with a different mean reflectance. Also, the model considers

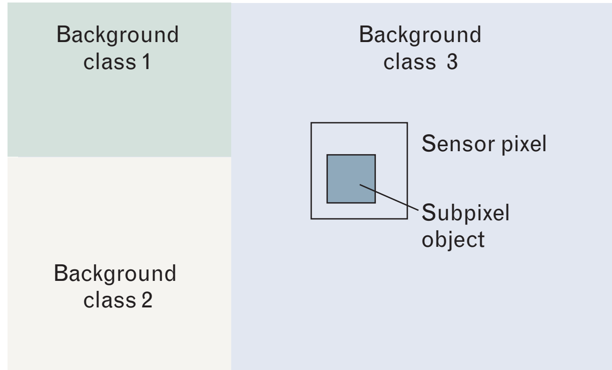


FIGURE 2. Notional scene geometry with multiple background classes and a single subpixel object. This model assumes a simple area-weighted linear mixing model for a subpixel object within a scene that contains one or more background surface classes.

the reflectance vectors to be hemispherical reflectance factors for completely diffuse surfaces. Effects related to the more complicated structure of the bidirectional reflectance distribution function are not considered.

Atmospheric Radiance and Transmittance

The model uses the Air Force Research Laboratory code MODTRAN [5] to compute the solar illumination and atmospheric effects. A number of calls are made to the code to calculate the various radiance vectors used to transform the reflectance statistics to radiance statistics. For convenience in the current version of the software, the sensor channel spectral response functions are convolved with the spectral radiances immediately after each MODTRAN run is completed. Thus the spectral radiance vectors at the output of the scene model have the same dimensionality as the sensor.

Mean Spectral Radiance†

The total mean spectral radiance for each class is the sum of $L_S(\rho)$, the total surface reflected radiance (diffuse and direct) for a mean reflectance ρ, and $L_P(\rho_{ave})$, which is the path-scattered radiance (adjacent and path) calculated with the scene average reflectance ρ_{ave} as the surface reflectance. Figure 3 shows the paths for these various radiance components. This

† All radiance calculations are performed as functions of wavelength, but for clarity in presentation, the subscript λ has been dropped.

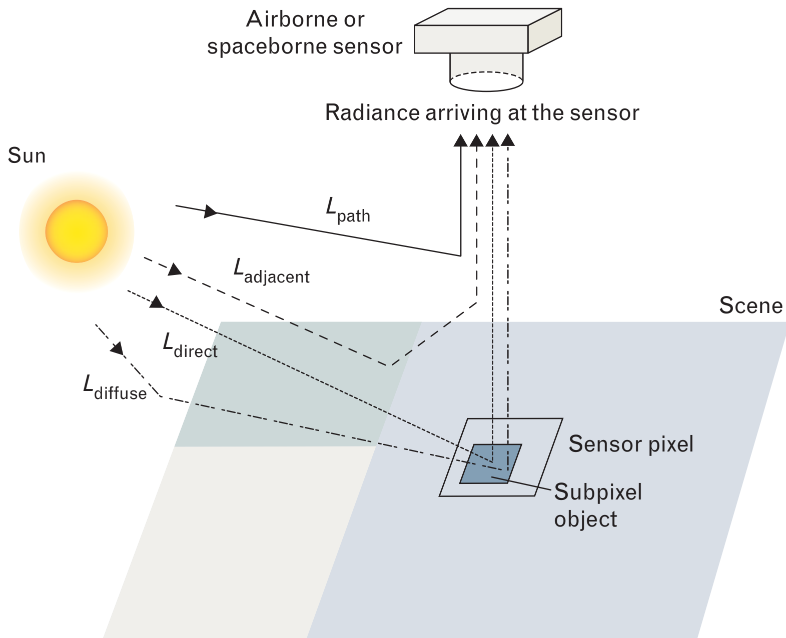


FIGURE 3. Sources of illumination and their paths from the sun to the scene and into the sensor. The total surface-reflected radiance arriving at the sensor consists of direct and diffuse components. The total path-scattered radiance arriving at the sensor consists of adjacent and path components.

formulation for the path radiance models the "adjacency effect," which is discussed below. The model has been developed for sensors operating in the reflective solar portion of the optical spectrum, with scenes near room temperature. Thermal emission effects are not considered.

Background Classes. A separate call to MODTRAN is made for each background class m, as well as one for the background scene average. The total mean spectral radiance L_B for each case is

$$L_{B_m} = L_S(\rho_m) + L_P(\rho_{ave}) \,,$$

and

$$L_{B_{ave}} = L_S(\rho_{ave}) + L_P(\rho_{ave}) \,.$$

The scene average reflectance ρ_{ave} is computed by using the class fractions f_m:

$$\rho_{ave} = \sum_{m=1}^{M} f_m \rho_m \,. \tag{1}$$

Object Class. The mean spectral radiance $\tilde{L}_T$ for the object class is

$$\tilde{L}_T = L_S(\tilde{\rho}_T) + L_P(\rho_{ave}) \,. \tag{2}$$

The surface reflectance used in the MODTRAN call to generate the first term of Equation 2 is computed as

$$\tilde{\rho}_T = f_T \rho_T + (1 - f_T)\rho_{m^*} \,.$$

The weighted sum of the object-class mean reflectance and the background-class m^* mean reflectance implements the linear model described above.

Path Radiance Calculation. For all mean spectral radiance calculations, the path-scattered contribution L_P is calculated by using the scene fractional area-weighted average reflectance ρ_{ave}, as shown in Equation 1. (Note that when MODTRAN runs with the "multiple scattering" option on, the atmospheric path-scattered radiance term depends on surface reflectance.) This approach accounts for atmospheric adjacency effects caused by the scattering of nearby surface-reflected radiance into the sensor's instantaneous field of view. The assumption here is that the scattering scale of the effect covers the entire scene being collected. Studies have shown this adjacency effect can occur out to several hundred meters [6], typical of the scenes considered by the model.

Spectral Radiance Covariance

The transformation of the spectral reflectance covariance statistics Γ_ρ to spectral radiance covariance statistics Γ_L follows the same linear atmospheric model assumed in the mean calculations. We interpolate spectral radiances calculated for surface reflectances equal to zero and one by using the entries of the reflectance covariance matrices.

These transformations use the following diagonal matrices, with the described vectors along the diagonals and zeros elsewhere: Λ_{LS1} is the total surface-reflected spectral radiance for a surface reflectance of 1, Λ_{LP1} is the atmospheric path-scattered spectral radiance for a surface reflectance of 1, and Λ_{LP0} is the atmospheric path-scattered spectral radiance for a surface reflectance of 0.

Background Classes. The background spectral radiance covariance matrices for each background class m and the scene average are computed as

$$\Gamma_{LB_m} = \Lambda_{LS1}\Gamma_{\rho B_m}\Lambda_{LS1} + \left[\Lambda_{LP1} - \Lambda_{LP0}\right]\Gamma_{\rho B_{ave}}\left[\Lambda_{LP1} - \Lambda_{LP0}\right],$$

and

$$\Gamma_{LB_{ave}} = \Lambda_{LS1}\Gamma_{\rho B_{ave}}\Lambda_{LS1} + \left[\Lambda_{LP1} - \Lambda_{LP0}\right]\Gamma_{\rho B_{ave}}\left[\Lambda_{LP1} - \Lambda_{LP0}\right].$$

Object Class. The object-class spectral radiance covariance matrix Γ_{LT} is computed by using the total surface-reflected radiance output from MODTRAN:

$$\Gamma_{LT} = f^2\Lambda_{LS1}\Gamma_{\rho T}\Lambda_{LS1} + (1 - f_T)^2\Lambda_{LS1}\Gamma_{\rho B_{m^*}}\Lambda_{LS1} + \left[\Lambda_{LP1} - \Lambda_{LP0}\right]\Gamma_{\rho B_{ave}}\left[\Lambda_{LP1} - \Lambda_{LP0}\right].$$

Sensor Module

The sensor module takes the spectral radiance mean and covariance statistics of the various ground classes and applies sensor effects to produce signal mean and covariance statistics that describe the scene as imaged by an imaging spectrometer. The sensor module includes a limited number of radiometric noise sources, with no spatial or spectral sources of error. Also, as we noted earlier, the channel spectral response of the sensor is applied during the input radiance calculations described in the previous section.

Radiometric noise processes are modeled by adding variance to the diagonal entries of the spectral covariance matrices. Off-diagonal entries are not modified because it is assumed that no channel-to-channel correlation exists in the noise processes.

The radiometric noise sources come from detector noise processes, including photon (shot) noise, thermal noise, and multiplexer/readout noise [7]. The total detector noise σ_n is then calculated as the root-sum-square of the photon noise, the thermal noise, and the multiplexer/readout noise. Because detector parameters are often specified in terms of electrons, the noise terms are root-sum-squared in that domain before being converted to noise equivalent spectral radiance.

After the total detector noise σ_n (in electrons) has been converted back to the noise equivalent spectral radiance σ_{Ln}, it is then scaled by a user-specified noise factor g_n. Next $(g_n\sigma_{L_n})^2$ is added to the diagonal entries of the spectral covariance matrices for each sensor spectral channel.

Another noise source is relative calibration error c_R. This error is also assumed to be uncorrelated between spectral channels; it is described by its standard deviation σ_{c_R} as a percentage of the mean signal level. Expressed as a variance, it is added to the diagonal entries of the covariance matrices of each class, as with the other noise sources.

The last two noise sources are quantization noise in the analog-to-digital conversion and bit errors in the communications or data recording system. These sources depend on the assumed dynamic range of the sensor $L_{\max}$. The quantization error variance σ_{nq}^2 is calculated for a system with Q radiometric bits as

$$\sigma_{nq}^2 = \frac{1}{12}\left(\frac{L_{\max}}{2^Q - 1}\right)^2 .$$

The model for the effect of bit errors in the data link (or onboard storage) assumes that bit errors are uniformly distributed across the data word and could be of either sign. Thus, for Q bits, the error will take on one of $2Q$ values, $\pm 2^i$ for $i = 0,\ldots,Q-1$, with equal probability of $1/(2Q)$. The noise variance $\sigma_{nB_e}^2$, caused by a bit error rate of B_e, is

$$\sigma_{nB_e}^2 = \frac{B_e}{Q}\sum_{q=0}^{Q-1}\left(2^q \frac{L_{\max}}{2^Q - 1}\right)^2 .$$

These last two noise terms are also added to the diagonal entries of the spectral covariance matrices.

For reporting as a performance metric, the class-dependent sensor signal-to-noise ratio is calculated as the ratio between the mean signal and the square root of the sum of the noise variance terms.

Processing Module

The signal means and covariances computed by the sensor module are then transformed by options within the processing module to produce outputs for evaluating the modeled hyperspectral system. An atmospheric compensation algorithm may be applied to the signal statistics to retrieve class reflectance statistics, reduced-dimensionality feature vectors may be extracted from the class signal vectors, and scalar performance metrics may be calculated. Each of these processing options is described below.

Atmospheric Compensation

Atmospheric compensation is accomplished by defining surrogate low- and high-reflectance calibration panels and computing the slope and offset of a two-point linear fit between the mean panel signals and the known reflectances. This approach models the empirical line method (ELM) often used for atmospheric compensation. The slopes and offsets are applied to the mean signal and covariance matrices of the object and background classes to compute the retrieved (or estimated) class reflectance mean $\hat{\rho}$ and covariance $\hat{\Gamma}$ statistics.

Feature Selection

Several options exist for extracting a reduced-dimensionality feature vector F from the signal vector: (1) all channels within contiguous regions (e.g., to avoid water-vapor absorption spectral regions), (2) principal components, and (3) band averaging to simulate multispectral channels. Each option is implemented as a linear transformation by applying an appropriate feature-selection matrix Ψ to the mean vectors and covariance matrices of the object class and each background class. This matrix Ψ can be applied in either the retrieved reflectance domain (if atmospheric compensation was performed) or the signal domain (directly on the statistics output by the sensor model):

$$F = \Psi^T X \tag{3}$$

and

$$\Gamma_F = \Psi^T \Gamma_X \Psi , \tag{4}$$

where X refers to the signal type (retrieved reflectance or sensor signal) for the object class and each background class.

Performance Metrics

Three algorithms are available to determine a performance metric for a given scenario: (1) spectral characterization accuracy, a measure of how well the spectral reflectance can be retrieved from the sensor measurements; (2) a version of a spectral matched filter, known as constrained energy minimization (CEM) [8], that can be used to predict probability of detection versus probability of false-alarm curves (P_D/P_{FA}); and (3) total error, which approximates the sum of false-alarm and missed-detection probabilities to produce a scalar performance metric.

The first performance metric, spectral characterization accuracy, is quantified by both the mean difference SC_{bias} between the retrieved surface reflectance of the object and its initial known reflectance, and by the standard deviation σ_{SC} of the difference for each spectral channel l:

$$SC_{bias}(l) = \hat{\rho}(l) - \rho(l) ,$$

and

$$\sigma_{SC}(l) = \sqrt{\hat{\sigma}_\rho^2(l) - \sigma_\rho^2(l)} .$$

The second performance metric, the matched filter, uses a known object spectral "signature" and an estimate of the background spectral covariance to minimize the energy from the background and to emphasize the desired object. In the model implementation, the known "signature" is the object's original mean spectral reflectance used at the input to the model. The filter operator w is

$$w = \frac{\hat{\Gamma}^{-1}_{\rho FB_{ave}}(\rho_{FT} - \hat{\rho}_{FB_{ave}})}{(\rho_{FT} - \hat{\rho}_{FB_{ave}})^T \hat{\Gamma}^{-1}_{\rho FB_{ave}}(\rho_{FT} - \hat{\rho}_{FB_{ave}})} ,$$

where the subscript F indicates the signal means and covariances have been transformed to the desired feature space by using Equations 3 and 4. The filter is applied to the combined object/background class features and to the features of each background class. The operator w transforms the mean and covariance from each class feature space to a scalar test statistic with mean θ and variance σ_θ^2:

$$\theta_T = w^T(\hat{\rho}_{FT} - \hat{\rho}_{B_{ave}}) ,$$

$$\theta_{Bm} = w^T(\hat{\rho}_{FBm} - \hat{\rho}_{FB_{ave}}) \text{ for } m = 1 \ldots M ,$$

$$\sigma^2_{\theta_T} = w^T \hat{\Gamma}_{\rho FT} w ,$$

and

$$\sigma^2_{\theta_{Bm}} = w^T \hat{\Gamma}_{\rho FB_m} w \text{ for } m = 1 \ldots M .$$

After this transformation, the P_D/P_{FA} curve can then be calculated. The probability of detection P_{D_m} (computed separately for each background class m) is computed for a user-specified probability of false alarm P_{FA} by assuming a Gaussian probability density function for the test statistic output. This assumption is somewhat justified by the Central Limit Theorem because the operator is a summation of a large number of random variables. The threshold h_m is determined from the variance $\sigma^2_{\theta_{B_m}}$ and mean θ_{B_m} for each background class m and the desired probability of false alarm:

$$h_m = \theta_{B_m} + \sigma_{\theta_{B_m}} \Phi^{-1}(P_{FA}) .$$

The function Φ^{-1} returns the cutoff value such that the area under the standard normal curve to the right of the cutoff is equal to the argument of the function. Then, using h_m, the probability of detecting the object in background class m is

$$P_{D_m} = \frac{1}{\sigma_{\theta_T}\sqrt{2\pi}} \int_{h_m}^{\infty} \exp\left[-\frac{(x - \theta_T)^2}{2\sigma^2_{\theta_T}}\right] dx .$$

For scenarios with multiple backgrounds, the threshold h^* yielding the minimum P_D is used to recompute the false-alarm probabilities for the other classes. These new probabilities are then summed by using the background class area fractions f_m to yield a combined P_{FA}:

$$P_{FA} = \sum_{m=1}^{M} f_m P_{FA_m}(h^*) .$$

The combined P_D is simply the minimum P_{D_m}:

$$P_D = \min P_{D_m} .$$

The third performance metric included in the model approximates the total error P_e (i.e., the overlap between multivariate distributions) in a two-class, equal *a priori* probability case:

$$P_e \approx \frac{1}{2}(1 - P_D) + \frac{1}{2} P_{FA} .$$

P_e is approximated by using the standard normal error function and the Bhattacharyya distance B_{dist} in the following function, which was found to provide a good estimate of the true error value [9]:

$$P_e = \frac{1}{\sqrt{2\pi}} \int_{\sqrt{2B_{dist}}}^{\infty} \exp\left(-\frac{x^2}{2}\right) dx ,$$

where

$$B_{dist} = \frac{1}{8}\left(F_T - F_B\right)^T \left(\frac{\Gamma_{FT} + \Gamma_{FB}}{2}\right)^{-1} (F_T - F_B) + \frac{1}{2} \ln \frac{\left|\frac{\Gamma_{FT} + \Gamma_{FB}}{2}\right|}{\sqrt{|\Gamma_{FT}||\Gamma_{FB}|}} .$$

While P_e does not distinguish between errors caused by false alarms and those caused by missed detections, it does provide a single scalar metric that can be used for relative performance comparisons. It is normally used in the model to assess the relative contribution to system error from the various system parameters, as shown in the section on example results.

Implementation

The model has been implemented as part of a software package named Forecasting and Analysis of Spectroradiometric System Performance (FASSP). The package is written in the Interactive Data Language (IDL) of Research Systems Incorporated to take advantage of IDL's integrated graphical user interface and plotting capabilities, as well as the portability between computing platforms. The model has been set up to run on UNIX, PC, and Macintosh platforms. A typical parameter trade study takes only a few minutes to run.

FASSP contains several routines for reading the information necessary for executing runs. Files that contain the material reflectance statistics and the sensor models are stored in ASCII files. The parameters that define a model run are stored outside of the FASSP environment in ASCII files called parameter files. Parameter files can be created and edited either inside FASSP or outside and then imported.

When FASSP is run, the individual scene, sensor, and processing modules (see Figure 1) are executed in series, without user interaction, using the model values in the parameter file as their inputs. It is not unusual to test several different sensor parameters or processing methods against a single scenario. In this case, the scene module results are static and only the sensor and processing modules need to be repeated. Because executing the scene module dominates the model run time, this approach allows analyses to be conducted quickly.

Run output is saved in an IDL native binary format. The output of a model run can be restored either within FASSP or it can be restored and manipulated in IDL outside the FASSP environment. Graphical displays are also available for plotting results from various stages of the model, such as material reflectances, at-sensor spectral radiance, signal-to-noise ratio, P_D/P_{FA} curves, and other processing metrics. FASSP allows the plots to be saved in JPEG, PostScript, or ASCII formats.

Validation

The model and its implementation have been validated with airborne hyperspectral imagery. A recent publication [10] provides comparisons of measured data and model predictions at points in the end-to-end spectral imaging process. Comparisons include the spectral radiance at the sensor input aperture, the sensor signal-to-noise ratio, and the detection performance (P_D/P_{FA}) after applying a spectral matched filter. At all points in the process, the model predictions compare favorably with the empirical data.

Example Results

One advantage of an end-to-end system model is that the user controls all of the scene, sensor, and processing parameters and can specify nearly arbitrary configurations. With these capabilities, our model can be applied to investigate which parameters have the most impact on a system's performance and to study the sensitivity of a given system's performance to changes in the scene and/or sensor. We present one example of a "relative impact" analysis, followed by two examples of "sensitivity" analyses.

Relative Roles of System Parameters

Our model includes an option to study automatically the relative roles of system parameters in a quantitative manner. We considered, as an example, the problem of detecting subpixel roads in a forested background. This application could arise in the use of moderate spatial-resolution hyperspectral imagery to derive a road network layer for a Geographic Information System (GIS). For our example, we used a sensor model of a generic hyperspectral imager. Reflectance statistics for a gravel road and an area with trees were derived from atmospherically compensated radiance data collected by the airborne Hyperspectral Digital Imagery Collection Experiment (HYDICE) instrument [11].

The analysis was conducted by calculating the total probability of error P_e with all parameters at their nominal values, and then recalculating P_e as each of

Table 2. Relative Importance of System Parameters in Detecting Subpixel Roads in a Forest

Scenario Parameter	*Nominal Value*	*Excursion Value*	*P_e for Excursion Value**	*Relative Role*
Number of spectral channels	30	60	0.1245	28%
Background variability scaling factor	1	0	0.1740	24%
Meteorological range	5 km	50 km	0.1894	22%
Sensor view angle (nadir = 0°)	60°	0°	0.3002	12%
Object subpixel fraction	0.5	1.00	0.3557	7%
Sensor relative calibration error	2%	0%	0.3778	4%
Solar zenith angle	60°	0°	0.3974	3%
Number of radiometric bits	8	16	0.4235	0%
Object variability scaling factor	1	0	0.4249	0%
Sensor noise scaling factor	1	0	0.4254	0%
Bit-error rate	10^{-9}	0	0.4255	0%
Total				100%

*(P_e = 0.4255 with all nominal values)

the parameters was changed to an "excursion" value, while leaving all other parameters at their nominal values. The parameter excursion values were chosen to represent an ideal value or one that would have a minimal effect on system performance. Table 2 lists the system parameters studied, their nominal values, their excursion values, the model-calculated probability of error, and a quantitative estimate of the relative importance to system performance of each parameter. The P_e metric was used as a measure of the separability of the subpixel road class and the forest background. The relative importance of each parameter was computed by taking the difference between the error calculated by using the excursion value of that parameter and the system nominal error, and then normalizing by the sum of all such differences.

The results indicate that the most important system parameters are the number of spectral channels, the forest background covariance scaling, and the meteorological range (atmospheric visibility). The importance of the number of spectral channels is not only because the dimensionality of the measurements is increased, but also because the additional channels cover a broader spectral range. In this example, the nominal 30 channels covered only 0.4 to 0.7 μm, while the excursion 60 channels covered 0.4 to 1.0 μm. After these three parameters, the sensor view angle has the next biggest influence, then the subpixel fraction, and so on. In this scenario, the impact of off-nadir viewing is predicted to be more significant than off-zenith solar angle. Although the model does not take into account the bidirectional reflectance distribution function of the surface, this result can be explained by the increase in the path radiance scattered into the sensor aperture, which, because of the adjacency effect [6], reduces separability between the road and the forest classes.

It is important to note that the conclusions from this type of analysis are extremely dependent upon the scenario configuration and the system parameters considered. The results shown in Table 2 are not intended to apply to the general case, but are shown to

Table 3. Detection Scenario for Road in Forest (Nominal)

Parameter	*Value*
Object subpixel fraction	Variable
Background	Trees
Meteorological range	10 km
Solar zenith angle	45°
Sensor view angle (nadir = 0°)	0°
Sensor relative calibration error	1%
Sensor noise scaling factor	1
Number of radiometric bits	12
Bit-error rate	10^{-9}
Number of spectral channels	121
Spectral processing algorithm	CEM spectral matched filter
False-alarm rate (per pixel)	10^{-5}

illustrate a use of the model and a methodology for quantitatively assessing the relative importance of diverse system parameters in an end-to-end spectral imaging system.

Subpixel Detection Sensitivity to Various System Parameters

Another use of the model is to study how sensitive a given sensor might be to either design changes or scene conditions. This type of analysis can be helpful when setting system requirements or when setting data collection parameters for systems that have controllable parameters.

To illustrate this use of the model, we present two examples showing parameter sensitivities. The scenario for both is the detection of subpixel roads in a forested background, as previously described, but with the use of a road spectrum (from a library or previous data collection) and a spectral matched filter. The analyses use a model of the Hyperion [12] sensor, and an atmospheric compensation algorithm with

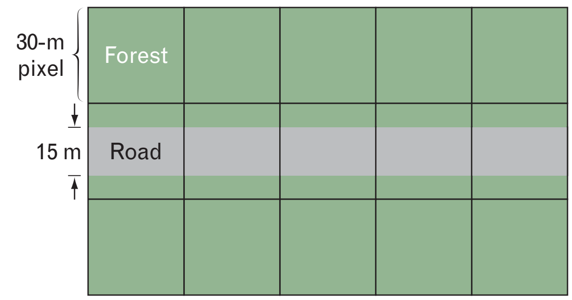


FIGURE 4. Overhead view showing 50% pixel fill factor for a 15-m road in a forest. This scene geometry ignores the detailed effects of sensor and atmospheric point spread functions.

1% (1 σ) accuracy. Table 3 presents other parameters and values assumed in the analyses.

In the examples given below, the probability of detection (at a constant false-alarm rate) is presented as a function of the pixel fill factor f_T, the fraction of a pixel occupied by the road. This pixel fill factor can be loosely translated to an actual object size, given a pixel ground resolution, and assuming the object is not split across adjacent pixels. As an example, Figure 4 illustrates how a 15-m-wide road running through the middle of a 30-m resolution pixel would roughly correspond to a 50% pixel fill factor (ignoring the detailed effects of sensor and atmospheric point spread functions). Because the model assumes a linear mixing of the object and the background class, we can present P_D as a function of the pixel fill factor and avoid specifying a particular sensor spatial resolution and object size. Thus the analysis results can be applied to a range of absolute pixel ground resolutions and object sizes. The exact performance, however, will vary because of sensor noise considerations and the variability of the spectral statistics at the various resolutions.

Sensitivity to Atmospheric Meteorological Range. Changes in atmospheric visibility, specified by the meteorological range parameter, will affect the signal level in a spectrally dependent manner, as well as affect the amount of radiance scattered from the background into the object pixel. Even though the scenario includes an atmospheric compensation step, this scattered radiance can affect performance. Figure 5 shows that the required pixel fill factor f_T for a high

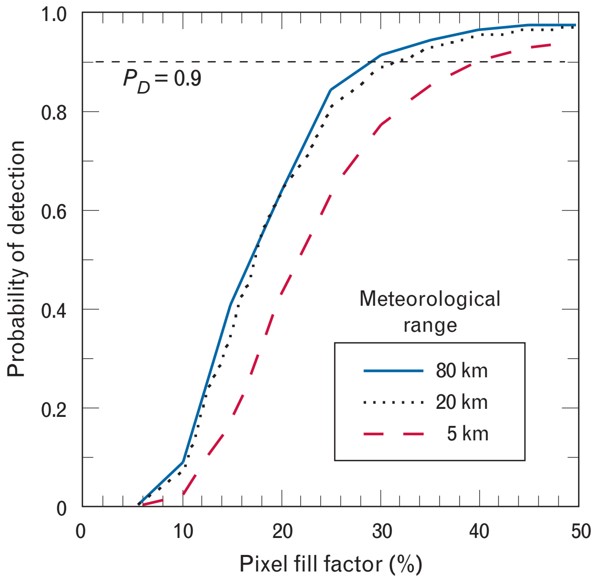


FIGURE 5. Sensitivity of road detection to atmospheric meteorological range as a function of pixel fill factor f_T. Note that the required fill factor for a high detection probability ($P_D \geq 0.9$) increases with decreasing meteorological range.

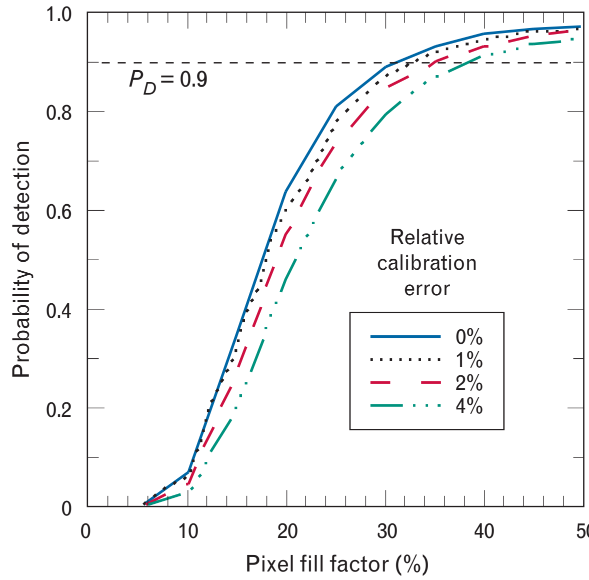


FIGURE 6. Sensitivity of road detection to additional (beyond sensor detector and electronics noise) random error (with a standard deviation expressed as a percentage of the mean signal) as a function of pixel fill factor.

detection probability ($P_D \geq 0.9$) increases with decreasing meteorological range. However, the increase in required f_T is relatively moderate (from 30% to 40%) for a significant decrease (80 km to 5 km) in the meteorological range. Thus, in this scenario, we can conclude that detection performance is only moderately affected by atmospheric haze over a reasonable range of visibility values.

Sensitivity to Random Calibration or Compensation Errors. The effects of random error in this analysis are studied by adding zero-mean "noise" with a standard deviation σ equal to a user-specified percentage of the mean radiance. In a real sensor system, this random error could come from a number of sources, such as residual nonuniformity correction error or random errors in the atmospheric compensation.[†] Figure 6 presents the sensitivity of detection probability to random errors of 0%, 1%, 2%, and 4% of the mean signal level. This range of additive error is typical for state-of-the-art sensors, nonuniformity correction routines, and atmospheric compensation algorithms. As in the previous example, for this range of values, the detection performance sensitivity is moderate, with the required ($P_D \geq 0.9$) fill fraction changing from 30% to 40%, as 4% random error is added.

[†] Another way to interpret these error levels is to recognize that with the additive error, the spectral radiance in each spectral channel has a signal-to-noise ratio hard limited to the inverse of the additive error. For example, with 2% random error added, the signal-to-noise ratio cannot be higher than 50.

Summary and Conclusions

We have presented an approach to predict detection performance, analyze sensitivities, and determine relative contributions of system parameters for multispectral or hyperspectral sensors used in the detection of subpixel objects. The end-to-end system model builds on a previously developed approach using first- and second-order spectral statistics and transformations of those statistics to predict performance. Enhancements include a linear mixing model for the subpixel objects, additional sensor modeling capability, atmospheric compensation approaches, and a matched filter detection algorithm. Unlike image simulation models, this model does not produce a simulated image, but rather predicts detection or error probabilities. Thus our model avoids the computational complexity of pixel-by-pixel ray tracing simulation approaches.

The model has been, and continues to be, validated by showing good agreement between predictions and measurements of spectral radiances, sensor signal-to-noise ratios, and detection probabilities derived from airborne hyperspectral sensor data.

We presented an example model analysis that showed which system parameters were most important in a given scenario. Also, we presented examples that predicted detection performance sensitivity to atmospheric and sensor parameters. In the cases examined, the most important and most sensitive model parameters were characteristics of the surface classes and environmental conditions of the scene, rather than sensor parameters or algorithmic choices. Certainly, though, situations exist in which these other parameters or choices can have significant effect.

The model has many advantages over other performance prediction tools, including quick execution, which enables extensive sensitivity studies to be conducted expeditiously. The model does, however, have a number of limitations, some of which can be addressed with further development, while others are inherent in the analytical approach. Inherent limitations include the inability to model specific geometries of scene objects or materials, especially those that involve multiple reflections, and sensor artifacts or processing algorithms that involve nonlinear operations.

Limitations of the model that we plan to address through additional development include implementation of additional sensor types and artifacts (e.g., Fourier-transform instruments, spectral jitter) and processing algorithms (e.g., physics-based atmospheric compensation, anomaly detection, linear unmixing, material identification). Also, we continue to assess the appropriate statistical distributions for various classes of hyperspectral data, as well as develop confidence intervals for detection probability predictions using appropriate models for the variability of the contributing factors.

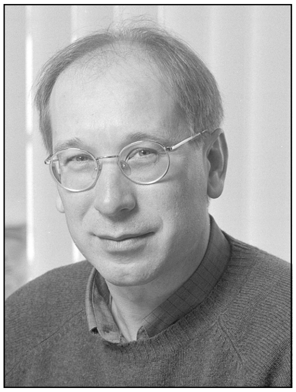
Even as the model is evolving and improving, it has already been useful in understanding the potential performance and limiting factors in spectral imaging scenarios supported by its current status. Thus it represents a step along the path toward the ultimate goal of a comprehensive understanding necessary for the optimal design and use of remote sensing systems.

Acknowledgments

The authors gratefully acknowledge the Office of the Deputy Undersecretary of Defense (Science and Technology) and the Spectral Information Technology Applications Center for supporting this work. The authors would also like to thank their colleagues in the Sensor Technology and System Applications group—Kris Farrar and Seth Orloff—for their contributions to the development of the model.

REFERENCES

1. J.P. Kerekes and D.A. Landgrebe, "An Analytical Model of Earth-Observational Remote Sensing Systems," *IEEE Trans. Syst. Man Cybern.* **21** (1), 1991, pp. 125–133.
2. B.V. Shetler, D. Mergens, C. Chang, F.C. Mertz, J.R. Schott, S.D. Brown, R. Strunce, F. Maher, S. Kubica, R.K. de Jonckheere, and B.C. Tousley, "Comprehensive Hyperspectral System Simulation: I. Integrated Sensor Scene Modeling and the Simulation Architecture," *SPIE* **4049**, 2000, pp. 94–104.
3. R.J. Bartell, C.R. Schwartz, M.T. Eismann, J.N. Cederquist, J.A. Nunez, L.C. Sanders, A.H. Ratcliff, B.W. Lyons, and S.D. Ingle, "Comprehensive Hyperspectral System Simulation: II. Hyperspectral Sensor Simulation and Preliminary VNIR Testing Results," *SPIE* **4049**, 2000, pp. 105–119.
4. J.R. Schott, *Remote Sensing: The Image Chain Approach* (Oxford University Press, New York, 1997).
5. A. Berk, L.S. Bernstein, and D.C. Robertson, "MODTRAN: A Moderate Resolution Model for LOWTRAN 7," GL-TR-89-0122, Spectral Sciences, Burlington, Mass. (30 Apr. 1989).
6. Y.J. Kaufman, "Atmospheric Effect on Spatial Resolution of Surface Imagery," *Appl. Opt.* **23** (19), 1984, pp. 3400–3408.
7. J.S. Accetta and D.L. Schumaker, *The Infrared and Electro-Optical Systems Handbook* **4**: *Electro-Optical Systems Design, Analysis, and Testing,* M.C. Dudzik, ed. (Environmental Research Institute of Michigan, Ann Arbor, Mich., 1993).
8. W.H. Farrand and J.C. Harsanyi, "Mapping the Distribution of Mine Tailings in the Coeur d'Alene River Valley, Idaho, through the Use of a Constrained Energy Minimization Technique," *Rem. Sens. Environ.* **59** (1), 1997, pp. 64–76.
9. S.J. Whitsitt and D.A. Landgrebe, "Error Estimation and Separability Measures in Feature Selection for Multiclass Pattern Recognition," LARS Publication 082377, Laboratory for Applications of Remote Sensing, Purdue University, W. Lafayette, Ind. (Aug. 1977).
10. J.P. Kerekes and J.E. Baum, "Spectral Imaging System Analytical Model for Subpixel Object Detection," *IEEE Trans. Geosci. Rem. Sens.* **40** (5), 2002, pp. 1088–1101.
11. L.J. Rickard, R. Basedow, and M. Landers, "HYDICE: An Airborne System for Hyperspectral Imaging," *SPIE* **1937**, 1993, pp. 173–179.
12. J. Pearlman, C. Segal, L. Liao, S. Carman, M. Folkman, B. Browne, L. Ong, and S. Ungar, "Development and Operations of the EO-1 Hyperion Imaging Spectrometer," *SPIE* **4135**, 2000, pp. 243–253.



JOHN P. KEREKES
is a staff member in the Sensor Technology and System Applications group. His primary area of research has been the development of remote sensing system performance models for geophysical parameter retrieval and object detection and discrimination applications. After joining Lincoln Laboratory in 1989, he worked on performance analyses for ballistic missile discrimination problems. He then developed analysis models supporting weather-satellite instrument development. In particular, he developed models to study the accuracy of proposed sensor systems in retrieving the atmospheric vertical temperature and water vapor profiles. Recently, he has developed modeling approaches for the prediction of the system performance of hyperspectral imaging systems. He is a senior member of the Institute of Electrical and Electronic Engineers (IEEE), and a member of the American Geophysical Society, the American Meteorological Society, and the American Society for Photogrammetry and Remote Sensing. He is an associate editor of the *IEEE Transactions on Geoscience and Remote Sensing,* and chairs the Boston Chapter of the IEEE Geoscience and Remote Sensing Society. He received his B.S., M.S., and Ph.D. degrees in electrical engineering from Purdue University.

JERROLD E. BAUM
is an associate staff member in the Sensor Technology and System Applications group, where he works on modeling and analyzing hyperspectral remote sensing systems. Jerry joined Lincoln Laboratory in 1989. He has analyzed Geostationary Operational Environmental Satellite (GOES) imager data, modeled infrared sensor performance in battlefield simulations, and evaluated active- and passive-infrared detection algorithms. Before coming to the Laboratory, he worked as a senior engineer at Textron Defense Systems, Wilmington, Massachusetts, for three and a half years, where he helped develop the detection algorithm for an airborne infrared sensor. In a previous life, Jerry taught high school physics in Maryland and Massachusetts for almost ten years. He earned an A.B. degree from Brandeis University in 1975 and an M.S. degree from the University of Maryland, College Park in 1980, both in physics.